

Лабораторные работы – 58 часов

№	Тема	ч
I Раздел. Программы, позволяющие создание контролирующих проектов (MyTestX, HotPot).		6
1.	Создание контролирующих проектов с помощью программы MyTestX	2
2.	Тестовая оболочка Hot Potatoes: основные сведения, установка и настройка конфигурации	2
3.	Создание контролирующих проектов с помощью программы Hot Potatoes	2
II Раздел. Программы для создания образовательных Web-ресурсов (Macromedia Dreamweaver)		10
4.	Разработка ЭУП с помощью программы Dreamweaver: разработка структуры ЭУП и оформление стартовой страницы	2
5.	Разработка ЭУП с помощью программы Dreamweaver: разработка страниц ЭУП с лекционным материалом и установление связи с главной страницей посредством гиперссылок	2
6.	Разработка ЭУП с помощью программы Dreamweaver: подготовка и внедрение страниц с практическими заданиями к главной странице ЭУП	2
7.	Разработка ЭУП с помощью программы Dreamweaver: подготовка и внедрение системы диагностики и контроля к главной странице ЭУП	2
8.	Разработка ЭУП с помощью программы Dreamweaver: внедрение на главную страницу ЭУП данных об авторе и списка литературы, звукового сопровождения	2
III Раздел. Создания электронных учебных материалов на основе технологии программирования (Delphi).		16
9.	Составляющие элементы ЭУП, разрабатываемого в среде Delphi	2
10.	Разработка главного меню ЭУП	2
11.	Разработка панели инструментов ЭУП	2
12.	Разработка оглавления ЭУП	2
13.	Оформление области вывода содержимого ЭУП	2
14.	Составление тестовых заданий для ЭУП	2
15.	Написание кода программы ЭУП	4
IV Раздел. Создания электронных учебных материалов на основе Flash -технологии		18
16.	Создание интерактивных учебных пособий во Flash: технология создания кнопки и использования языка ActionScript для работы с ней	2
17.	Создание интерактивных заданий в среде Flash на установление соответствия	2
18.	Создание интерактивных заданий с проверкой правильности их решения и внедрением звукового сопровождения в среде Flash	2

19.	Технология создания тестовых заданий с использованием текстовых полей во Flash	2
20.	Создание дидактических игр с помощью программы Flash	2
21.	Создание электронного пера и теста на установление соответствия при помощи цвета во Flash	2
22.	Технология разработки ЦОР на установление хронологической последовательности с использованием технологии перемещения объектов	2
23.	Использование приема создания маски в качестве курсора при разработке ЦОР	2
24.	Технология объединения нескольких ЦОР в один авторский ресурс	2
V Раздел. Технологии создания мультимедийных обучающих видеокурсов (Camtasia Studio).		8
25.	Выполнение предварительных настроек и запись видео с экрана монитора в программе Camtasia Studio	2
26.	Редактирование видеоклипов, вставка выносок и использование эффекта увеличения (Zoom) в программе Camtasia Studio	2
27.	Вставка эффектов переходов и титульного клипа в видеоурок в программе Camtasia Studio	2
28.	Вставка субтитров и тестовых заданий в видеоурок в программе Camtasia Studio	2
	Итого:	58

Раздел I. Программы, предназначенные для создания контролирующих проектов

Модель технологии обучения

Тема 1	Создание контролирующих проектов с помощью программы MyTestX
---------------	---

1.1. Технология обучения на лекции

<i>Учебное время:</i> 2 часа.	
<i>Форма учебного занятия:</i> лабораторная работа	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Обзор системы MyTest X - система программ для создания и проведения компьютерного тестирования. 2. Установка и запуск программы MeTestX. 3. Создание банка тестовых вопросов с помощью программы MyTestX.
<i>Цель учебного занятия:</i>	научить использовать тестовые оболочки для создания и проведения компьютерного тестирования.
<i>Педагогические задачи:</i> <ul style="list-style-type: none"> • Провести обзор системы MyTest X - система программ для создания и проведения компьютерного тестирования. • Показать, как производится установка и запуск программы MeTestX. • Организовать работу студентов по созданию банка тестовых вопросов с помощью программы MyTestX. 	<i>Результаты учебной деятельности:</i> <i>Студент должен знать:</i> <ul style="list-style-type: none"> • Составные элементы системы MyTest X - системы программ для создания и проведения компьютерного тестирования. <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • устанавливать и запускать программы MeTestX. • создавать банк тестовых вопросов с помощью программы MyTestX.
<i>Методы обучения</i>	Мозговой штурм, слайдовая презентация
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.

Условия обучения	Кабинет вычислительной техники
------------------	--------------------------------

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.1. Сообщает, тему, цель, планируемые результаты учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (65 мин)	<p>2.1. Проводит мозговой штурм, с помощью которых актуализируются знания о формах тестовых заданий, а также программ, с помощью которых возможно создание данных заданий (<i>приложение 2</i>).</p> <p>2.2. С помощью слайдовой презентации проводит обзор системы MyTest X - система программ для создания и проведения компьютерного тестирования (<i>приложение 3</i>).</p> <p>2.3. Организует установку программы MyTestX и запуск программы (<i>приложение 4</i>).</p> <p>2.4. Раздает методические рекомендации по созданию банка тестовых вопросов с помощью программы MyTestX. (<i>приложение 5</i>).</p>	<p>2.1. Отвечают на вопросы.</p> <p>2.2. Слушают, просматривают слайды, задают вопросы.</p> <p>2.3. Устанавливают и запускают программу.</p> <p>2.4. Выполняют задания лабораторной работы.</p>
3-й этап. Заключительная-результатирующая (10 мин)	3.1. Дает и разъясняет задания для домашней работы: Создать банк тестовых вопросов по предмету «Информатика» по различным темам (<i>приложение 6</i>).	Задают уточняющие вопросы.

Приложение 1

Тема: Создание контролирующих проектов с помощью программы MyTestX

Цель занятия: научить использовать тестовые оболочки для создания и проведения компьютерного тестирования.

План занятия:

1. Обзор системы MyTest X - система программ для создания и проведения компьютерного тестирования.
2. Установка и запуск программы MeTestX.
3. Создание банка тестовых вопросов с помощью программы MyTestX.

Приложение 2

Вопросы для проведения мозгового штурма

1. Что такое «тест»?
2. Какие виды тестовых заданий вам известны?
3. С помощью каких программ можно создавать тесты и проводить их?
4. Какие программы позволяют проводить автоматическую обработку результатов тестирования?

Приложение 3

Система работы по составлению тестовых заданий и проведению тестирования с использованием программы MyTestX Мастер-класс	Термин «тест» <ul style="list-style-type: none">• (от англ. «test») – в широком смысле слова <i>испытание, исследование, опыт.</i>• В педагогике «тест» определяется как <i>система заданий специфической формы, определенного содержания, возрастающей трудности, позволяющая объективно оценить структуру и качественно измерить уровень подготовленности учащихся.</i>
Преимущества тестирования <ul style="list-style-type: none">• строгость;• объективность;• технологичность;• применимость ко всем группам испытуемых;• интегрируемость;• междисциплинарность;• стимулирование мотивации и др.	Недостатки тестирования <ul style="list-style-type: none">• Разработка качественного тестового инструментария• Данные, получаемые в результате тестирования, не позволяют судить о причинах пробелов в знаниях обучающихся• Тест не позволяет проверять и оценивать уровни знаний, связанные с творчеством• Недостаток времени для глубокого анализа темы• Конфиденциальность тестовых заданий• В тестировании присутствует элемент случайности

Тестовые задания

- это дидактические и технологические средства объективного контроля подготовленности ученика.



Требования к составу тестового задания

1. Инструкция
2. Текст задания или вопроса
3. Правильный ответ



Тестовые задания закрытого типа

♦ альтернативных ответов

Форма задания

Текст задания (вопрос)	Ответ	
Утверждение 1	да	нет
Утверждение 2	да	нет
Утверждение 3	да	нет

Инструкция для задания альтернативных ответов:
Вам необходимо выбрать один вариант ответа, который Вы считаете правильным.

Тестовые задания закрытого типа

♦ альтернативных ответов

Ех: Выберите один вариант ответа, который Вы считаете правильным.

Программа Paint является программой для работы с электронными таблицами?
Варианты ответов:

- 1) да; 2) нет.



Формы тестовых заданий



Задания закрытой формы

Предполагают выбор правильного ответа из предложенных вариантов ответов.



Тестовые задания закрытого типа

♦ множественного выбора

Форма задания:

Вопрос (утверждение)

A. Вариант ответа 1

B. Вариант ответа 2

C. Вариант ответа 3

Инструкция для задания множественного выбора:
Выберите букву (и), (ие) соответствующую варианту (ам) правильного (ых) ответа (ов).

Тестовые задания закрытого типа

♦ множественного выбора

Ех: Выберите букву, соответствующую варианту правильного ответа.

Вопрос: Для какой операции с таблицами служит окно, фрагмент которого представлен на рисунке:



- A. Суммирование по строкам.
- B. Суммирование по столбцам.
- C. Фильтрация таблицы.
- D. Консолидация данных.

Тестовые задания закрытого типа

♦ множественного выбора

Инструкция: выберите буллы, соответствующие вариантам правильных ответов.

Вопрос: Выберите из приведенного списка типы объектов, с которыми работает Access.

Варианты ответов:

- A. Таблицы.
- B. Сведения.
- C. Запросы.
- D. Формы.
- E. Стили.
- F. Отчеты.
- G. Макросы.
- H. Модули.



Тестовые задания закрытого типа

♦ восстановления соответствия

Форма задания

Инструкция: Сопоставьте написанные в столбцах 1 и 2.

Вопрос:

Варианты ответа:

Столбец 1	Столбец 2
A.	1.
B.	2.
C.	3.
D.	4.
E.	5.

Тестовые задания закрытого типа

♦ восстановления соответствия

Инструкция: Сопоставьте написанное в столбцах 1 и 2. Вопрос: При создании таблицы СУБД можно использовать 5 возможностей. Выберите из приведенных определений те, которые им соответствуют.

Варианты ответа:

Столбец 1	Столбец 2
A. Равное значение	1. Если таблица не содержит данных.
B. Экстернатор	2. В поле ввода можно ввести только один символ.
C. Мастер таблиц	3. Создание таблицы можно использовать только в режиме конструктора.
D. Визуальная таблица	4. Создание таблицы и изменение ее структуры можно использовать только в режиме конструктора.
E. Связь с таблицей	5. Создание таблицы и изменение ее структуры можно использовать только в режиме конструктора.

Ответ:

- A. 1.
- B. 2.
- C. 5.
- D. 1.
- E. 4.

Тестовые задания закрытого типа

♦ восстановления последовательности

Форма задания

Инструкция: Расположите в правильной последовательности.

Вопрос:

Варианты ответа:

- 1. A.
- 2. B.
- 3. C.
- 4. D.

Тестовые задания закрытого типа

♦ восстановления последовательности

Ех: Установить правильную последовательность дней недели.

- 1. вторник
- 2. среда
- 3. воскресенье
- 4. понедельник
- 5. четверг
- 6. пятница
- 7. суббота



Ответ: 4-1-2-5-6-7-3.

Задания открытой формы

Испытуемый должен сам дописать ответ.



Тестовые задания открытого типа

♦ дополнения

Инструкция для заданий дополнения:

Вместо многоточия впишите только одно слово (символ, знак и т.д.).

Тестовые задания открытого типа

♦ дополнения

Инструкция: Вместо многоточия впишите только одно слово.

Ех: Силы, с которыми взаимодействуют два тела равны по _____ и противоположны по _____.



Ответ: величина; направление.

Тестовые задания открытого типа

♦ свободного изложения

Инструкция для заданий свободного изложения:

Закончите предложение (фразу), впишите вместо многоточия правильный ответ (словосочетание, фразу, предложение или несколько предложений).

21

Тестовые задания открытого типа

♦ свободного изложения

Ех: Задания части С в ЕГЭ.



Использование программы MyTestX для составления тестовых заданий и проведения тестирования

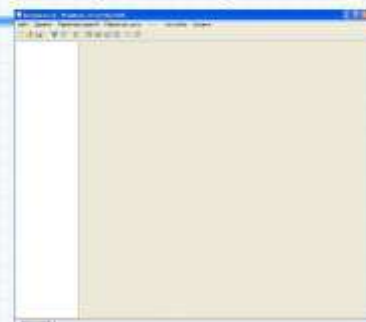
22

Запуск программы MyTestX



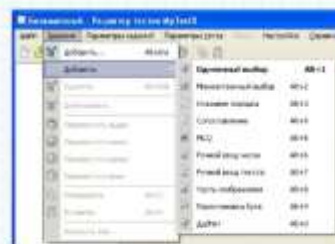
23

Окно редактора тестов



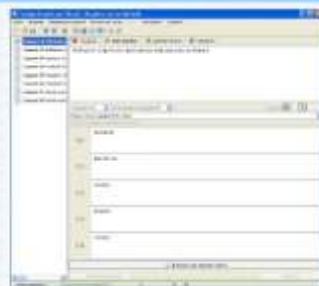
24

Выбор формы тестового задания



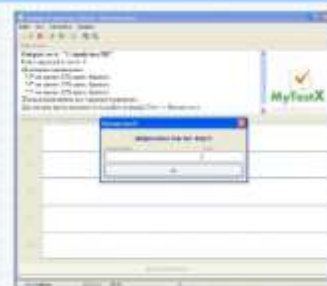
25

Готовый тест может выглядеть так:



26

Запуск теста



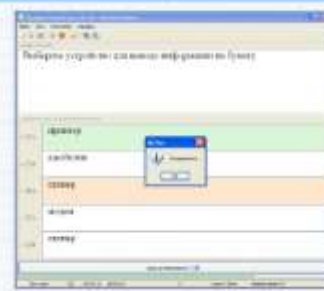
27

Проведение тестирования



28

Анализ неверных ответов



30

Получение результата тестирования



31

**Желаем успехов в
составлении
компьютерных
тестов!**

32

Приложение 4

Программа MyTestX доступна в двух вариантах - в виде инсталлятора и в виде архива.

Версия с инсталлятором - самый лучший и надёжный выбор для начинающего пользователя. Всё что вам нужно - скачать файл, и запустить его, после этого процесс установки произойдёт без вашего вмешательства. Инсталлятор поможет вам легко установить программу, а также создать ярлыки, ассоциировать файлы, указать нужный язык интерфейса...

1. Скачайте программу с сайта.
2. Запустите файл установки программы.
3. Последовательно выполните все предложенные шаги (укажите место установки, выберите нужные компоненты, язык интерфейса...). Для большинства пользователей подойдут настройки по умолчанию, т.е. достаточно проверить предложенные варианты и нажать далее...
4. Программа установлена. Запустить можно либо с помощью ярлыков на Рабочем столе, либо через меню Пуск.

Внимание! Для установки программы требуются права администрирования.

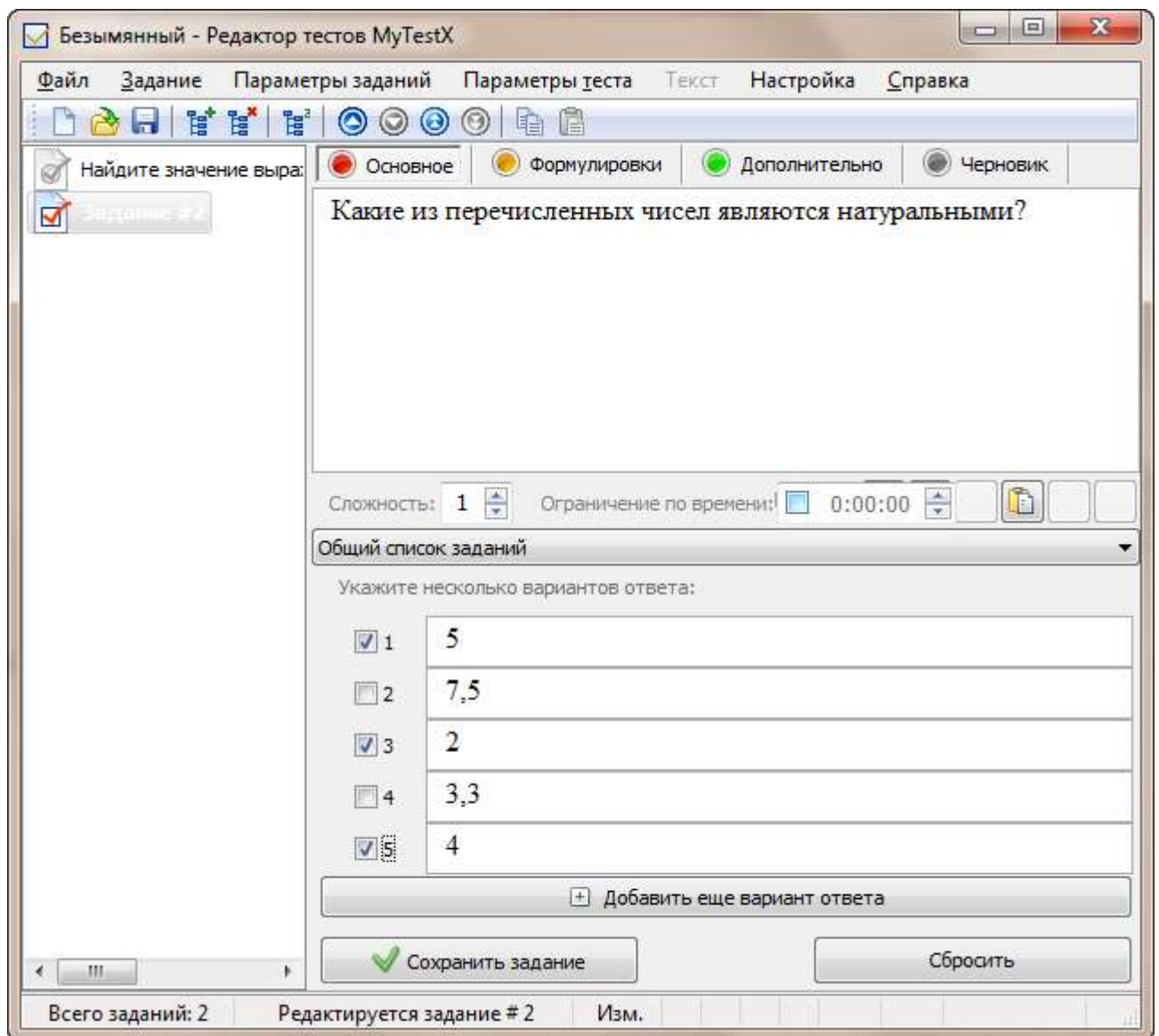
Приложение 5

Лабораторная работа по теме «Создание банка тестовых вопросов с помощью программы MyTestX»

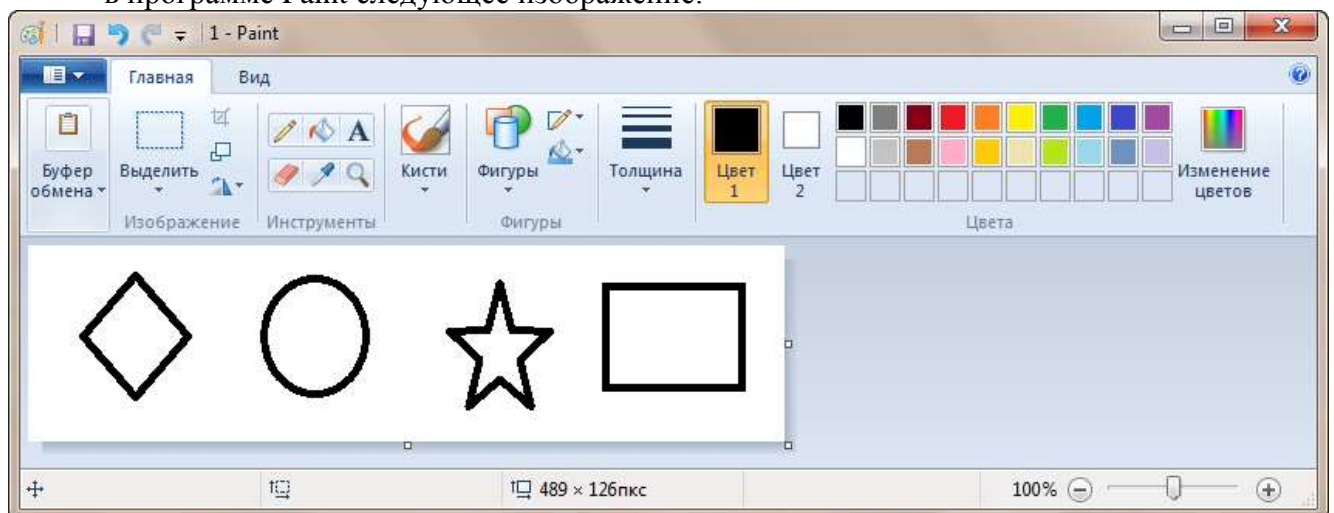
1. Запускаем программу MyTest Editor.
2. В меню Задание выбираем команду Добавить задание. В открывшемся окне можно выбрать десять видов вопросов. Создадим тест с одиночным выбором ответа. Вводим вопрос в верхнее поле. В нижние поля - варианты ответов. По умолчанию программа предлагает до 5 ответов (можно меньше). Такое количество наиболее оптимальное при тестировании. Если мало - то нажмите "добавить вариант". Напротив правильного ответа зажимаем зелёную точку, то есть фиксируем правильный вариант и жмем на кнопке Сохранить задание.

The screenshot shows the 'Безымянный - Редактор тестов MyTestX' window. The menu bar includes 'Файл', 'Задание', 'Параметры заданий', 'Параметры теста', 'Текст', 'Настройка', and 'Справка'. The toolbar contains icons for file operations and question management. The 'Задание #1' tab is active, showing a question editor. The question text is 'Найдите значение выражения $26*25-25*24+24*23-23*22-12*8$ '. Below the question, there are five radio button options: 1 (106), 2 (1), 3 (54), 4 (8), and 5 (0). The fifth option is selected. A button '+ Добавить еще вариант ответа' is below the options. At the bottom, there are buttons 'Сохранить задание' (with a green checkmark) and 'Сбросить'. The status bar at the bottom shows 'Всего заданий: 1', 'Редактируется задание #1', and 'Изм.'.


3. Повторим процедуру добавления задания, но выберем второй вид вопроса - Множественный выбор. Все процедуры повторяются:
 - текст вопроса
 - тексты ответов

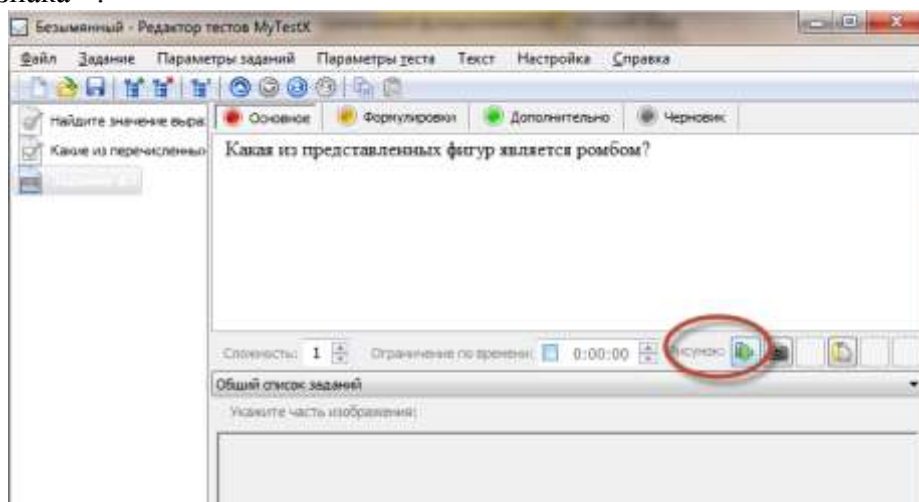


4. Теперь добавим вопрос "Часть изображения". Для этого предварительно создайте в программе Paint следующее изображение:

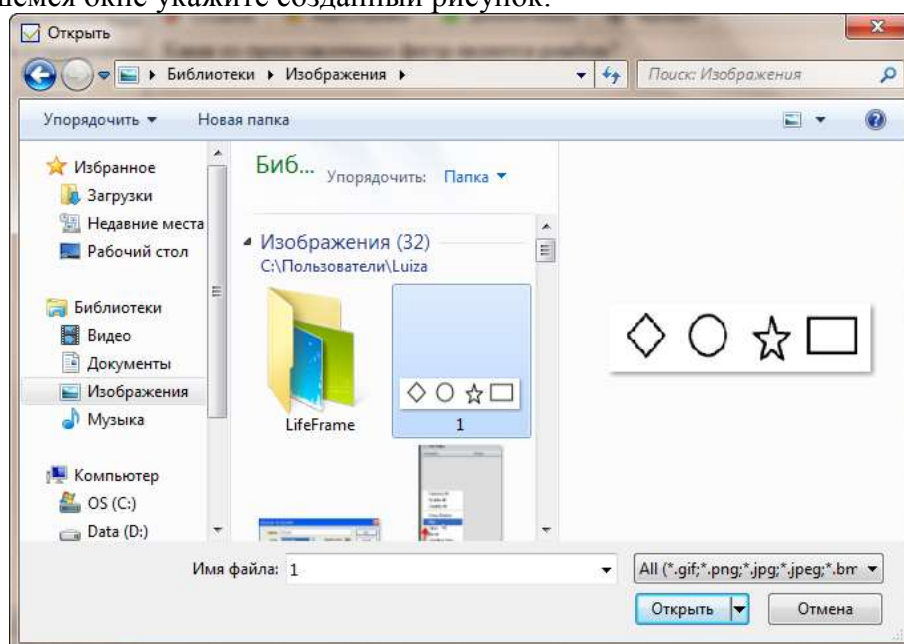


и сохраните его под именем 1.gif

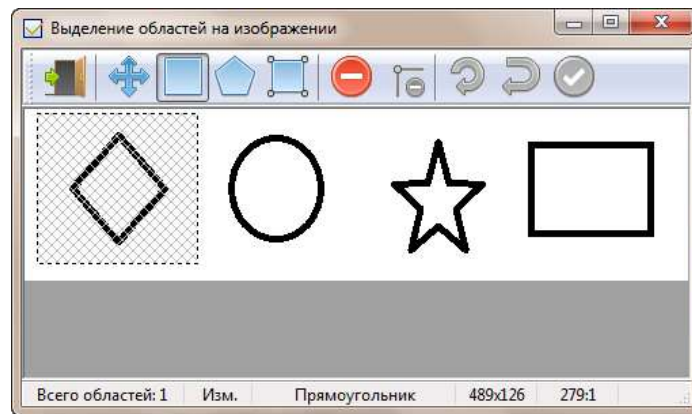
Далее в программе MyTestEditor щелкните на кнопке  Добавить задание и в открывшемся окне выберите пункт Часть изображения. Введите вопрос и в качестве ответа вставьте созданный рисунок. Для этого щелкните на кнопке с изображением зеленого знака +:



В открывшемся окне укажите созданный рисунок:

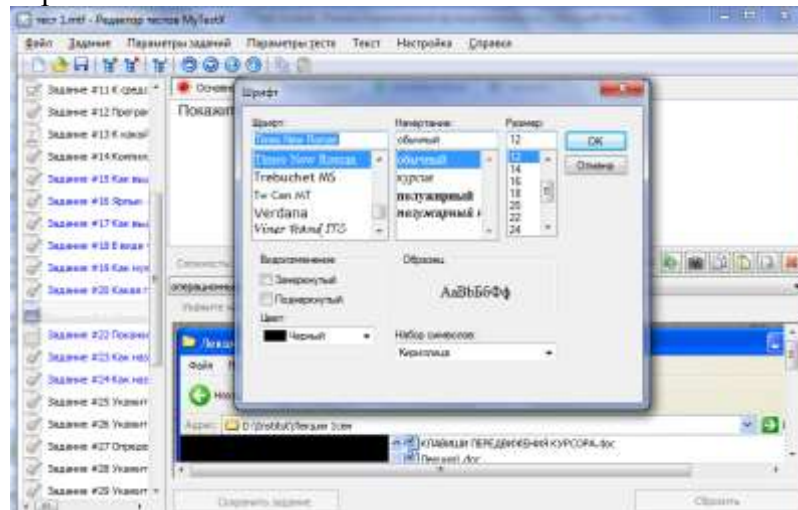


Дважды щелкнув по изображению, вызываем его редактор. В верхней строчке жмем на инструмент "Создать прямоугольник" и на рисунке выделяем нужную область рисунка.

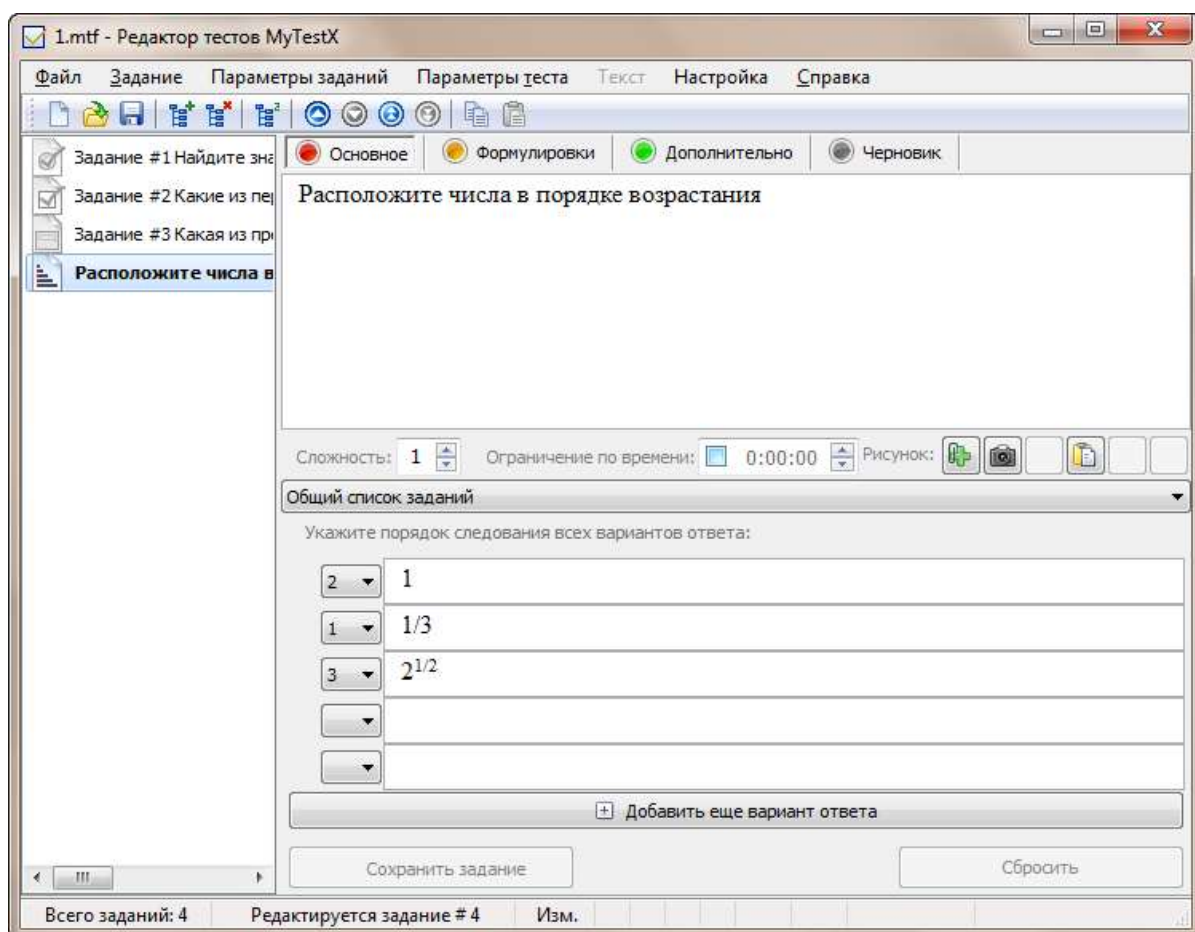


Теперь эта область рисунка является правильным ответом.

5. Для установки шрифтов для всех вопросов в строке меню выбираем пункт Параметры – Для всех - Изменить шрифт всех заданий. После этого откроется диалоговое окно, в котором настраиваем размер шрифта, цвет, гарнитуру шрифта, начертание.



6. Вставьте тестовое задание типа «Указание порядка». Для перевода чисел в верхний регистр войдите в меню Текст и в открывшемся диалоговом окне выберите кнопку верхний регистр.



7. Для сохранения теста необходимо выполнить команду Файл - Сохранить как...Файл сохранится с расширением mtf.
8. Запустите полученный тест с помощью MyTestStudent и протестируйте полученное.
9. Откройте файл вновь в режиме редактирования и добавьте тестовые вопросы типа Сопоставление и Перестановка букв. Сохраните выполненную работу.

Приложение 6

№	Тема
1.	История развития вычислительной техники.
2.	Программное обеспечение ЭВМ
3.	Аппаратное обеспечение ЭВМ
4.	Основы работы в программе MS Word
5.	Основы работы в программе MS Excel
6.	Основы работы в программе MS Access
7.	Основы работы в программе MS Power Point
8.	Компьютерные сети
9.	Язык программирования Pascal
10.	Язык программирования Delphi

Тема 2	Тестовая оболочка Hot Potatoes: основные сведения, установка и настройка конфигурации
---------------	--

1.1. Технология обучения на лабораторном занятии

<i>Учебное время: 2 часа.</i>	
<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Программа Hot Potatoes. Общие сведения. 2. Выгрузка и установка программы. 3. Регистрация программы и установка языка интерфейса программы. 4. Основные блоки программы. 5. Панель инструментов. 6. Настройка конфигурации.
<i>Цель учебного занятия:</i>	научить устанавливать, загружать и выполнять настройки в тестовой оболочке Hot Potatoes.
<i>Педагогические задачи:</i> <ul style="list-style-type: none"> • Дать общие сведения о программе Hot Potatoes. • Научить устанавливать программу. • Показать способы регистрации программы и установки языка интерфейса программы. • Ознакомить с основными блоками программы. • Охарактеризовать кнопки панели инструментов; • Научить выполнять настройку конфигурации тестовых заданий. 	<i>Результаты учебной деятельности:</i> <i>Студент должен знать:</i> <ul style="list-style-type: none"> • общие сведения о программе Hot Potatoes. • основные блоки программы; • назначение кнопок на панели инструментов. <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • устанавливать и запускать программу Hot Potatoes. • Проводить регистрацию программы и устанавливать язык интерфейса программы; • Выполнять настройку конфигурации тестового задания.
<i>Методы обучения</i>	слайдовая презентация, диаграмма Венна
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.2. Сообщает, тему, цель, планируемые результаты учебного занятия и план его проведения. <i>(приложение 1)</i> .	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (66 мин)	<p>2.1.С помощью слайдовой презентации дает краткую характеристику системы Hot Potatoes - <i>(приложение 2)</i>.</p> <p>2.2.Организует установку программы <i>(приложение 3)</i>.</p> <p>2.3.Инструктирует процесс регистрации программы и установки языка интерфейса программы. <i>(приложение 4)</i>.</p> <p>2.4.Предлагает выполнить запуск программы и знакомит с основными блоками программы <i>(приложение 5)</i>.</p> <p>2.5.Характеризует панели инструментов окна программы <i>(приложение 6)</i>.</p> <p>2.6.Показывает, как выполнять настройку конфигурации тестовых заданий <i>(приложение 7)</i>.</p>	<p>2.1. Просматривают слайды, слушают.</p> <p>2.2. Устанавливают программу на ПК.</p> <p>2.3. Регистрируют программу и устанавливают язык интерфейса программы.</p> <p>2.4. Запускают программу и знакомятся с основными блоками программы.</p> <p>2.6. Выполняют настройку конфигурации тестовых заданий.</p>
3-й этап. Заключитель но-результатирующ ий (10 мин)	3.1.Дает и разъясняет задания для домашней работы: Составить диаграмму Венна и выполнить сравнение тестовых оболочек MyTestX и Hot Potatoes <i>(приложение 8)</i>	Задают уточняющие вопросы.

Тема: Тестовая оболочка Hot Potatoes: основные сведения, установка и настройка конфигурации

Цель занятия: научить устанавливать, загружать и выполнять настройки в тестовой оболочке Hot Potatoes.

План занятия:

1. Программа Hot Potatoes. Общие сведения.
2. Выгрузка и установка программы.
3. Регистрация программы и установка языка интерфейса программы.
4. Основные блоки программы.
5. Панель инструментов.
6. Настройка конфигурации.





Программа Hot Potatoes. Общие сведения

Hot Potatoes - инструментальная программа-оболочка, предоставляющая преподавателям возможность самостоятельно создавать интерактивные задания и тесты для контроля и самоконтроля учащихся без знания языков программирования и привлечения специалистов в области программирования.

С помощью программы можно создать 10 типов упражнений и тестов по различным дисциплинам с использованием текстовой, графической, аудио- и видеoinформации.

Особенностью этой программы является то, что созданные задания сохраняются в стандартном формате веб-страницы: для их использования ученикам необходим только веб-браузер (например, Internet Explorer).

Ученикам не нужна программа Hot Potatoes: она требуется только преподавателям для создания и редактирования упражнений.

Программа широко используется во всем мире для создания заданий для изучения любых дисциплин.

Условия использования программ

Программы предлагаются в бесплатном (при соблюдении определенных условий) и платном вариантах.

Бесплатно пять указанных программ для разработки заданий и тестов могут использоваться государственными некоммерческими образовательными учреждениями при условии, что созданные с помощью программы учебные материалы будут находиться в Интернете в свободном доступе.

Программа **The Masher** является платной – она приобретается и регистрируется отдельно от других программ.

Приложение 3

Выгрузка и установка программы

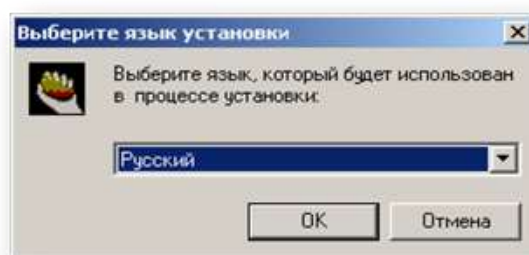
Hot Potatoes можно скачать с главной страницы сайта <http://web.uvic.ca/hrd/hotpot>



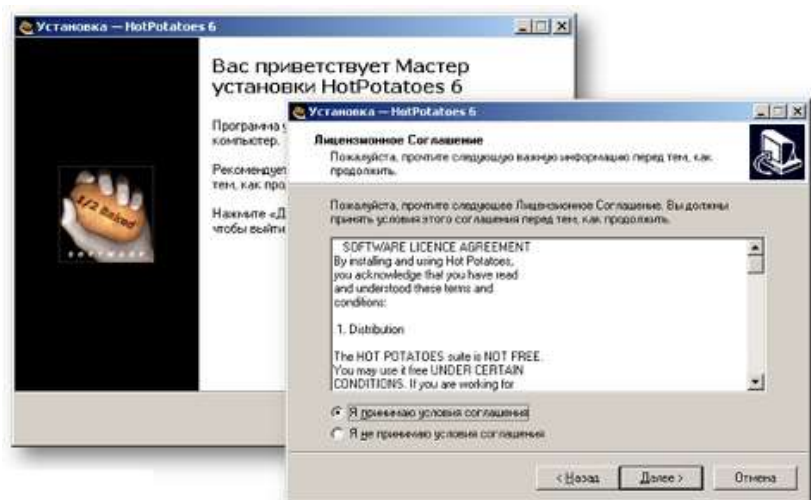
В разделе Downloads при обращении к ссылке **Hot Potatoes 6.2 installer** выполнится копирование саморазархивирующегося файла *setup_hotpot_6240.exe*. Размер файла – около 10 МБ.

Возможно, что в момент Вашего скачивания версия файла будет другая.

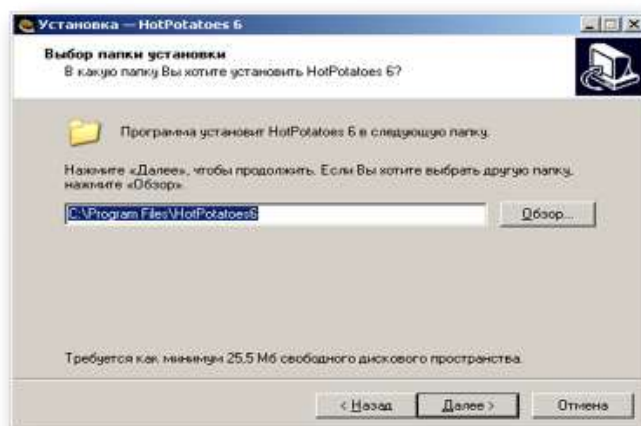
Для установки программы просто запустите этот файл на выполнение. Вы можете выбрать язык установки.



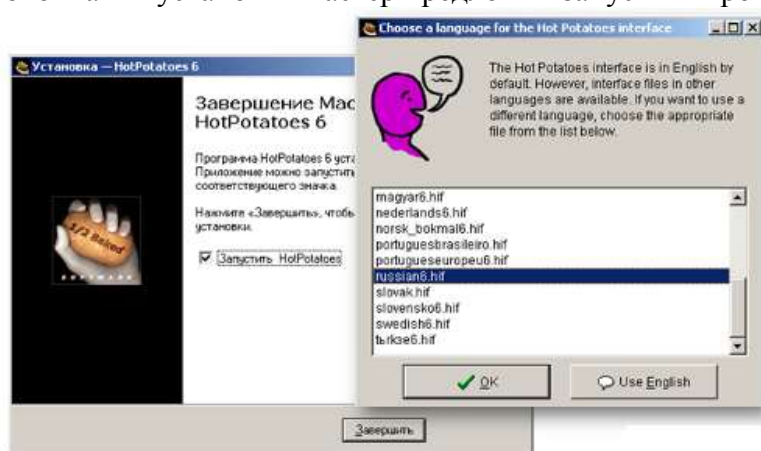
Затем просто следуйте инструкциям мастера:



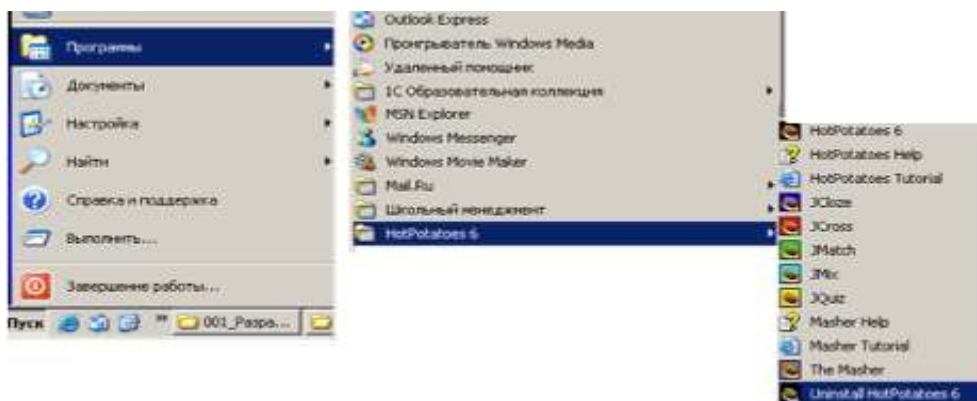
Выберите папку для установки программы:



После окончания установки мастер предложит запустить программу.



Для удаления программы Вы можете использовать главное меню – выбрать команду **Uninstall Hot Potatoes 6** или стандартную процедуру удаления программ (**Пуск – Панель управления – Установка и удаление программ**).



Приложение 4

Регистрация программы

Для использования пяти программ по созданию заданий и тестов без всяких ограничений Вы должны зарегистрировать Hot Potatoes. Регистрация программы бесплатная.

Незарегистрированная копия программы предоставляет ограниченные возможности (например, можно создать не более 2 вопросов в упражнении).

Зарегистрировать программу можно на странице регистрации на сайте. На страницу регистрации можно выйти:

- ✓ с главной страницы сайта программы <http://hotpot.uvic.ca/reg/register.htm>;
- ✓ по адресу <http://web.uvic.ca/hrd/hotpot/register.htm>;
- ✓ после установки программы – из меню **Помощь - Регистрация**.

После прочтения лицензионного соглашения и согласия с ним вы должны заполнить форму, где указываются Ваши имя, фамилия, адрес электронной почты и выбирается страна проживания. После заполнения формы (на английском языке) надо нажать кнопку **Submit**.

По указанному адресу Вы получите сообщение о логине и регистрационном коде.

После получения по электронной почте имени пользователя и регистрационного кода необходимо:

запустить программу;

выбрать в меню **Помощь** пункт **Регистрация** и ввести полученные имя и регистрационный код.



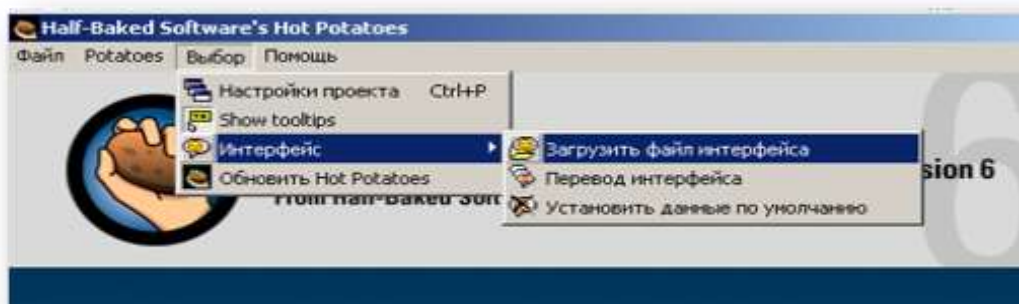
После регистрации в меню **Помощь** не будет пункта **Регистрация**, а в пункте **О программе** будет указано имя пользователя, на которого зарегистрирована программа.



Использование различных языков

Язык интерфейса программы по умолчанию английский. При первом запуске программа предлагает выбрать язык. Но Вы можете изменить язык в любой момент. Для

смены языка можно в меню **Выбор – Интерфейс – Загрузить файл интерфейса** выбрать нужный файл.



Папка, где лежат интерфейсы на различных языках, расположена в каталоге установки программы, по умолчанию - C:\Program Files\HotPotatoes6\translations.



Можно также сделать собственный перевод интерфейса, открыв окно **Выбор-Интерфейс-Перевод интерфейса**.

Приложение 5

Основные блоки программы

В состав Hot Potatoes входят 5 блоков программ для составления заданий и тестов разных видов. Каждый блок может быть использован как самостоятельная программа.

1. **JQuiz – Викторина** – вопросы с множественным выбором ответа (4 типа заданий).
2. **JCloze – Заполнение пропусков**.
3. **JMatch – Установление соответствий** (3 типа заданий).
4. **JCross – Кроссворд**.
5. **JMix – Восстановление последовательности**.

Все упражнения выполняются в режиме самоконтроля (режим тестирования предусмотрен только для вопросов с множественным выбором ответа). Результат выполнения заданий оценивается в процентах. Неудачные попытки приводят к снижению оценки.

Приложение 6

Описание панелей инструментов

Внешний вид панелей инструментов показан на рисунке.



Кнопки панелей инструментов позволяют реализовать быстрый доступ к основным операциям создания упражнений.

Вывод подсказки (к сожалению, на английском языке).

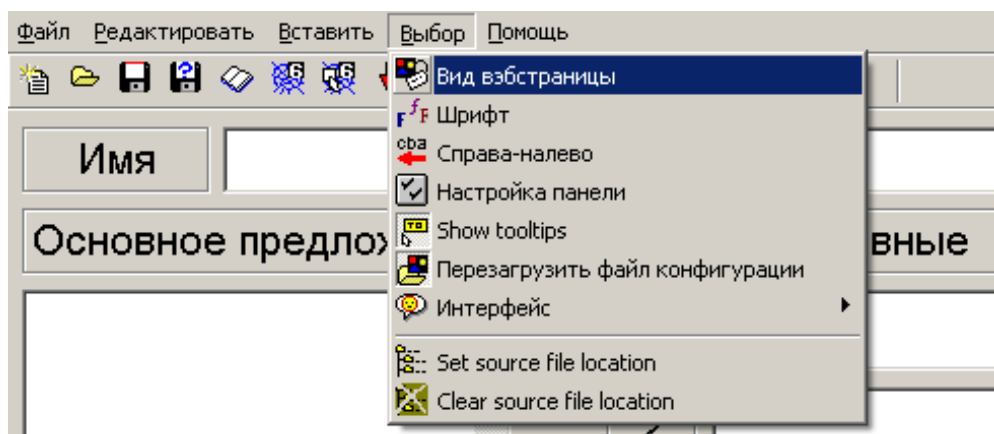
В других программах в панелях инструментов могут появиться и другие кнопки, предназначенные для выполнения действий, связанных именно с этими программами, например, автоматическое создание сетки в кроссвордах.

Приложение 7

Настройка конфигурации

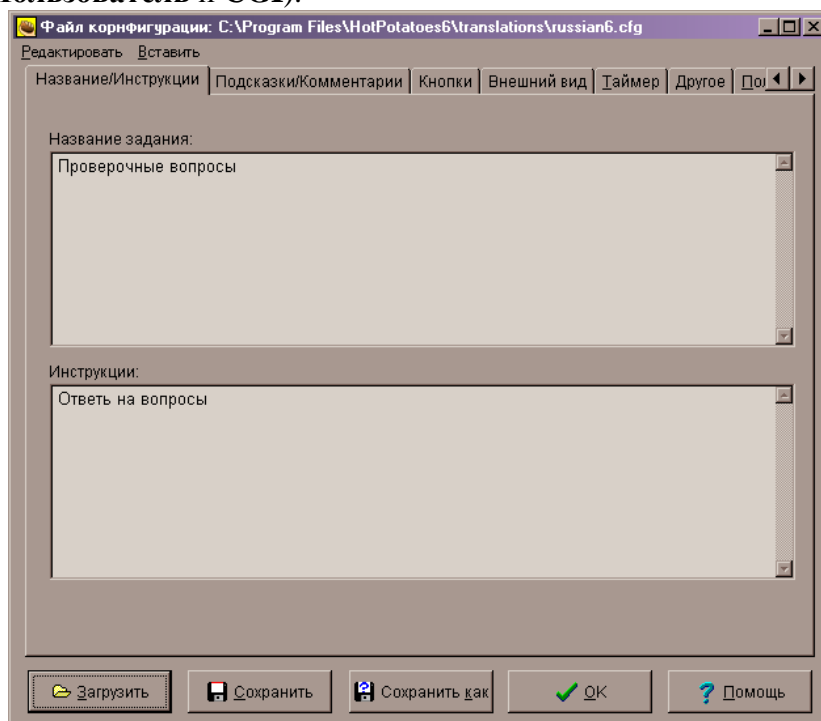
При создании заданий и тестов в каждой из пяти программ предусмотрена возможность настройки конфигурации страниц, общей для всех программ, а также и специфических для каждой программы параметров.

Для запуска диалога, в котором можно изменять конфигурацию, Вы можете использовать либо соответствующую кнопку панели инструментов, либо пункт меню **Выбор-Вид вебстраницы**.



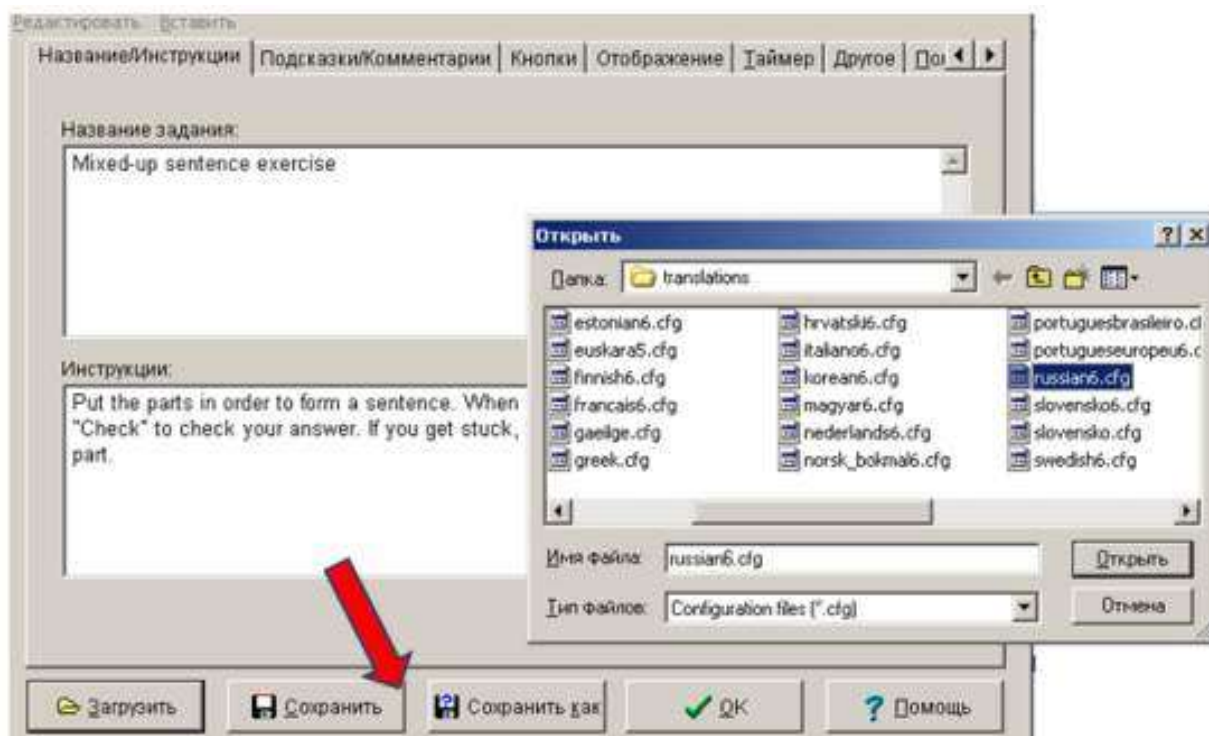
Общие для всех программ настройки позволяют изменить:

- ✓ подзаголовки и инструкции к заданиям (вкладка **Название/Инструкции**);
- ✓ тексты помощи и обратной связи (вкладка **Подсказки/Комментарии**);
- ✓ названия кнопок и их использование в заданиях (вкладка **Кнопки**);
- ✓ оформление заданий – шрифт, цвет фона, текста и т.п. (вкладка **Внешний вид**);
- ✓ установку таймера, ограничивающего время выполнения задания (вкладка **Таймер**);
- ✓ параметры, специфичные для каждой программы (вкладка **Другое**);
- ✓ дополнительные параметры для опытных пользователей (вкладки **Пользователь** и **CGI**).



Большая часть настроек появляется на языке установки, но Вы можете в любой момент поменять язык. Для этого Вы должны щелкнуть на кнопке **Load/Загрузить**, в

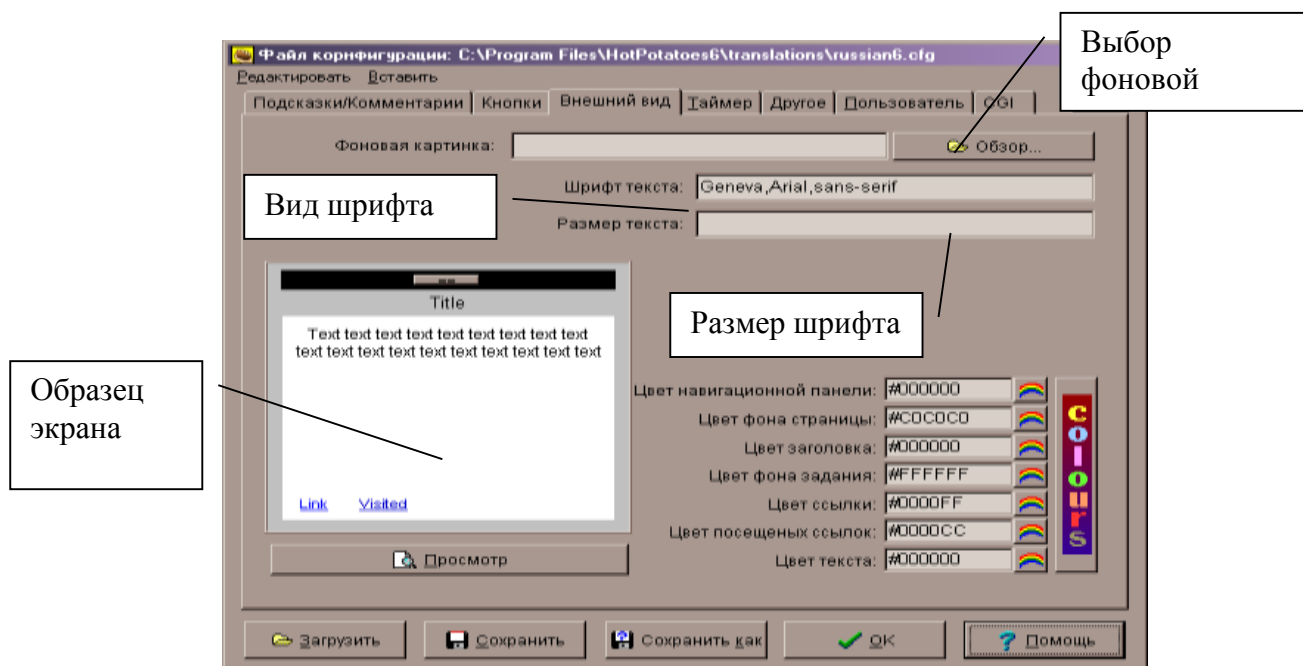
открывшемся окне выбрать файл russian6.cfg. Вы можете изменить все тексты по своему желанию. Большая часть текстовых настроек используется во всех пяти программах. Для сохранения настроек Вы просто сохраните изменения, нажав на кнопку **Сохранить** или создайте файл с новым именем, нажав на кнопку **Сохранить как**.



Внешний вид заданий

Вы можете изменять внешний вид задания с помощью вкладки **Внешний вид**. По умолчанию для задания установлен белый фон страницы, серый фон текста задания и черный цвет букв.

Цвет каждого элемента задания задается с помощью стандартной цветовой палитры, которая вызывается нажатием кнопки с радугой, расположенной рядом с полем нужного элемента (в поле цвет обозначен цифровым шестнадцатеричным кодом). Все вносимые изменения сразу же отражаются на образце экрана в левой части окна:



Можно установить фоновую картинку для задания, но помните, что фоновая картинка должна лежать в папке, где будет находиться страница с заданием. Вы можете также изменить шрифт по гарнитуре и размеру. Выбор гарнитуры не предусмотрен, поэтому Вы должны будете впечатать название гарнитуры. Если Вы не знаете это название, то просто не изменяйте его. Размер шрифта можно задать в пикселях.

Измененную конфигурацию можно использовать только для данного задания – для этого нужно нажать на кнопку «ОК».

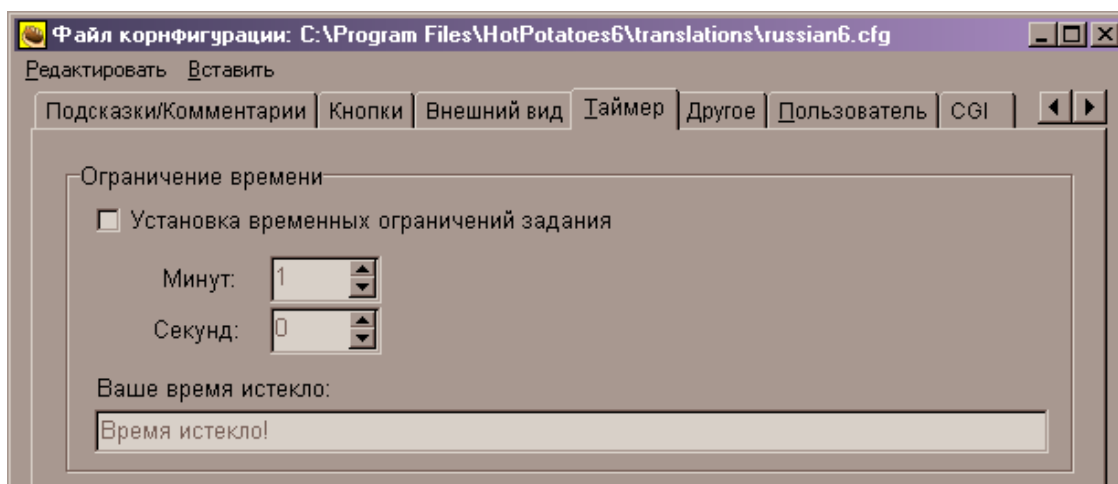
Чтобы использовать созданную конфигурацию в других заданиях программы, нужно сохранить файл конфигурации – кнопки **Сохранить** и **Сохранить как**, а затем загружать его – кнопка **Загрузить** в окне **Файл конфигурации**.



Настройка таймера

Для ограничения времени на выполнение задания или теста необходимо в окне настройки конфигурации

- ✓ выбрать вкладку **Таймер**;



- ✓ в разделе **Ограничение времени** установить флажок **Установка временных ограничений задания**;
- ✓ ввести временные показатели в минутах и/или секундах;
- ✓ впечатать в поле **Ваше время истекло** фразу, которая сообщала бы ученику о том, что у него больше нет времени для выполнения задания.

Замечание. Если учащийся не укладывается в отведенное для выполнения упражнения время, упражнение загружается еще раз и выполняется сначала.

Преобразование заданий и тестов в веб-страницу

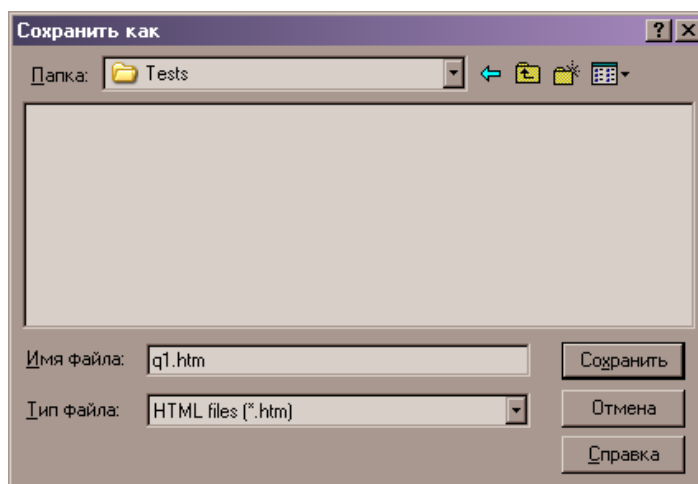
Введенные данные (задания, тесты и настройки конфигурации) необходимо преобразовать в формат веб-страницы и сохранить.

Для этого в пункте меню **Файл** используется пункт **Создать Вэбстраницу** или соответствующая кнопка на панели инструментов.

Если задание предусматривает несколько форматов упражнения, в этом пункте меню и, соответственно на панели инструментов, появляются дополнительные подпункты.

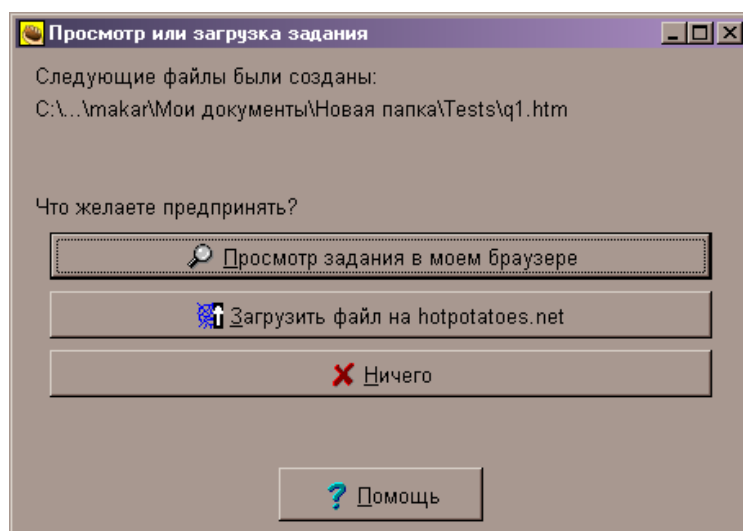


После выбора подпункта меню или нажатия на кнопку **Создать веб-страницу** появляется окно **Сохранить как**, в котором необходимо ввести имя файла страницы и выбрать или создать папку, в которой оно будет сохраняться. Удобнее, если Вы создадите папку, куда будете сохранять именно страницы в формате HTML отдельно, туда же поместите все рисунки и медиаобъекты, которые хотите поместить в задании.



***Замечание.** В имени файла желательно обозначить тему и порядковый номер задания, также в имени файла не рекомендуется использовать русские буквы и пробелы, поскольку это может вызвать проблемы при размещении заданий и тестов на серверах.*

Затем появится еще один диалог – **Просмотр или загрузка задания.**



Вы можете выбрать один из трех вариантов:

- ✓ посмотреть созданное упражнение в веб-браузере (**Просмотр задания в моем браузере**);
- ✓ разместить упражнение на сайте hotpotatoes.net (**Загрузить файл на hotpotatoes.net**);
- ✓ ничего не делать (**Ничего**).

При выборе первого варианта задание будет представлено в том виде, в котором с ним будет работать обучающийся.

Задание можно не только просмотреть, но и выполнить. После закрытия окна просмотра происходит возврат к рабочему экрану программы создания задания.

Если после просмотра в задание необходимо внести какие-то изменения, это можно сделать после просмотра, а потом обязательно еще раз сохранить (заменить) созданный

ранее файл страницы (эта процедура повторяется после любого внесения изменений в задание).

При выборе второго варианта необходимо соблюдать условия использования сайта www.hotpotatoes.net. Вы должны выполнить регистрацию на сайте, при этом Ваши задания смогут бесплатно размещаться только в течение ограниченного времени.

При выборе третьего варианта файл сохраняется без предварительного просмотра задания, и происходит возврат к рабочему экрану создания задания.

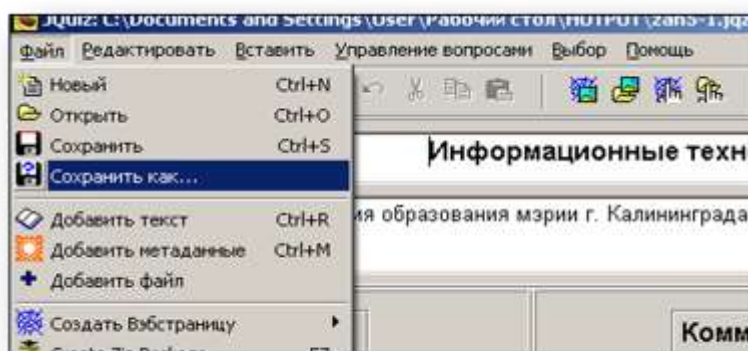
Сохранение проекта с заданием

В веб-страницы с заданиями изменения внести уже нельзя, поэтому необходимо сохранить проект задания.

Это можно сделать в любой момент:

- ✓ после завершения создания задания и тестов и настройки конфигурации;
- ✓ после просмотра созданной веб-страницы;
- ✓ при выходе из программы.

Файл проекта сохраняется стандартным способом: с помощью меню **Файл** и подпунктов **Сохранить** или **Сохранить как** или соответствующих кнопок на панели инструментов.



Файл проекта для каждой из пяти программ имеет свое расширение и свою пиктограмму.

Замечание. Файл проекта заданий, объединенных в один урок с помощью программы *The Masher*, обозначается пиктограммой программы *The Masher* и также имеет свое расширение.

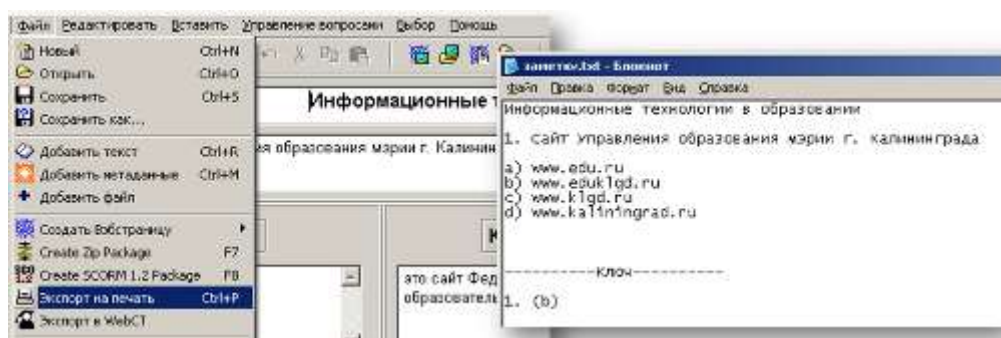
Задание или тест		Пиктограмма	Расширение имени файла
JQuiz	Викторина (множественный выбор ответа)		*.jqz
JCloze	Заполнение пропусков		*.jcl
JCross	Кроссворд		*.jcw
JMix	Восстановление последовательности		*.jmx
JMatch	Установление соответствий		*.jmt
The Masher	Инструменты		*.jms

Для того, чтобы можно было легко ориентироваться в файлах проектов и страниц, рекомендуется файлам данных и файлам страниц давать одинаковые имена и хранить файлы проектов в одной папке с созданными страницами.

Печать заданий

Все задания можно распечатать.

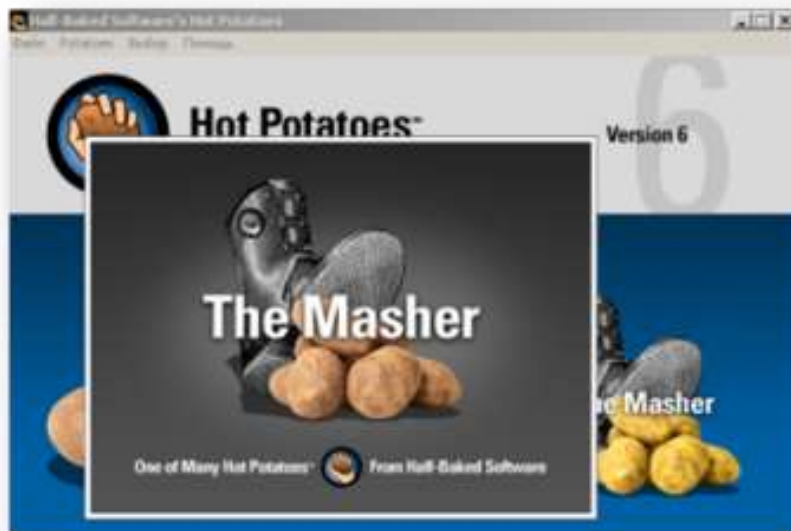
Для печати из программ просто надо выбрать в меню **Файл** пункт **Экспорт на печать**. Вся необходимая информация в текстовом виде сохраняется в буфере обмена. Затем просто надо запустить любой текстовый редактор и выполнить вставку из буфера. Все задания и тесты печатаются с ответами.



Кроссворд распечатывается из веб-браузера как веб-страница, специально созданная для печати.

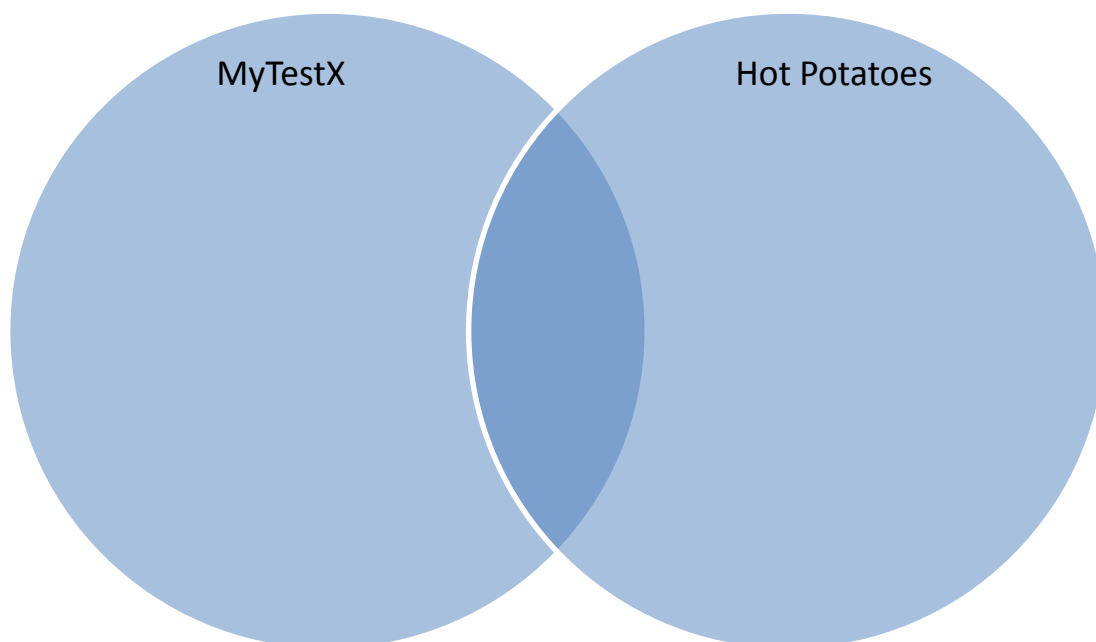
Объединение заданий в уроки и тематические блоки

Созданные задания и тесты можно объединить в блоки, уроки и серии уроков различными способами. Самым простым способом является использование программы The Masher, входящей в состав HotPotatoes. Правда, для использования этой программы с любыми возможностями необходимо за нее заплатить.



Другие (бесплатные) способы объединения уроков в блоки мы рассмотрим позже.

Приложение 8



Тема 3	Создание контролирующих проектов с помощью программы Hot Potatoes
---------------	--

1.1. Технология обучения на лабораторном занятии

<i>Учебное время: 2 часа.</i>	
<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	1. Кроссворд (JCross) 2. Установление соответствий (JMatch) 3. Викторина (JQuiz)
<i>Цель учебного занятия:</i>	научить устанавливать, загружать и выполнять настройки в тестовой оболочке Hot Potatoes.
<i>Педагогические задачи:</i> Организовать работу студентов по созданию <ul style="list-style-type: none"> Кроссвордов (JCross) Заданий на установление соответствий (JMatch) Викторины (JQuiz) 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь составлять:</i> <ul style="list-style-type: none"> Кроссворд (JCross) Задания на установление соответствий (JMatch) Викторину (JQuiz)
<i>Методы обучения</i>	Мозговой штурм, техника Зиз-заг
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.3. Сообщает, тему, цель, планируемые результаты учебного занятия и план его проведения. (приложение 1).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть	2.1.С помощью мозгового штурма актуализирует знания студентов о приложении Hot Potatoes. 2.2.Техника «Зиг-заг». Делит студентов на 3 группы и раздает экспертные листы для ознакомления с	2.1. Отвечают на вопросы.

(67 ин)	методом создания заданий в среде Hot Potatoes (приложение 2).	<p>2.2.1. Знакомятся с учебным заданием и индивидуально выполняют за ПК.</p> <p>2.2.2. «Встреча экспертов» - изучающие один и тот же материал в разных группах встречаются и обмениваются информацией как эксперты, прорабатывают свой вопрос, сообщают, как планируют, как наиболее эффективно изложить эту информацию членам своих исходных групп.</p> <p>2.2.3. «Эксперты» возвращаются в свои исходные группы и обучают других всему новому, что узнали сами: каждый докладывает о своей части задания.</p> <p>2.2.4. Каждый возвращается к своему компьютеру и выполняет задания.</p>
<p>3-й этап.</p> <p>Заключительные-результатирующие</p> <p>(10 мин)</p>	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <p>Составить диаграмму Венна и выполнить сравнение тестовых оболочек MyTestX и Hot Potatoes (приложение 8)</p>	<p>Задают уточняющие вопросы.</p>

Приложение 1

Тема: Создание контролируемых проектов с помощью программы Hot Potatoes

Цель занятия: научить использовать тестовую оболочку Hot Potatoes для создания и проведения компьютерного тестирования.

План занятия:

7. Программа Hot Potatoes. Общие сведения.
8. Выгрузка и установка программы.
9. Регистрация программы и установка языка интерфейса программы.
10. Основные блоки программы.
11. .

Приложение 2

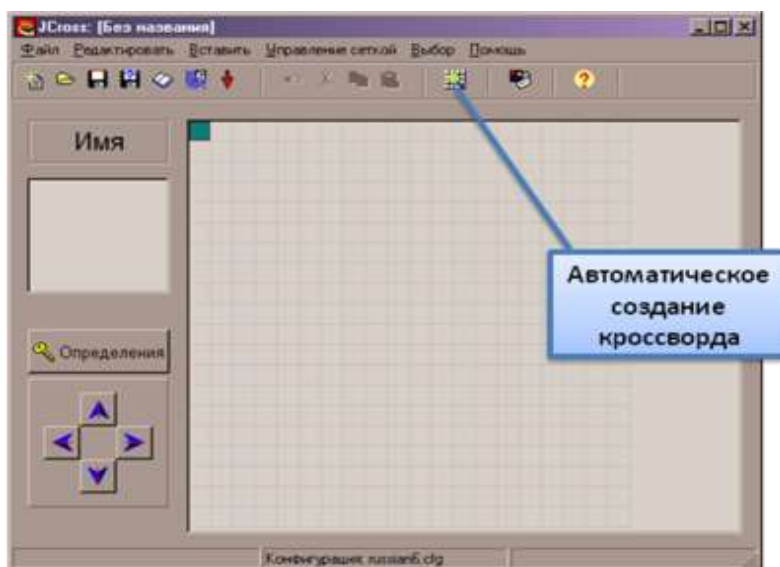
Экспертный лист1

Кроссворд (JCross)

Программа JCross предназначена для быстрого составления кроссвордов. Этот вид заданий очень популярен в средней школе.

3.1. Составление кроссворда

Для составления кроссворда надо запустить программу **HotPotatoes**, а затем выбрать программу **JCross** из меню **Potatoes** или на экране **HotPotatoes** щелкнуть на соответствующей картошке.



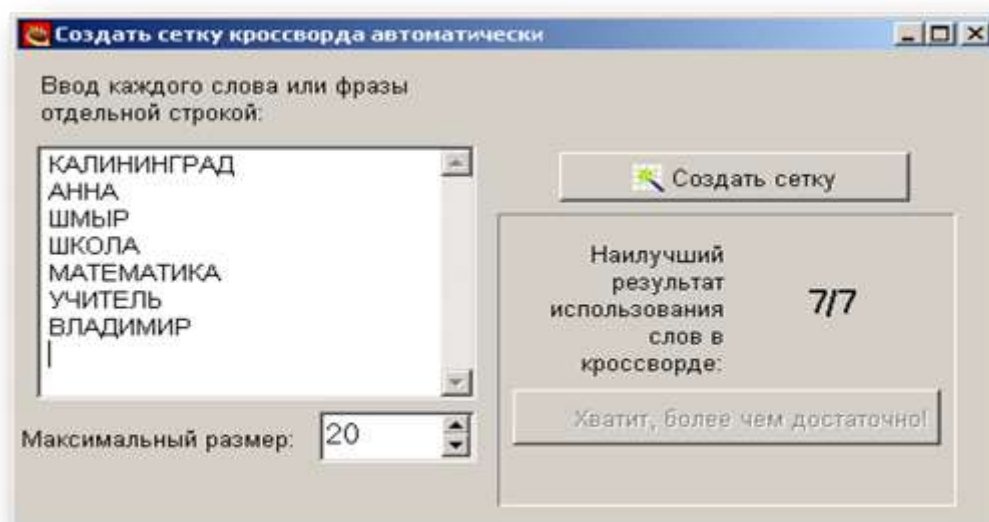
В поле **Имя** надо ввести имя кроссворда.

Для размещения слов в кроссворде существуют два режима:

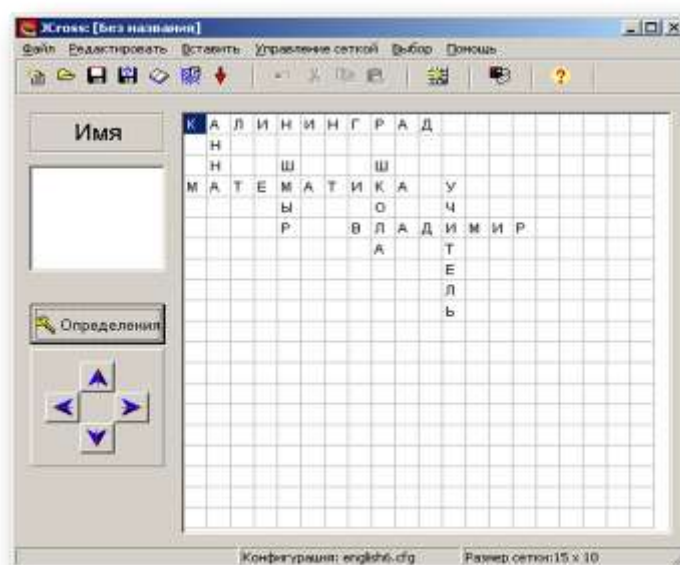
- ✓ ручное размещение слов;
- ✓ автоматическое размещение слов.

При ручном размещении слов положение слов в сетке определяется пользователем: слова буква за буквой вводятся в поле кроссворда.

Для автоматического создания кроссворда надо щелкнуть на кнопке **Автоматическое создание** в панели инструментов или выбрать пункт меню **Управление сеткой – Автоматическое создание**.

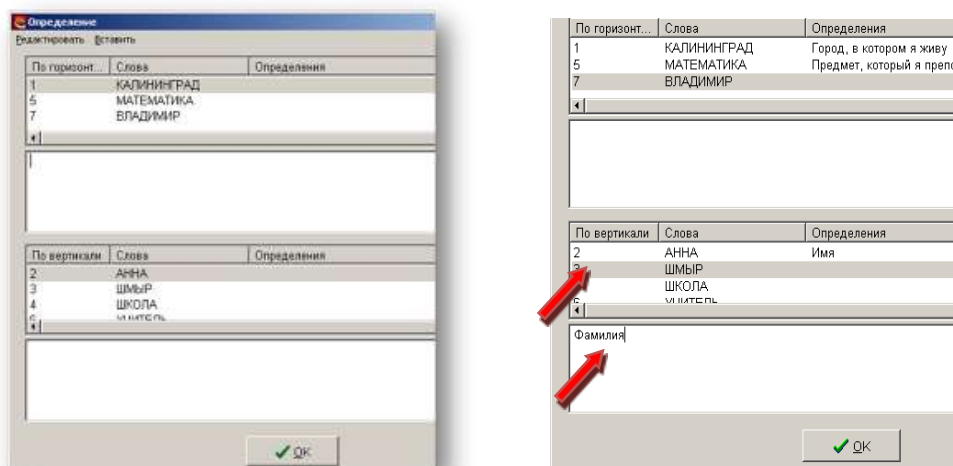


При автоматическом размещении слов все слова вводятся списком в поле ввода слов в открывшемся диалоге, при этом все слова вводятся в отдельные строки, а затем программа подбирает варианты их размещения. В поле **Максимальный размер** указывается максимальное число ячеек в кроссворде. Щелчок на кнопке **Создать сетку** приведет к созданию кроссворда.



***Замечание.** С помощью кнопок-стрелок в левом нижнем углу экрана можно перемещать созданный кроссворд в рамках поля.*

Следующий шаг заключается в том, что надо ввести определения к словам. Эти определения описывают указанные слова. Для ввода определений надо нажать кнопку **Определения**. Появится диалог **Определения**:



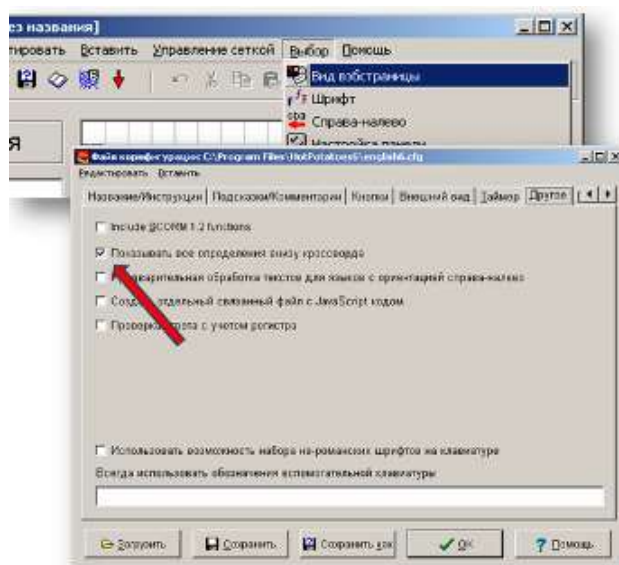
В верхней части экрана расположены слова, которые в кроссворде размещены по горизонтали, в нижней – по вертикали. Для ввода определения надо выделить нужное слово в списке, в поле под списком ввести определение и нажать кнопку **ОК**.

Последовательность ввода определений не имеет значения.

После введения определений можно вносить любые изменения в сетку – при перемещении слов программа будет сохранять соответствующие определения.

3.2. Дополнительные настройки

Дополнительные настройки в конфигурации задания можно задать через меню - **Выбор – Вид вебстраницы**.

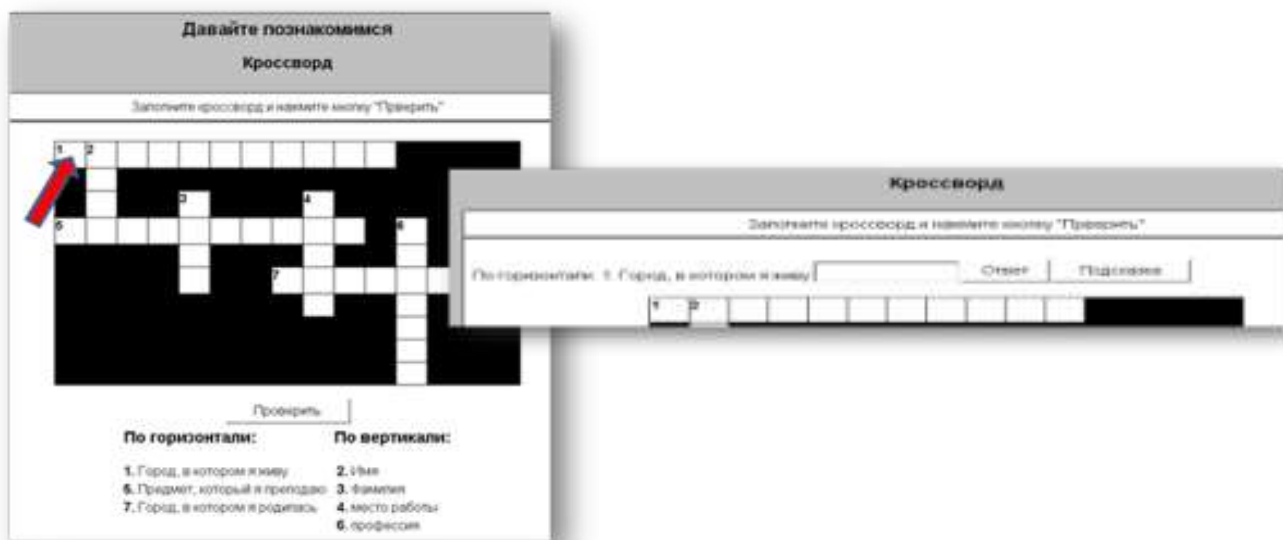


Определения к каждому слову могут появляться все сразу, как в кроссвордах, публикуемых в газетах и журналах, или же по одному определению. Для выбора этого режима отображения используется флажок **Показывать все определения внизу**

кроссворда. А затем надо сохранить проект и опубликовать кроссворд в виде HTML-страницы.

3.3. Выполнение задания

Выполнение задания – кроссворда заключается в следующих действиях. Учащийся выбирает первую клетку слова в кроссворде. При щелчке на этой кнопке появляется определение соответствующего слова, вне зависимости от того, есть полный список слов и определений или нет:

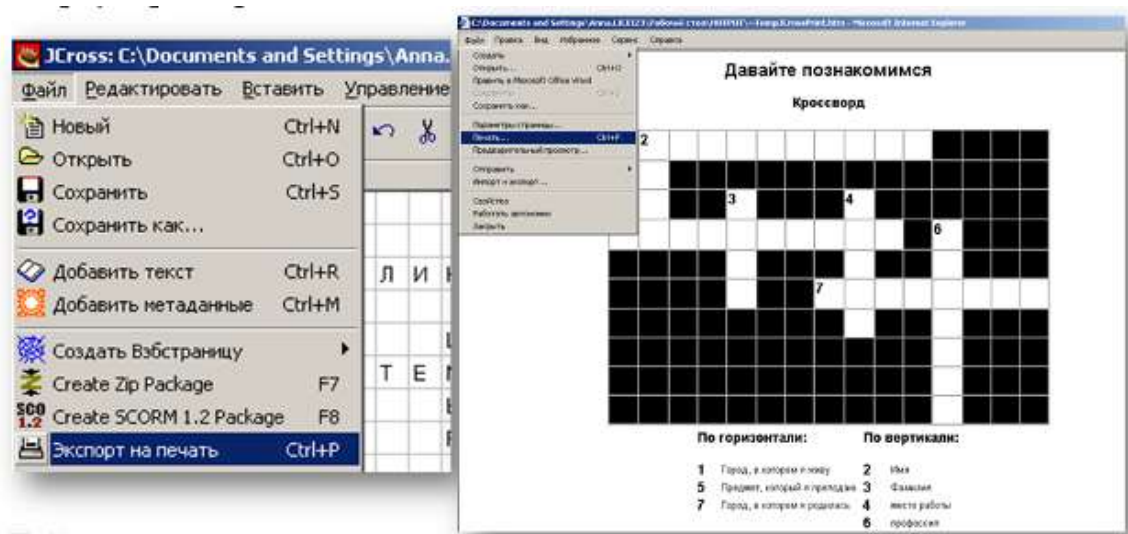


Последовательность заполнения кроссворда произвольная. После заполнения всех клеток кроссворда можно проверить верность заполнения.

3.4. Распечатка кроссворда

Распечатать кроссворд можно в двух вариантах: незаполненный и заполненный. Для создания версии для распечатки можно использовать меню **Файл – Экспорт для печати**.

На экране появится веб-страница с кроссвордом в черно-белом варианте. Распечатка страницы может быть выполнена из браузера через меню **Файл – Печать**.



Чтобы распечатать вариант страницы с решенным кроссвордом, нужно на полученной странице щелкнуть левой кнопкой мыши по названию кроссворда. Повторный щелчок по названию возвращает к исходному варианту – странице с нерешенным кроссвордом.



Приложение 3

Экспертный лист 2

Установление соответствий (JMatch)

Вопросы на соответствие – очень популярный тип вопросов. Примером такого типа вопроса является вопрос вида:

❖ *Установите соответствие между наборами слов:*

корова	• яйца
курица	• шерсть
овца	• молоко

Такой же тип вопросов позволяет указать порядок следования слов, фраз или действий.

Например, вопрос может иметь вид:

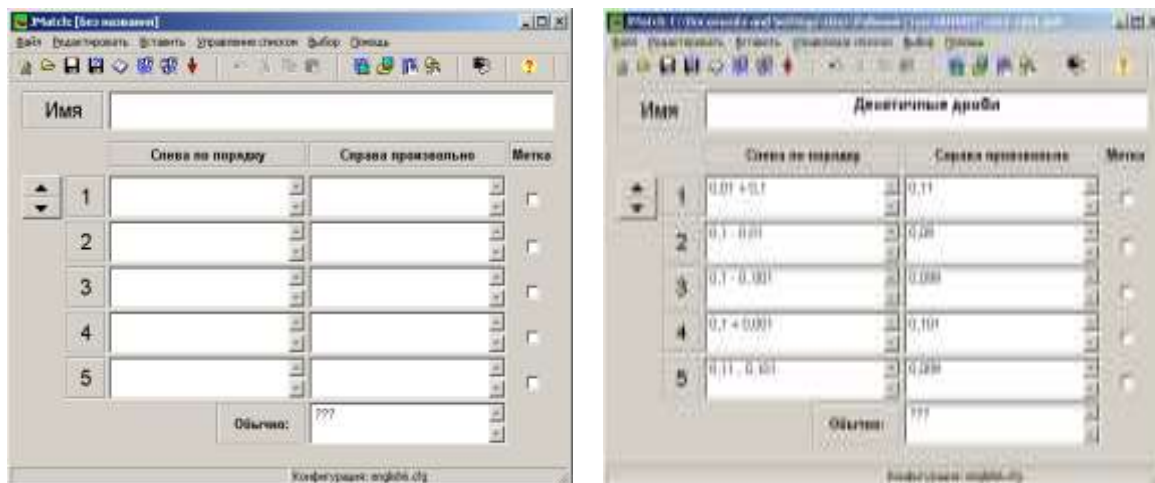
❖ *Установить слова в алфавитном порядке: яблоко, ананас, слива, тыква.*
Задания могут быть представлены в трех форматах:

- ✓ установление соответствий с помощью перемещения элементов мышью;
- ✓ выбор варианта соответствия из раскрывающегося списка;
- ✓ карточки для запоминания соответствий.

Тип задания выбирается на этапе сохранения в виде страниц HTML.

Ввод задания

Запустите программу HotPotatoes: выберите программу **JMatch** на экране или в меню **Potatoes**– откроется рабочее окно программы.

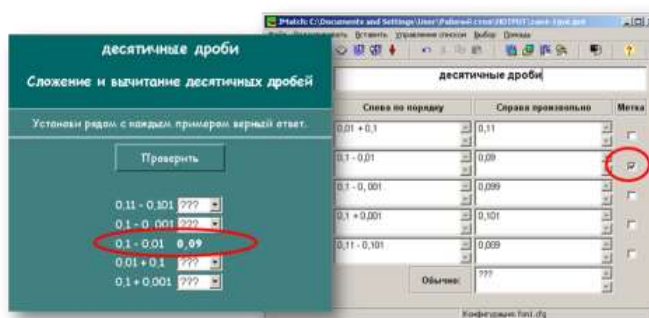


Вы можете ввести заголовок задания или теста в поле **Имя**.

В столбик слева вводится фиксированный список элементов, а в столбик справа – элементы, которые могут быть перепутаны, и из них будет выполняться подбор соответствий к элементам фиксированного списка.

Элементы вводятся в правильной последовательности. Они будут перемешиваться автоматически при загрузке страницы.

Для первых двух типов вопросов при вводе ответов можно выполнить фиксацию одного из перемещаемых элементов списка, установив метку – флажок. Соответствие, отмеченное как «зафиксированное», будет устанавливаться на экране автоматически и являться как бы ориентиром ответа.



Если метку убрать, то вопрос будет выглядеть по-другому:



В заданиях второго типа используется поле **Обычно**. Это то поле, которое показано в списке элементов для выбора, в примере это поле **???**.



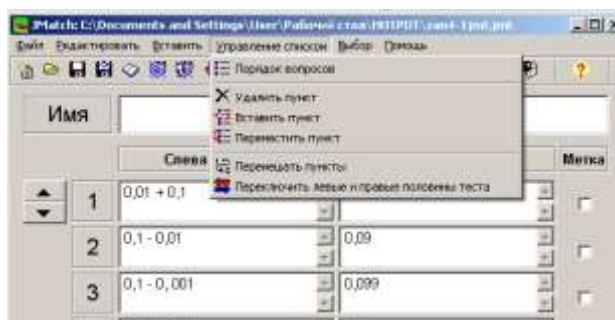
В качестве такого текста лучше использовать:

- ✓ один или несколько вопросительных знаков (как это сделано в программе по умолчанию)
- ✓ не заполненное поле – таким оно останется на экране
- ✓ дополнительный неправильный ответ
- ✓ один из правильных ответов



Редактирование задания

Редактирование задания можно выполнить из меню **Управление списком**. Вы можете удалить выбранный пункт, вставить ответ, переместить ответы, перемешать ответы.



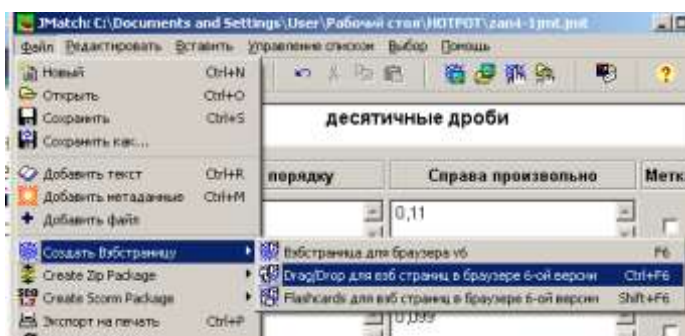
Необходимо выполнить сохранение проекта, а затем опубликовать вопрос в виде HTML-страницы.

Публикация в виде страницы HTML

В этой программе возможно применение трех разных форматов заданий/тестов:

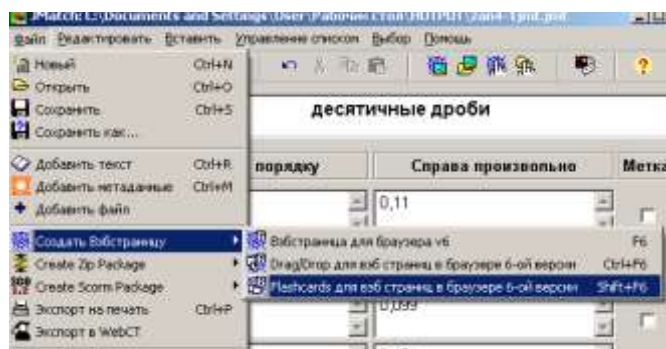
- ✓ использование механизма drag&drop, когда для установления соответствия используется перетаскивание мышью;
- ✓ использование раскрывающегося списка, из которого выбирается нужный пункт;
- ✓ использование карточек для запоминания слов и выражений.

Для создания задания, в котором соответствие устанавливается с помощью перемещения элементов мышью, необходимо выбрать пункт в меню **Файл – Создать Вебстраницу-Drag/Drop** для **вэбстраниц** или соответствующую кнопку на панели инструментов.



Для создания задания, в котором выбирается вариант соответствия из раскрывающегося списка, необходимо в меню **Файл - Создать веб-страницу** выбрать подпункт **Веб-страница** или соответствующую кнопку на панели инструментов.

Карточки для запоминания слов и выражений создаются только с помощью меню: **Файл –Создать веб-страницу – Flashcards для вэбстраниц**.



Внешний вид задания, использующего механизм Drag&Drop, показан на рисунке:



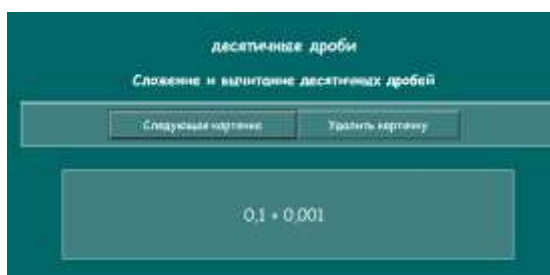
Для установления соответствия учащийся нажимает левую кнопку мыши на любом элементе справа и переносит его к левому столбцу.

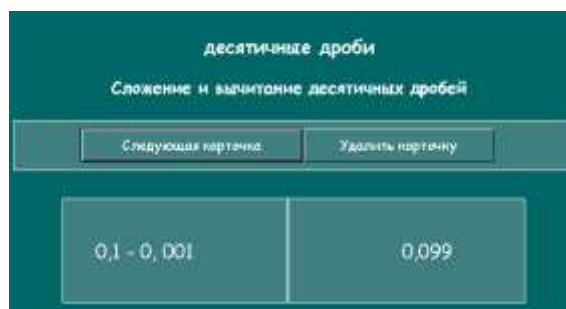
Задание, созданное с использованием выпадающего списка, показано на рисунке:



Для установки соответствия просто выбирается нужный пункт из раскрывающегося списка.

При выполнении задания, созданного с использованием карточек на экране сначала появляется карточка с текстом из одного поля задания, после нажатия на кнопку **Следующая карточка** – две карточки с правильным вариантом соответствия.





С помощью кнопки **Следующая карточка** осуществляется переход к следующим парам соответствий. Карточки с выученными соответствиями можно убирать из списка с помощью кнопки **Удалить карточку**.

Приложение 5

Экспертный лист3

Викторина (JQuiz)

Возможности

С помощью этой программы могут быть созданы тесты на основе вопросов с различными вариантами представления ответов:

- ✓ альтернативный выбор – один правильный ответ из нескольких вариантов ответов;



- ✓ множественный выбор – несколько правильных ответов;



- ✓ короткий ответ – короткий ответ открытого типа (вводится с помощью клавиатуры);

- ✓ смешанный вариант ответа – объединяет вопросы с ответом открытого типа и множественным выбором ответа: после ввода неверного ответа (возможное количество неверных ответов устанавливается при создании задания) задание с кратким ответом трансформируется в задание с множественным выбором.

В один тест или задание можно включить несколько вопросов разных типов.

Количество вопросов в упражнении не ограничено.

При выполнении задания учащийся может выбрать режим просмотра задания по одному вопросу или всех вопросов сразу. Для этого используется кнопка в правом верхнем углу теста **Покажи все вопросы**.

Ввод задания на викторину

Запустите программу HotPotatoes, выберите программу **JQuiz** на экране или в меню **Potatoes**– откроется рабочее окно программы.

Выбор типа

Введите заголовок упражнения (поле **Имя**).

Выберите номер вопроса в поле **В**, введите текст вопроса, выберите тип вопроса в списке справа и введите ответы (поля **Ответы**).

Отметьте правильные ответы в поле **Верно**.

Замечание. В вопросе с коротким открытым ответом может быть как один, так и несколько вариантов правильных ответов.

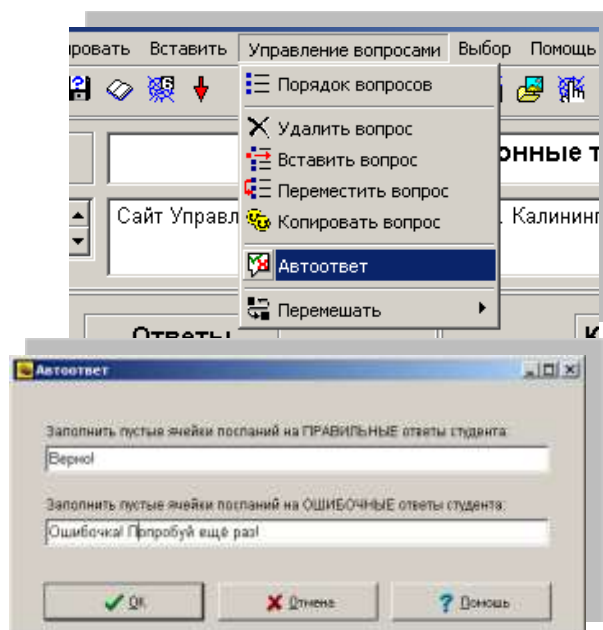
Введите комментарии к отдельным ответам в поле **Комментарии**.

Для перехода к следующему вопросу используются стрелки рядом с полем номера вопроса.

Редактирование заданий/тестов

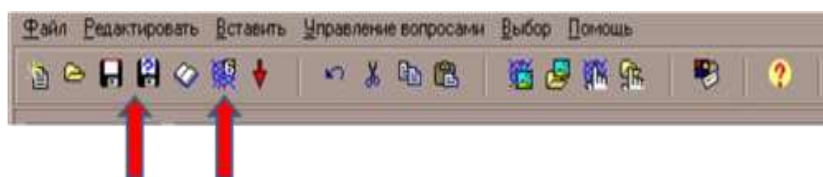
Редактирование задания выполняется с помощью меню **Управление вопросами**. Этот пункт меню включает несколько подпунктов, которые позволяют изменить порядок вопросов в задании, удалить вопрос, вставить новый, переместить или скопировать вопрос, а также перемешать вопросы в задании/тесте.

Подпункт меню **Автоответ** позволяет ввести комментарий на правильные ответы и ошибочные ответы учащегося.



Внести изменения в конкретный вопрос можно, выбрав в поле номера вопроса нужный вопрос и изменив необходимые поля.

После ввода вопросов не забудьте сохранить проект и создайте веб-страницу.

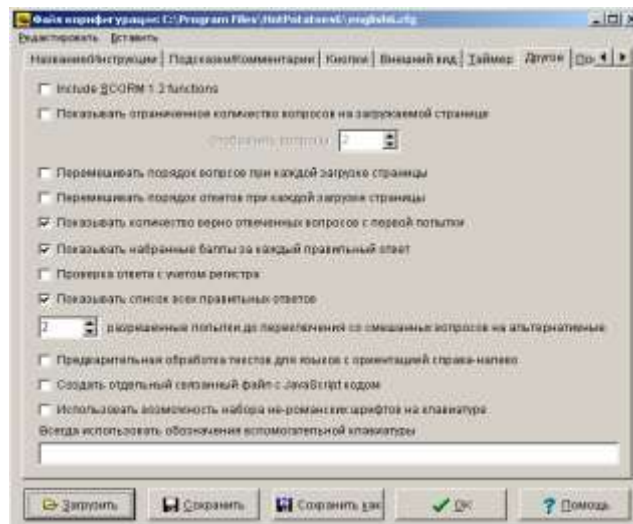


Теперь можно создать новое задание или тест, используя меню **Файл-Новый**, или закрыть эту программу и перейти к другой программе из набора HotPotatoes, или завершить работу с HotPotatoes.

Настройка конфигурации

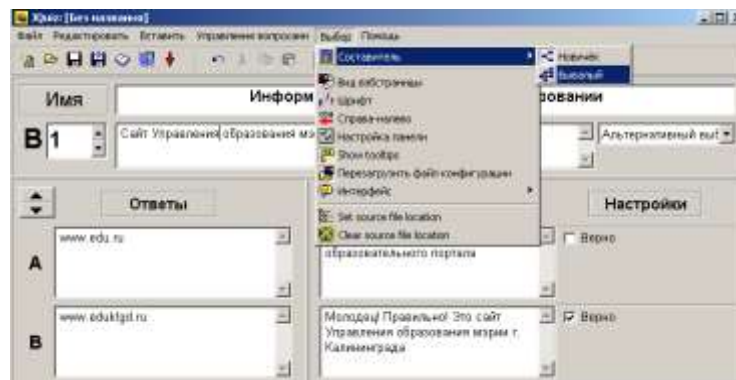
В программе Quiz предоставляются дополнительные возможности для настройки конфигурации заданий. Выбрав вкладку **Другое** в диалоге **Файл конфигурации** можно:

- ✓ менять порядок вопросов при каждой загрузке задания;
- ✓ показывать количество вопросов, на которых был дан правильный ответ с первой попытки;
- ✓ показывать общую оценку после каждого правильного ответа;
- ✓ учитывать регистр букв (строчные, прописные) при проверке ответа.
- ✓ показывать список других возможных верных ответов;
- ✓ установить количество попыток ввода короткого ответа, после которых задание трансформируется в вопрос с множественным выбором ответа (настройка используется в вопросах смешанного типа);
- ✓ использовать экранную клавиатуру (для коротких ответов);
- ✓ выводить на экран заданное количество вопросов.



Кроме того, продвинутые пользователи могут использовать возможности оценки степени правильности ответа; учета частично правильных ответов; включения режима тестирования. Для использования дополнительных возможностей в настройке заданий необходимо выбрать режим продвинутого пользователя

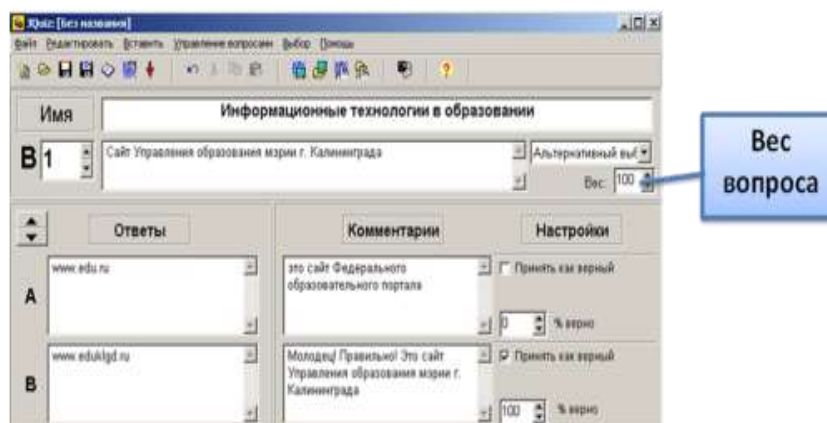
Бывалый в пункте меню **Выбор – Составитель**.



Вес вопроса

При установке режима для продвинутых пользователей (бывалый) Вы можете задавать относительную важность вопросов. Это делается с помощью элемента **Вес** в диалоге. Вы устанавливаете вес вопроса в диапазоне от 0 до 100. Чем выше вес, тем важнее вопрос. При этом очки, полученные при верном ответе на вопрос, увеличиваются во столько раз, каково значение веса.

Вы можете создать два-три вопроса в начале теста, которые имеют вес 0 и не учитываются в общем счете теста.

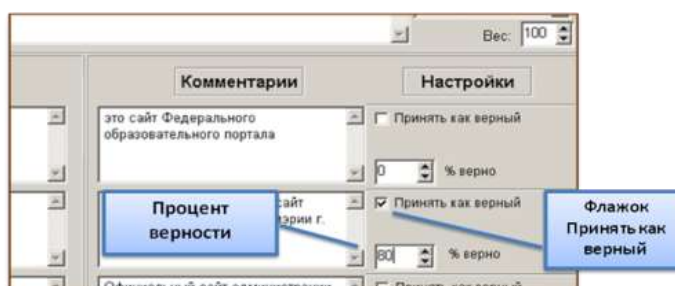


Процентная верность ответов

При такой возможности можно установить процент верности ответа. Некоторые вопросы могут иметь ответ со 100% верностью – это абсолютно правильный ответ, а также несколько ответов – частично правильных. Так можно поощрить учащихся, давших частично правильные ответы.

Установка Принять как верный

Этот флажок позволяет принять ответ как верный с оценкой в процентах верности. Даже если ответ не совсем верный, он принимается, никакие попытки повторного ответа на такой вопрос не допускаются.



Раздел II. Web-технологии для создания электронных учебных пособий (ЭУП)

Модель технологии обучения

Тема 4	Разработка ЭУП с помощью программы Dreamweaver: разработка структуры ЭУП и оформление стартовой страницы
---------------	---

1.1. Технология обучения на лекции

<i>Учебное время: 2 часа.</i>	
<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	1. Разработка структуры ЭУП. 2. Оформление главной страницы
<i>Цель учебного занятия:</i>	Сформировать умения разработки структуры ЭУП с помощью web-технологий и разработки стартовой страницы ЭУП.
<i>Педагогические задачи:</i> <ul style="list-style-type: none"> Организовать работу по разработке структуры ЭУП; Дать рекомендации по оформлению главной страницы ЭУП. 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> Разрабатывать структуру ЭУП. Оформлять главную страницу ЭУП.
<i>Методы обучения</i>	Мозговой штурм
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов

1-й этап. Введение в учебное занятие (5 мин)	1.4. Сообщает, тему, цель, планируемые результаты учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (68 мин)	<p>2.1. Проводит мозговой штурм, с помощью которых актуализируются знания о структурных элементах ЭУП (<i>приложение 2</i>).</p> <p>2.2. Приводит примеры структур ЭУП по различным учебным предметам (<i>приложение 3</i>) и предлагает разработать структуру создаваемого студентами ЭУП согласно полученным темам (<i>приложение 4</i>).</p> <p>2.3. Приводит примеры оформления стартовых страниц готовых ЭУП (<i>приложение 5</i>) и предлагает разработать их в среде Dreamweaver.</p>	<p>2.1. Отвечают на вопросы.</p> <p>2.2. Знакомятся с основами структурирования учебного материала в ЭУП и составляют структуру своего учебного пособия согласно выбранной тематике.</p> <p>2.3. Оформляют в среде Dreamweaver стартовую страницу ЭУП.</p>
3-й этап. Заключитель но-результатирующий (10 мин)	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> • Повторить из курса «Компьютерная графика и web-дизайн» основы работы в программе Dreamweaver; • Доработать главную страницу ЭУП; • Подготовить лекционный материал для своего ЭУП. 	Задают уточняющие вопросы.

Приложение 1

Тема: Разработка ЭУП с помощью программы Dreamweaver: разработка структуры ЭУП и оформление стартовой страницы

Цель занятия: сформировать умения разработки структуры ЭУП с помощью web-технологий и разработки стартовой страницы ЭУП

План занятия:

1. Разработка структуры ЭУП.
2. Оформление главной страницы

Приложение 2

Любое ЭУП должно включать в себя следующие обязательные компоненты:

- Средства изучения теоретических основ дисциплины;
- Средства поддержки практических занятий;
- Средства контроля знаний;
- Средства взаимодействия между учителем и учащимися;
- Методические рекомендации по изучению дисциплины;
- Средства управления процессом изучения дисциплины.

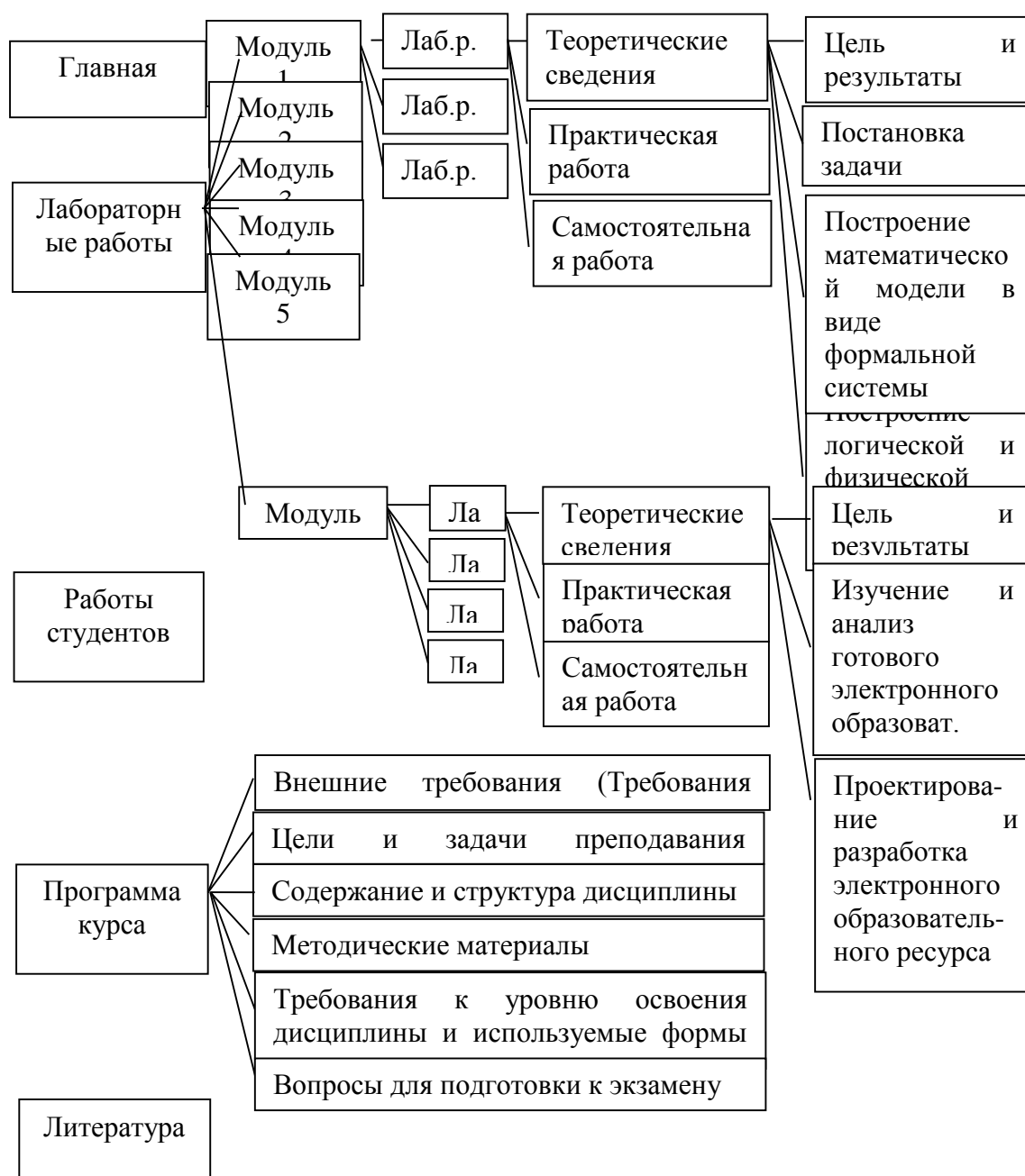
Приложение 3

Пример1



Пример 2

Структура Web-ресурса

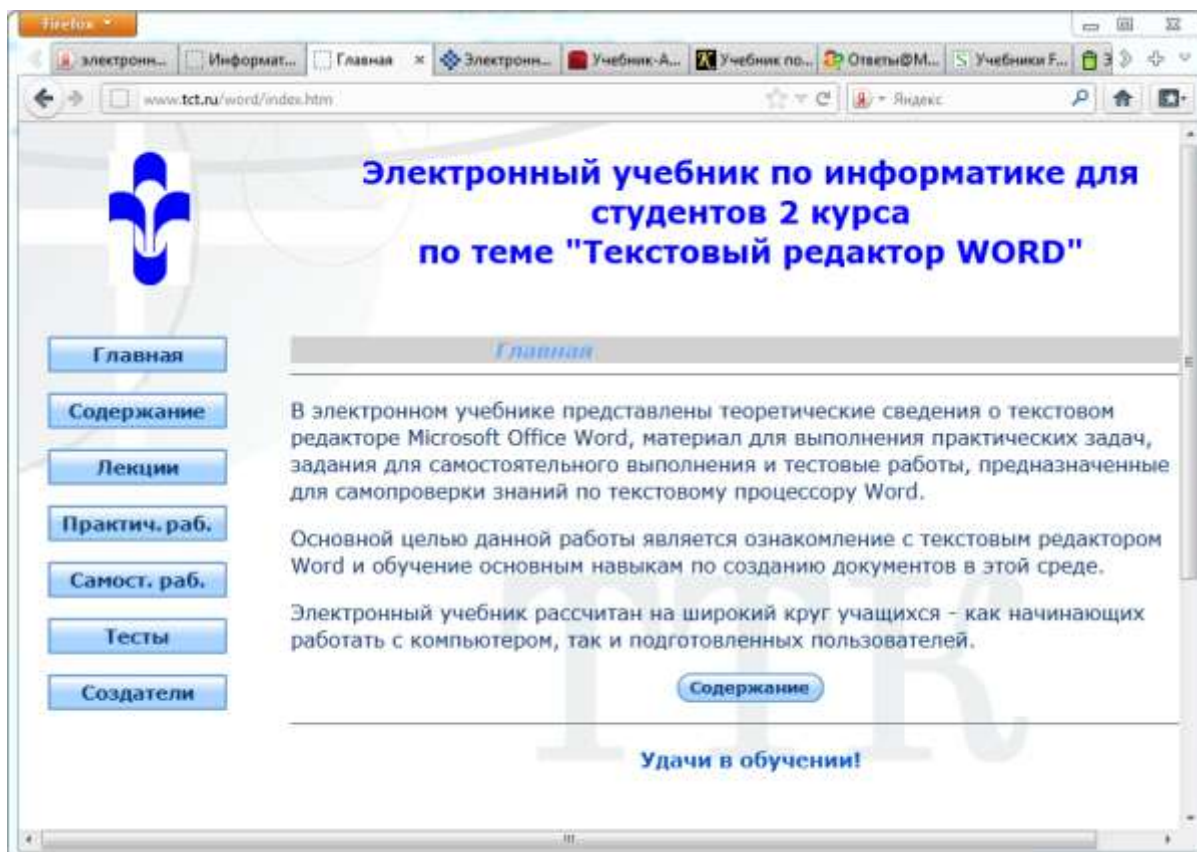



Приложение 4



№	Тема ЭУП
1.	Системы счисления
2.	Язык программирования Паскаль
3.	Язык программирования Delphi
4.	Основы языка HTML
5.	Основы работы в программе MS Word

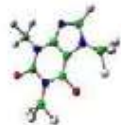
6.	Основы работы в программе MS Excel
7.	Основы работы в программе MS Power Point
8.	Разработка баз данных в MS Access
9.	Компьютерная графика
10.	Компьютерные сети

Приложение 5




SSORO

Современные средства оценивания результатов обучения



Современные средства оценивания результатов обучения



Главная


Курс лекции


Семинарские занятия


Словарь терминов


Тестирование


Темы лекций


 **Лекция №1. Тема: "Качество образования"**


1.1.  Понятие о качестве образования. Оценка как элемент управления качеством


1.2.  Традиционные и новые средства оценки результатов обучения


 **Лекция №2. Тема: "История развития системы тестирования в России и за рубежом"**


2.1.  История развития системы тестирования за рубежом


2.2.  История развития системы тестирования в России


 **Лекция №3. Тема: "Психолого-педагогические аспекты тестирования"**

3.1.  Таксономия образовательных целей и результаты образования

3.2.  Тесты школьных достижений (педагогические тесты), их отличия от психологических и преимущества перед иными формами аттестации учащихся

3.3.  Подходы к структурированию учебных достижений

 **Лекция №4. Тема: "Педагогические тесты. Термины и определения"**

4.1.  Понятийный аппарат тестологии

Введение в статистический анализ на основе системы Statistica - Windows Internet Explorer - [Автоматическая работа]

Оглавление курсКурсУчебник.html

Введение в статистический анализ на основе системы Statistica

Учебное пособие Лабораторные работы

1. Введение

1.1. Принимая создание файлов данных

1.2. Однофакторные статистики

1.3. Вычисление корреляций

1.4. Построение простейших статистических графиков

2. Первичный анализ в системе Statistica - модуль Basic Statistics/Tables (Основные статистики/Таблицы)

2.1. Вероятностный калькулятор

2.2. Генерация случайных чисел в STATISTICA

2.3. Анализ данных ЦЕНА РЕКЛАМЫ

3. Регрессионный анализ в системе Statistica - модуль Multiple Regression (Множественная регрессия)

3.1. Введение

3.2. Пример курса акций ПЕРУТОКСНЕТО и КРАСНОЯРСКОБАНКО

3.3. Пример со стоимостью эксплуатации самолетов


3.4. Пример цены жилого дома в США

3.5. Примеры использования средства КРИСЬ для анализа

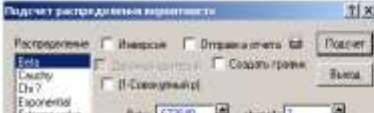
2. Первичный анализ в системе Statistica - модуль Basic Statistics/Tables (Основные статистики/Таблицы)

2.1. Вероятностный калькулятор

Запустите модуль Basic Statistics/Tables (Основные статистики/таблицы) из Переключателя модулей. Выделите в стартовой панели модуля Basic Statistics/Tables (Основные статистики/таблицы) строку Probability calculator (Вероятностный калькулятор).



Нажмите кнопку ОК. Перед вами откроется окно Probability Distribution Calculator (Калькулятор вероятностных распределений) (рис. 2.1).



Проектирование баз данных

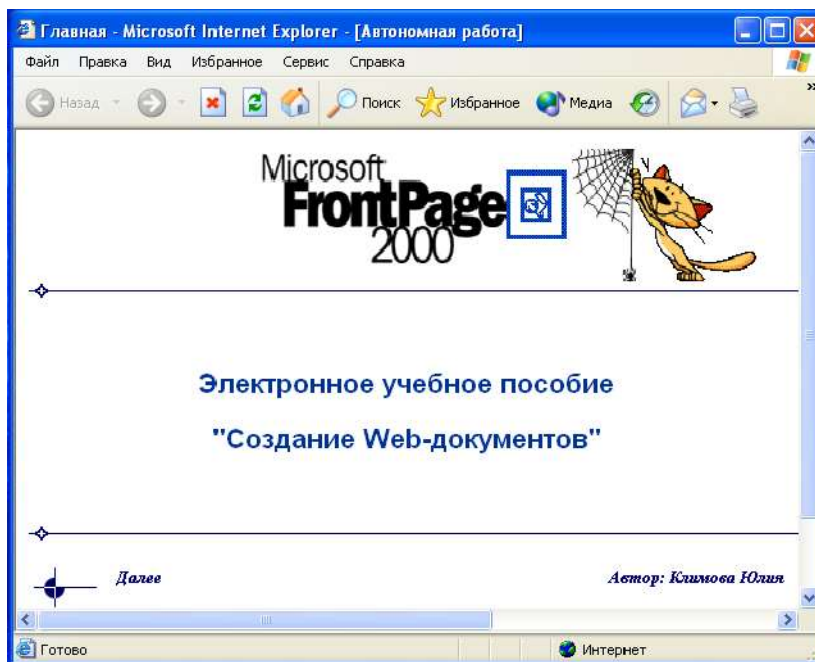
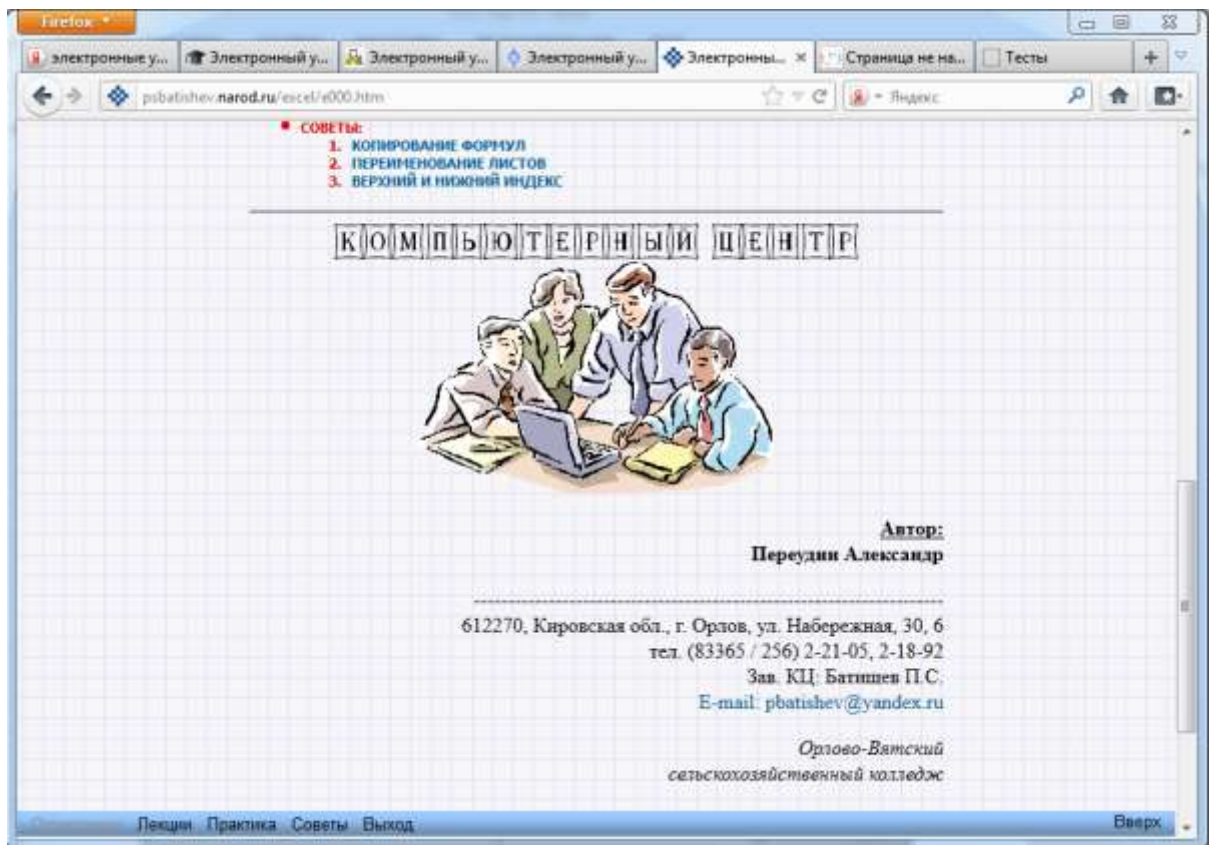
Электронный учебник

Проектирование баз данных

Данный учебник предназначен для студентов изучающих дисциплину "Проектирование баз данных". Это очень удобное и эффективное средство обучения, поскольку содержит весь нужный материал для изучения данной дисциплины. Здесь приведены весь курс лекций, лабораторные и практические работы.

Желаем Вам успеха!

... | Оглавление | Лекции | Лабораторные работы | Практические работы | Тест | Разработки | ...



Сборник задач по физике - Microsoft Internet Explorer

Файл Правка Вид Избранное Сервис Справка

Готово

Адрес: G:\Депон\INDEX.html

Перезагрузить Отменить

Раздел 1.
Механика

Кинематика

- Теоретическая часть
- Решение задач

Динамика



- Теоретическая часть
- Решение задач

Раздел 2.
Молекулярная физика



Первоначальные сведения о тепловых явлениях

Основы молекулярно-кинетической теории идеального газа

Электронное учебное пособие по предмету "Физика"



Сборник задач по предмету "Физика" (профильного (противопоставленного) содержания) за курс среднего(полного) общего образования для обучающихся специальности 190604 "Техник по ремонту и техническому обслуживанию автомобильного транспорта".



Преподаватель
Колабина Г.А.

Готово

ПУСК

Диск D - Microsoft IE...

Сборник задач по ф...

Мой компьютер

5 июня 2010 г.

Тема 5	Разработка ЭУП с помощью программы Dreamweaver: разработка страниц ЭУП с лекционным материалом и установление связи с главной страницей посредством гиперссылок
---------------	--

1.1. Технология обучения на лабораторном занятии

<i>Учебное время: 2 часа.</i>	
<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Разработка лекционных материалов ЭУП в виде web-страничек в среде Dreamweaver. 2. Установление взаимосвязей между главной страницей и лекционными материалами в среде Dreamweaver посредством гиперссылок.
<i>Цель учебного занятия:</i>	Сформировать умения разработки лекционных материалов ЭУП и установление связи с главной страницей посредством гиперссылок.
<i>Педагогические задачи:</i> Организовать работу студентов по: <ul style="list-style-type: none"> • разработке лекционных материалов ЭУП в виде web-страничек в среде Dreamweaver. • установлению взаимосвязей между главной страницей и лекционными материалами в среде Dreamweaver посредством гиперссылок. 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • Разрабатывать лекционный материал ЭУП в виде web-страничек в среде Dreamweaver. • Устанавливать взаимосвязи между главной страницей и лекционными материалами в среде Dreamweaver посредством гиперссылок.
<i>Методы обучения</i>	Блиц-опрос, схемы и графы
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.5. Сообщает, тему, цель, планируемые результаты учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (69 мин)	<p>2.1. Проводит блиц-опрос по особенностям программы Dreamweaver.</p> <p>2.2. Напоминает об особенностях восприятия информации человеком (<i>приложение 2</i>).</p> <p>2.3. Предлагает оформить в среде Dreamweaver лекционный материал.</p> <p>2.4. Напоминает способы вставки гиперссылок в среде Dreamweaver (<i>приложение 3</i>) и предлагает связать стартовую страницу ЭУП со страничками, содержащими лекционный материал.</p>	<p>2.1. Отвечают на вопросы.</p> <p>2.2. Оформляют в среде Dreamweaver лекционный материал ЭУП</p> <p>2.3. Устанавливают взаимосвязи между главной страницей и лекционными материалами в среде Dreamweaver посредством гиперссылок</p>
3-й этап. Заключительный о- результатирующий (10 мин)	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> Повторить из курса «Компьютерная графика и web-дизайн» основы работы в программе Dreamweaver; Доработать страницы ЭУП; Подготовить материалы по практическим работам для своего ЭУП. 	Задают уточняющие вопросы.

Приложение 1

Тема: Разработка ЭУП с помощью программы Dreamweaver: разработка страниц ЭУП с лекционным материалом и установление связи с главной страницей посредством гиперссылок

Цель занятия: сформировать умения разработки лекционных материалов ЭУП и установление связи с главной страницей посредством гиперссылок

План занятия:

1. Разработка лекционных материалов ЭУП в виде web-страничек в среде Dreamweaver.
2. Установление взаимосвязей между главной страницей и лекционными материалами в среде Dreamweaver посредством гиперссылок.

Приложение 2





Установлены следующие допустимые комбинации цвета символа с цветом фона (в порядке убывания четкости восприятия)	
	Синий на белом
	Черный на желтом
	Зеленый на белом
	Черный на белом
	Белый на синем
	Зеленый на красном
	Красный на желтом

Красный на белом
Оранжевый на черном
Черный на пурпурном
Оранжевый на белом
Красный на зеленом

Приложение 3

Создание гиперссылок на web-страницы

Для создания гиперссылки, необходимо выделить текст, который в дальнейшем будет гиперссылкой. Далее в редакторе свойств в поле Link (рис. 1) ввести адрес сайта или файла, и нажать клавишу <Enter>.



Рис.1. Поле ввода интернет-адреса в редакторе свойств

Например, мы хотим создать гиперссылку на сайт фирмы Macromedia — создателя Dreamweaver. Выделяем это слово и в поле Link вводим: <http://www.macromedia.com>

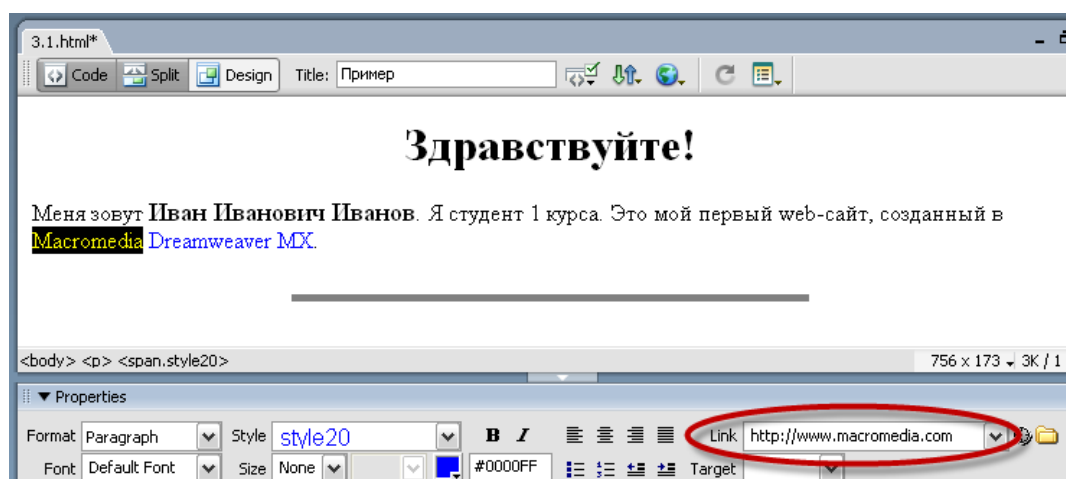


Рис. 2. Пример вставки гиперссылки на сайт

Для создания гиперссылки из выделенного текста также можно воспользоваться пунктом **Make Link** меню **Modify** или контекстного меню или нажать комбинацию клавиш Ctrl+L. После этого на экране появится диалоговое окно **Select File**, показанное на рис. 3. Нужный адрес вводится в поле ввода URL этого окна. Не забудьте нажать кнопку **ОК**, чтобы Dreamweaver создал эту гиперссылку.

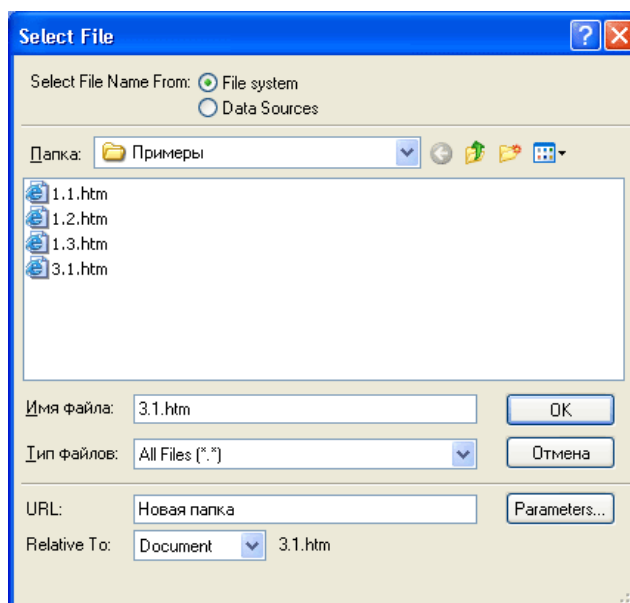


Рис. 1. Диалоговое окно Select File

Тема 6	Разработка ЭУП с помощью программы Dreamweaver: подготовка и внедрение страниц с практическими заданиями к главной странице ЭУП
---------------	--

1.1. Технология обучения на лабораторном занятии

<i>Учебное время: 2 часа.</i>	
<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Разработка материалов с практическими заданиями ЭУП в виде web-страничек в среде Dreamweaver. 2. Установление взаимосвязей между главной страницей и практическими материалами в среде Dreamweaver посредством гиперссылок.
<i>Цель учебного занятия:</i>	Сформировать умения разработки материалов с практическими заданиями ЭУП и установление связи с главной страницей посредством гиперссылок.
<i>Педагогические задачи:</i> Организовать работу студентов по: <ul style="list-style-type: none"> • разработке материалов с практическими заданиями ЭУП в виде web-страничек в среде Dreamweaver. • установлению взаимосвязей между 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • Разрабатывать материалы с практическими заданиями ЭУП в виде web-страничек в среде Dreamweaver. • Устанавливать взаимосвязи между главной страницей и практическими материалами в среде Dreamweaver посредством гиперссылок.

главной страницей и практическими материалами в среде Dreamweaver посредством гиперссылок.	
<i>Методы обучения</i>	Частично-поисковая работа за ПК
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.6. Сообщает, тему, цель, планируемые результаты учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (70 мин)	<p>2.1. С помощью проектора приводит примеры оформления web-страничек с практическими заданиями по различным темам (<i>приложение 2</i>).</p> <p>2.2. Предлагает оформить в среде Dreamweaver практический материал.</p> <p>2.3. Предлагает связать стартовую страницу ЭУП со страничками, содержащими практический материал.</p>	<p>2.1. Отвечают на вопросы.</p> <p>2.2. Оформляют в среде Dreamweaver практический материал ЭУП</p> <p>2.3. Устанавливают взаимосвязи между главной страницей и практическими материалами в среде Dreamweaver посредством гиперссылок</p>
3-й этап. Заключительный о- результатирующий (10 мин)	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> Доработать страницы ЭУП. Подготовить материалы с тестовыми заданиями в среде MyTest или Hot Potatoes для своего ЭУП. 	Задают уточняющие вопросы.

Приложение 1

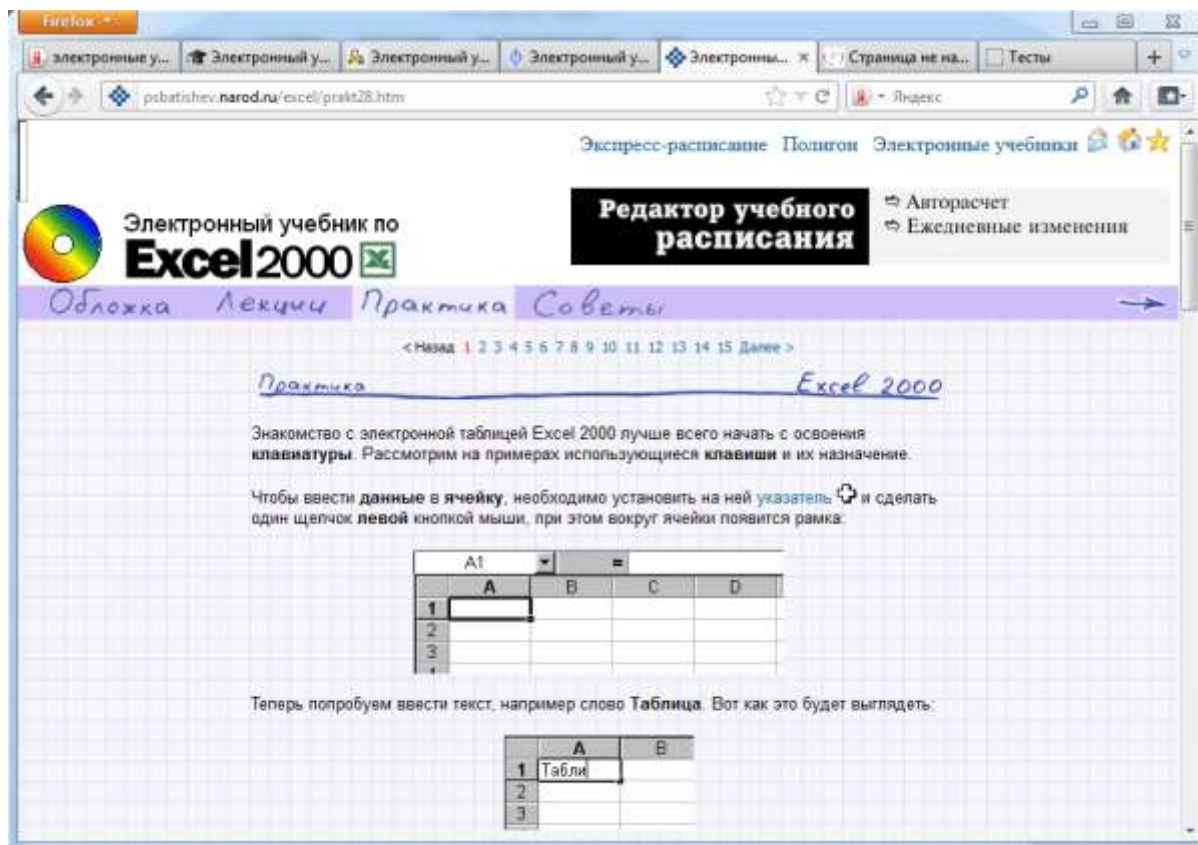
Тема: Разработка ЭУП с помощью программы Dreamweaver: подготовка и внедрение страниц с практическими заданиями к главной странице ЭУП

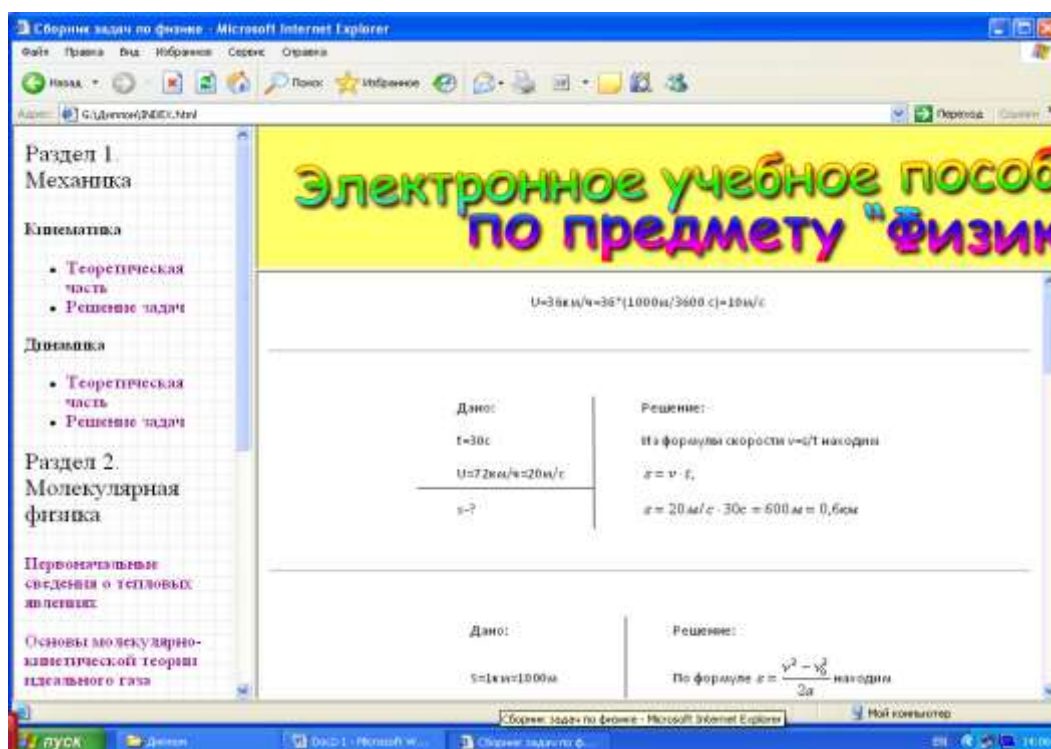
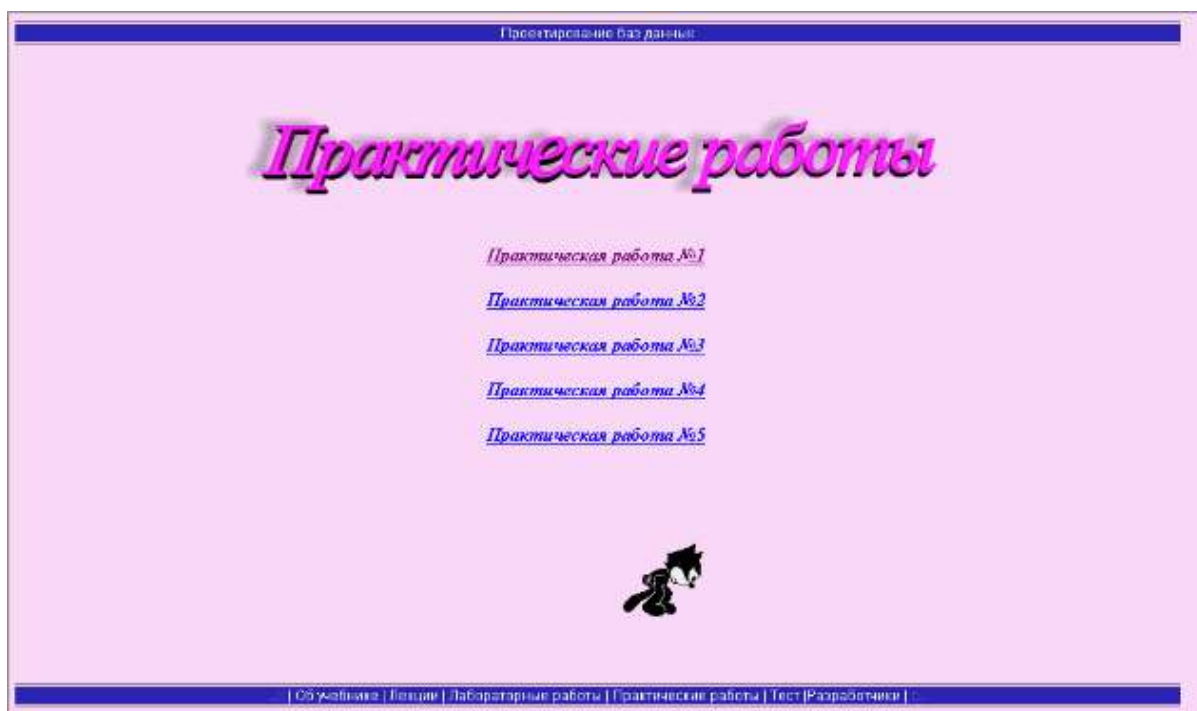
Цель занятия: сформировать умения разработки материалов с практическими заданиями ЭУП и установление связи с главной страницей посредством гиперссылок.

План занятия:

1. Разработка материалов с практическими заданиями ЭУП в виде web-страничек в среде Dreamweaver.
2. Установление взаимосвязей между главной страницей и практическими материалами в среде Dreamweaver посредством гиперссылок

Приложение 2





Тема 7	Разработка ЭУП с помощью программы Dreamweaver: подготовка и внедрение системы диагностики и контроля к главной странице ЭУП
--------	--

1.1. Технология обучения на лабораторном занятии

Учебное время: 2 часа.

<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Разработка тестовых заданий ЭУП в оболочках MyTest или Hot Potatoes. 2. Установление взаимосвязей между главной страницей и тестовыми заданиями в среде Dreamweaver посредством гиперссылок.
<i>Цель учебного занятия:</i>	Сформировать умения внедрения тестовых материалов в ЭУП посредством гиперссылок.
<i>Педагогические задачи:</i> Организовать работу студентов по: <ul style="list-style-type: none"> • разработке тестовых заданий ЭУП в оболочках MyTest или Hot Potatoes. • установлению взаимосвязей между главной страницей и тестовыми заданиями в среде Dreamweaver посредством гиперссылок. 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • Разрабатывать тестовых заданий ЭУП в оболочках MyTest или Hot Potatoes. • Устанавливать взаимосвязи между главной страницей и тестовыми заданиями в среде Dreamweaver посредством гиперссылок.
<i>Методы обучения</i>	Частично-поисковая работа за ПК
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.7. Сообщает, тему, цель учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть	2.1. Актуализирует знания студентов о программах, с помощью которых можно создавать тестовые задания, а также их видах (<i>приложение 2</i>).	2.1. Отвечают на вопросы.

(71 ин)	<p>2.2. Показывает пример вставки тестового задания на главную страницу ЭУП (приложение 3).</p> <p>2.3. Предлагает связать стартовую страницу ЭУП со тестовыми заданиями, разработанными в качестве домашнего задания в одной из тестовых оболочек.</p>	<p>2.2. Просматривают, задают вопросы, уточняют.</p> <p>2.3. Устанавливают взаимосвязи между главной страницей и тестовыми заданиями в среде Dreamweaver посредством гиперссылок</p>
<p>3-й этап.</p> <p>Заключительный о- результатирующий (10 мин)</p>	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> • Доработать страницы ЭУП. • Подготовить страницу с личными данными и литературой по теме своего ЭУП, а также звуковые аудиофайлы, которые необходимо встроить в ЭУП. 	<p>Задают уточняющие вопросы.</p>

Приложение 1

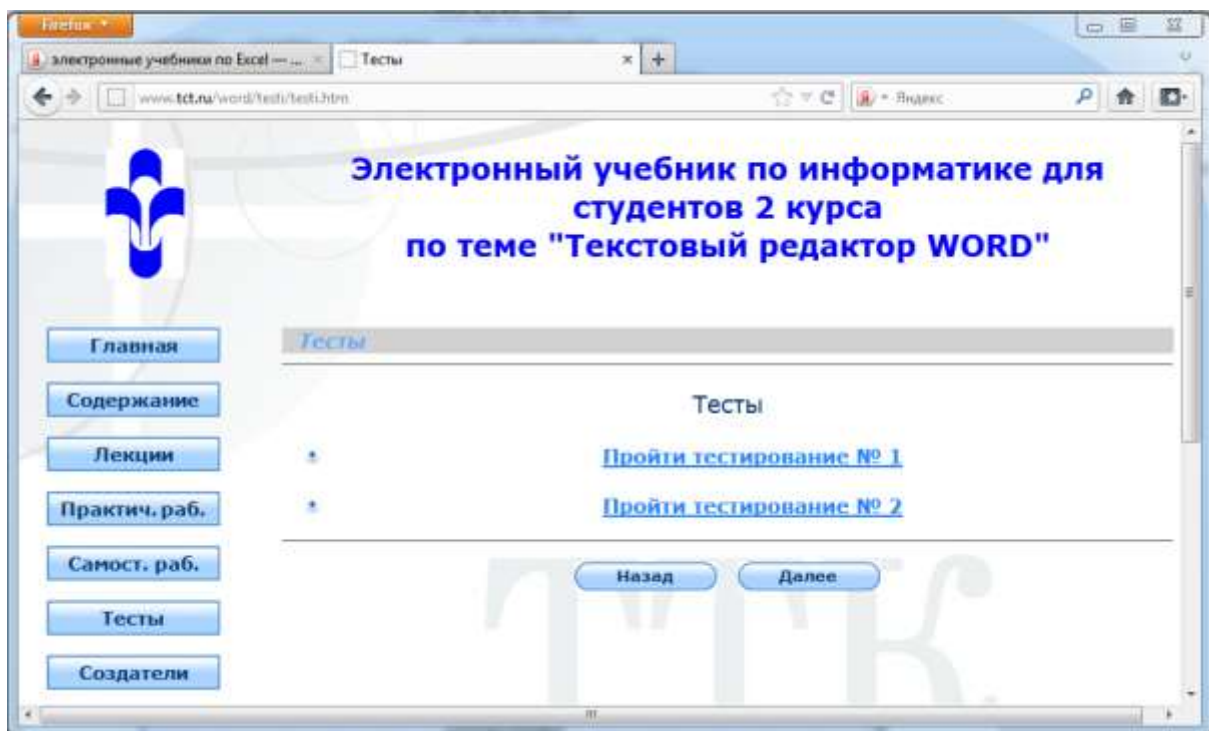
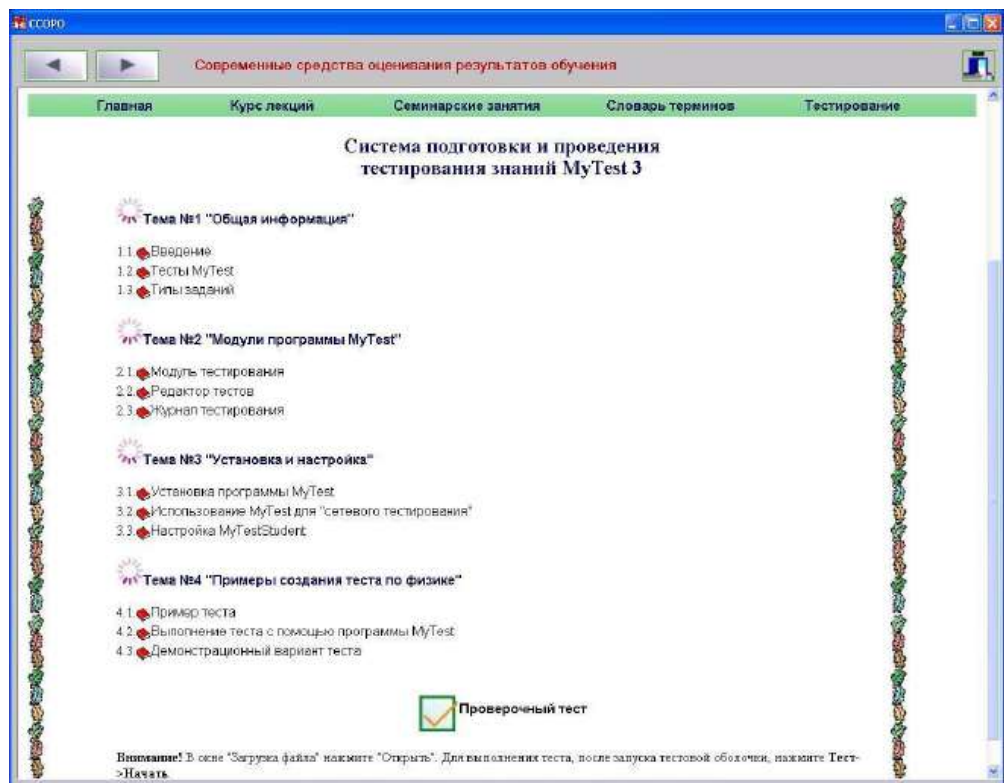
Тема: Разработка ЭУП с помощью программы Dreamweaver: подготовка и внедрение системы диагностики и контроля к главной странице ЭУП

Цель занятия: сформировать умения внедрения тестовых материалов в ЭУП посредством гиперссылок.

План занятия:

1. Разработка тестовых заданий ЭУП в оболочках MyTest или Hot Potatoes.
2. Установление взаимосвязей между главной страницей и тестовыми заданиями в среде Dreamweaver посредством гиперссылок.





Тема 8	Разработка ЭУП с помощью программы Dreamweaver: внедрение на главную страницу ЭУП данных об авторе и списка литературы, звукового сопровождения
---------------	--

1.1. Технология обучения на лабораторном занятии

<i>Учебное время: 2 часа.</i>	
<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Разработка web-странички с личными данными в среде Dreamweaver. 2. Разработка web-странички со списком литературы в среде Dreamweaver. 3. Встраивание звуковых файлов в ЭУП средствами программы Dreamweaver. 4. Установление взаимосвязей между главной страницей и страницами, содержащими данные об авторе и литературе в среде Dreamweaver посредством гиперссылок.
<i>Цель учебного занятия:</i>	Сформировать умения по оформлению личных данных и списка литературы в ЭУП посредством гиперссылок, а также встраиванию звуковых данных.
<i>Педагогические задачи:</i> Организовать работу студентов по: <ul style="list-style-type: none"> • Разработке web-странички с личными данными в среде Dreamweaver. • Разработке web-странички со списком литературы в среде Dreamweaver. • Встраиванию звуковых файлов в ЭУП средствами программы Dreamweaver. • Установлению взаимосвязей между главной страницей и страницами, содержащими данные об авторе и 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • Разрабатывать web-странички с личными данными в среде Dreamweaver. • Разрабатывать web-странички со списком литературы в среде Dreamweaver. • Встраивать звуковых файлов в ЭУП средствами программы Dreamweaver. • Устанавливать взаимосвязи между главной страницей и страницами, содержащими данные об авторе и литературе в среде Dreamweaver посредством гиперссылок

литературе в среде Dreamweaver посредством гиперссылок	
<i>Методы обучения</i>	Частично-поисковая работа за ПК
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.8. Сообщает, тему, цель учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (65 мин)	<p>2.1. Показывает пример оформления странички с личными данными ЭУП (<i>приложение 2</i>) и предлагает оформить web-странички с личными данными и списком литературы.</p> <p>2.2. Предлагает связать стартовую страницу ЭУП со страницами с личными данными и списком литературы в среде Dreamweaver с помощью гиперссылок.</p> <p>2.3. Знакомит студентов со способом вставки аудиофайла на web-страницу (<i>приложение 3</i>).</p> <p>2.4. Предлагает вставить необходимые звуковые сопровождения на разрабатываемое ЭУП</p>	<p>2.1. Просматривают, уточняют, работают в среде Dreamweaver.</p> <p>2.2. Устанавливают взаимосвязи между стартовой страницей ЭУП со страницами с личными данными и списком литературы в среде Dreamweaver с помощью гиперссылок.</p> <p>2.3. Слушают, уточняют.</p> <p>2.4. Вставляют подготовленные звуковые файлы в ЭУП.</p>
3-й этап.	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> Доработать страницы ЭУП. 	Задают уточняющие вопросы.

Заключитель- но-результати- рующий (10 мин)	<ul style="list-style-type: none"> Подготовить страницу с личными данными и литературой по теме своего ЭУП. 	
--	--	--

Приложение 1

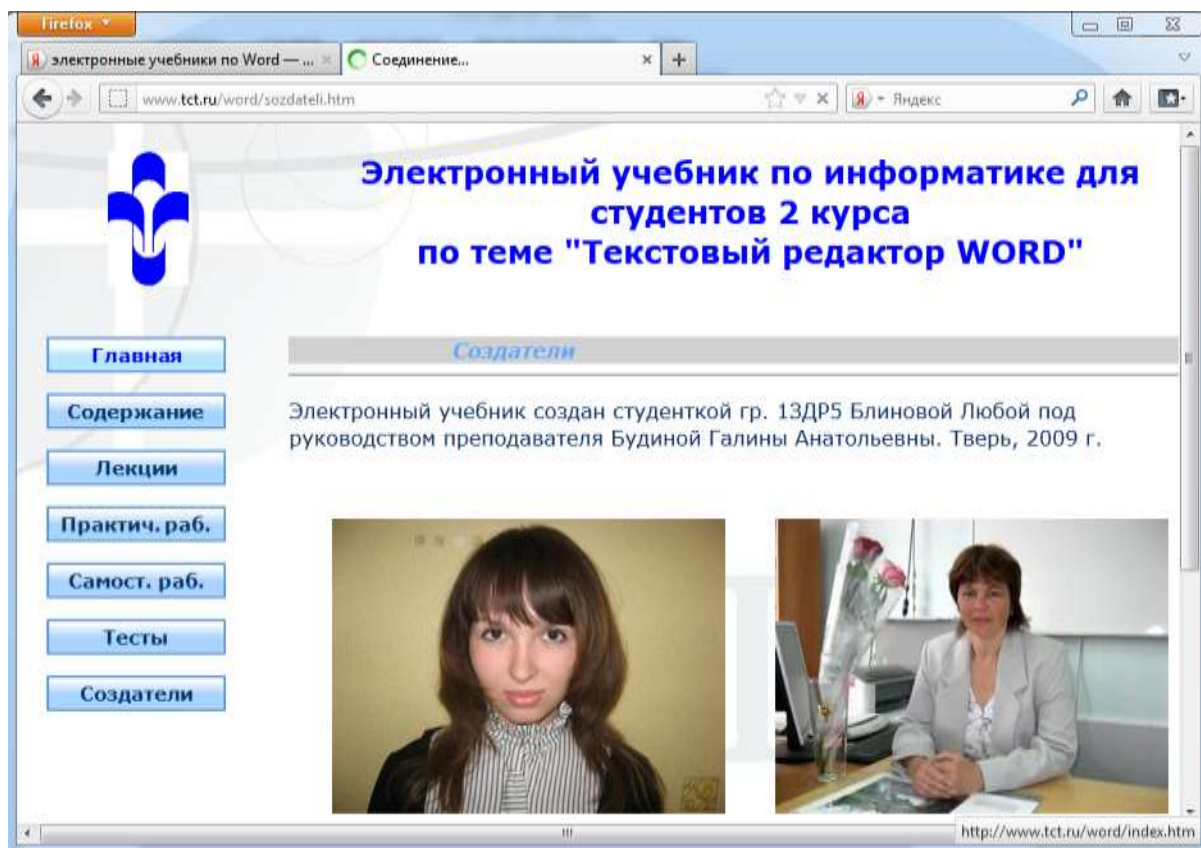
Тема: Разработка ЭУП с помощью программы Dreamweaver: внедрение на главную страницу ЭУП данных об авторе и списка литературы, звукового сопровождения

Цель занятия: сформировать умения по оформлению личных данных и списка литературы в ЭУП посредством гиперссылок, а также встраиванию звуковых данных.

План занятия:


1. Разработка web-странички с личными данными в среде Dreamweaver.
2. Разработка web-странички со списком литературы в среде Dreamweaver.
3. Встраивание звуковых файлов в ЭУП средствами программы Dreamweaver.
4. Установление взаимосвязей между главной страницей и страницами, содержащими данные об авторе и литературе в среде Dreamweaver посредством гиперссылок.


Приложение 2



Внедрение звукового файла

Внедрение звука означает встраивание его непосредственно в страницу, однако звук воспроизводится только в том случае, если у пользователя сайта установлен подключаемый модуль, соответствующий звуковому файлу. Внедрять файлы можно, если требуется использовать звук в качестве фоновой музыки или управлять громкостью, отображением проигрывателя на странице или начальной и конечной точками звукового файла.

 При встраивании звуковых файлов в веб-страницы следует внимательно оценить целесообразность их использования на веб-сайте и то, как его посетители используют данные мультимедийные ресурсы. Следует всегда предоставлять возможность включить или отключить воспроизведение звука в случае, если посетители не желают прослушивать аудиосодержимое.

1. В представлении «Дизайн» установите курсор вставки в том месте, куда необходимо внедрить файл, и выполните одно из следующих действий.
 - В категории «Общие» на панели «Вставка» нажмите кнопку «Мультимедиа» и во всплывающем меню выберите значок «Подключаемый модуль» .
 - Выберите команды «Вставка» > «Мультимедиа» > «Подключаемый модуль».
2. Перейдите к аудиофайлу и нажмите кнопку «ОК».
3. Введите значения ширины и высоты в соответствующих полях в инспекторе свойств или путем изменения размера местозаполнителя подключаемого модуля в области «Окно документа».

Эти значения определяют размер отображаемых элементов управления звуком в браузере.

Раздел III. Технологии создания электронных учебных материалов на основе технологии программирования (Delphi).

Тема 9	Составляющие элементы ЭУП, разрабатываемого в среде Delphi
---------------	---

1.1. Технология обучения на лабораторном занятии

<i>Учебное время: 2 часа.</i>	
<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Общее описание обучающей системы, разрабатываемой в среде Delphi. 2. Составление технического описания задания. 3. Компоненты Delphi, используемые при создании ЭУП.
<i>Цель учебного занятия:</i>	Сформировать умение проектировать работу по созданию ЭУП в среде Delphi.
<i>Педагогические задачи:</i> <ul style="list-style-type: none"> • Дать общее описание обучающей системы, разрабатываемой в среде Delphi. • Организация работы студентов по составлению технического описания задания. • Провести обзор компонент Delphi, используемые при создании ЭУП. 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • Представлять общий вид обучающей системы, разрабатываемой в среде Delphi. • составлять техническое описание задания. • Работать с компонентами Delphi, используемые при создании ЭУП.
<i>Методы обучения</i>	Слайдовая презентация, мозговой штурм
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.9. Сообщает, тему, цель учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (65 мин)	<p>2.1. Показывает с помощью слайдов презентации пример оформления ЭУП в среде Delphi (<i>приложение 2</i>).</p> <p>2.2. Раздает темы ЭУП (<i>приложение 3</i>), приводит пример оформления технического задания (<i>приложение 4</i>) и организует работу по его оформлению индивидуально каждым студентами.</p> <p>2.3. Проводит мозговой штурм, посредством которого определяются компоненты, используемые для разработки ЭУП. (<i>приложение 5</i>).</p>	<p>2.1. Просматривают, уточняют, анализируют.</p> <p>2.2. Разрабатывают техническое задание.</p> <p>2.3. Предлагают названия компонентов и их свойства.</p>
3-й этап. Заключительно-результативный (10 мин)	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> Повторить компоненты Delphi. Подготовить материал согласно полученной теме ЭУП. 	Задают уточняющие вопросы.

Приложение 1

Тема: Составляющие элементы ЭУП, разрабатываемого в среде Delphi

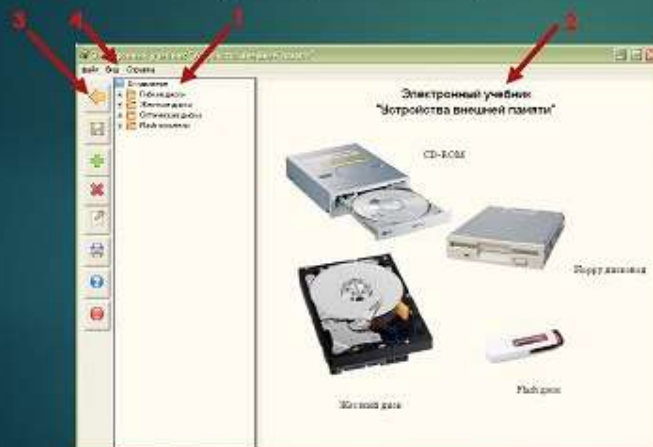
Цель занятия: Сформировать умение проектировать работу по созданию ЭУП в среде Delphi.

План занятия:

- Общее описание обучающей системы, разрабатываемой в среде Delphi.
- Составление технического описания задания.
- Компоненты Delphi, используемые при создании ЭУП.

Создание ЭУП с помощью языка программирования Delphi

Пример ЭУП, созданного средствами языка программирования Delphi

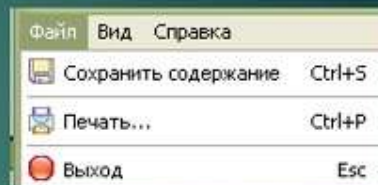


Главное окно программы состоит из четырех основных частей:




1. Область оглавления;
2. Область вывода обучения;
3. Панель инструментов;
4. Главное меню.

Главное меню

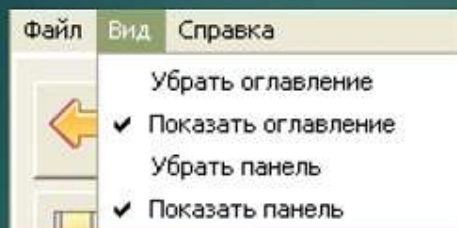
Главное меню состоит из пунктов: Файл, Вид и Помощь.



Меню «Файл» содержит следующие пункты:

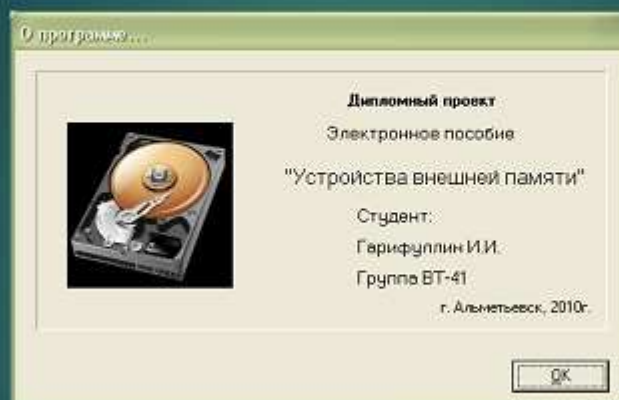
-  Сохранить содержание – сохраняет содержание. Так же можно использовать комбинацию клавиш Ctrl+S.
-  Печать... – открывает диалоговое окно принтера. Так же можно использовать комбинацию клавиш Ctrl+P.
-  Выход – кнопка выхода из программы. Так же можно использовать клавишу Esc.

► Меню «Вид» содержит следующие пункты:



- При выборе «Убрать оглавление» или «Показать оглавление» оглавление убирается или показывается соответственно.
- При выборе «Убрать панель» или «Показать панель» панель с кнопками убирается или показывается соответственно.

Меню «Справка» содержит пункт «О программе...»



Приложение 3

№	Тема
1.	Аппаратное обеспечение ПК
2.	Программное обеспечение ПК
3.	Работа с массивами в языке Паскаль
4.	Подпрограммы в Паскаль
5.	Языки программирования
6.	Программа Corel Draw
7.	Программа Adobe PhotoShop
8.	Программа Adobe Dreamweaver
9.	Программа Adobe Flash
10.	Компьютерная безопасность

Приложение 4

Техническое задание

Введение:

Наименование программы - "Электронный учебник " _____ ""

Наименование темы разработки - "Разработка электронного учебника "Энциклопедия интернет"".

Условное обозначение темы разработки (шифр темы) - "ЭУП-___"

Основание для разработки:

Основанием для проведения разработки является разработка программы для учебного заведения.

Назначение разработки:

Функциональным назначением электронного учебника является предоставление пользователю возможности поиска интересующей информации.

Требования к электронному учебнику или программному изделию:

Требования к составу выполняемых функций:

Электронный учебник должен обеспечивать возможность выполнения перечисленных ниже функций:

- Функции поиска.
- Функцией сортировки данных.
- Обеспечивать корректную работу данного программного обеспечения.

Требования к организации входных данных:

Входные данные программы должны быть организованы корректно.

Требования к организации выходных данных:

Требования к обеспечению надежного (устойчивого) функционирования программы:

- Организовать корректную работу программы.
- Приятный внешний вид.
- Простое и легко усвоение информации и оформление.

Условия эксплуатации:

Требования к составу и параметрам технических средств:

• В состав технических средств должен входить IBM-совместимый персональный компьютер (ПЭВМ), включающий в себя:

- Процессор с тактовой частотой 800 ГГц;
- Оперативную память объемом 256 мб;
- Клавиатуру, мышь.

Требования к информационной и программной совместимости:

Требования к исходным кодам и языкам программирования

Исходные коды программы должны быть реализованы на языке Delphi. В качестве интегрированной среды разработки программы должна быть использована среда Borland Delphi 7.

Требования к программным средствам, используемым программой:

Системные программные средства, которые будут использоваться программой, должны быть представлены лицензионной локализованной версией операционной системы.

Требования к упаковке отсутствуют.

Специальные требования:

Должно быть обеспечено взаимодействие с пользователем (оператором) посредством графического пользовательского интерфейса.

Предварительный состав программной документации:

Состав программной документации должен включать в себя:

- Техническое задание.
- Листинг программы.

Стадии разработки:

Разработка должна быть проведена в три стадии:

- техническое задание;
- технический (и рабочий) проекты;
- внедрение.

Этапы разработки

На стадии "Техническое задание" должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.

На стадии "Технический (и рабочий) проект" должны быть выполнены перечисленные ниже этапы работ:

- разработка программы;
- разработка программной документации;
- испытания программы.

На стадии "Внедрение" должен быть выполнен этап разработки "Подготовка и передача программы".

Содержание работ по этапам:

- Постановка задачи;
- Определение и уточнение требований к техническим средствам;

- Определение требований к программе;
- Определение стадий, этапов и сроков разработки программы и документации на нее;
- Выбор языков программирования;
- Согласование и утверждение технического задания.

Приложение 5

При создании электронного учебника могут использоваться следующие компоненты:

TreeView - служит для показа иерархических данных. Отображение имеет древовидную структуру, в которой можно выбрать необходимый узел или узлы. Информация может быть разнообразной от структуры большого учреждения до структуры простого документа. Каждый узел имеет свои значения и параметры. Данные компонента TreeView содержатся в свойстве Items. На этапе разработки проекта дерево формируется в окне редактора узлов дерева. Редактор вызывается двойным щелчком на компоненте TreeView или в Инспекторе Объектов нажать на кнопку с многоточием рядом со свойством Items.

- Кнопка New Item - добавляет новый узел. Новый узел будет на том же уровне, где в момент добавления находилась мышь.
- Кнопка New SubItem - добавляет дочерний узел. Он будет на уровень ниже чем тот узел на котором располагалась мышь.
- Delete - удалит узел на который указывает мышь.
- В панели Item Properties можно задать такие свойства узла как Text (это надпись узла), и пиктограммы рядом с узлом.
- Image Index - картинка если пользователь выделил узел,
- Selected Index - узел не выделен,
- State Index - это вторая миниатюра, которая остается с узлом постоянно не зависимо от его состояния.

WebBrowser - Данный компонент представляет собой элемент управления ActiveX. Это значит, что на самом деле программа, используя данный компонент, обращается к программным ресурсам компьютера и использует Internet Explorer (IE), установленный в операционной системе Windows. Этот компонент может выполнить все те операции, которые заложены в IE, поэтому его использование позволяет сделать создаваемую программу очень функциональной. Для того чтобы созданная программа корректно работала на компьютере конечного пользователя, в его системе должен быть установлен IE. Разумеется, это требование автоматически выполняется, если используется операционная система Windows 98 или более новая.

Свойства объекта **WebBrowser**:

- Busy - Булевское свойство, которое принимает значение true, когда элемент управления находится в состоянии выполнения текущей операции
- Document - Обеспечивает доступ к интерфейсу управления исходным кодом Web-страницы
- LocationName - Содержит короткое имя текущего документа
- LocationURL - Содержит URL текущего документа
- Offline - Позволяет перевести компонент в offline-режим работы
- Silent - Определяет, может ли WebBrowser отображать диалоговые окна

- **Visible** - Булевское свойство, определяющее, отображается ли компонент на форме.

Edit - представляет из себя однострочный редактор для ввода, отображения и изменения текста. Основное свойство, содержащее данные введенные пользователем это **Text**. Оно имеет тип **string** и может хранить и передавать только текстовые данные. Если необходимо записать какое либо число, то оно будет представлено в текстовом виде. При создании нового компонента **Edit** на форме свойство **Text** будет совпадать с именем (**Edit1**, **Edit2** и далее по нарастающей). Желательно его сразу очистить.

У компонента **Edit** нет управляющих реакций на нажатие клавиш **ENTER** и **ESC**, хотя в остальном он имеет свойства редакторов. Перемещение курсора по строке клавишами со стрелками, удаление ненужных символов **DELETE** и **Backspace** выделение текста, это неполный перечень его возможностей.

Button - это простая командная кнопка.

Свойства компонента **Button**:

- **Caption** - содержит название которое отображается на кнопке, стандартное название **"Button1"**;
- **Enabled** - свойство позволяющее "отключить" (свойство **false**) или "включить" (свойство **true**) кнопку. В первом случае кнопка будет отображаться серым цветом и никак не будет реагировать на действия пользователя. Полезна когда требуется чтобы пользователь прежде чем нажмет кнопку выполнил определенные действия. Например ввел данные в поле ввода (компонент **Edit**). Значение по умолчанию **True**;
- **Height** - принимает числовое значение, устанавливает высоту кнопки. Значение по умолчанию **25**;
- **Width** - принимает аналогичное вышеописанному свойству значение, только устанавливает длину кнопки. Значение по умолчанию **75**;
- **Hint** - редко используемое свойство, но очень полезное пользователю - подсказка. Которая высвечивается при наведении указателя мышки на кнопку. Свойство принимает строковое значение;
- **Left** - регулирует положение компонента относительно левой границы родительского компонента, если компонент расположен на форме **"Form1"**, то относительно его левой границы. Свойство принимает числовое значение, расстояние измеряется в пикселах;
- **Right** - свойство аналогично вышеописанному только задает расстояние от правой границы родительского компонента;
- **Top** - задается расстояние от верхней границы компонента;
- **Bottom** - задается расстояние от нижней границы компонента;
- **Visible** - принимает два значения **True** и **False**. При установке второго значения компонент **"Button1"** делается невидимым;
- **Name** - имя компонента используемое для работы с ним в редакторе кода среды разработки **Delphi**.

Тема 10	Разработка главного меню ЭУП
----------------	-------------------------------------

1.1. Технология обучения на лабораторном занятии

<i>Учебное время: 2 часа.</i>	
<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Создание основной формы приложения и настройка ее свойств. 2. Вставка и разработка главного меню на форму приложения.
<i>Цель учебного занятия:</i>	Сформировать умение проектировать работу по созданию ЭУП в среде Delphi.
<i>Педагогические задачи:</i> Организовать работу студентов по: <ul style="list-style-type: none"> • созданию основной формы приложения и настройки ее свойств. • вставке и разработке главного меню на форме приложения. 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • Создавать основную форму приложения и настраивать ее свойства. • Вставлять и разрабатывать главное меню на форме приложения.
<i>Методы обучения</i>	иерархическая диаграмма КАК?
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.10. Сообщает, тему, цель учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (65 мин)	2.1. Предлагает каждому студенту заполнить иерархическую диаграмму КАК?, которая позволит получить общее представление о способах и средствах решения проблемы:	2.1. Просматривают, уточняют, анализируют.

	<p>как организовать Главное меню приложения ЭУП? (<i>приложение 2</i>).</p> <p>2.2. Организует оформление главной формы и настройку ее свойств.</p> <p>2.3. Раздает методические рекомендации и организует оформление главного меню приложения (<i>приложение 3</i>).</p>	<p>2.2. Создают главную форму приложения.</p> <p>2.3. Оформляют главное меню приложения.</p>
<p>3-й этап.</p> <p>Заключительно-результативный (10 мин)</p>	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> • Повторить компоненты Delphi. • Подготовить изображения в формате .ico для создания кнопок панели инструментов. 	<p>Задают уточняющие вопросы.</p>

Приложение 1

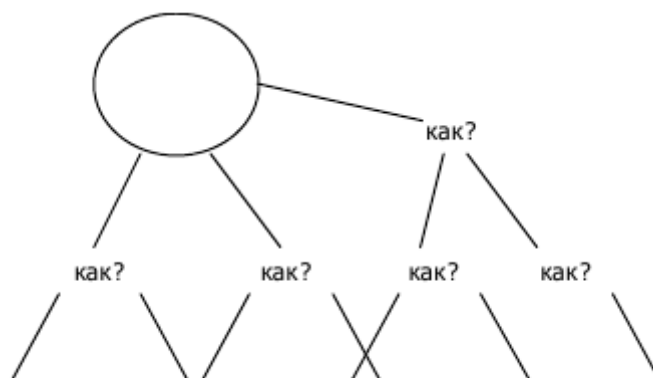
Тема: Составляющие элементы ЭУП, разрабатываемого в среде Delphi

Цель занятия: Сформировать умение проектировать работу по созданию ЭУП в среде Delphi.

План занятия:

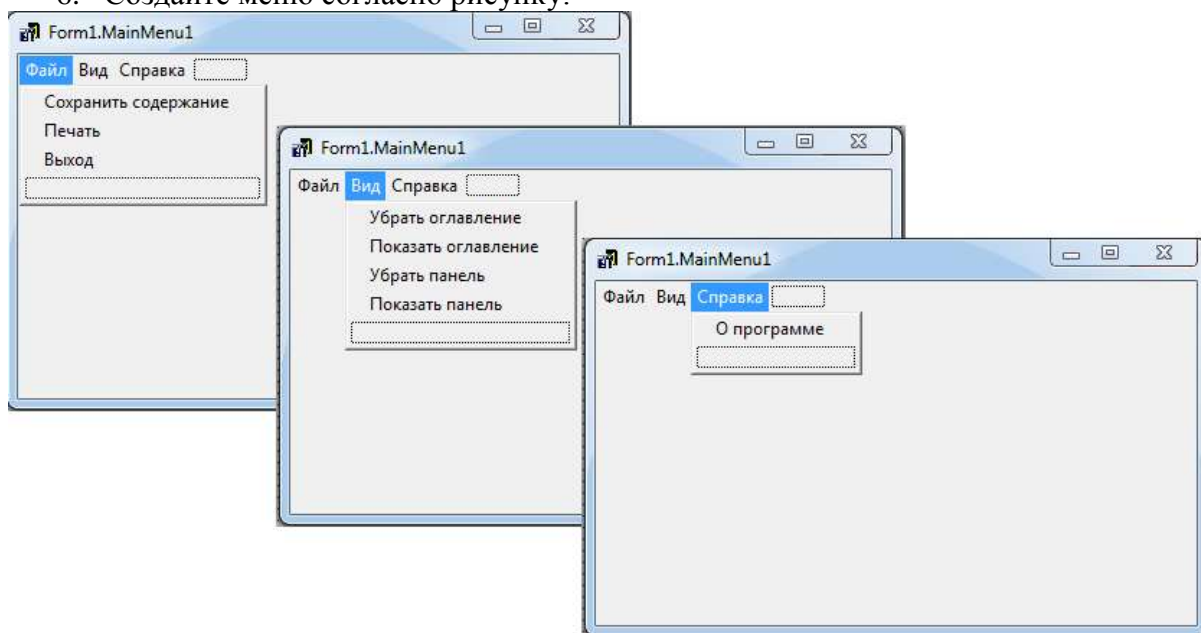
1. Создание основной формы приложения и настройка ее свойств.
2. Вставка и разработка главного меню на форму приложения.

Приложение 2

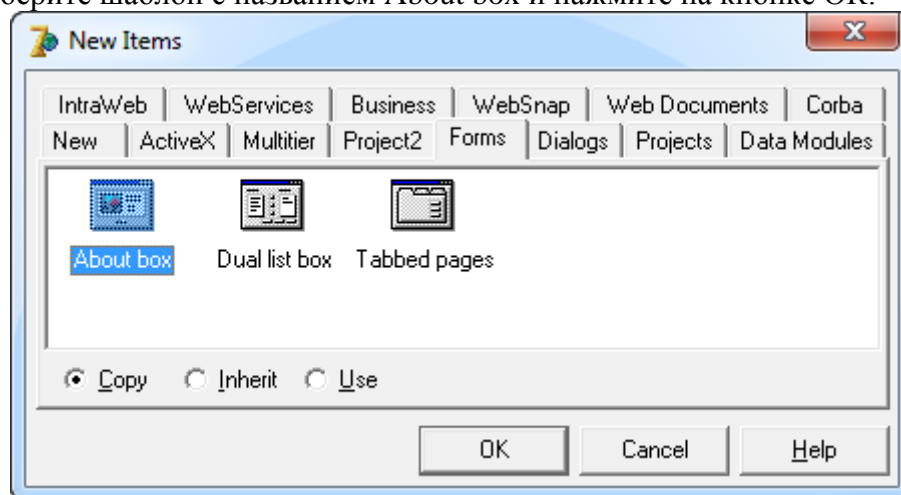


Приложение 3

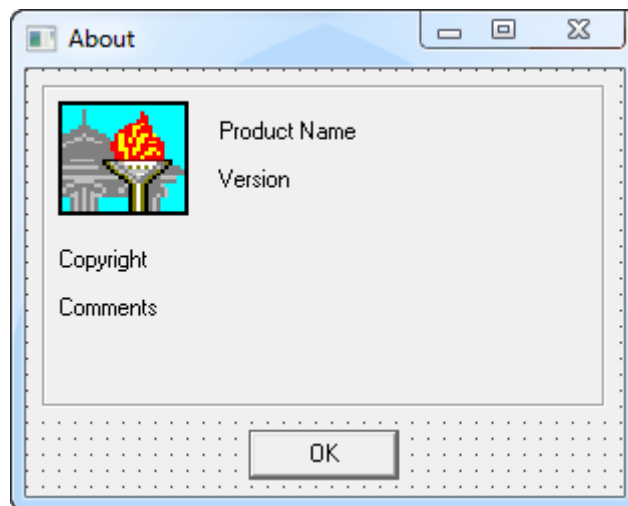
1. Создайте папку и назовите ее Main.
2. Создайте новое приложение командой File – New – Application.
3. Введите в свойство Caption формы текст: “Электронный учебник «_____» (вместо многоточия укажите название ЭУП).
4. Вставьте компонент MainMenu на форму.
5. Выполнив двойной щелчок на значке MainMenu, вызовите диалоговое окно Конструктора меню.
6. Создайте меню согласно рисунку:



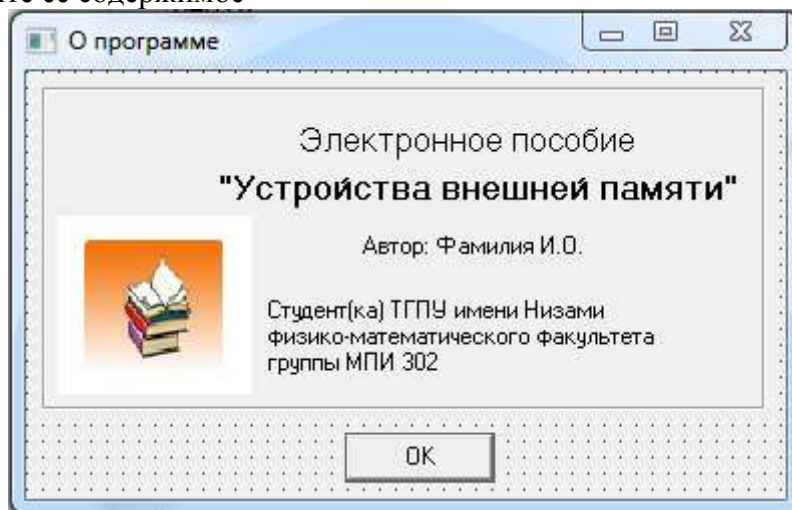
7. Чтобы оформить пункт Справка – О программе дайте команду File – New – Other и в открывшемся диалоговом окне New Items перейдите на вкладку Forms. Выберите шаблон с названием About box и нажмите на кнопке OK.



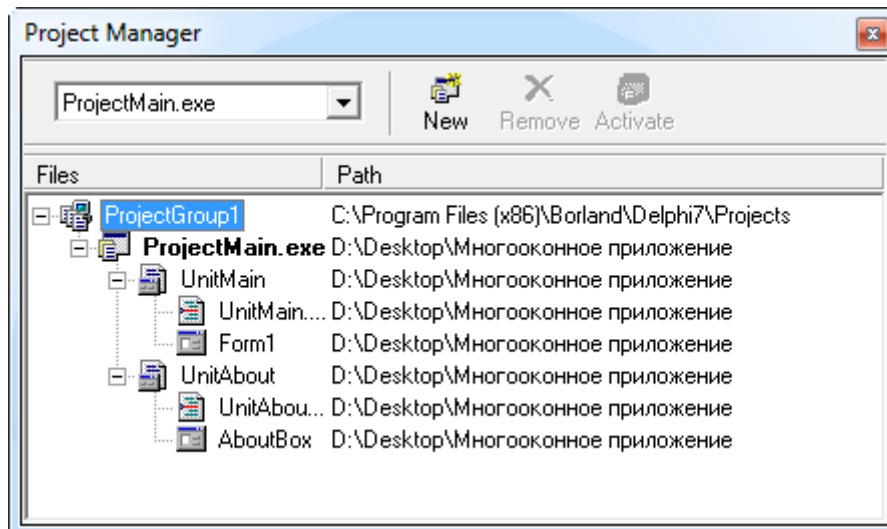
8. В проект добавится форма с именем AboutBox.



9. Измените ее содержимое



10. Для командной кнопки ОК напишите код, закрывающий данное окно (команда Close).
11. Сохраните файлы в папке Main дав следующие имена: для Unit1 – UnitMain; Unit2 – UnitAbout; Project1 – ProjectMain.
12. Выполните команду View – Project Manager... Откроется окно, отображающее все элементы данного проекта. Щелкнув на кнопке со знаком плюс можно просмотреть полное содержимое проекта. Закройте данное окно.



13. Теперь напомним код для пункта строки меню О программе. Выполните двойной щелчок на пункте О программе. Откроется код программы. После раздела Implementation введите строку

```
uses UnitAbout;
```

а в процедуре обработки пункта меню следующую команду:

```
AboutBox.ShowModal;
```

```
implementation
uses UnitAbout;

{$R *.dfm}

procedure TForm1.N5Click(Sender: TObject);
begin
    AboutBox.Show;
end;
```

14. Сохраните введенные изменения и запустите проект. Проверьте, что происходит при нажатии на пункте меню Об авторе.

Тема 11	Разработка панели инструментов ЭУП
----------------	---

1.1. Технология обучения на лабораторном занятии

<i>Учебное время: 2 часа.</i>	
<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Разработка структуры панели инструментов и подбор изображений для кнопок 2. Выбор компонента для создания кнопок панели инструментов, их размещение на форме и настройка их свойств. 3. Настройка событий кнопок панели инструментов.
<i>Цель учебного занятия:</i>	Сформировать умение по созданию панели инструментов ЭУП в среде Delphi.
<i>Педагогические задачи:</i> Организовать работу студентов по: <ul style="list-style-type: none"> • разработке структуры панели инструментов и подбору изображений для кнопок • выбору компонента для создания кнопок панели инструментов, их размещению на форме и настройке их свойств. • Настройке событий кнопок панели инструментов. 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • Разрабатывать структуру панели инструментов и подбирать изображения для кнопок • Выбирать компоненты для создания кнопок панели инструментов, размещать их на форме и настраивать их свойства. • Настраивать события кнопок панели инструментов..
<i>Методы обучения</i>	структурно-логическая схема КАСКАД
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, парная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.11. Сообщает, тему, цель учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (65 мин)	<p>2.1. Предлагает разбиться студентам на пары и заполнить структурно-логическую схему КАСКАД которая позволит структурировать действия пользователя, выполняемые с помощью панели инструментов ЭУП (<i>приложение 2</i>).</p> <p>2.2. Предлагает рассмотреть кнопки, которые размещены в электронном учебнике и их назначения (<i>приложение 3</i>). Из подготовленных изображений в формате .ico отобрать необходимые.</p> <p>2.3. Размещают кнопки панели инструментов на форме, выполняют настройку их свойств.</p> <p>2.4. Разбирают основные события, которые должны происходить при щелчке на кнопках меню (<i>приложение 4</i>). Дополняют необходимыми элементами панель инструментов.</p>	<p>2.1. Составляют структурно-логическую схему КАСКАД.</p> <p>2.2. Просматривают, уточняют, подбирают изображения.</p> <p>2.3. размещают кнопки панели инструментов.</p> <p>2.4. Разрабатывают и настраивают события кнопок панели инструментов.</p>
3-й этап. Заключительно-результативный (10 мин)	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> • Доработать панель инструментов; • Повторить компоненты Delphi для отображения иерархических данных. 	Задают уточняющие вопросы.

Приложение 1

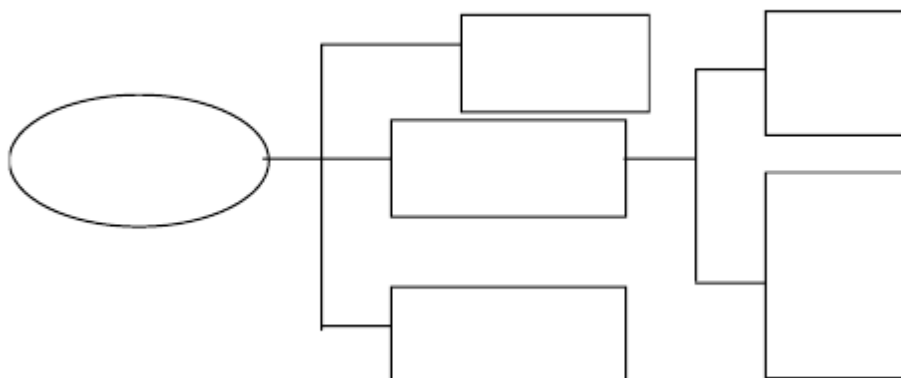
Тема: Разработка панели инструментов ЭУП

Цель занятия: Сформировать умение по созданию панели инструментов ЭУП в среде Delphi.

План занятия:

1. Разработка структуры панели инструментов и подбор изображений для кнопок
2. Выбор компонента для создания кнопок панели инструментов, их размещение на форме и настройка их свойств.
3. Настройка событий кнопок панели инструментов


Приложение 2



Приложение 3

ПАНЕЛЬ ИНСТРУМЕНТОВ

На панели инструментов находятся следующие кнопки:

 или  – показывают или скрывают оглавление;


 – сохраняет текущее содержание в файл;


 – выводит меню с выбором:


 – добавляет новый раздел;


 – добавляет новую тему к выделенному разделу;

 – удаляет выделенный элемент;



 – редактирование текущего элемента;

 – вызывает диалоговое окно печати;

 – вызывает окно о программе;

 – выход из программы.

Приложение 4

Для добавления в содержание нового раздела нужно нажать на кнопку  и из выпадающего меню выбрать . Откроется диалоговое окно добавления нового раздела (рисунок 1).

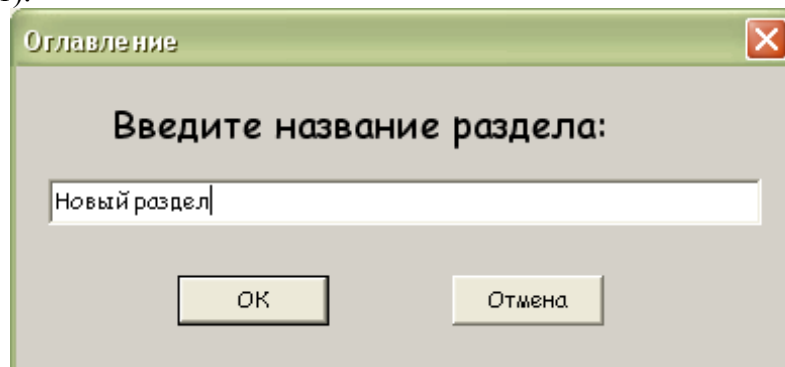


Рисунок 1 Диалоговое окно добавления нового раздела.

Введем название главы и подтвердим кнопкой ОК. Новая глава появится в конце содержания (рис. 2).

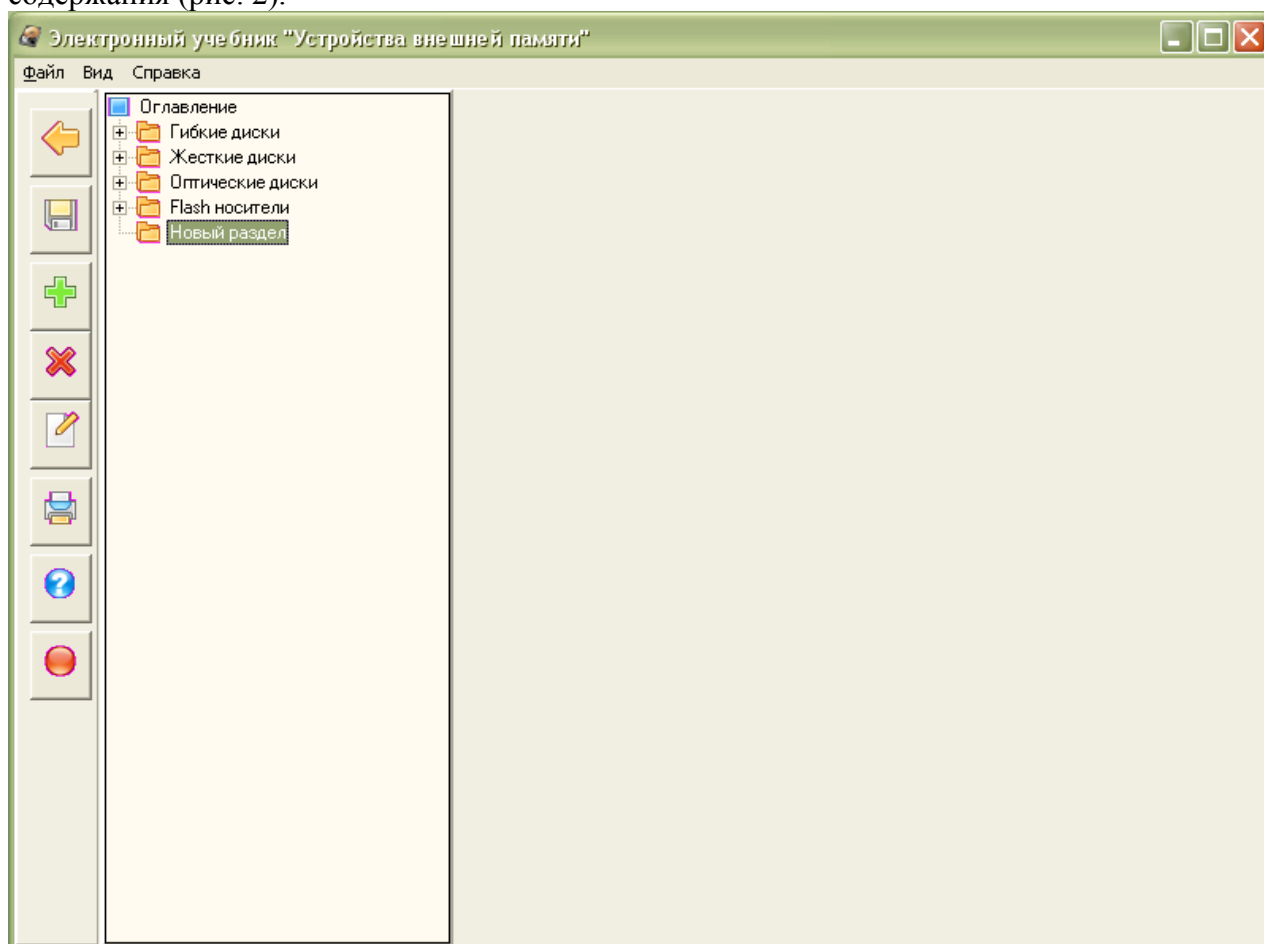



Рисунок 2 Новый раздел.

Для добавления новой темы нужно выделить в содержании главу и нажать на кнопку добавления новой темы . Откроется диалоговое окно добавления темы (рисунок 3).

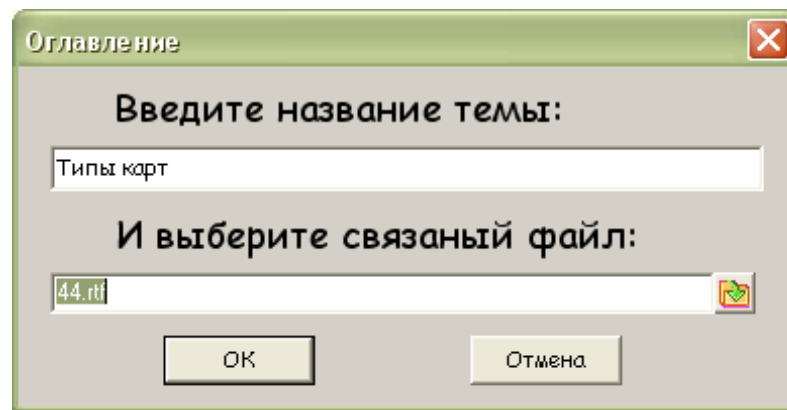



Рисунок 3 Добавление темы.

Введем название темы и выберем связанный с ней файл кнопкой  (рисунок 4).

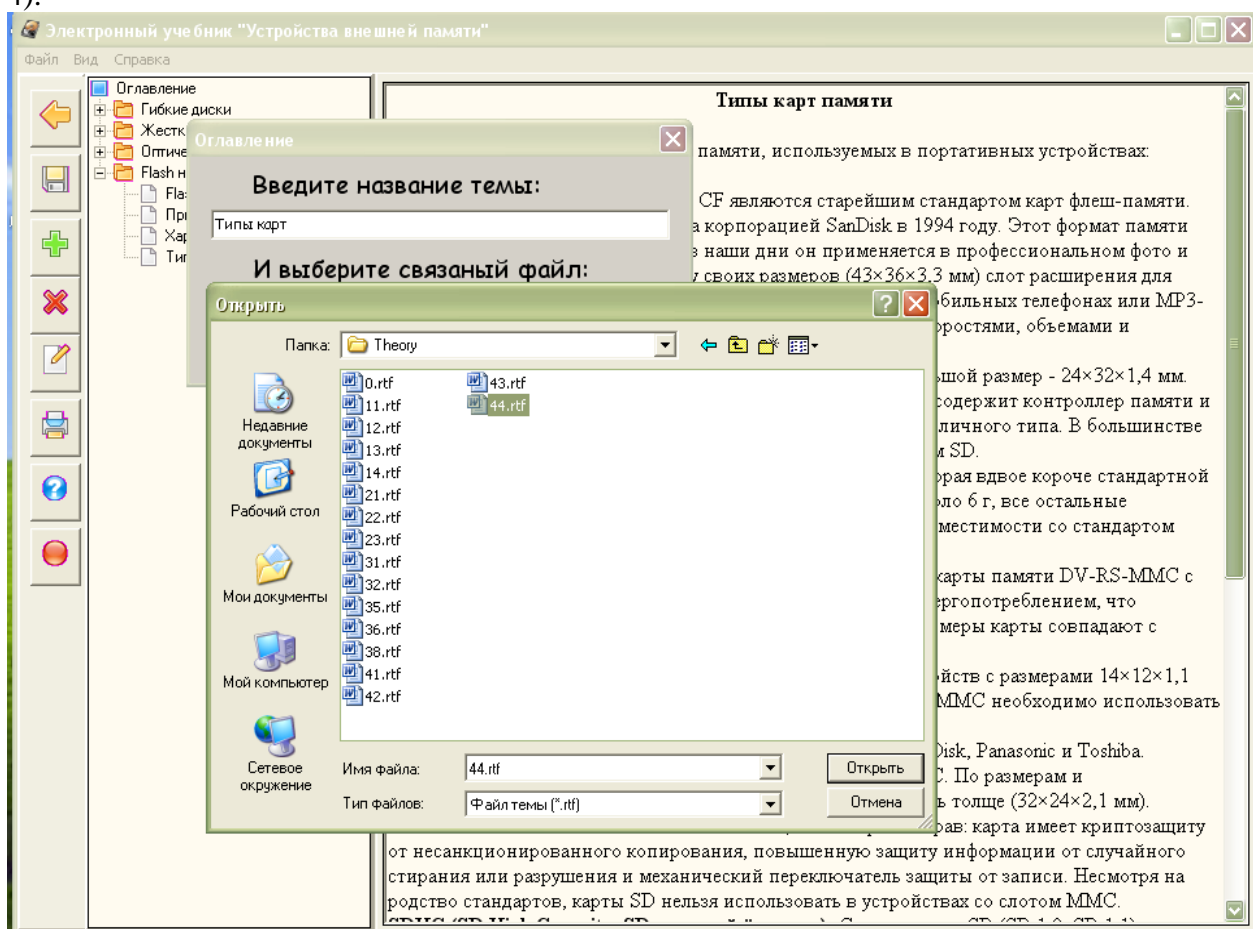





Рисунок 4 Выбор файла темы.

Сохраним изменения. Тема появится в указанной главе.

Для сохранения текущих изменений нужно нажать на сохранить содержание .

Для редактирования элемента содержания используется кнопка редактирования . Выделяем нужный элемент содержания и нажимаем на нее. В зависимости от выбранного типа элемента откроется окно изменения темы или главы.

Для удаления из содержания какого-либо элемента нужно выделить этот элемент и нажать на кнопку удаления элемента . Для предотвращения случайного удаления элемента программа выдаст запрос (рисунок 5).

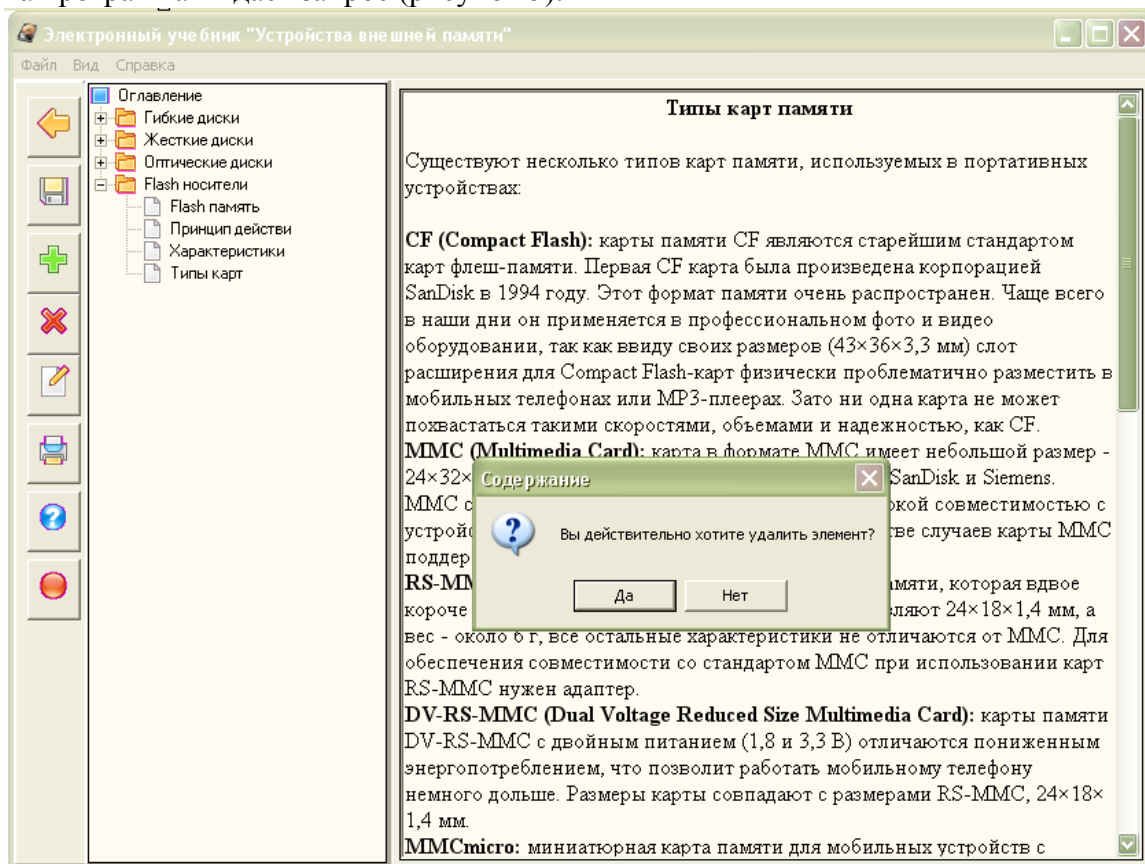


Рисунок 5 Удаление элемента.

Тема 12	Разработка оглавления ЭУП
----------------	----------------------------------

1.1. Технология обучения на лабораторном занятии

<i>Учебное время: 2 часа.</i>	
<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Выбор компонента для отображения содержания ЭУП. 2. Встраивание, заполнение и написание кода.
<i>Цель учебного занятия:</i>	Сформировать умение по созданию оглавления ЭУП в среде Delphi.
<i>Педагогические задачи:</i> Организовать работу студентов по: <ul style="list-style-type: none"> • Выбору компонента для отображения содержания ЭУП. • Встраиванию, заполнению и написанию кода. 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • Выбирать компонент для отображения содержания ЭУП. • Встраивать, заполнять и писать код.
<i>Методы обучения</i>	схема ПИРАМИДА
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, парная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.12. Сообщает, тему, цель учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (65 мин)	2.1. Приводит пример содержания созданного ЭУП и ознакомиться с их характеристиками (<i>приложение 2</i>). 2.2. Предлагает студентам используя схему ПИРАМИДА представить иерархическую	2.1. Составляют структурно-логическую схему КАСКАД.

	<p>структуру содержания ЭУП (<i>приложение 3</i>).</p> <p>2.3. Проводит мозговой штурм, в ходе которого актуализирует знания студентов о компонентах Delphi, с помощью которых можно отобразить иерархические данные. Напоминает о свойствах данных компонентов (<i>приложение 4</i>).</p> <p>2.4. Организует работу студентов по созданию содержания ЭУП.</p>	<p>2.2. Просматривают, уточняют, подбирают изображения.</p> <p>2.3. размещают кнопки панели инструментов.</p> <p>2.4. Разрабатывают и настраивают события кнопок панели инструментов.</p>
<p>3-й этап.</p> <p>Заключительно-результативный (10 мин)</p>	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> • Доработать содержание ЭУП; • Подготовить содержимое теоретических материалов в виде документов в формате RTF согласно содержанию ЭУП. 	<p>Задают уточняющие вопросы.</p>

Приложение 1

Тема: Разработка оглавления ЭУП

Цель занятия: сформировать умения по созданию оглавления ЭУП в среде Delphi.

План занятия:

1. Выбор компонента для отображения содержания ЭУП.
2. Встраивание, заполнение и написание кода.

Приложение 2

В левой части основного окна находится содержание обучения. Оно состоит из разделов (📁), каждый из которых содержит параграфы или другие разделы.

Параграф характеризуется одним из следующих типов:

1. Теория (📄);
2. Практика (🧑‍🔬);
3. Тестирование (🔍).

Каждый из вышеперечисленных типов определяет режим изучения материала. При выборе параграфа правая часть основного окна меняется в зависимости от типа выбранного параграфа.

В системе реализовано 4 режима обучения:

- а) режим изучения (теория);
- б) демонстрационный режим (практика);
- в) режим тестирования (тестирование).

Режим изучения.

Переход в режим изучения осуществляется при выборе параграфа, который характеризуется типом «Теория» (📄).

В данном режиме пользователь может изучить необходимую литературу. Вся информация выдается в правой части основного окна. Внешний вид системы в данном режиме показан на рис. 2.

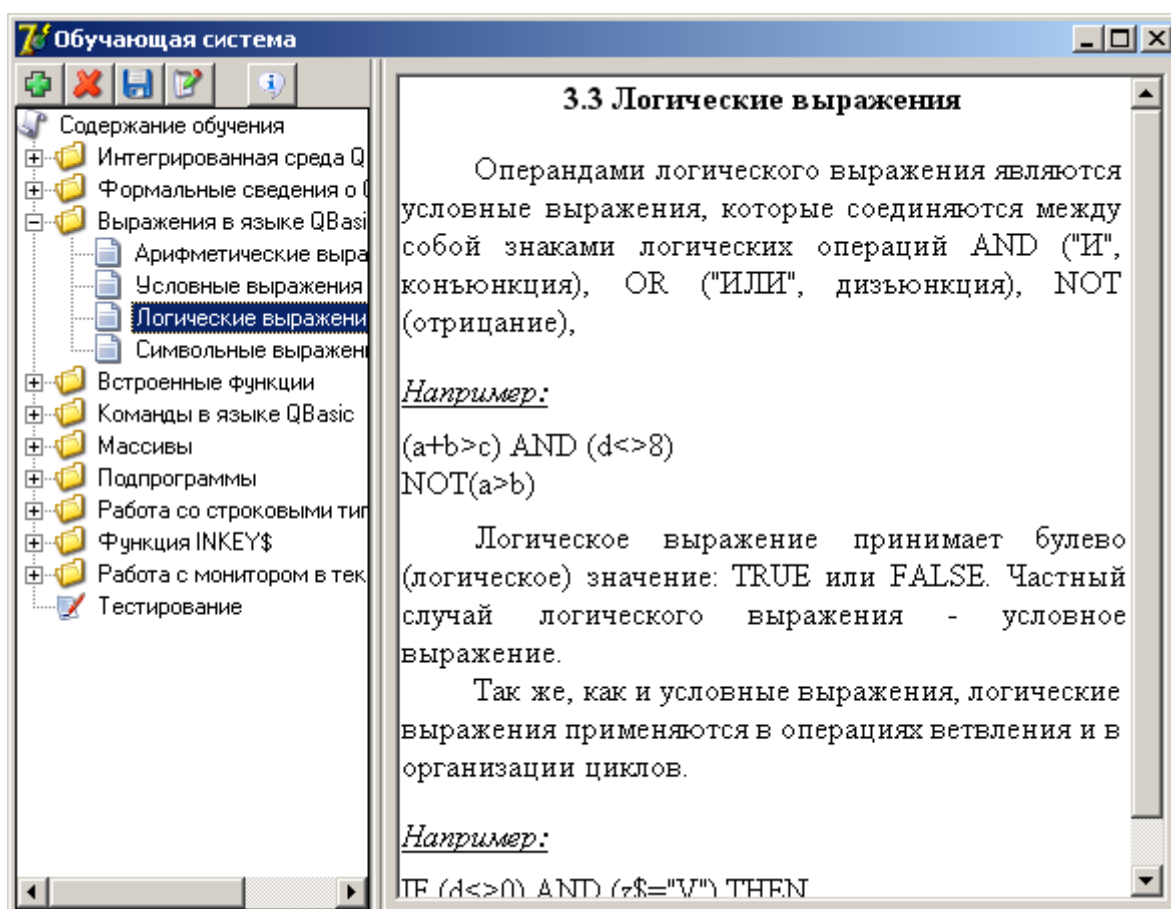


Рис. 2. Режим изучения.

Демонстрационный режим.

Переход в демонстрационный режим осуществляется при выборе параграфа, который характеризуется типом «Практика» (🎮).

В данном режиме пользователю предоставляется возможность запуска уже готовых программ в QBASIC.

В области «Описание программы» выводится постановка решаемой с помощью данного кода задачи.

В области «Код программы» отображается код программы на языке QBasic.

Демонстрационный режим предназначен для наглядного изучения. Внешний вид системы в данном режиме показан на рис. 3.

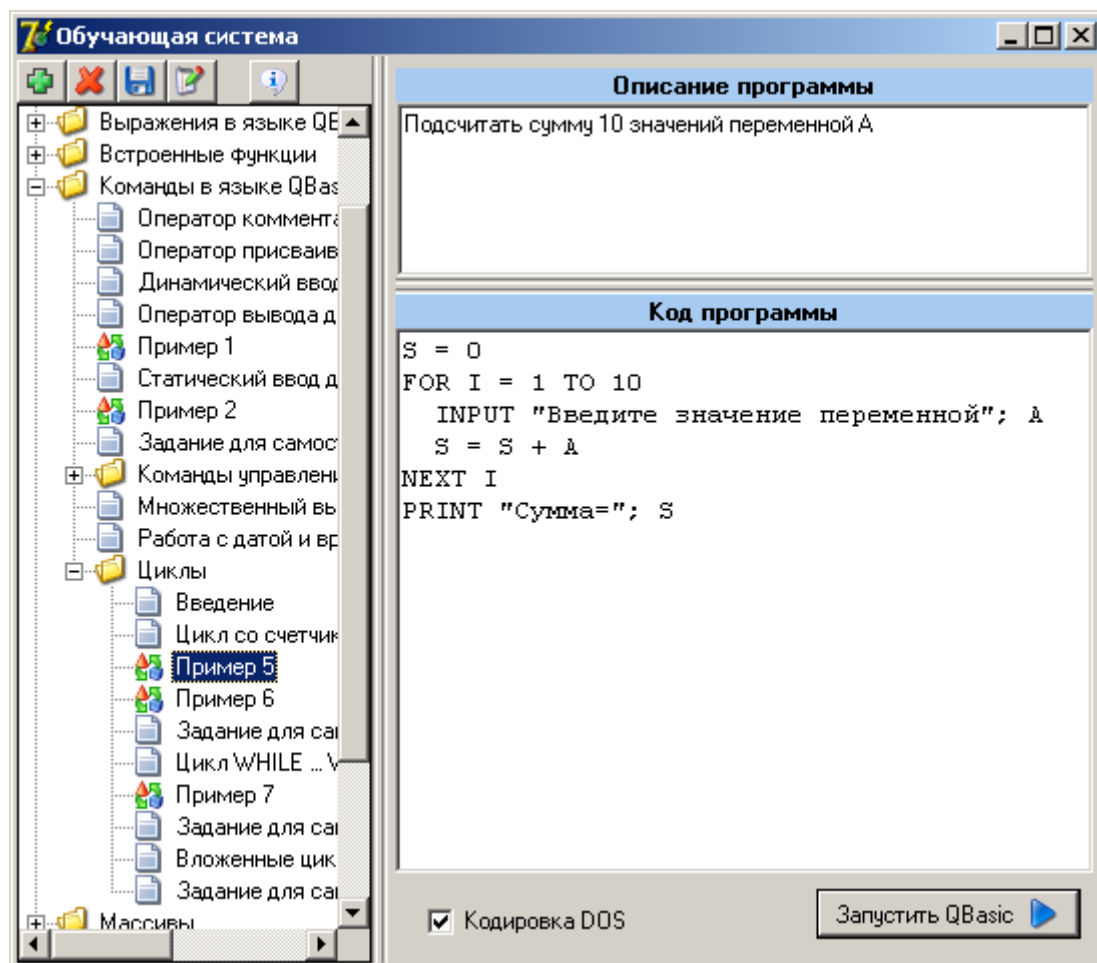


Рис. 3. Демонстрационный режим.

При нажатии на кнопку «Запустить QBasic» происходит открытие текущей программы в редакторе QBasic.

Режим тестирования.

Переход в режим тестирования осуществляется при выборе параграфа, который характеризуется типом «Тестирование» (🔍). После нажатия на кнопку «Начать тестирование» пользователь переходит в режим ответа на вопросы (рис. 4).

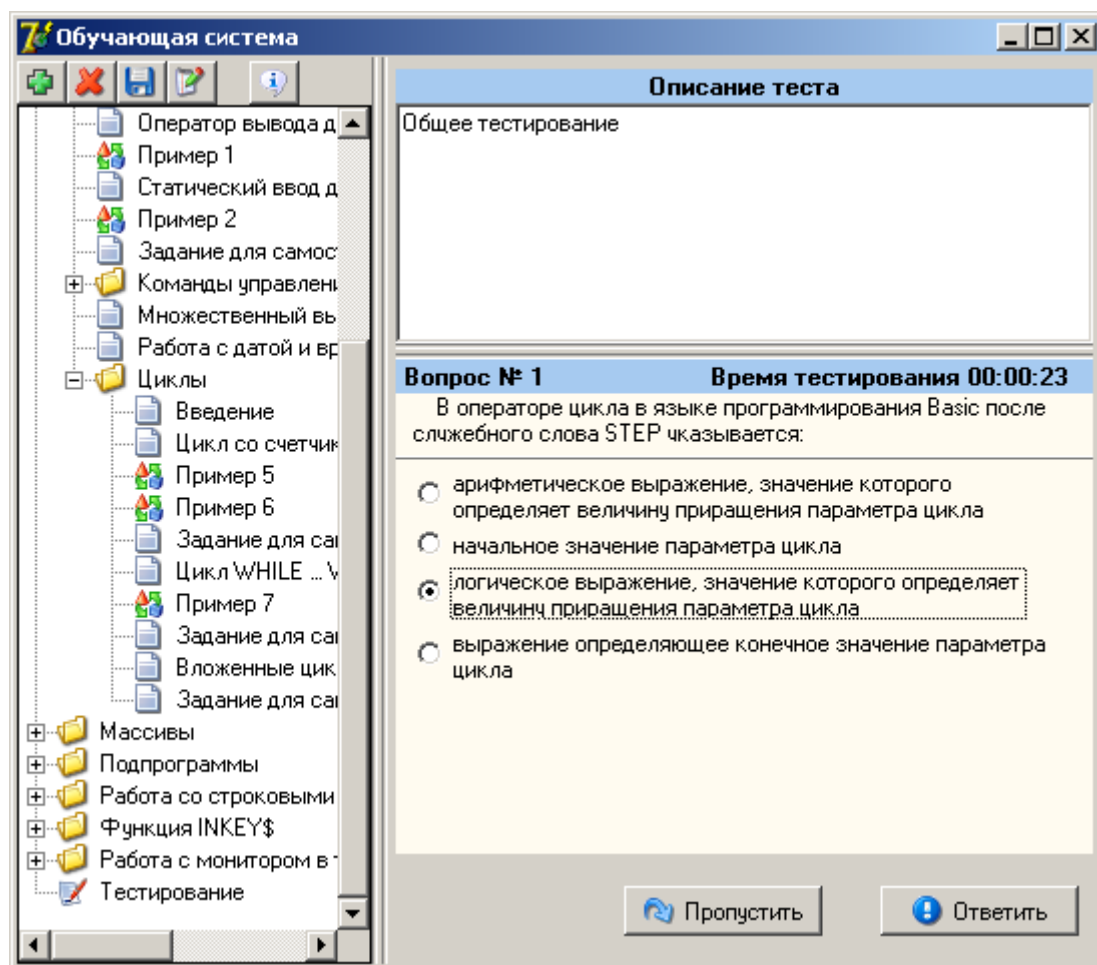


Рис. 4. Режим тестирования.

В верхней части области вопроса отображается его номер и общее время теста.

На каждый вопрос предоставляются варианты ответов, из которых необходимо выбрать один и нажать кнопку «Ответить» или «Пропустить» (если пользователь не знает ответа на вопрос).

По окончании тестирования выводится окно результата контроля (рис. 5)

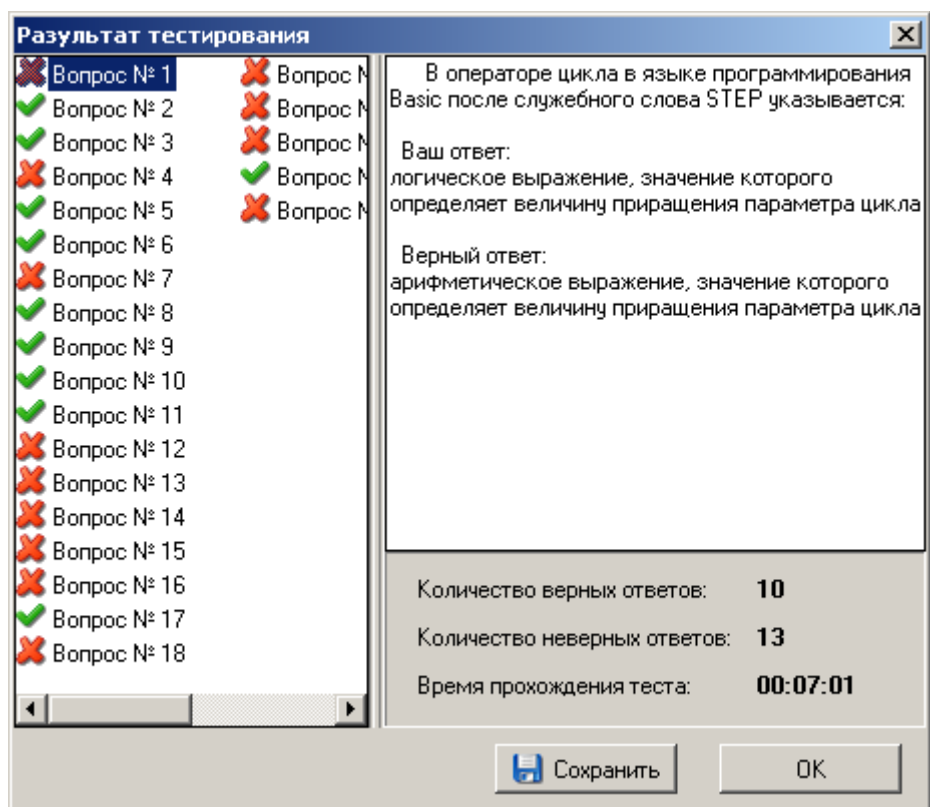


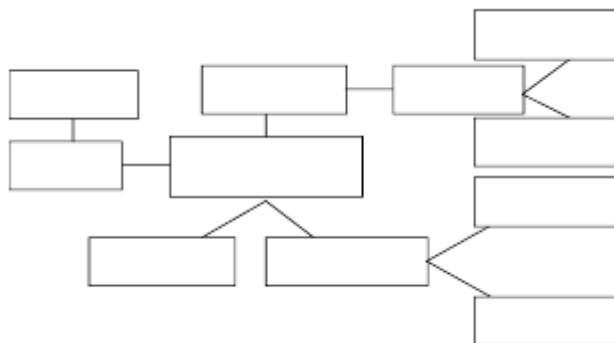
Рис. 5. Окно результата тестирования.

В этом окне пользователю предоставляется следующая информация:

1. Количество верных ответов;
2. Количество неверных ответов;
3. Время прохождения контроля;
4. Перечень всех вопросов с признаком верности ответа;
5. Выбранный вариант ответа на вопрос;
6. Верный вариант ответа на вопрос.

Примечание. При необходимости можно добавить в программу возможность сохранения результата тестирования/контроля.

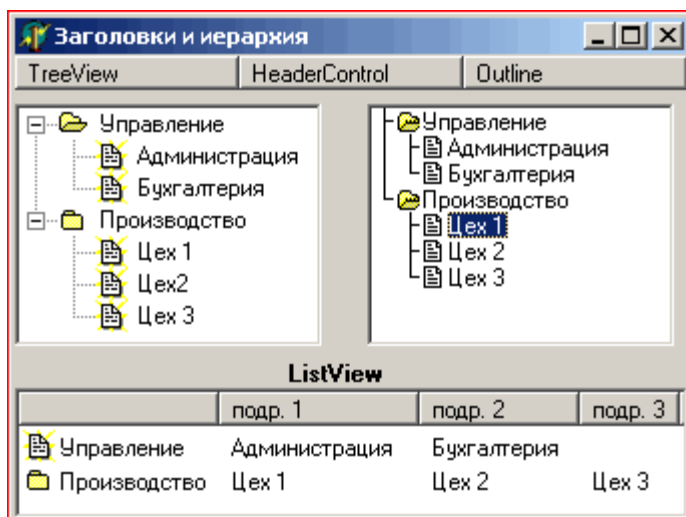
Приложение 3



Компоненты отображения иерархических данных — TreeView, Outline, ListView

Компоненты **TreeView** и **Outline** служат для отображения иерархических данных в виде дерева (см. пример на рис. 6), в котором пользователь может выбрать нужный ему узел или узлы. Иерархическая информация может быть самой разной: структура некоторого предприятия, структура документации учреждения, структура отчета и т.п. С каждым узлом дерева могут быть связаны некоторые данные.

Рис. 6
Примеры
компонентов
заголовков и
отображения
иерархических
данных

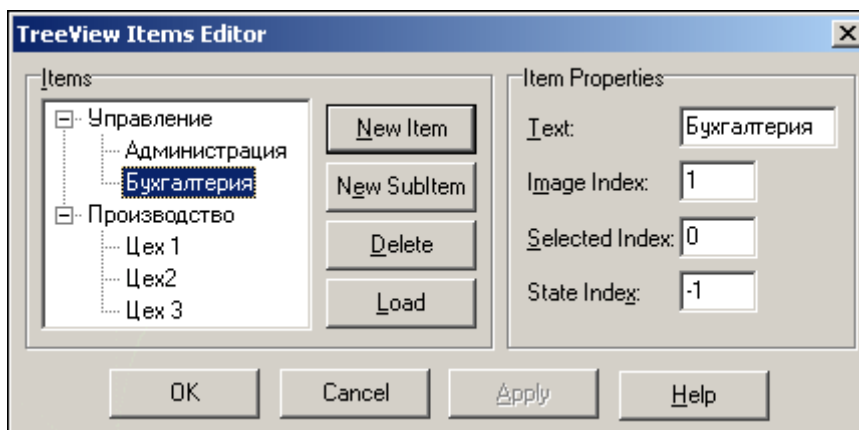


Возможности компонента **TreeView** несколько шире, чем компонента **Outline**. К тому же **TreeView** — 32-разрядный компонент, а **Outline** — 16-разрядный. Поэтому **Outline** целесообразно использовать только в приложениях, создаваемых с помощью Delphi 1.

Основным свойством **TreeView**, содержащим информацию об узлах дерева, является **Items**. Доступ к информации об отдельных узлах осуществляется через этот индексный список узлов. Например, **TreeView1.Items[1]** — это узел дерева с индексом 1 (второй узел дерева). Каждый узел является объектом типа **TTreeNode**, обладающим своими свойствами и методами.

Во время проектирования формирование дерева осуществляется в окне редактора узлов дерева, представленном на рис. 3.20. Это окно вызывается двойным щелчком на компоненте **TreeView** или нажатием кнопки с многоточием около свойства **Items** в окне Инспектора Объектов.

Рис. 7
Окно редактора
узлов дерева
компонента
TreeView



Кнопка **New Item** (новый узел) позволяет добавить в дерево новый узел. Он будет расположен на том же уровне, на котором расположен узел, выделенный курсором в момент щелчка на кнопке **New Item**.

Кнопка **New SubItem** (новый дочерний узел) позволяет добавить в дерево дочерний узел. Он будет расположен на уровень ниже уровня того узла, который выделен курсором в момент щелчка на кнопке **New SubItem**.

Кнопка **Delete** (удалить) удаляет выделенный узел дерева. Кнопка **Load** позволяет загрузить структуру дерева из файла. Файл, хранящий структуру дерева — это обычный текстовый файл, содержащий тексты узлов. Уровни узлов обозначаются отступами. Например, файл дерева, изображенного на рис. 6 и 7, может иметь вид:

```
производство
  цех 1
  цех 2
  цех 3
управление
  администрация
  бухгалтерия
```

Для каждого нового узла дерева можно указать ряд свойств в панели **Item Properties** окна на рис. 7. Это прежде всего свойство **Text** — надпись, появляющаяся в дереве около данного узла. Свойства **Image Index** и **Selected Index** определяют индекс пиктограммы, отображаемой для узла, который соответственно не выделен и выделен пользователем в данный момент. Эти индексы соответствуют списку изображений, хранящихся в отдельном компоненте **ImageList**. Указание на этот компонент вы можете задать в свойстве **Images** компонента **TreeView**. Индексы начинаются с 0. Если вы укажете индекс -1 (значение по умолчанию), пиктограммы изображаться не будут. Последнее свойство — **State Index** в панели **Item Properties** позволяет добавить вторую пиктограмму в данный узел, не зависящую от состояния узла. Подобная пиктограмма может просто служить дополнительной характеристикой узла. Индекс, указываемый как **State Index**, соответствует списку изображений, хранящихся в отдельном компоненте **ImageList**, указанном в свойстве **StateImages** компонента **TreeView**.

Мы рассмотрели формирование дерева в процессе проектирования. Однако, дерево можно формировать или перестраивать и во время выполнения приложения. Для этого

служит ряд методов объектов типа **TTreeNode**. Следующие методы позволяют вставлять в дерево новые узлы:

function Add(Node: TTreeNode; **const** S: **string**): TTreeNode;

Добавляет новый узел с текстом **S** как последний узел уровня, на котором расположен **Node**.

function AddFirst(Node: TTreeNode; **const** S: **string**): TTreeNode;

Вставляет новый узел с текстом **S** как первый из узлов уровня, на котором находится **Node**. Индексы последующих узлов увеличиваются на 1.

function Insert (Node: TTreeNode; **const** S: **string**): TTreeNode;

Вставляет новый узел с текстом **S** сразу после узла **Node** на тот же уровень. Индексы последующих узлов увеличиваются на 1.

function AddChild(Node: TTreeNode; **const** S: **string**): TTreeNode;

Добавляет узел с текстом **S** как последний дочерний узла **Node**.

function AddChildFirst(Node: TTreeNode; **const** S: **string**): TTreeNode;

Вставляет новый узел с текстом **S** как первый из дочерних узлов узла **Node**. Индексы последующих узлов увеличиваются на 1.

Каждый из этих методов возвращает вставленный узел. Ниже в качестве примера приведен код, формирующий то же дерево, которое вы можете видеть на рис. 3.18 и 3.20.

```
TreeView1.Items.Clear;    // очистка списка
// добавление корневого узла 'производство' (индекс 0)
TreeView1.Items.Add(nil, 'производство');
{добавление дочерних узлов 'цех 1' – 'цех 3' (индексы 1 – 3)}
TreeView1.Items.AddChild(TreeView1.Items.Item[0], 'цех 1');
TreeView1.Items.AddChild(TreeView1.Items.Item[0], 'цех 2');
TreeView1.Items.AddChild(TreeView1.Items.Item[0], 'цех 3');
{добавление корневого узла 'управление' после узла
 'производство' (индекс 4) }
TreeView1.Items.Add(TreeView1.Items.Item[0], 'управление');
{добавление дочерних узлов 'администрация' и
 'бухгалтерия' узла 'управление' }
TreeView1.Items.AddChild(TreeView1.Items.Item[4], 'администрация');
TreeView1.Items.AddChild(TreeView1.Items.Item[4], 'бухгалтерия');
```

Дерево может быть сколь угодно разветвленным. Например, следующие операторы добавляют дочерние узлы «бригада 1» и «бригада 2» в сформированный ранее узел «цех 1»:

```
TreeView1.Items.AddChild(TreeView1.Items.Item[1], 'бригада 1');
TreeView1.Items.AddChild(TreeView1.Items.Item[1], 'бригада 2');
```

Текст, связанный с некоторым узлом, можно найти с помощью его свойства **Text**. Например, **TreeView1.Items.Item[1].Text** — это надпись «цех 1».

С каждым узлом может быть связан некоторый объект. Добавление таких узлов осуществляется методами **AddObject**, **AddObjectFirst**, **InsertObject**, **AddChildObject**, **AddChildObjectFirst**, аналогичными приведенным выше, но содержащими в качестве параметра еще указатель на объект:

function AddObject(Node: TTreeNode; const S: string ; Ptr: Pointer): TTreeNode;	Добавляет новый узел с текстом S и объектом Ptr как последний узел уровня, на котором расположен Node .
--	--

function AddObjectFirst(Node: TTreeNode; const S: string ; Ptr: Pointer): TTreeNode;	Вставляет новый узел с текстом S и объектом Ptr как первый из узлов уровня, на котором находится Node . Индексы последующих узлов увеличиваются на 1.
---	--

function InsertObject(Node: TTreeNode; const S: string ; Ptr: Pointer): TTreeNode;	Вставляет новый узел с текстом S и объектом Ptr сразу после узла Node на тот же уровень. Индексы последующих узлов увеличиваются на 1.
---	---

function AddChildObject(Node: TTreeNode; const S: string ; Ptr: Pointer): TTreeNode;	Добавляет узел с текстом S и объектом Ptr как последний дочерний узла Node .
---	---

function AddChildObjectFirst(Node: TTreeNode; const S: string ; Ptr: Pointer): TTreeNode;	Вставляет новый узел с текстом S и объектом Ptr как первый из дочерних узлов узла Node . Индексы последующих узлов увеличиваются на 1.
--	---

Объект, связанный с некоторым узлом, можно найти с помощью его свойства **Data**. Например, **TreeView1.Items.Item[1].Data**.

Для удаления узлов имеется два метода: **Clear**, очищающий все дерево, и **Delete(Node: TTreeNode)**, удаляющий указанный узел **Node** и все его узлы — потомки. Например, оператор

```
TreeView1.Items.Clear;
```

удалит в нашем примере все узлы, а оператор

```
TreeView1.Items.Delete(TreeView1.Items.Item[1]);
```

удалит узел «цех 1» и его дочерние узлы (если они имеются).

При удалении узлов, связанных с объектами, сами эти объекты не удаляются.

Реорганизация дерева, связанная с созданием или удалением многих узлов, может вызывать неприятное мерцание изображения. Избежать этого можно с помощью методов **BeginUpdate** и **EndUpdate**. Первый из них запрещает перерисовку дерева, а второй — разрешает. Таким образом, изменение структуры дерева может осуществляться по следующей схеме:

```
TreeView1.Items.BeginUpdate;  
<операторы изменения дерева>  
TreeView1.Items.EndUpdate;
```

Если метод **BeginUpdate** применен подряд несколько раз, то перерисовка дерева произойдет только после того, как столько же раз будет применен метод **EndUpdate**.

Среди свойств узлов следует отметить **Count** — число узлов, управляемых данным, т.е. дочерних узлов, их дочерних узлов и т.п. Если значение **Count** узла равно нулю, значит у узла нет дочерних узлов, т.е. он является листом дерева.

Вернемся к свойствам компонента **TreeView**. Важным свойством компонента **TreeView** является **Selected**. Это свойство указывает узел, который выделен пользователем. Пользуясь этим свойством можно запрограммировать операции, которые надо выполнить для выбранного пользователем узла. Если ни один узел не выбран, значение **Selected** равно **nil**. При выделении пользователем нового узла происходят события **OnChanging** (перед изменением выделения). В обработчик события передаются параметры **Node: TTreeNode** — узел, который выделен в данный момент, и **var AllowChange: Boolean** — разрешение на перенос выделения. Если в обработчике задать **AllowChange = false**, то переключение выделения не произойдет.

У компонента **TreeView** имеется свойство **RightClickSelect**, разрешающее (при значении равном **true**) выделение узла щелчком как левой, так и правой кнопкой мыши.

Ряд событий компонента **TreeView** связан с разворачиванием и свертыванием узлов. При разворачивании узла происходят события **OnExpanding** (перед разворачиванием) и **OnExpanded** (после разворачивания). В обработчики обоих событий передается параметр **Node: TTreeNode** — разворачиваемый узел. Кроме того в обработчик **OnExpanding** передается параметр **var AllowExpansion: Boolean**, который можно задать равным **false**, если желательно запретить разворачивание. При свертывании узла происходят события **OnCollapsing** (перед свертыванием) и **OnCollapsed** (после свертывания). Так же, как и в событиях, связанных с разворачиванием, в обработчики передается параметр **Node: TTreeNode** — свертываемый узел, а в обработчик **OnCollapsing** дополнительно передается параметр **var AllowCollapse: Boolean**, разрешающий или запрещающий свертывание.

Свойство **ReadOnly** компонента **TreeView** позволяет запретить пользователю редактировать отображаемые данные. Если редактирование разрешено, то при редактировании возникают события **OnEditing** и **OnEdited**, аналогичные

рассмотренным ранее (в обработчике **OnEditing** параметр **var AllowEdit: Boolean** позволяет запретить редактирование).

Ряд свойств компонента **TreeView**: **ShowButtons**, **ShowLines**, **ShowRoot** отвечают за изображение дерева и позволяют отображать или убирать из него кнопки, показывающие раскрытия узла, линии, связывающие узлы, и корневой узел. Поэкспериментируйте с этими свойствами и увидите, как они влияют на изображение.

Свойство **SortType** позволяет автоматически сортировать ветви и узлы дерева. По умолчанию это свойство равно **stNone**, что означает, что дерево не сортируется. Если установить **SortType** равным **stText**, то узлы будут автоматически сортироваться по алфавиту. Возможно также проводить сортировку по связанным с узлами объектам **Data** (значение **SortType** равно **stData**), одновременно по тексту и объектам **Data** (значение **SortType** равно **stBoth**) или любым иным способом. Для использования этих возможностей сортировки надо написать обработчик события **OnCompare**, в который передаются, в частности, параметры **Node1** и **Node2** — сравниваемые узлы, и по ссылке передается целый параметр **Compare**, в который надо заносить результат сравнения: отрицательное число, если узел **Node1** должен располагаться ранее **Node2**, 0, если эти узлы считаются эквивалентными, и положительное число, если узел **Node1** должен располагаться в дереве после **Node2**. Например, следующий оператор в обработчике события **OnCompare** обеспечивает обратный алфавитный порядок расположения узлов:

```
Compare := - AnsiCompareText(Node1.Text, Node2.Text);
```

События **OnCompare** наступают после задания любого значения **SortType**, отличного от **stNone**, и при изменении пользователем свойств узла (например, при редактировании им надписи узла), если значение **SortType** не равно **stNone**. После сортировки первоначальная последовательность узлов в дереве теряется. Поэтому последующее задание **SortType = stNone** не восстанавливает начальное расположение узлов, но исключает дальнейшую генерацию событий **OnCompare**, т.е. автоматическую перестановку узлов, например, при редактировании их надписей. Если же требуется изменить характер сортировки или провести сортировку с учетом новых созданных узлов, то надо сначала задать значение **SortType = stNone**, а затем задать любое значение **SortType**, отличное от **stNone**. При этом будут сгенерированы новые обращения к обработчику событий **OnCompare**.

Имеются и другие возможности сортировки. Например, метод **AlphaSort** обеспечивает алфавитную последовательность узлов независимо от значения **SortType**, но при отсутствии обработчика событий **OnCompare** (если обработчик есть, то при выполнении метода **AlphaSort** происходит обращение к этому обработчику). Отличие метода **AlphaSort** от задания значения **SortType = stText** заключается в том, что изменение надписей узлов приводит к автоматической пересортировке дерева только при **SortType = stText**.

Мы рассмотрели основные возможности компонента **TreeView**. Компонент **Outline** похож на него. Структура дерева тоже содержится в свойстве **Items** и доступ к отдельным узлам также осуществляется через этот индексный список узлов. Но индексы начинаются с 1. Например, **Outline1.Items[1]** — это узел дерева с индексом 1 (первый узел дерева). Свойство **Items** имеет тип **TOutlineNode**. Его свойства и методы отличаются от свойств и методов типа узлов в **TreeView**. И заполняется структура

дерева иначе: через свойство **Lines** типа **TStrings**. Редактор этого свойства можно вызвать, щелкнув на кнопке с многоточием около свойства **Lines** в окне Инспектора Объектов. Вы попадете в окно, в котором можете записать тексты всех узлов, делая отступы, чтобы выделить уровни узлов. Текст должен выглядеть так, как выше описывалось представление структуры в текстовом файле.

Пиктограммы, сопровождающие изображения узлов, задаются параметрами **PictureOpen** (пиктограмма развернутого узла), **PictureClosed** (пиктограмма свернутого узла), **PictureMinus** (пиктограмма символа «-» около развернутого узла, имеющего наследников), **PicturePlus** (пиктограмма символа «+» узла, имеющего наследников, но не развернутого), **PictureLeaf** (пиктограмма узла, не имеющего наследников — листа дерева). Основное отличие пиктограмм компонента **Outline** заключается в том, что они одинаковы для всех узлов одного типа, тогда как в **TreeView** можно задавать пиктограммы индивидуально для каждого узла.

Программно изменять структуру дерева можно с помощью методов **Add**, **AddObject** (добавление узла в дерево), **Insert**, **InsertObject** (вставка узла в заданную позицию), **AddChild**, **AddChildObject** (вставка дочернего узла), **Delete** (удаление узла).

Индекс выделенного пользователем узла можно определить через свойство **SelectedItem**. Если **SelectedItem = 0**, значит ни один узел не выделен. Текст выделенного узла определяется свойством **Text**: например,

```
Outline1.Items[Outline1.SelectedItem].Text
```

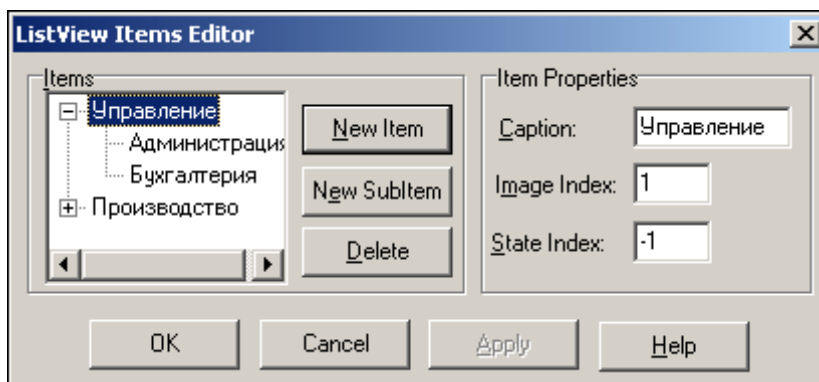
Впрочем, тот же самый текст даст и выражение

```
Outline1.Lines[Outline1.SelectedItem-1]
```

В заключение коротко рассмотрим компонент **ListView**. Он позволяет отображать в стиле Windows 95/98 данные в виде списков, таблиц, крупных и мелких пиктограмм. С подобным отображением все вы сталкиваетесь, раскрывая папки Windows.

Стиль отображения информации определяется свойством **ViewStyle**, которое может устанавливаться в процессе проектирования или программно во время выполнения. Свойство может принимать значения: **vsIcon** — крупные значки, **vsSmallIcon** — мелкие значки, **vsList** — список, **vsReport** — таблица. Что означает каждое из этих значений вы можете посмотреть в любой папке Windows на рабочем столе.

Рис. 8
Окно редактора
списка объектов
компонента
ListView



Если вы хотите предоставить пользователю возможность изменять вид списка, сделайте следующее:

- введите в приложение разделы меню Крупные значки (пусть его имя будет **MIcon**), Мелкие значки (имя **MSmallIcon**), Список (имя **MList**) и Таблица (имя **MReport**)
- установите во всех этих разделах одинаковый отличный от нуля индекс **GroupIndex** и свойства **RadioItem** в **true**
- один из разделов пометьте как **Checked** и в свойстве списка **ViewStyle** установите значение, соответствующее этому разделу
- напишите следующие обработчики щелчков для этих разделов:
- **procedure** TForm1.MIconClick(Sender: TObject);
- **begin**
- ListView1.ViewStyle:=vsIcon;
- MIcon.Checked:=true;
- **end;**
-
- **procedure** TForm1.MSmallIconClick(Sender: TObject);
- **begin**
- ListView1.ViewStyle:=vsSmallIcon;
- MSmallIcon.Checked:=true;
- **end;**
-
- **procedure** TForm1.MListClick(Sender: TObject);
- **begin**
- ListView1.ViewStyle:=vsList;
- MList.Checked:=true;
- **end;**
-
- **procedure** TForm1.MReportClick(Sender: TObject);
- **begin**
- ListView1.ViewStyle:=vsReport;
- MReport.Checked:=true;
- **end;**

Основное свойство компонента, описывающее состав отображаемой информации — **Items**. Во время проектирования оно может быть установлено специальным редактором (рис. 8), вызываемом щелчком на кнопке с многоточием рядом с этим свойством в окне Инспектора Объектов. Оно похоже на окно, описанное для компонента **TreeView** (рис. 7). Точно так же в нем задаются новые узлы кнопкой New Item и дочерние узлы — кнопкой New SubItem. Только смысл дочерних узлов другой: это информация, которая появляется только в режиме **vsReport** — в виде таблицы.

Для каждого узла задается свойство **Caption** — надпись, появляющаяся около пиктограммы. Для дочерних узлов это свойство соответствует надписи, появляющейся, в ячейках таблицы в режиме **vsReport**.

Свойство **Image Index** определяет индекс пиктограммы. Индекс соответствует спискам изображений, хранящимся в отдельных компонентах **ImageList**. Указания на эти компоненты вы можете задать в свойствах **LargeImages** для режима **vsIcon** и **SmallImages** для режимов **vsSmallIcon**, **vsList** и **vsReport**. Индексы начинаются с 0. Если вы укажете

индекс -1 (значение по умолчанию), пиктограммы изображаться не будут. Свойство **State Index** в панели **Item Properties** позволяет добавить вторую пиктограмму в данный объект. Подобная пиктограмма может просто служить дополнительной характеристикой объекта. Индекс, указываемый как **State Index**, соответствует списку изображений, хранящихся в отдельном компоненте **ImageList**, указанном в свойстве **StateImages** компонента **List View**.

Свойство **Columns** определяет список заголовков таблицы в режиме **vsReport** при свойстве **ShowColumnHeaders** (показать заголовки), установленном в **true**. Свойство **Columns** можно задать в процессе проектирования специальным редактором заголовков, вызываемом двойным щелчком на компоненте **List View** или щелчком на кнопке с многоточием рядом со свойством **Columns** в окне Инспектора Объектов.

Свойство **Checkboxes** определяет отображение индикатора с флажком около каждого элемента списка. Индикаторы можно устанавливать программно или их может изменять пользователь во время выполнения. Тогда узнать программно, установлен ли индикатор в некотором элементе **Items[i]**, можно проверкой его свойства **Checked**. Например:

```
for i:=0 to ListView1.Items.Count-1 do
  if ListView1.Items[i].Checked then
    ...;
```

Свойства **HotTrack** и **HotTrackStyles** определяют появление выделения при перемещении курсора над элементом списка и стиль этого выделения. Свойство **HoverTime** (только в Delphi 5) задает в миллисекундах задержку появления такого выделения.

Свойство списка **Selected** определяет выделенный пользователем элемент списка.

Свойство **DragMode** определяет режим перетаскивания элементов списка. Если вы хотите разрешить пользователю перетаскивать пиктограммы по площади списка в режимах **vsIcon** и **vsSmallIcon**, вам надо сделать следующее:

- установить **DragMode = dmAutomatic**
- установить **DragKind = dkDrag**
- написать обработчик события **OnDragOver** списка:
- `Accept := Source = ListView1;`
- написать обработчик события **OnDragDrop** списка:
- `(Source as TListView).Selected.Position:=Point(X,Y);`
- если вы предоставили пользователю возможность выбирать вид отображения списка с помощью описанных ранее разделов меню, то в обработчиках щелчка на **MIcon** и **MSmallIcon** надо написать операторы
- `ListView1.DragMode:=dmAutomatic;`

а в обработчиках щелчка на **MList** и **MReport** надо написать операторы

```
ListView1.DragMode:=dmManual;
```

Метод **Arrange**:

```
procedure Arrange(Code: TListArrangement);
```

позволяет упорядочить пиктограммы в режимах **vsIcon** и **vsSmallIcon**. Параметр **Code** определяет способ упорядочивания:

arAlignBottom	выравнивание вдоль нижнего края области
arAlignLeft	выравнивание вдоль левого края области
arAlignRight	выравнивание вдоль правого края области
arAlignTop	выравнивание вдоль верхнего края области
arDefault	выравнивание по умолчанию (вдоль верхнего края области)
arSnapToGrid	размещение каждой пиктограммы в ближайшем узле сетки

В Delphi 5 введено еще одно свойство **ListView** — **WorkAreas**.

Это свойство определяет рабочую область (прямоугольную рамку), в которой осуществляется выравнивание пиктограмм в режимах **vsIcon** и **vsSmallIcon**. Свойство **WorkAreas** представляет собой индексированный список, аналогичный **Items**, но совершенно независимый от него. Если **WorkAreas** — пустой список (ни одна область в него не добавлена), то упорядочивание пиктограмм производится в пределах всей клиентской области **Listview**. Добавление новой рабочей области осуществляется методом **Add**. Свойство рабочей области **Rect** типа **TRect** определяет границы области. Свойство **Color** определяет цвет рамки, обрамляющей область. Свойство **DisplayName** определяет подпись под рамкой. И рамка, и подпись отображаются только в случае, если свойство списка **ShowWorkAreas** установлено в **true**.

Пример операторов, создающих в площади списка две рабочие области:

```
Listview1.WorkAreas.Add;  
Listview1.WorkAreas[0].Rect := Rect(0,0,  
    Listview1.ClientWidth div 4, Listview1.ClientHeight div 2);  
Listview1.WorkAreas.Items[0].DispiayName := 'Область 0';  
Listview1.WorkAreas.Items[0].Color := clRed;  
Listview1.WorkAreas.Add;  
Listview1.WorkAreas[1].Rect := Rect(Listview1.ClientWidth div 2,  
    0, (Listview1.ClientWidth div 4)*3, Listview1.ClientHeight div 2);  
Listview1.WorkAreas.Items[1].DisplayName := 'Область 1';
```

Упорядочивание пиктограмм происходит в пределах той области, в которой они находятся. Если вы разрешили пользователю перетаскивать пиктограммы, как было описано ранее, то вид выравнивания будет зависеть от расположения пиктограмм. Если одна или несколько из них расположены в пределах одной области, а другие

размещаются вне рабочих областей, то при вызове метода **Arrange** все они расположатся в пределах той области, в которой была хоть одна пиктограмма. Если же пиктограммы были расположены в нескольких рабочих областях, то они будут упорядочиваться в пределах областей их размещения.

Способ упорядочивания определяется соответствующим заданием свойства **SortType**, которое уже рассматривалось нами для компонента **TreeView**.

Тема 13	Оформление области вывода содержимого ЭУП
----------------	--

1.1. Технология обучения на лабораторном занятии

<i>Учебное время: 2 часа.</i>	
<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	1. Выбор компонента для отображения содержимого ЭУП. 2. Встраивание, заполнение и написание кода.
<i>Цель учебного занятия:</i>	Сформировать умение по созданию оглавления ЭУП в среде Delphi.
<i>Педагогические задачи:</i> Организовать работу студентов по: <ul style="list-style-type: none"> Выбору компонента для отображения содержимого ЭУП. Встраиванию, заполнению и написанию кода. 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> Выбирать компонент для отображения содержимого ЭУП. Встраивать, заполнять и писать код.
<i>Методы обучения</i>	Мозговой штурм, диаграмма Венна
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.13. Сообщает, тему, цель учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (65 мин)	2.1. Приводит мозговой штурм с целью выявления знаний о компонентах, имеющих возможность вывода на экран многострочного текста	2.1. Выдвигают названия компонентов.

	<p>2.2. Предлагает вспомнить основные свойства и методы этих компонентов (<i>приложение 2</i>) и составить диаграмму Венна для их сопоставления (<i>приложение 3</i>).</p> <p>2.3. Организует работу студентов по созданию в приложении области для отображения содержимого ЭУП.</p>	<p>2.2. Просматривают, подготовленный материал и заполняют диаграмму Венна.</p> <p>2.3. Оформляют область для отображения содержимого ЭУП.</p>
<p>3-й этап.</p> <p>Заключитель- но-результат- ный (10 мин)</p>	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> • Доработать интерфейс ЭУП; • Подготовить тестовые задания для ЭУП. • Повторить методы и события компонента TreeView, в котором при выборе того или иного пункта соответствующее содержимое будет отображаться в компоненте Memo или RichEdit; а также каким образом можно загрузить текстовые данные из файла в компоненты Memo или RichEdit. 	<p>Задают уточняющие вопросы.</p>

Приложение 1

Тема: Разработка оглавления ЭУП

Цель занятия: сформировать умения по созданию оглавления ЭУП в среде Delphi.

План занятия:

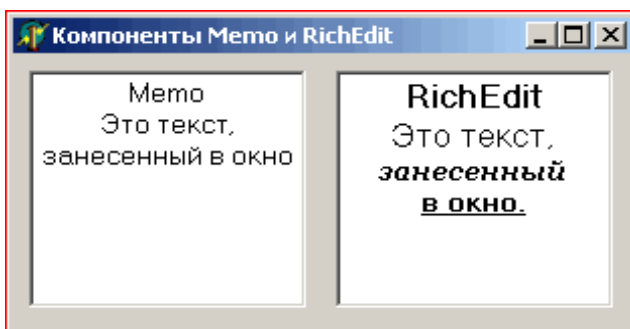
1. Выбор компонента для отображения содержимого ЭУП.
2. Встраивание, заполнение и написание кода.

Приложение 2

Многострочные окна редактирования Memo и RichEdit

Компоненты **Memo** и **RichEdit** (см. пример на рис. 1) являются окнами редактирования многострочного текста. Они так же, как и окно **Edit**, снабжены многими функциями, свойственными большинству редакторов. В них предусмотрены типичные комбинации «горячих» клавиш: **Ctrl-C** — копирование выделенного текста в буфер обмена Clipboard (команда Copy), **Ctrl-X** — вырезание выделенного текста в буфер Clipboard (команда Cut), **Ctrl-V** — вставка текста из буфера Clipboard в позицию курсора (команда Paste), **Ctrl-Z** — отмена последней команды редактирования.

Рис. 1
Примеры
компонентов
Memo и
RichEdit



В компоненте **Memo** формат (шрифт, его атрибуты, выравнивание) одинаков для всего текста и определяется свойством **Font**. Если вы сохраните в файле текст, введенный или отредактированный пользователем, то будет создан текстовый файл, содержащий только символы и не содержащий элементов форматирования. При последующем чтении этого файла в **Memo** формат будет определяться текущим состоянием свойства **Font** компонента **Memo**, а не тем, в каком формате ранее вводился текст.

Компонент **RichEdit** работает с текстом в обогащенном формате RTF. При желании изменить атрибуты вновь вводимого фрагмента текста вы можете задать свойство **SelAttributes**. Это свойство типа **TTextAttributes**, которое в свою очередь имеет подсвойства: **Color** (цвет), **Name** (имя шрифта), **Size** (размер), **Style** (стиль) и ряд других. Например, введите на форму компонент **RichEdit**, диалог выбора шрифта **FontDialog** и кнопку **Button**, которая позволит пользователю менять атрибуты текста. В обработчик щелчка кнопки можно было бы ввести текст:

```
if FontDialog1.Execute then
  with RichEdit1.SelAttributes do
    begin
      Color:=FontDialog1.Font.Color;
      Name:=FontDialog1.Font.Name;
      Size:=FontDialog1.Font.Size;
      Style:=FontDialog1.Font.Style;
    end;
  RichEdit1.SetFocus;
```

В приведенном коде присваивается поочередно значение каждого свойства. Но этот текст можно кардинально сократить, воспользовавшись тем, что объекты **SelAttributes** и **Font** совместимы по типу. Поэтому можно присвоить сразу все свойства одного объекта другому:

```
if FontDialog1.Execute then
  RichEdit1.SelAttributes.Assign(FontDialog1.Font);
RichEdit1.SetFocus;
```

Запустите приложение и увидите, что вы можете менять атрибуты текста, выполняя отдельные фрагменты различными шрифтами, размерами, цветами, стилями. Устанавливаемые атрибуты влияют на выделенный текст или, если ничего не выделено, то на атрибуты нового текста, вводимого начиная с текущей позиции курсора (позиция курсора определяется свойством **SelStart**).

В компоненте имеется также свойство **DefAttributes**, содержащее атрибуты по умолчанию. Эти атрибуты действуют до того момента, когда изменяются атрибуты в свойстве **SelAttributes**.

Но значения атрибутов в **DefAttributes** сохраняются и в любой момент эти значения могут быть методом **Assign** присвоены атрибутам свойства **SelAttributes**, чтобы вернуться к прежнему стилю.

Свойство **DefAttributes** доступно только во время выполнения. Поэтому его атрибуты при необходимости можно задавать, например, в обработчике события **OnCreate**.

За выравнивание, отступы и т.д. в пределах текущего абзаца отвечает свойство **Paragraph** типа **TParaAttributes**. Этот тип в свою очередь имеет ряд свойств:

Alignment	Определяет выравнивание текста. Может принимать значения taLeftJustify (влево), taCenter (по центру) или taRightJustify (вправо).
FirstIndent	Число пикселей отступа красной строки.
Numbering	Управляет вставкой маркеров, как в списках. Может принимать значения nsNone — отсутствие маркеров, nsBullet — маркеры ставятся.
LeftIndent	Отступ в пикселях от левого поля.
RightIndent	Отступ в пикселях от правого поля.
TabCount	Количество позиций табуляции.
Tab	Значения позиций табуляции в пикселях.

Значения подсвойств свойства **Paragraph** можно задавать только в процессе выполнения приложения, например, в событии создания формы или при нажатии какой-нибудь кнопки. Значения подсвойств свойства **Paragraph** относятся к тому абзацу, в котором находится курсор. Например, каждый из следующих операторов осуществит соответственное выравнивание текущего абзаца:

```
RichEdit1.Paragraph.Alignment:=taLeftJustify; // Влево
RichEdit1.Paragraph.Alignment:=taCenter;      // По центру
RichEdit1.Paragraph.Alignment:=taRightJustify; // Вправо
```

Следующий оператор приведет к тому, что текущий абзац будет отображаться как список, т.е. с маркерами:

```
RichEdit1.Paragraph.Numbering:=nsBullet;
```

Уничтожение списка в текущем абзаце осуществляется оператором

```
RichEdit1.Paragraph.Numbering:=nsNone;
```

В целом, если с помощью компонента **ActionList** определено некоторое действие ввода и уничтожения списка, названное **ABullet**, то операторы обработки соответствующего действия могут иметь вид:

```
if (ABullet.Checked) then  
  RichEdit1.Paragraph.Numbering:=nsNone  
else RichEdit1.Paragraph.Numbering :=nsBullet;  
ABullet.Checked:=not ABullet.Checked;
```

Они обеспечивают переключение соответствующей быстрой кнопки и раздела меню из нажатого состояния (отмеченного) в ненажатое с соответствующим изменением вида текущего абзаца.

Свойства **TabCount** и **Tab** имеют смысл при вводе текста только при значении свойства компонента **WantTabs = true**. Это свойство разрешает пользователю вводить в текст символ табуляции. Если **WantTabs = false**, то нажатие пользователем клавиши табуляции просто переключит фокус на очередной компонент и символ табуляции в текст не введется.

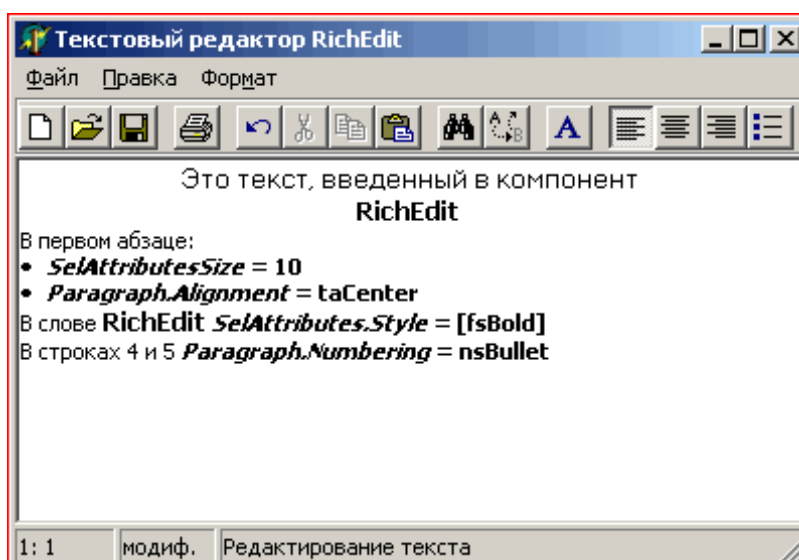
Мы рассмотрели основные отличия **Memo** и **RichEdit**. Теперь остановимся на общих свойствах этих окон редактирования.

Свойства **Alignment** и **WordWrap** имеют тот же смысл, что, например, в метках, и определяют выравнивание текста и допустимость переноса длинных строк. Установка свойства **ReadOnly** в **true** задает текст только для чтения. Свойство **MaxLength** определяет максимальную длину вводимого текста. Если **MaxLength = 0**, то длина текста не ограничена. Свойства **WantReturns** и **WantTab** определяют допустимость ввода пользователем в текст символов перевода строки и табуляции.

Свойство **ScrollBars** определяет наличие полос прокрутки текста в окне. По умолчанию **ScrollBars = ssNone**, что означает их отсутствие. Пользователь может в этом случае перемещаться по тексту только с помощью курсора. Можно задать свойству **ScrollBars** значения **ssHorizontal**, **ssVertical** или **ssBoth**, что будет соответственно означать наличие горизонтальной, вертикальной или обеих полос прокрутки.

В качестве примера на рис. 2 приведен пример текстового редактора, использующего описанные выше свойства компонента **RichEdit**. Текст в окне редактора частично поясняет атрибуты шрифта, использованные при его написании.

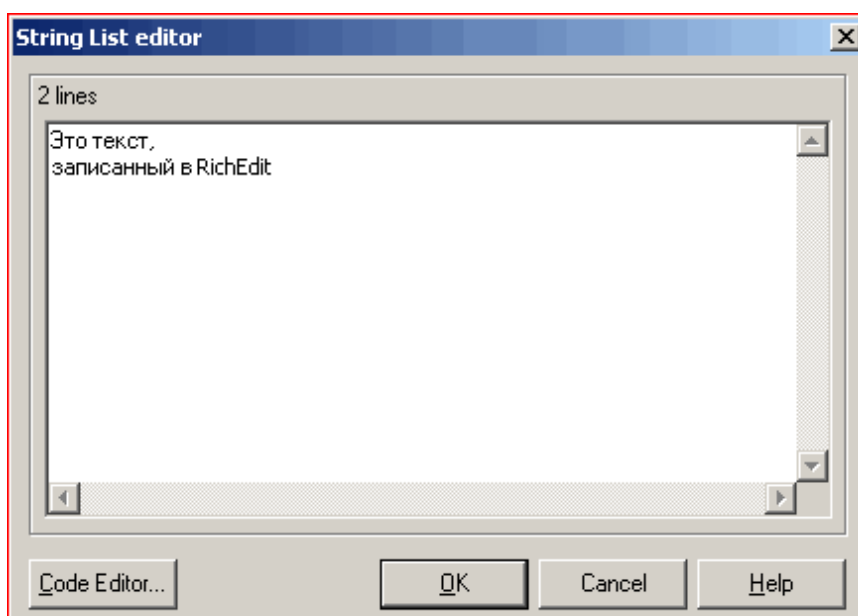
Рис. 2
Пример редактора на основе компонента RichEdit



Основное свойство окон **Memo** и **RichEdit** — **Lines**, содержащее текст окна в виде списка строк и имеющее тип **TStrings**. Начальное значение текста можно установить в процессе проектирования, нажав кнопку с многоточием около свойства **Lines** в окне Инспектора Объектов. Перед вами откроется окно редактирования списков строк, представленное на рис. 3. Вы можете редактировать или вводить текст непосредственно в этом окне, или нажать кнопку **CodeEditor** и работать в обычном окне Редактора Кода. В этом случае, завершив работу с текстом, выберите из контекстного меню, всплывающего при щелчке правой кнопкой мыши, команду **Close Page** и ответьте утвердительно на вопрос, хотите ли вы сохранить текст в соответствующем свойстве окна редактирования.

В Delphi 1 аналогичное окно имеет еще кнопки, позволяющие загрузить текст из файла или сохранить его в файле.

Рис. 3
Окно редактирования списков строк



Во время выполнения приложения вы можете заносить текст в окно редактирования с помощью методов свойства **Lines** типа **TStrings**. Этот тип широко используется в свойствах многих компонентов и его описание вы можете найти в во встроенной справке Delphi. Здесь коротко укажем только на его основные свойства и методы, используемые в свойстве **Lines**.

Весь текст, представленный одной строкой типа **String**, внутри которой используются разделители типа символов возврата каретки и перевода строки, содержится в свойстве **Text**.

Доступ к отдельной строке текста вы можете получить с помощью свойства **Strings[Index: Integer]**. Индексы, как и везде в Delphi, начинаются с **0**. Так что **Memo1.Lines.Strings[0]** — это текст первой строки. Учтите, что если окно редактирования изменяется в размерах при работе с приложением и свойство **WordWrap = true**, то индексы строк будут изменяться при переносах строк, так что в этих случаях индекс мало о чем говорит.

Свойство только для чтения **Count** указывает число строк в тексте.

Для очистки текста в окне надо выполнить процедуру **Clear**. Этот метод относится к самому окну, а не к его свойству **Lines**.

Для занесения новой строки в конец текста окна редактирования можно воспользоваться методами **Add** или **Append** свойства **Lines**. Для загрузки текста из файла применяется метод **LoadFromFile**. Сохранение текста в файле осуществляется методом **SaveToFile**.

Пусть, например, в вашем приложении имеется окно редактирования **Edit1**, в котором пользователь вводит имя сотрудника, и есть кнопка, при щелчке на которой в окно **Memo1** должна занестись шапка характеристики этого сотрудника, после чего пользователь может заполнить текст характеристики.

Обработчик щелчка на кнопке может иметь вид:

```
Memo1.Clear;  
Memo1.Lines.Add ('Х А Р А К Т Е Р И С Т И К А');  
Memo1.Lines.Add ('Сотрудник '+Edit1.Text) ;  
Memo1.SetFocus;
```

Загрузка в окно **Memo1** текста из файла (например, хранящейся в файле характеристики сотрудника) может осуществляться командой

```
Memo1.Lines.LoadFromFile ('text.txt');
```

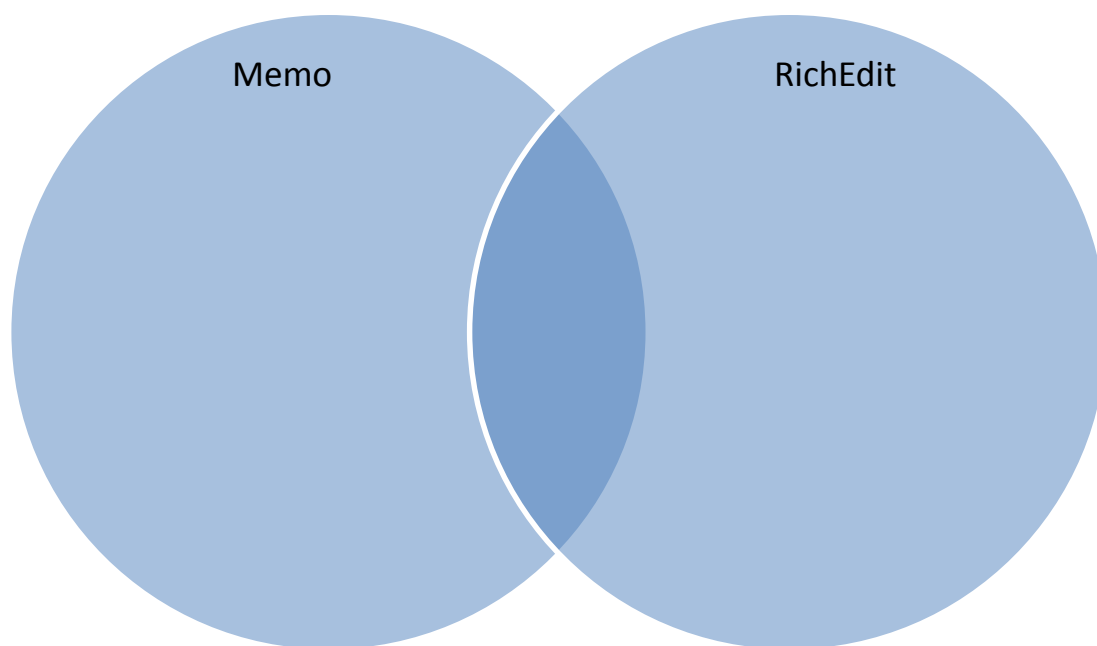
Сохранение текста в файле может осуществляться командой

```
Memo1.Lines.SaveToFile ('text.txt');
```

Свойство **SelStart** компонентов **Memo** и **RichEdit** указывает позицию курсора в тексте или начало выделенного пользователем текста. Свойство **CaretPos** указывает на запись, поле **X** которой содержит индекс символа в строке, перед которым расположен курсор, а поле **Y** — индекс строки, в которой находится курсор (встроенная справка Delphi

утверждает другое — что свойство **CaretPos** содержит координаты курсора в пикселях; но, к счастью, это не так). Таким образом, учитывая, что индексы начинаются с 0, значения **Memo1.CaretPos.Y+1** и **Memo1.CaretPos.X+1** определяют соответственно номер строки и символа в ней, перед которым расположен курсор. В редакторе на рис. 2 именно эти значения (только не для **Memo**, а для **RichEdit**) использованы, чтобы отображать в строке состояния позицию курсора.

Приложение 3



Тема 14	Составление тестовых заданий для ЭУП
----------------	---

1.1. Технология обучения на лабораторном занятии

<i>Учебное время: 2 часа.</i>	
<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Выбор метода создания и внедрения в ЭУП тестовых заданий. 2. Создание и внедрение тестовых заданий в приложение.
<i>Цель учебного занятия:</i>	Сформировать умения по созданию и внедрению тестовых заданий в ЭУП в среде Delphi.
<i>Педагогические задачи:</i> Организовать работу студентов по: <ul style="list-style-type: none"> • выбору метода создания и внедрения в ЭУП тестовых заданий. • созданию и внедрению тестовых заданий в приложение. 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • Выполнять выбор метода создания и внедрения в ЭУП тестовых заданий. • Создавать и внедрять тестовые задания в приложение
<i>Методы обучения</i>	Таблица ЗХУ
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.14. Сообщает, тему, цель учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап.	2.1. Предлагает заполнить таблицу ЗХУ, в котором должны ответить на вопросы: Что вы знаете о способах создания тестовых	2.1. Заполняют таблицу ЗХУ.

Основная часть (65 мин)	<p>заданий в среде Delphi или внедрении сторонних тестовых программ в приложения Delphi? Что бы вы хотели узнать об этом? (<i>приложение 2</i>).</p> <p>2.2. Проводит мозговой штурм с целью выявления знаний о методах создания и внедрения в приложение Delphi тестовых заданий</p> <p>2.3. Приводит пример оформления тестового задания в среде Delphi (<i>приложение 3</i>)</p> <p>2.4. Организует работу студентов по созданию и внедрению тестовых заданий в приложения Delphi.</p>	<p>2.2. Высказывают предположения.</p> <p>2.2.Просматривают, подготовленный материал, анализируют, делают выводы.</p> <p>2.3.Работают в среде Delphi над созданием тестовых заданий и их внедрению в приложение..</p>
3-й этап. Заключитель- но-результат- ный (10 мин)	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> • Доработать интерфейс ЭУП; • Повторить методы и события компонента TreeView, в котором при выборе того или иного пункта соответствующее содержимое будет отображаться в компоненте Memo или RichEdit; а также каким образом можно загрузить текстовые данные из файла в компоненты Memo или RichEdit. 	Задают уточняющие вопросы.

Приложение 1

Тема: Составление тестовых заданий для ЭУП

Цель занятия: сформировать умения по созданию и внедрению тестовых заданий в ЭУП в среде Delphi.

План занятия:

1. Выбор метода создания и внедрения в ЭУП тестовых заданий.
2. Создание и внедрение тестовых заданий в приложение.

Приложение 2

Таблица 3/Х/У

Знаем	Хотим узнать	Узнали

Приложение 3

Режим тестирования.

Переход в режим тестирования осуществляется при выборе параграфа, который характеризуется типом «Тестирование» (📝). После нажатия на кнопку «Начать тестирование» пользователь переходит в режим ответа на вопросы (рис. 4).

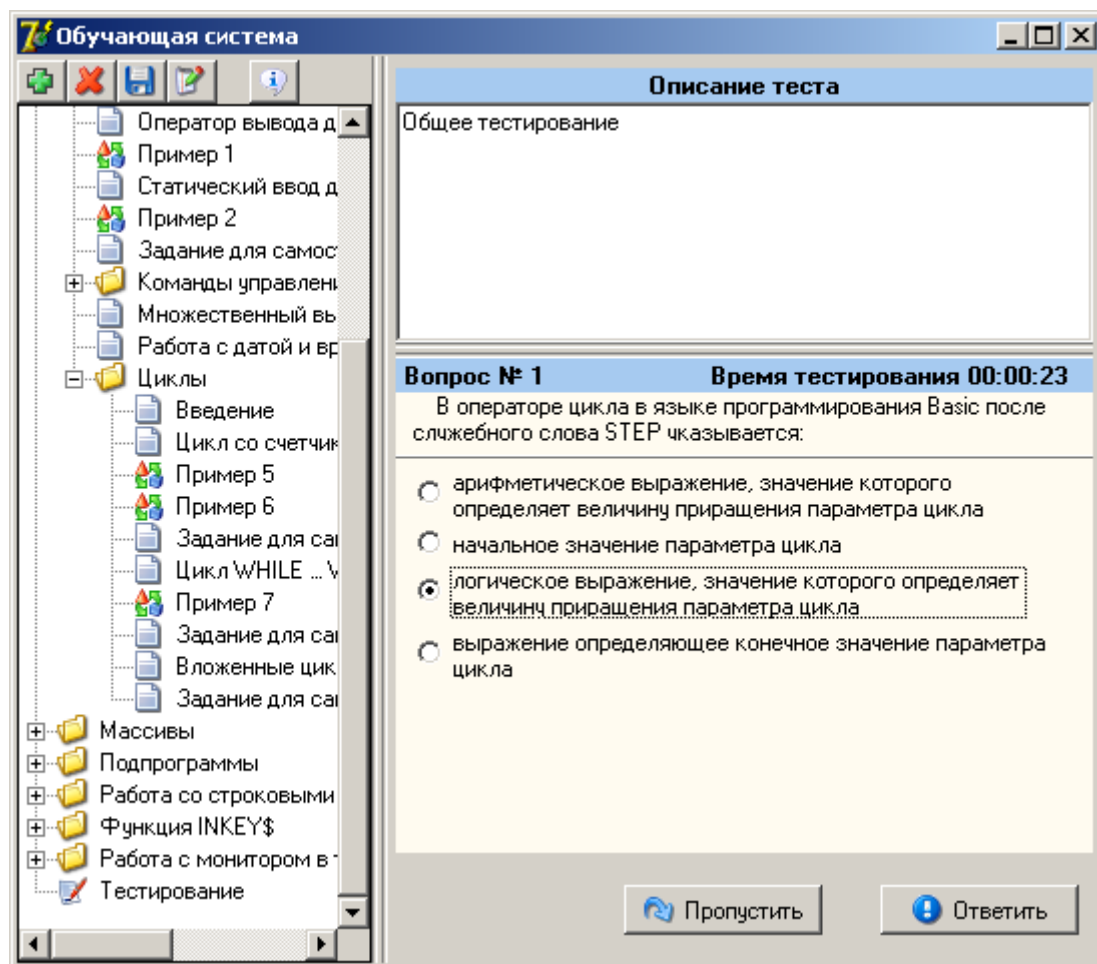


Рис. 4. Режим тестирования.

В верхней части области вопроса отображается его номер и общее время теста.

На каждый вопрос предоставляются варианты ответов, из которых необходимо выбрать один и нажать кнопку «Ответить» или «Пропустить» (если пользователь не знает ответа на вопрос).

По окончании тестирования выводится окно результата контроля (рис. 5)

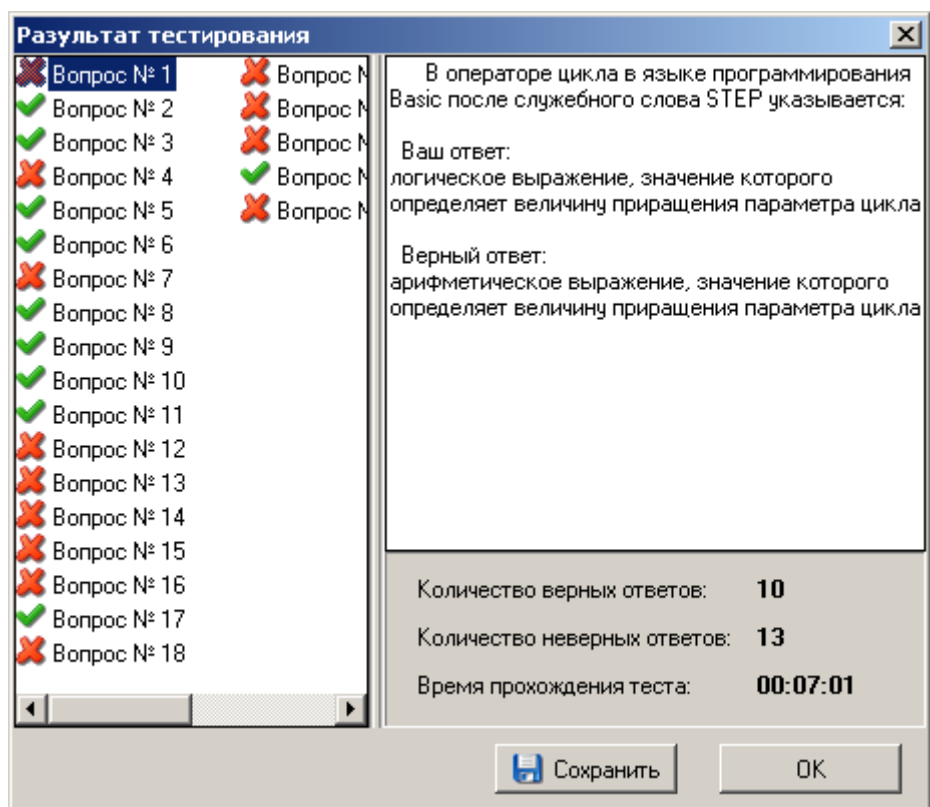


Рис. 5. Окно результата тестирования.

В этом окне пользователю предоставляется следующая информация:

1. Количество верных ответов;
2. Количество неверных ответов;
3. Время прохождения контроля;
4. Перечень всех вопросов с признаком верности ответа;
5. Выбранный вариант ответа на вопрос;
6. Верный вариант ответа на вопрос.

Тема 15	Написание кода программы ЭУП
----------------	-------------------------------------

1.1. Технология обучения на лабораторном занятии

<i>Учебное время:</i> 4 часа.	
<i>Форма учебного занятия:</i> лабораторная работа	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Код программы, поддерживающий работу содержания ЭУП. 2. Код программы, поддерживающий работу кнопок панели инструментов.
<i>Цель учебного занятия:</i>	Сформировать умения по написанию кода для организации работы ЭУП в среде Delphi.
<i>Педагогические задачи:</i> Организовать работу студентов по: <ul style="list-style-type: none"> • Составлению кода программы, поддерживающий работу содержания ЭУП. • Составлению кода программы, поддерживающий работу кнопок панели инструментов. 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • Составлять код программы, поддерживающий работу содержания ЭУП. • Составлять код программы, поддерживающий работу кнопок панели инструментов.
<i>Методы обучения</i>	Обучение сообща
<i>Средства обучения</i>	Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение
<i>Формы организации учебной деятельности</i>	Групповая, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.15. Сообщает, тему, цель учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап.	2.1. «Обучение сообща». Делит студентов на группы и организует обсуждение общих подходов по написанию кода программы.	2.1. В группах обсуждают,

Основная часть (65 мин)	2.2. После обсуждения организует индивидуальную работу студентов по написанию кода в приложении Delphi за компьютерами.	высказывают идеи по написанию кода. 2.2.Работают в среде Delphi.
3-й этап. Заключительно-результативный (10 мин)	3.1.Дает и разъясняет задания для домашней работы: <ul style="list-style-type: none"> Доработать код программы; 	Задают уточняющие вопросы.

Приложение 1

Тема: Написание кода программы ЭУП

Цель занятия: сформировать умения по написанию кода для организации работы ЭУП в среде Delphi.

План занятия:

- Код программы, поддерживающий работу кнопок панели инструментов.

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.1. Сообщает, тему, цель учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (65 мин)	2.1. Напоминает кнопки панели инструментов и действия, выполняемые с их помощью (<i>приложение 2</i>). 2.2. Организует работу студентов по написанию кода в приложении Delphi.	2.1. Высказывают предположения. 2.2.Работают в среде Delphi.
3-й этап. Заключительно-результативный (10 мин)	3.1.Дает и разъясняет задания для домашней работы: <ul style="list-style-type: none"> Доработать код программы; 	Задают уточняющие вопросы.

Приложение 1

Тема: Написание кода программы ЭУП

Цель занятия: сформировать умения по написанию кода для организации работы ЭУП в среде Delphi.


План занятия:

2. Код программы, поддерживающий работу содержания ЭУП.


Приложение 2

ПАНЕЛЬ ИНСТРУМЕНТОВ

На панели инструментов находятся следующие кнопки:

 или  – показывают или скрывают оглавление;


 – сохраняет текущее содержание в файл;

 – выводит меню с выбором:


 – добавляет новый раздел;


 – добавляет новую тему к выделенному разделу;

 – удаляет выделенный элемент;

 – редактирование текущего элемента;

 – вызывает диалоговое окно печати;

 – вызывает окно о программе;

 – выход из программы.

Раздел IV. Технологии создания электронных учебных материалов на основе Flash -технологии

Тема 16	Создание интерактивных учебных пособий во Flash: технология создания кнопки и использования языка ActionScript для работы с ней
----------------	--

1.1. Технология обучения на лабораторном занятии

<i>Учебное время: 2 часа.</i>	
<i>Форма учебного занятия: лабораторная работа</i>	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Технология создания кнопки во Flash и использования языка ActionScript 2.0. для работы с ней. 2. Добавление кнопки из библиотеки стандартных элементов, поставляемой вместе с Flash; 3. Создание собственной клип-кнопки.
<i>Цель учебного занятия:</i>	освоить технологию создания кнопки во Flash и использования языка ActionScript 2.0. для работы с ней, добавления кнопки из библиотеки стандартных элементов, поставляемой вместе с Flash; научиться создавать собственную клип-кнопку.
<i>Педагогические задачи:</i> Организовать работу студентов по: <ul style="list-style-type: none"> • созданию кнопки во Flash и использования языка ActionScript 2.0. для работы с ней. • добавлению кнопки из библиотеки стандартных элементов, поставляемой вместе с Flash; • созданию собственной клип-кнопки. 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • создавать кнопки во Flash и использовать язык ActionScript 2.0. для работы с ней. • Добавлять кнопки из библиотеки стандартных элементов, поставляемой вместе с Flash; • Создавать собственные клип-кнопки.
<i>Методы обучения</i>	Слайдовая презентация
<i>Средства обучения</i>	персональные компьютеры с установленной программой Adobe Flash, печатная версия задания
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.2. Сообщает, тему, цель учебного занятия и план его проведения. (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (65 мин)	<p>2.1. Показывает с помощью слайдов презентации возможность использования кнопок из библиотеки (<i>приложение 2</i>).</p> <p>2.2. Организует работу студентов по созданию кнопки во Flash и использованию языка ActionScript 2.0. для работы с ней (<i>приложение 3</i>).</p> <p>2.3. Раздает задания для индивидуального выполнения. (<i>приложение 4</i>) и разъясняет форму отчета по лабораторной работе (<i>приложение 5</i>).</p>	<p>2.1. Просматривают, уточняют, анализируют.</p> <p>2.2. Выполняют задания согласно приложению 3.</p> <p>2.3. Выполняют индивидуальные задания и подготавливают отчет о проделанной работе..</p>
3-й этап. Заключительно-результативный (10 мин)	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> Изучить источники Интернет по работе с Flash (<i>приложение 6</i>). 	Задают уточняющие вопросы.

Приложение 1

Тема: Создание интерактивных учебных пособий во Flash: технология создания кнопки и использования языка ActionScript для работы с ней.

Цель занятия: освоить технологию создания кнопки во Flash и использования языка ActionScript 2.0. для работы с ней, добавления кнопки из библиотеки стандартных элементов, поставляемой вместе с Flash; научиться создавать собственную клип-кнопку

План занятия:

1. Технология создания кнопки во Flash и использования языка ActionScript 2.0. для работы с ней.
2. Добавление кнопки из библиотеки стандартных элементов, поставляемой вместе с Flash;
3. Создание собственной клип-кнопки.

Кнопки из стандартной библиотеки компонентов Flash



Кнопка важный элемент любого интерактивного ресурса. Разрабатывая ЦОР во Flash можно использовать кнопки из стандартной библиотеки компонентов Flash. Для этого нужно:

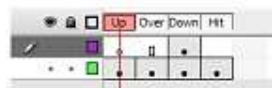
1. Выбрать в меню *Window* → *Common Libraries* → *Buttons* (*Окно - Общие библиотеки - Buttons*).

2. Открыть папку "Arcade buttons", и перетащить кнопку "arcade button - orange" на рабочее пространство вашего стола.



Большинство кнопок в стандартной библиотеке являются уже готовыми и рабочими. Всё что останется - это добавить действия для кнопки.

В кнопке четыре кадра, которые соответствуют четырём её состояниям:



- ▶ **Up** - первоначальное (пассивное) состояние кнопки. Кнопка находится в этом состоянии, когда курсор мышки находится за её пределами.
- ▶ **Over** - это состояние соответствует виду кнопки, когда курсор находится над кнопкой. Многие в этом состоянии используют эффекты "подсветки", "приподнимания" и т.п. чтобы показать, что кнопка "готова к нажатию".
- ▶ **Down** - состояние кнопки, когда она нажата. В этом состоянии используются эффекты "вдавливания", более тёмной "подсветки" и т.п.
- ▶ **Hit** - это состояние используется для области кнопки, которая будет реагировать на курсор. Контур, нарисованный в этом состоянии, будет обозначать активную область кнопки. Например, в символе *arcade button - orange*, активной является область кнопки, и окружающего её ободка - тень не входит в активную область (хотя тоже присутствует в символе кнопки), поэтому при нажатии на неё ничего не происходит.

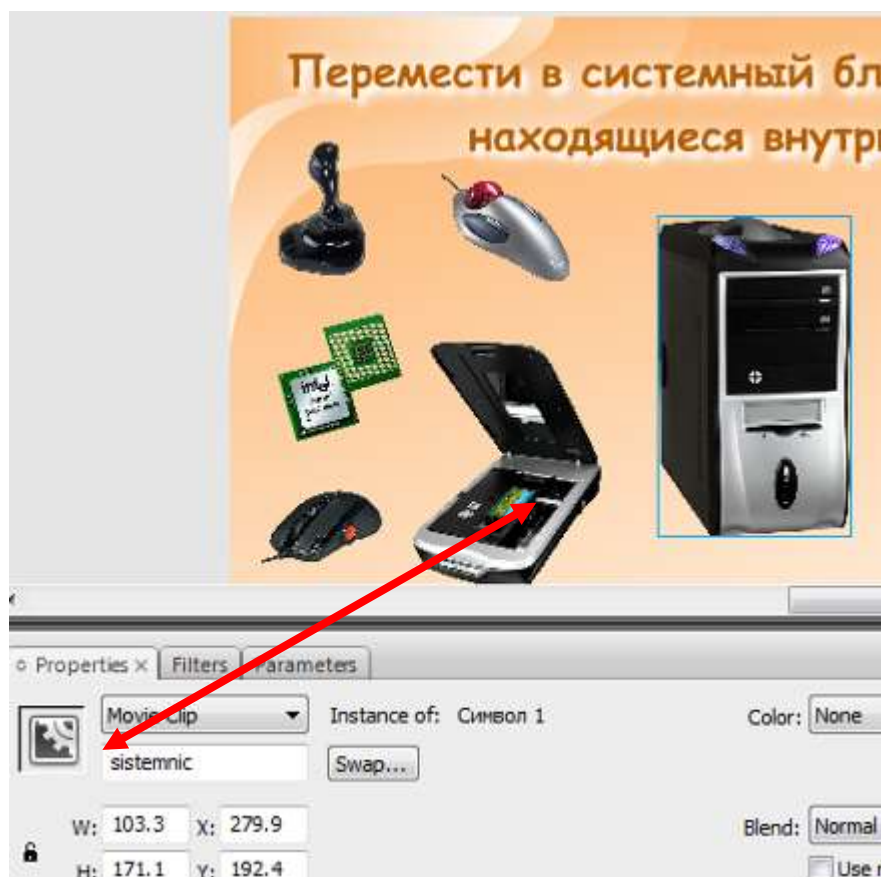


В эти четыре состояния можно помещать всё что угодно, включая другие символы (даже объекты MovieClip), тем самым создавая всевозможные кнопки. Цель нашего занятия - научиться создать клип-кнопку самостоятельно. Помимо этого мы познакомимся с одним из возможных вариантов моментальной проверки правильности перемещения объектов.

Приложение 3

Технология создания клипа-кнопки

1. Открываем новый документ Adobe Flash CS3.
2. Называем слой именем «Фон» и устанавливаем цвет фона по желанию.
3. Введите в слое «Фон» статический текст *Перемести в системный блок устройства, находящиеся внутри него.*
4. Импортируем в библиотеку графические изображения устройств компьютера, которые можно отрисовать, а можно использовать и исходные изображения с указанием источника. Обратите внимание, что фон картинок должен быть прозрачным.
5. Создаем новый слой с именем «Клипы» и размещаем на нем графические изображения.
6. Изображение системного блока преобразовываем в символ – тип клип и задаем имя инстанции *sistemnic*.



7. Изображения, входящие в состав системного блока преобразовываем в символ – тип клип. Двойным щелчком заходим внутрь клипа и вновь преобразовываем изображение в символ – тип кнопка и для каждой кнопки прописываем **код свободного перемещения с исчезновением**:

```
on (press) //когда кнопка мыши нажата
{
    this.startDrag(true); //начинаем перемещать текущий объект, ключевое слово
    true в данном случае задает привязку центра объекта к курсору
}
on (release) //когда кнопка мыши отпущена
{
    this.stopDrag(); //заканчиваем перемещать текущий объект
    if (this._droptarget == "/системник") //осуществляем проверку условия - если
    есть столкновение с объектом по имени системник, то
    {
        this._visible = 0; //текущий объект становится невидимым
    } //конец выполнения условного оператора
}
```

Комментарий к представленному выше скрипту

Ключевое слово `this` применяется вместо указания имени перетаскиваемого объекта - этот код более универсальный, его можно копировать и не оглядываться на сам объект, он подойдет для любого объекта.

Свойство `_droptarget`, доступное только для чтения, возвращает абсолютный путь к мувиклипу (объекту `системник`), на который был "брошен" (перетащен) любой

мувиклип (`this`), в слэш-нотации. Свойство `_droptarget` всегда возвращает путь, начинающийся со слэша (`/`).

Операторы языка ActionScript

Оператор	Назначение
<code>+</code>	Сложение
<code>-</code>	Вычитание
<code><</code>	Меньше
<code><=</code>	Не больше
<code>></code>	Больше
<code>>=</code>	Не меньше
<code>==</code>	Равно
<code>!=</code>	Не равно
<code>=</code>	Присваивание

Свойство `_visible` определяет видимость объекта.

Как в обычных языках программирования, так и в языках сценариев предусмотрены структуры, позволяющие реализовывать механизм сравнений и записывать ряд альтернативных вариантов действий, и ActionScript в этом отношении не является исключением. В языке ActionScript существуют три основные структуры условных инструкций: `if...`, `if...else` и `if... else if`. Каждая из них служит некой вполне определенной цели и может применяться в соответствующих ситуациях.

В приведенном выше сценарии условный оператор `if...` (если) предполагает исполнение единственной последовательности действий, не имеющей альтернативы, при выполнении определенного условия: *если есть столкновение объекта с системным блоком, то он исчезает*.

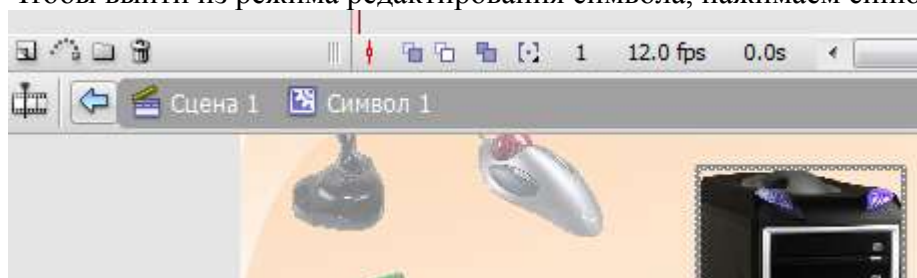
А что же будет при перемещении объектов без столкновения с системным блоком? В этом случае при использовании конструкции `if...` не произойдет вообще ничего. Если вы хотите, чтобы при невыполнении заданного условия сценарий производил те или иные действия, следует добавить конструкцию `else` (иначе).

Как и во многих других объектно-ориентированных языках, в ActionScript для оформления операций, которые нужно выполнить при выполнении указанного условия, применяются фигурные скобки `{ }`, а само условие записывается в круглые скобки `()`, например, `if (this._droptarget == "/sistemnic")`

Структура `if...else` — позволяет записать в сценарии необходимость выполнить одну из двух последовательностей операций. С данной конструкцией мы познакомимся более подробно в следующих занятиях мастер-класса.

Продолжим...

Чтобы выйти из режима редактирования символа, нажимаем синюю стрелку:



8. Изображения, не входящие в состав системного блока преобразовываем в символ – тип клип. Двойным щелчком заходим внутрь клипа и вновь преобразовываем изображение в символ – тип кнопка и для каждой кнопки прописываем код свободного перемещения:

```
on (press) { //Кнопка мыши нажата
StartDrag (this); } //Начать перемещение
on (release) { //Кнопка мыши отпущена
StopDrag (); } //Закончить перемещение
```

Таким образом, устройства, входящие в состав системного блока, при перемещении на него будут исчезать, в отличие от устройств не входящих в системный блок.

9. В слое «Фон» прописываем код разворачивания клипа на весь экран и отключения меню.

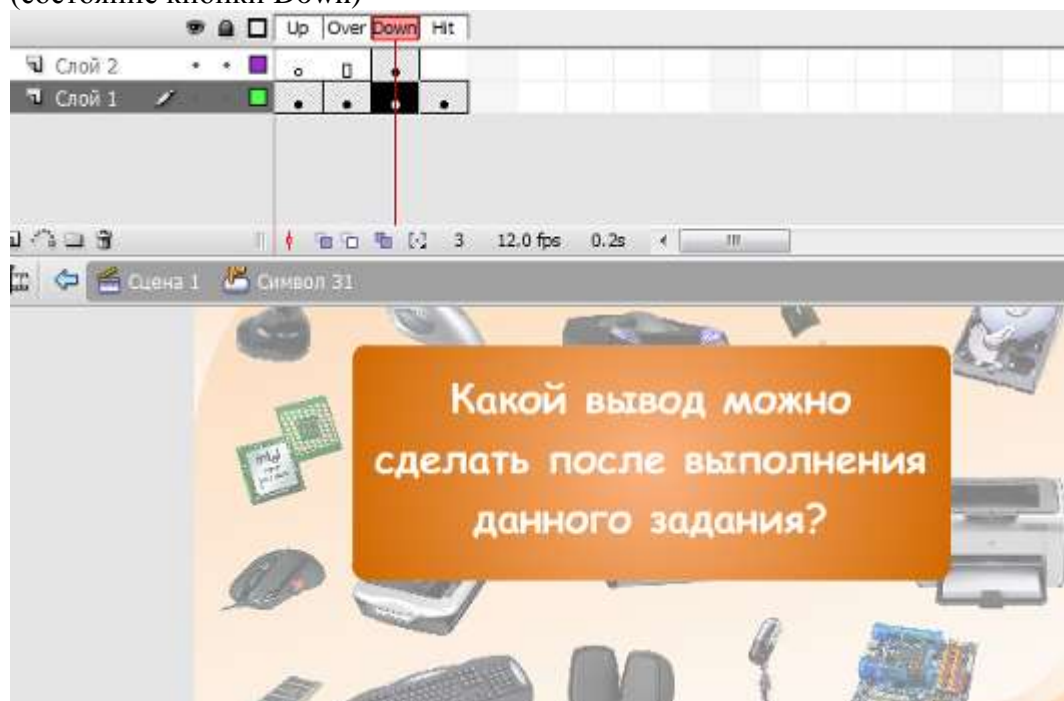
```
fscommand("showmenu", "false"); //Отключить большую часть выпадающего
меню, где есть и пункт «Проиграть сначала», которым ученик может
запустить, например, заново тест.
```

```
fscommand("fullscreen", "true"); //Развернуть ролик на весь экран
```

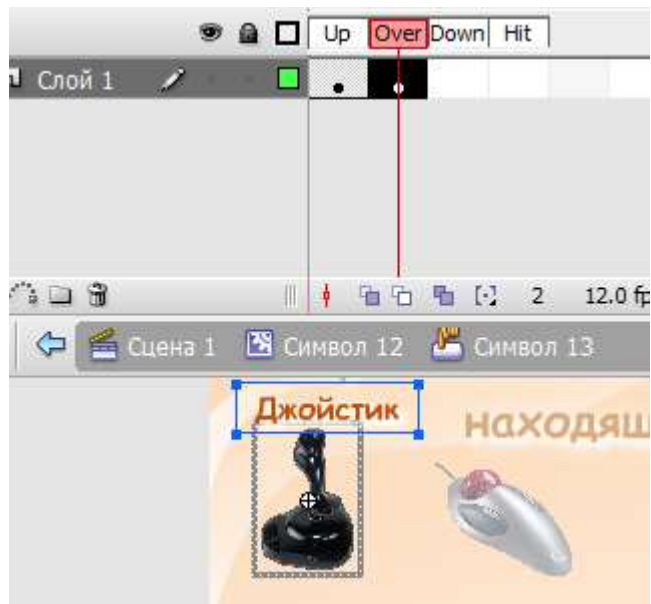
10. В слое «Клипы» добавляем кнопку выхода со следующим кодом:

```
on (release) { //Кнопка мыши отпущена
fscommand("quit"); //Выход
}
```

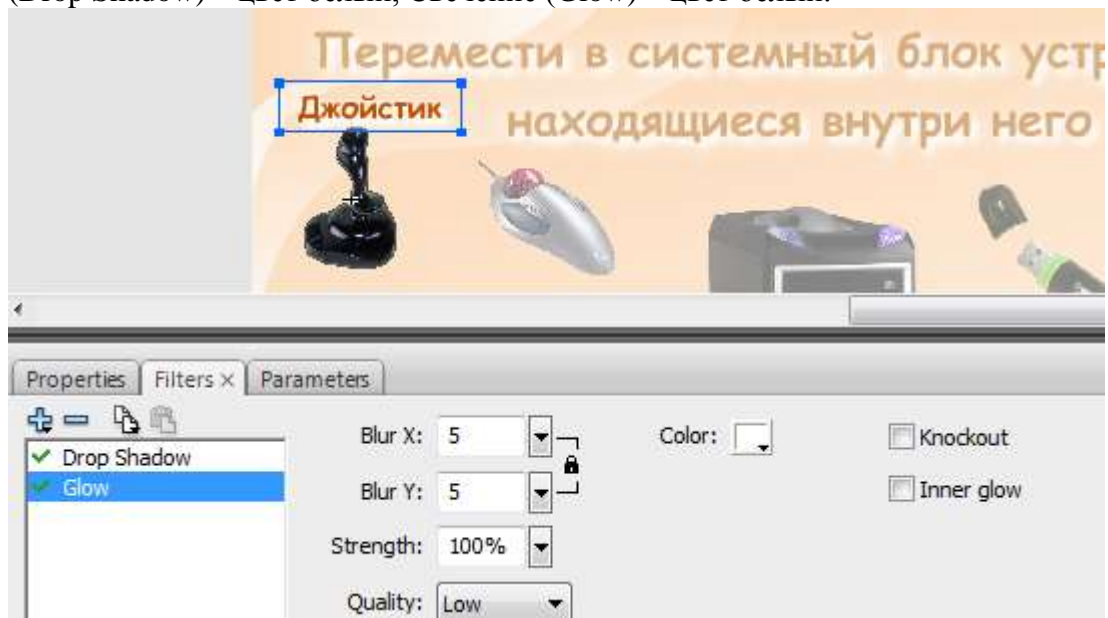
11. В слое «Клипы» добавляем кнопку, при нажатии на которую будет появляться вопрос: *Какой вывод можно сделать после выполнения данного задания?* (состояние кнопки Down)



12. Добавим подсказку - название устройства, которое будет появляться при наведении на него курсора мыши. Для этого заходим внутрь каждого символа клип, затем внутрь символа кнопка и во втором кадре (положение кнопки Over) нажимаем F6 (копировать кадр) и добавляем соответствующую надпись:



13. Переходим на вкладку «Фильтры» и, нажав на плюс, добавляем фильтры: Тень (Drop Shadow) – цвет белый, Свечение (Glow) – цвет белый.



14. Аналогичным образом поступаем со всеми клипами (п. 12-13).
 15. Нажимаем Ctrl+Enter - тестируем проект!
 16. Публикуем в формате swf.



Приложение 4

Задания для индивидуального выполнения

1. Спланируйте ЦОР в рамках одного из изучаемых или изученных ранее предметов.
2. Подберите необходимые изображения.
3. Создайте свой ресурс с использованием рассмотренной технологии.
4. Обязательно укажите в ролике ФИО, курс и факультет, на котором вы учитесь.
5. Сохраните ролик с именем ФИО_sr1, например, *StepanenkoOV_sr1 fla*
6. Опубликуйте ролик.

Приложение 5

Для зачета по лабораторному занятию:

1. Оформите пояснительную записку к ролику и сохраните под именем *ФИО_zapiska.doc*.
2. Файлы (*StepanenkoOV_sr1 fla*, *StepanenkoOV_sr1.swf*, *StepanenkoOV_zapiska.doc*) заархивируйте (*stepanenkoOV_zad1.rar*) и сдайте преподавателю.

Приложение 6

Источники для самостоятельного ознакомления:

1. Выявление коллизий <http://master.kkr.ru/Flash/master3/html/glava9.htm>
2. http://electrotech.gomart.ua/images/product/auto_complex_corpus/colorsit/colorsit_atx-l8005-c43_400w/colorsit_atx-l8005-c43_400w_000.jpg
3. http://images.anandtech.com/reviews/tradeshows/2005/tourdetaiwan/gigabyte_975x.jpg
4. <http://www.ultraprice.by/image/news/1/73382.jpg>
5. <http://www.computer-land.ru/page57.html>
6. Лекции дистанционного мастер-класса Е.А. Томиловой «Первые шаги во Flash»: http://it-n.ru/communities.aspx?cat_no=73740&lib_no=91030&tmpl=lib

Тема 17	Создание интерактивных заданий в среде Flash на установление соответствия
----------------	--

1.1. Технология обучения на лабораторном занятии

Учебное время: 2 часа.	
Форма учебного занятия: лабораторная работа	
План занятия	<ol style="list-style-type: none"> 1. Оформление фона, картинок, текста образовательного ресурса и запись кода Action Script. 2. Оформление кнопки для перехода между сценами.
Цель учебного занятия:	освоить технологию создания ресурса на установление соответствия в среде Flash.
Педагогические задачи: Организовать работу студентов по: <ul style="list-style-type: none"> • Оформлению фона, картинок, текста образовательного ресурса и записи кода Action Script. • Оформлению кнопки для перехода между сценами. 	Результаты учебной деятельности: Студент должен уметь: <ul style="list-style-type: none"> • Оформлять фон, картинки, текст образовательного ресурса и записывать код Action Script. • Оформлять кнопки для перехода между сценами.
Методы обучения	Таблица ЗХУ
Средства обучения	персональные компьютеры с установленной программой Adobe Flash, печатная версия задания
Формы организации учебной деятельности	Фронтальная, индивидуальная.
Условия обучения	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.1. Сообщает, тему, цель учебного занятия и план его проведения (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.
2-й этап. Основная часть (65 мин)	2.1. Раздает задания лабораторной работы (<i>приложение 2</i>) и организует их работу. 2.2. Объясняет, что в некоторых случаях ЦОРы могут содержать несколько сцен. Для перемещения между сценами понадобится алгоритм создания кнопки. Предлагает	2.1. Индивидуально выполняют задания лабораторной работы. 2.2. Слушают, выполняют задания.

	<p>ознакомиться с данным алгоритмом (<i>приложение 3</i>).</p> <p>2.3. Раздает задания для индивидуального выполнения (<i>приложение 4</i>) и разъясняет форму отчета по лабораторной работе (<i>приложение 5</i>).</p>	2.3. Выполняют индивидуальные задания и подготавливают отчет о проделанной работе.
3-й этап. Заключительно-результативный (10 мин)	<p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> Изучить источники Интернет по работе с Flash (<i>приложение 6</i>). Заполнить таблицу ЗХУ (<i>приложение 7</i>). 	Задают уточняющие вопросы.

Приложение 1

Тема: Создание интерактивных заданий в среде Flash на установление соответствия.

Цель занятия: освоить технологию создания ресурса на установление соответствия в среде Flash.

План занятия:

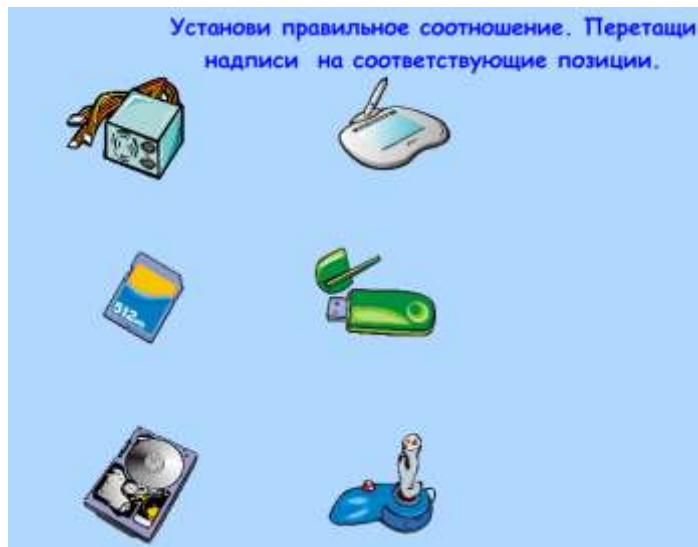
1. Оформление фона, картинок, текста образовательного ресурса и запись кода Action Script.
2. Оформление кнопки для перехода между сценами.
3. Создание собственного ресурса.

Приложение 2

Наша работа продолжается. Итак, мы уже освоили с Вами технологию свободного перемещения объектов, перемещать с исчезновением. Также технология свободного перемещения объектов во Flash может быть усовершенствована, если использовать возможность моментальной проверки правильности расположения перетаскиваемых объектов. Это можно осуществить, если использовать проверку принадлежности клипа той или иной области.

Моментальная проверка правильности расположения перетаскиваемых объектов

1. Открываем новый документ Flash CS3.
2. Называем слой именем «Фон» и устанавливаем цвет фона по желанию.
3. Устанавливаем размер рабочей области 800x600.
4. В 1-м кадре слоя «Фон» размещаем текст задания: *Установи правильное соотношение. Перетащи надписи на соответствующие позиции.*
5. Добавляем новый слой с именем «Картинки» и располагаем на нем изображения устройств.

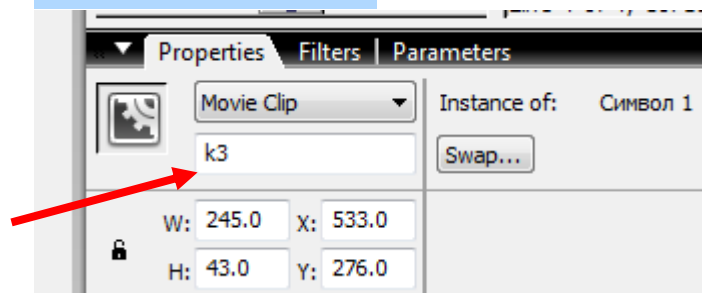
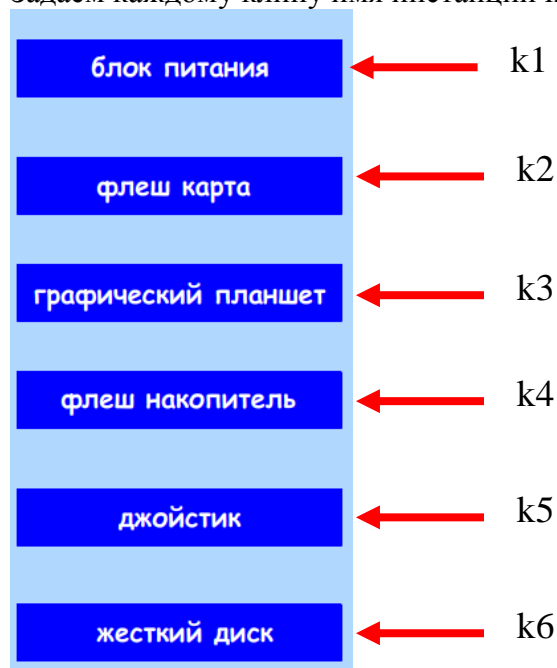


6. Добавляем новый слой с именем «Ответы». И на данном слое в 1-м кадре создаем объекты для перемещения - названия устройств.

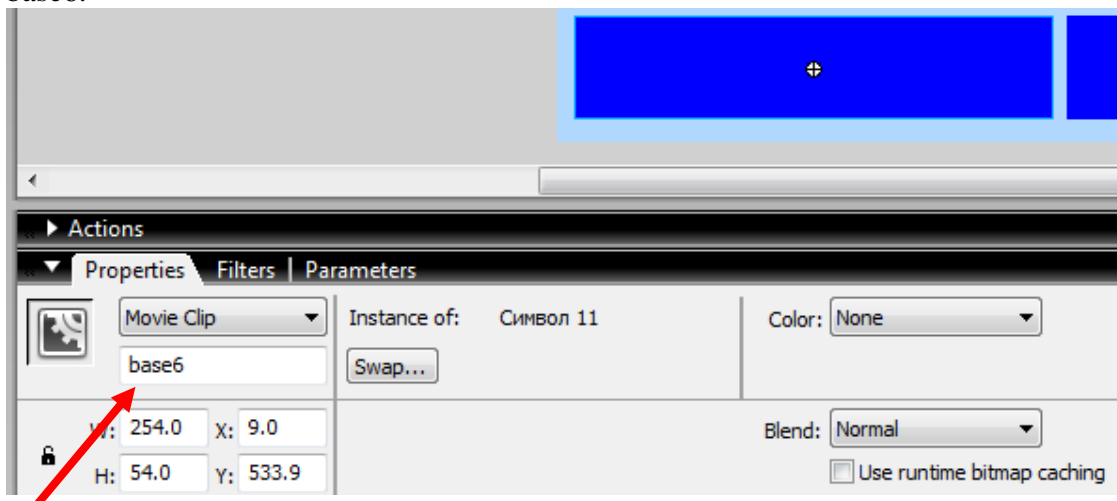
Код для свободного перемещения каждого объекта (клипа) нам уже знаком:

```
on (press) { //Кнопка мыши нажата
StartDrag (this); } //Начать перемещение
on (release) { //Кнопка мыши отпущена
StopDrag (); } //Закончить перемещение
```

7. Задаем каждому клипу имя инстанции k1, k2, k3, k4, k5, k6 соответственно:



8. Добавляем новый слой с именем «База» и располагаем на нем прямоугольные поля чуть большего размера, чем перемещаемые клипы, устанавливая их на задний план (щелкаем правой кнопкой мыши и в контекстном меню выбираем пункт *Расположить(Arrange)→На задний план(Send to Back)→Ок*.
9. Каждое прямоугольное поле (базу) преобразовываем в символ – тип клип и присваиваем соответствующее имя инстанции: base1, base2, base3, base4, base5, base6.



Обращаю Ваше внимание на то, что на клип base1 должен будет встать клип k1 и т.п.

10. Добавляем новый слой с именем «Код» и для 1 кадра (!) прописываем следующий код:

```
stop();
//Функция
function dropToBase(clip:MovieClip, initX, initY, base:MovieClip) {
    clip._x = initX;
```

```

clip._y = initY;
clip.onPress = function() { // При нажатии на клип...
    this.startDrag(); // Включаем перетаскивание
};
clip.onRelease = function() { // При отпускании кнопки мыши...
    this.stopDrag(); // Прекращаем перетаскивание
    if (eval(this._droptarget) == base) { //Если клип "сброшен" в
требуемую область (клип base),он центрируется по этой области //
        this._x = base._x;
        this._y = base._y;
    }
    else { // Иначе (если клип "сброшен" мимо цели) возвращаем его в
исходное положение
        this._x = initX;
        this._y = initY;
    }
};
}
//Основная программа
var clipNum:Number = 6; //Количество клипов
for (i = 1; i <= clipNum; i++) {
    dropToBase(this["k" + i], this["k" + i]._x, this["k" + i]._y, this["base" + i]);
}

```

Комментарий к представленному выше скрипту

Проанализируем сначала код функции.

function dropToBase(clip:MovieClip, initX, initY, base:MovieClip)

Строка описания функции, где dropToBase – имя функции, переменные clip и base, имеют тип MovieClip, переменные initX и initY определены для сохранения начального положения x и y текущего (this) экземпляра клипа:

```

clip._x = initX;
clip._y = initY;

```

Событию onPress экземпляра clip определена и присвоена функция. Она вызывается при каждом нажатии на клип (k1, k2, k3, k4, k5, k6) и включает перетаскивание:

```

clip.onPress = function() { // При нажатии на клип...
    this.startDrag(); // Включаем перетаскивание
};

```

Событию onRelease экземпляра clip также определена и присвоена функция. Она вызывается при каждом отпускании клипа (k1, k2, k3, k4, k5, k6) и прекращает перетаскивание.

```

clip.onRelease = function() { // При отпускании кнопки мыши...
    this.stopDrag(); // Прекращаем перетаскивание

```

При этом, если клип (k1, k2, k3, k4, k5, k6) "сброшен" в требуемую область (base1 и т.д.), он центрируется по этой области, иначе функция возвращает клип назад, в исходное положение. В данном случае используется структура условного оператора If (если) ...else (иначе), которая позволяет записать в сценарии необходимость выполнить одну из двух последовательностей операций:

```

if (eval(this._droptarget) == base) { //Если клип "сброшен" в требуемую область
(клип base),он центрируется по этой области //

```

```

        this._x = base._x;
        this._y = base._y;
    }
    else { //Иначе (если клип "сброшен" мимо цели) возвращаем его в
исходное положение
        this._x = initX;
        this._y = initY;
    }
}

```

Оператор `eval` позволяет нам обратиться к текущему перемещаемому объекту-клипу (`this`) и проверить его принадлежность данной базе:

```
(eval(this._droptarget) == base)
```

Следующее на, что стоит обратить внимание, это то, что для описания действий над объектами, а также для указания подчиненности объекта в языке `ActionScript` обычно применяется так называемая «точечная нотация». Например, в нашем случае чтобы указать, что к клипу `clip` следует применить метод `onPress()` используется конструкция: `clip.onPress = function()`

Основная программа начинается с описания переменной:

```
var clipNum:Number = 6; //Количество клипов
```

Слово `var` сообщает интерпретатору, что мы объявляем переменную, а последующий текст `clipNum` становится именем новой переменной, затем через двоеточие указывается тип переменной `Number` (число) и через знак равенства задается значение переменной - количество клипов, равное **6**. Понятие переменной (variable) во многих отношениях является одним из краеугольных камней программирования.

В следующей строке используется оператор цикла `for` - компактный и удобный способ программирования циклов, основанных на том, что значение некоторой переменной при каждой итерации увеличивается (как в нашем случае `i++`) или уменьшается на единицу. Удобен он тем, что инициализация переменной цикла, установление условия цикла, инкремент (или декремент) переменной – на все это требуется всего одна строка `ActionScript`:

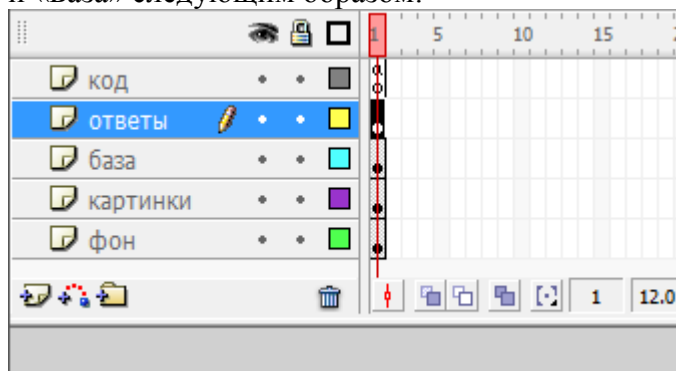
```
for (i = 1; i <= clipNum; i++) {
```

В следующей строке основной программы по имени `dropToBase` вызывается функция для каждого клипа (`k1, k2, k3, k4, k5, k6`) и базы (`base1, base2, base3, base4, base5, base6`):

```
dropToBase(this["k" + i], this["k" + i]._x, this["k" + i]._y, this["base" + i]);
```

Итак скрипт рассмотрен полностью. Думаю, что теперь все понятно... Продолжим создавать ЦОР дальше.

11. Для того, чтобы ответ размещался поверх базы меняем местами слои «Ответы» и «База» следующим образом:



12. Нажимаем Ctrl+Enter - тестируем проект!
13. Самостоятельно добавляем код, разворачивающий клип на весь экран, и кнопку выхода.
14. Публикуем в формате swf.

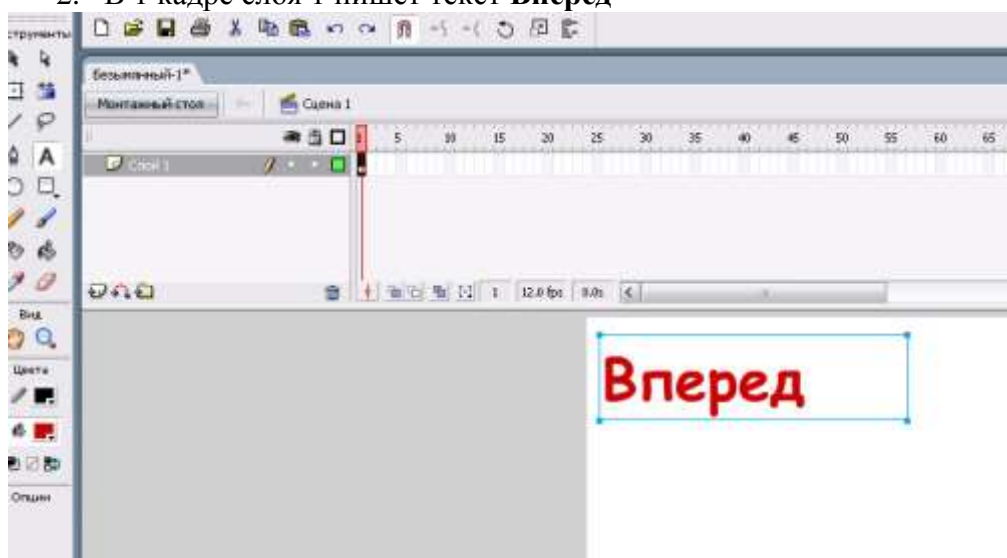
Приложение 3


Кнопка перемещения между сценами

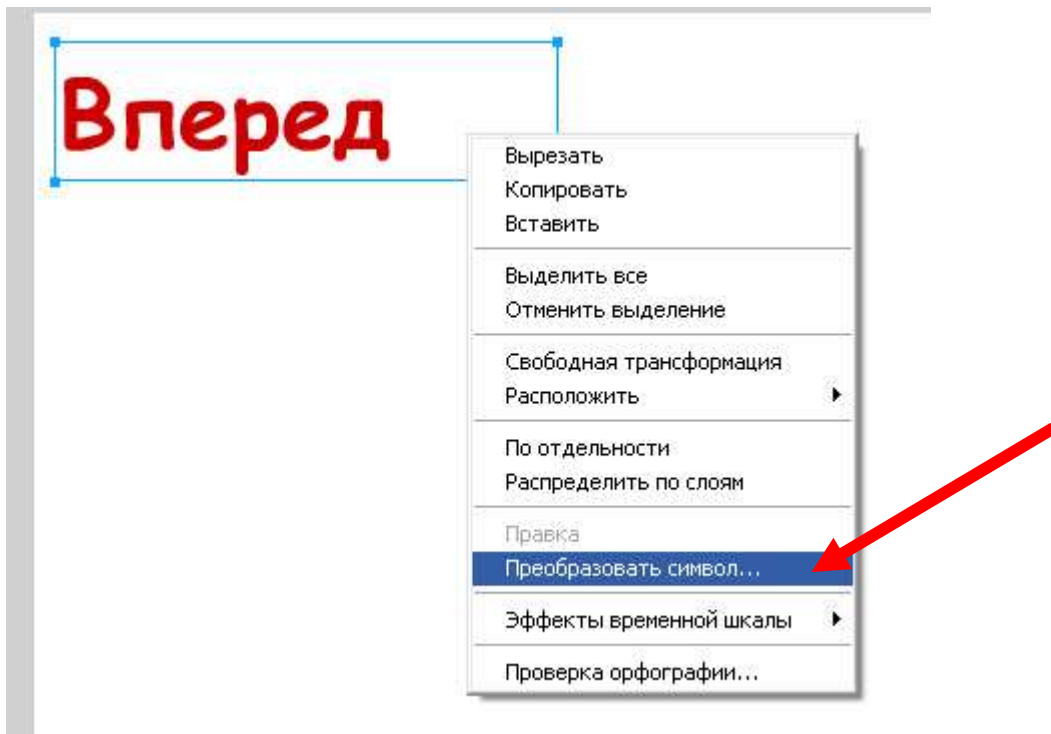
Для желающих создавать ЦОРы, содержащие несколько сцен предлагаем алгоритм создания кнопки перемещения между сценами.

Алгоритм выполнения работы:

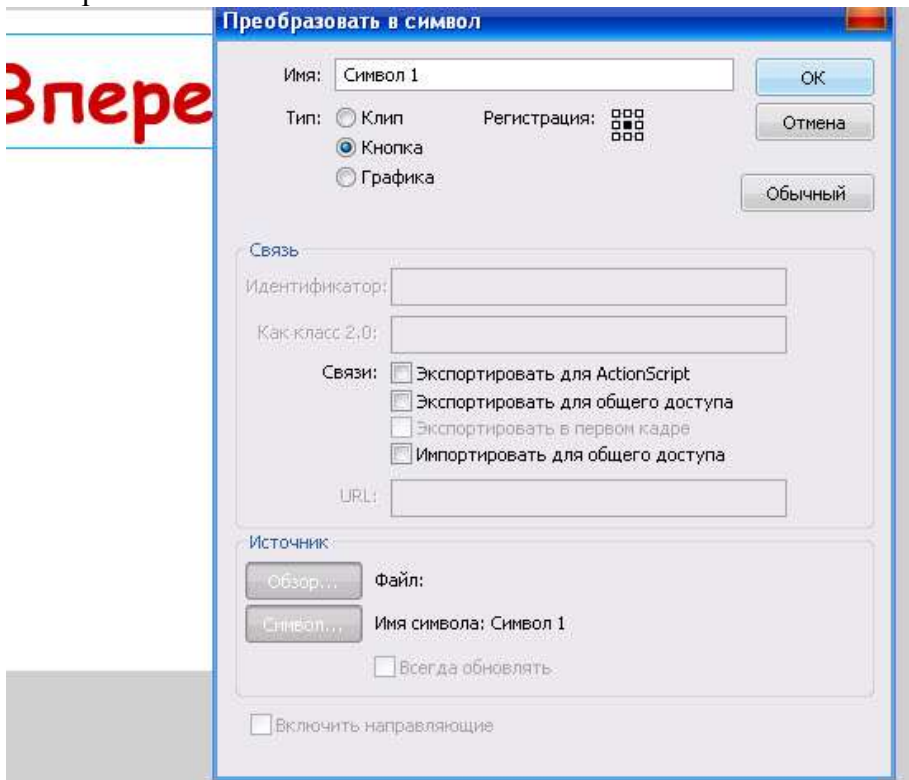
1. Открываем новый документ Flash.
2. В 1 кадре слоя 1 пишет текст **Вперёд**



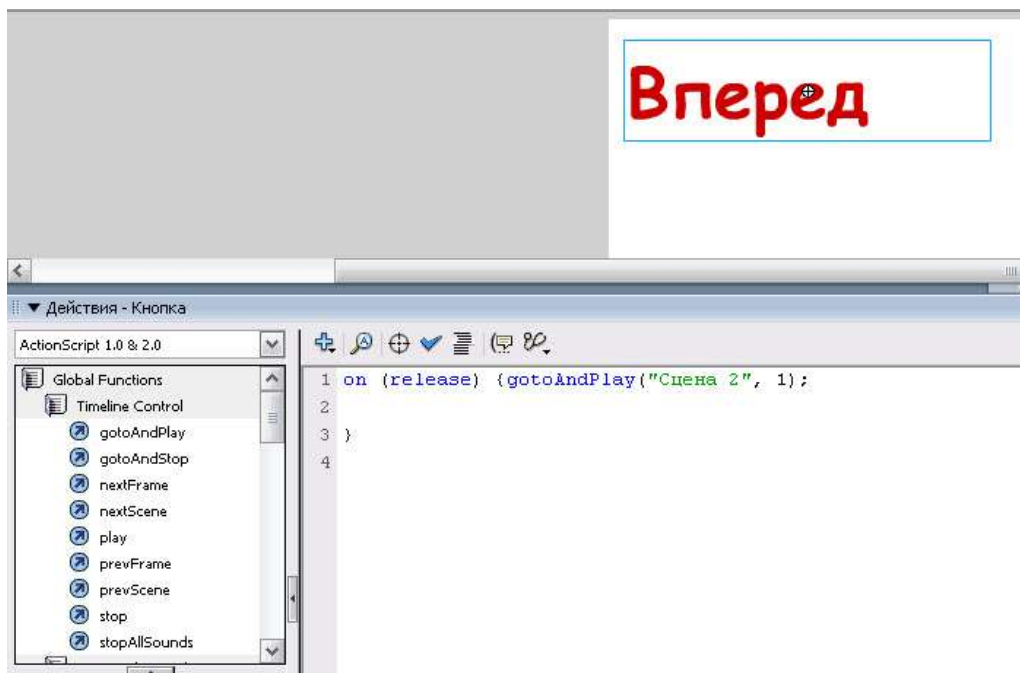
3. Выбираем инструмент  **Выделение (V)** и щелкаем по нашей надписи, нажимаем правую кнопку мыши, вызывая контекстное меню и выбираем **Преобразовать в символ:**



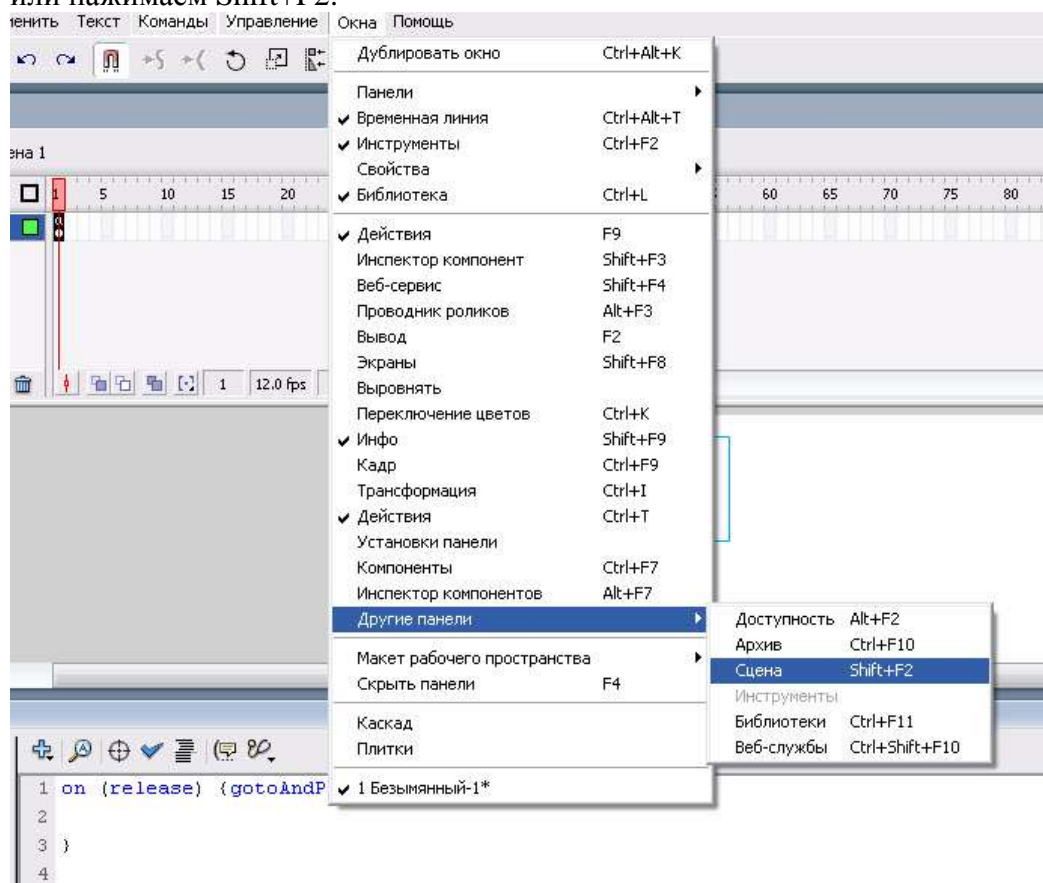
4. Выбираем тип символа – **Кнопка** и нажимаем **ОК**:



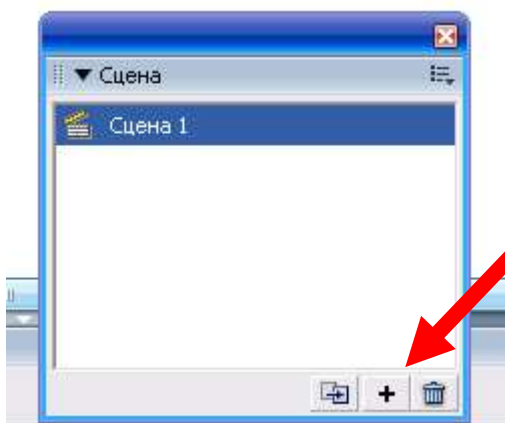
5. Теперь пропишем **действие–переход** для нашей созданной кнопки.
6. Открываем окно **Действия** или нажимаем клавишу **F9** и прописываем для кнопки переход на следующую сцену:
- ```
on (release) {gotoAndPlay("Сцена 2", 1);
```
- }



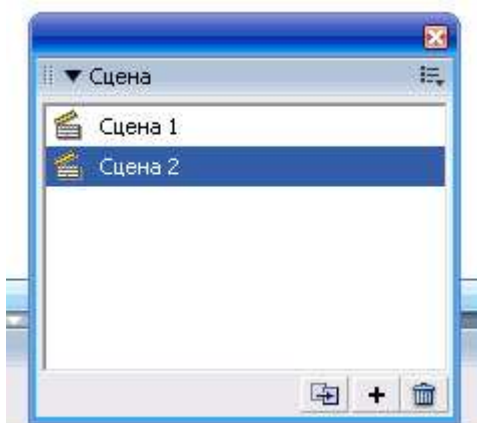
7. Создаем сцену 2. Для этого выбираем в пункте меню *Окна-Другие панели-Сцены* или нажимаем Shift+F2:



8. В открывшемся окне нажимаем + (добавить сцену):



9. Теперь у нас две сцены:



10. В сцене 2 делаем надпись **Назад** и выполняем все те же действия что и для кнопки **Вперед** (пункты 3-6). При этом действие для кнопки **Назад** будет соответственно таким:

```
on (release) {gotoAndPlay("Сцена 1", 1);
```

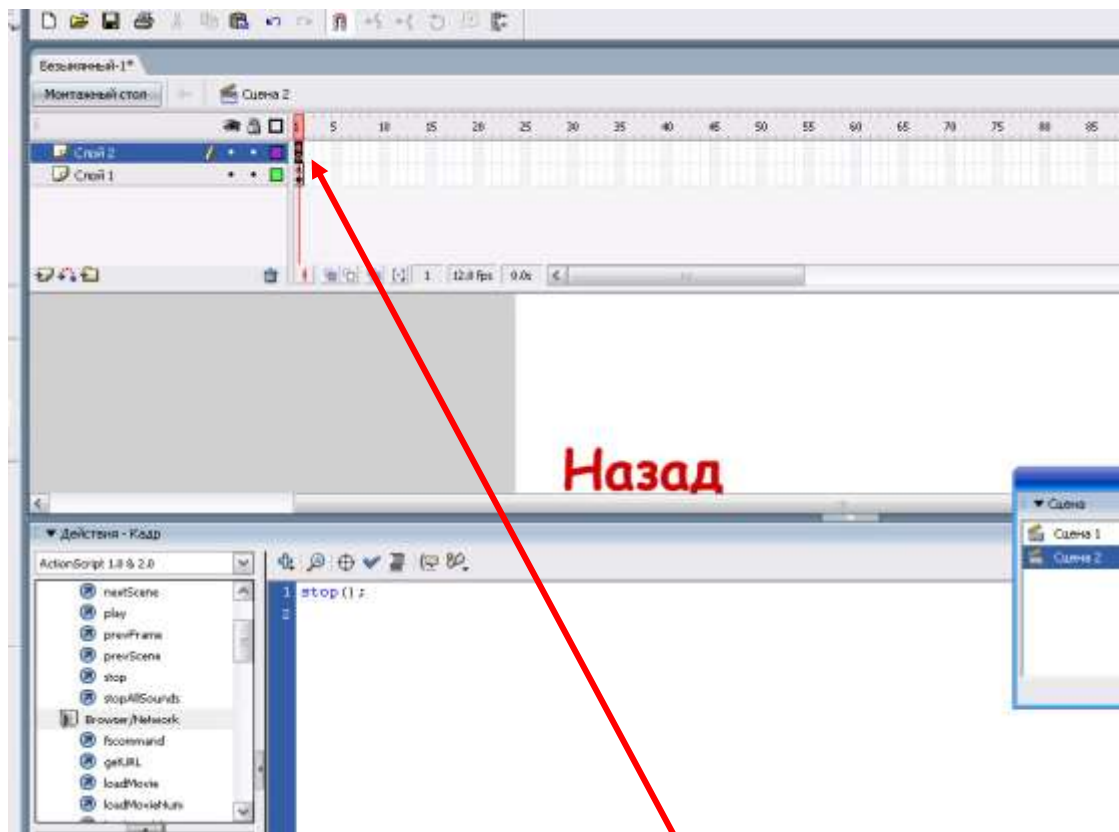
```
}
```

11. Тестируем проект, нажав Ctrl+Enter. Как видим, сцены мелькают, сменяя друг друга.

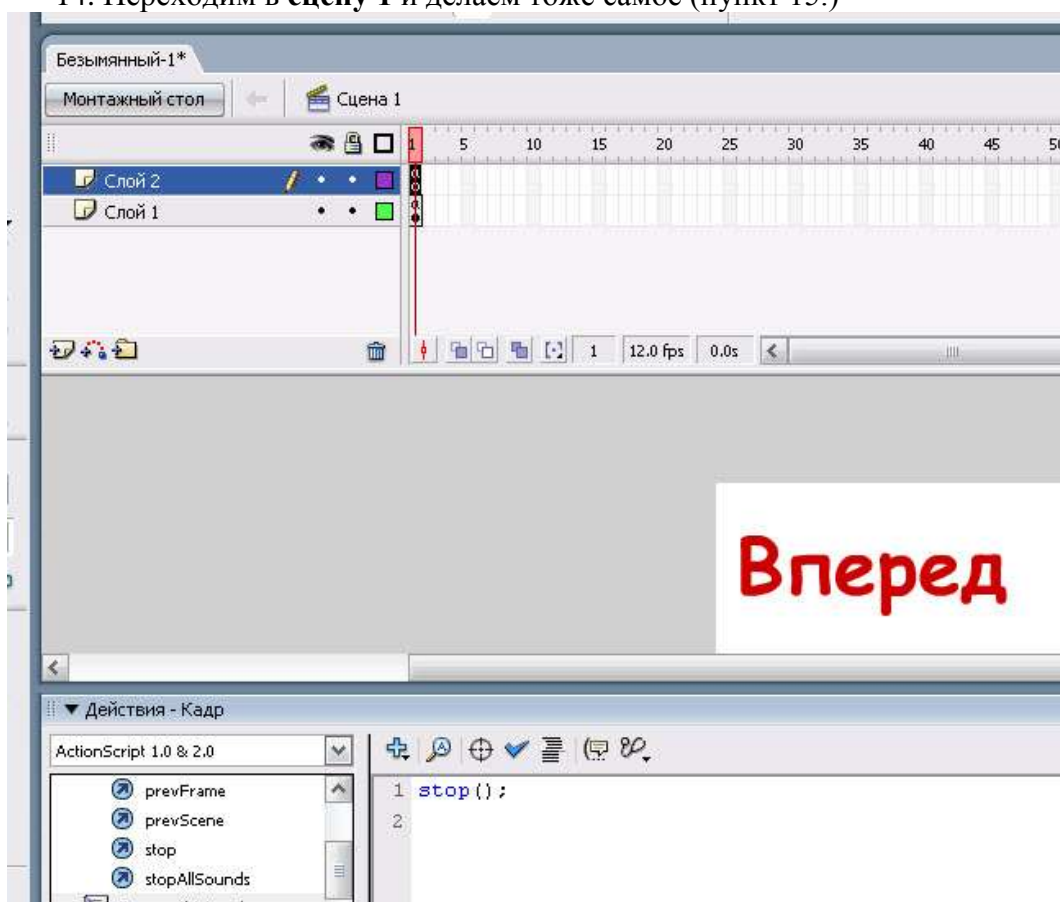
12. Теперь нам нужно прописать на обеих сценах действие **Стоп**.

13. На **сцене 2** вставляем новый слой (можно прописать и в 1-м слое), щелкаем по первому кадру и прописываем действие:

```
stop ();
```



Как видим, над кадром появилась буква **а**.  
 14. Переходим в **сцену 1** и делаем тоже самое (пункт 15.)



15. Тестируем проект, нажав Ctrl+Enter. Как видим, теперь сцены сменяются только после нажатия на созданные нами кнопки **Вперед** и **Назад**.

#### Приложение 4

##### **Задания для индивидуального выполнения**

1. Спланируйте ЦОР в рамках одного из изучаемых или изученных ранее предметов по специальности.
2. Подберите необходимые изображения.
3. Создайте свой ресурс с использованием рассмотренной технологии и количеством сцен, равным 2.
4. Обязательно укажите ФИО, курс и факультет, на котором вы учитесь.
5. Сохраните ролик с именем ФИО\_sr2, например, *StepanenkoOV\_2r2 fla*
6. Опубликуйте ролик.

#### Приложение 5

##### **Для зачета по лабораторному занятию:**

1. Оформите пояснительную записку к ролику.
2. Файлы (*StepanenkoOV\_sr2 fla*, *StepanenkoOV\_sr2.swf*, *StepanenkoOV\_zapiska.doc*) заархивируйте (*stepanenkoOV\_zad2.rar*) и сдайте преподавателю.

#### Приложение 6

##### **Источники:**

1. Векторный клипарт <http://allday.ru/index.php?newsid=24182>
2. Сидорка А.И. Как усовершенствовать Flash пособие [http://www.it-n.ru/communities.aspx?cat\\_no=106255&d\\_no=119608&ext=Attachment.aspx?Id=41074](http://www.it-n.ru/communities.aspx?cat_no=106255&d_no=119608&ext=Attachment.aspx?Id=41074)
3. Сидорка А.И. Система упражнений.
4. Лекции дистанционного мастер-класса Е.А. Томиловой «Первые шаги во Flash»: [http://it-n.ru/communities.aspx?cat\\_no=73740&lib\\_no=91030&tmpl=lib](http://it-n.ru/communities.aspx?cat_no=73740&lib_no=91030&tmpl=lib)
5. <http://dizigner.com/archives/flash/peremennyye-vo-flash-actionscript/>  
[http://www.daflash.ru/action\\_script.html](http://www.daflash.ru/action_script.html)

#### Приложение 7

Таблица ЗХУ

| Знал(а), как в программе Adobe Flash... | Хочу узнать, как в программе Adobe Flash... | Узнал(а), как в программе Adobe Flash... |
|-----------------------------------------|---------------------------------------------|------------------------------------------|
|                                         |                                             |                                          |

|                |                                                                                                                              |
|----------------|------------------------------------------------------------------------------------------------------------------------------|
| <b>Тема 18</b> | <b>Создание интерактивных заданий с проверкой правильности их решения и внедрением звукового сопровождения в среде Flash</b> |
|----------------|------------------------------------------------------------------------------------------------------------------------------|

### 1.1. Технология обучения на лабораторном занятии

|                                                                                                                                                                                                                                                                                                                                                                                                          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Учебное время: 2 часа.</i>                                                                                                                                                                                                                                                                                                                                                                            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| <i>Форма учебного занятия: лабораторная работа</i>                                                                                                                                                                                                                                                                                                                                                       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| <i>План занятия</i>                                                                                                                                                                                                                                                                                                                                                                                      | <ol style="list-style-type: none"> <li>1. Оформление образовательного ресурса и запись кода Action Script (Проверка перекрытия/пересечения клипов. Метод hitTest).</li> <li>2. Способы проверки правильности перемещения объектов с применением озвучивания ролика.</li> <li>3. Выполнение индивидуального задания.</li> </ol>                                                                                                                                                  |
| <i>Цель учебного занятия:</i>                                                                                                                                                                                                                                                                                                                                                                            | освоить технологию создания ресурса «Проверка перекрытия/пересечения клипов. Метод hitTest», научиться озвучивать ролик, освоить программный вызов клипа проверки.                                                                                                                                                                                                                                                                                                              |
| <i>Педагогические задачи:</i><br>Организовать работу студентов по: <ul style="list-style-type: none"> <li>• оформлению образовательного ресурса и записи кода Action Script (Проверка перекрытия/пересечения клипов. Метод hitTest);</li> <li>• освоению способов проверки правильности перемещения объектов с применением озвучивания ролика;</li> <li>• выполнению индивидуального задания.</li> </ul> | <i>Результаты учебной деятельности:</i><br><i>Студент должен уметь:</i> <ul style="list-style-type: none"> <li>• оформлять образовательный ресурс и записывать код Action Script (Проверка перекрытия/пересечения клипов. Метод hitTest);</li> <li>• применять различные способы проверки правильности перемещения объектов с применением озвучивания ролика;</li> <li>• самостоятельно разрабатывать и выполнять образовательный ресурс по аналогии с предложенным.</li> </ul> |
| <i>Методы обучения</i>                                                                                                                                                                                                                                                                                                                                                                                   | Схема «Рыбий скелет»                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <i>Средства обучения</i>                                                                                                                                                                                                                                                                                                                                                                                 | персональные компьютеры с установленной программой Adobe Flash, печатная версия задания                                                                                                                                                                                                                                                                                                                                                                                         |
| <i>Формы организации учебной деятельности</i>                                                                                                                                                                                                                                                                                                                                                            | Фронтальная, индивидуальная.                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <i>Условия обучения</i>                                                                                                                                                                                                                                                                                                                                                                                  | Кабинет вычислительной техники                                                                                                                                                                                                                                                                                                                                                                                                                                                  |

### Технологическая карта лабораторного занятия

| Этап работы, время | Содержание деятельности |           |
|--------------------|-------------------------|-----------|
|                    | Преподавателя           | Студентов |

|                                                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                                                                                                                                                                                                    |
|----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1-й этап.<br>Введение в учебное занятие (5 мин)    | 1.2. Сообщает, тему, цель учебного занятия и план его проведения ( <i>приложение 1</i> ).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | Слушают, уточняют, задают вопросы.                                                                                                                                                                 |
| 2-й этап.<br>Основная часть (65 мин)               | <p>2.1. Озвучивает постановку задачи (<i>приложение 2</i>).</p> <p>2.2. С помощью схемы «Рыбий скелет» знакомит с основными этапами проектирования задания (<i>приложение 3</i>).</p> <p>2.3. Раздает задания лабораторной работы (<i>приложение 4</i>) и организует их работу.</p> <p>2.4. Объясняет, что в некоторых случаях ЦОРы могут содержать несколько сцен. Для перемещения между сценами понадобится алгоритм создания кнопки. Предлагает ознакомиться с данным алгоритмом (<i>приложение 5</i>).</p> <p>2.5. Раздает задания для индивидуального выполнения (<i>приложение 6</i>) и разъясняет форму отчета по лабораторной работе (<i>приложение 7</i>).</p> | <p>2.1. Индивидуально выполняют задания лабораторной работы.</p> <p>2.2. Слушают, выполняют задания.</p> <p>2.3. Выполняют индивидуальные задания и подготавливают отчет о проделанной работе.</p> |
| 3-й этап.<br>Заключительно-результативный (10 мин) | <p>3.1. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> <li>• Доработать задания;</li> <li>• Изучить источники Интернет по работе с Flash (<i>приложение 8</i>).</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                       | Задают уточняющие вопросы.                                                                                                                                                                         |

## Приложение 1

### Тема: Создание интерактивных заданий с проверкой правильности их решения и внедрением звукового сопровождения в среде Flash

**Цель занятия:** освоить технологию создания ресурса «Проверка перекрытия/пересечения клипов. Метод hitTest», научиться озвучивать ролик, освоить программный вызов клипа проверки.

#### План занятия:

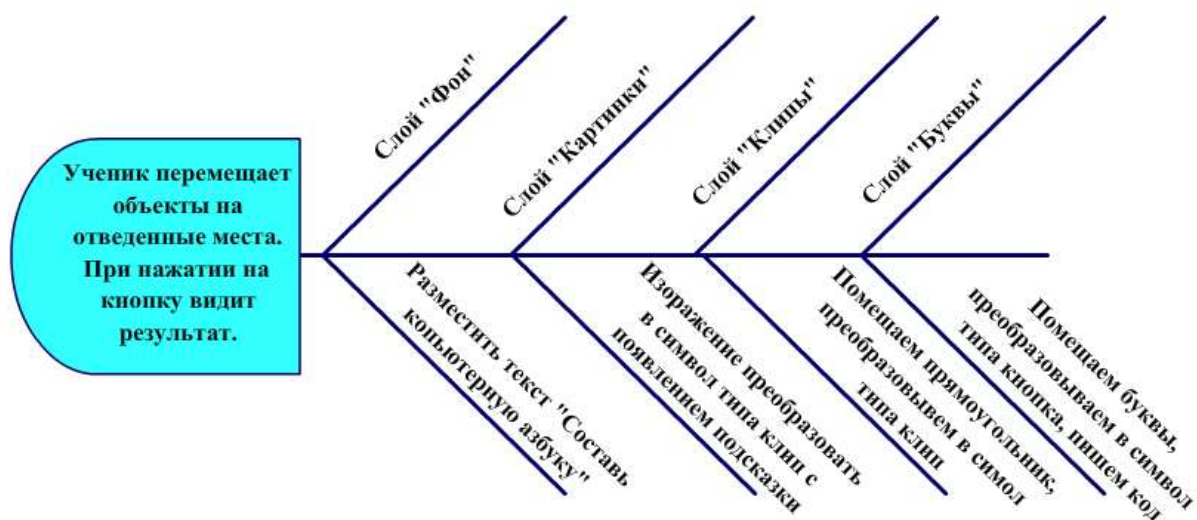
1. Оформление образовательного ресурса и запись кода Action Script (Проверка перекрытия/пересечения клипов. Метод hitTest).
2. Способы проверки правильности перемещения объектов с применением озвучивания ролика.
3. Выполнение индивидуального задания.

## Приложение 2

### Постановка задачи:

Представим ситуацию: ученик выполняет задание – перемещает объекты на отведенные для этого места. После того, как все объекты перемещены, он нажимает на кнопку проверки и сразу видит результат своей работы: верно или неверно он выполнил задание.

## Приложение 3

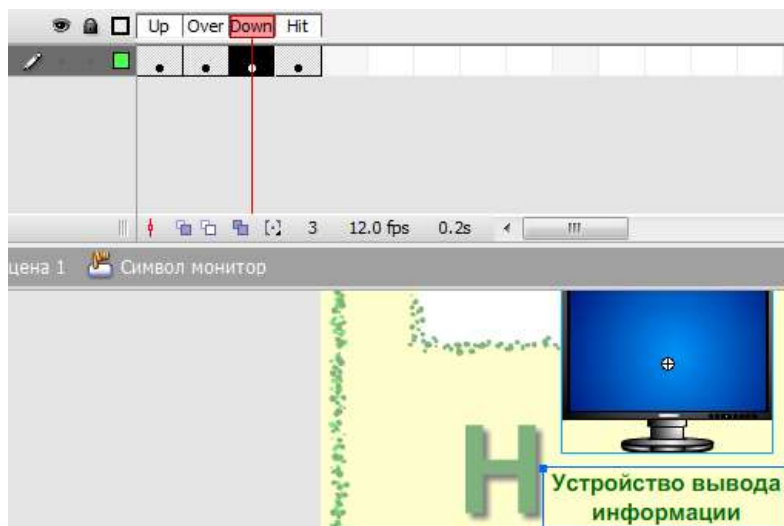


## Приложение 4

### Проверка перекрытия/пересечения клипов. Метод hitTest

#### Алгоритм выполнения работы:

1. Открываем новый документ Flash.
2. Называем слой именем «Фон» и устанавливаем цвет фона по желанию. Устанавливаем размер рабочей области 800x700.
3. В 1-м кадре слоя «Фон» размещаем текст задания: *Составь компьютерную азбуку*.
4. Добавляем новый слой с именем «Картинки» и располагаем на нем изображения.
5. Преобразовываем каждое изображение в символ – тип кнопка. Заходим внутрь символа кнопка и делаем так, чтобы при щелчке по кнопке появлялась подсказка. Для этого копируем содержимое первого кадра с помощью кнопки F6 во второй, третий и четвертый кадры, а потом в третий кадр кнопки добавляем текст подсказки:



6. Аналогичным образом (п. 5) поступаем с остальными изображениями.
7. Добавляем новый слой с именем «Клипы». На данном слое размещаем объекты, на которые нужно будет перемещать буквы. В нашем случае это прямоугольники (стиль штриха линии – точки). Внутри большого прямоугольника добавляем по центру маленький, преобразовываем его в символ – тип клип и задаем каждому клипу (базе) имя инстанции соответственно: b1, b2, b3, b4, b5.



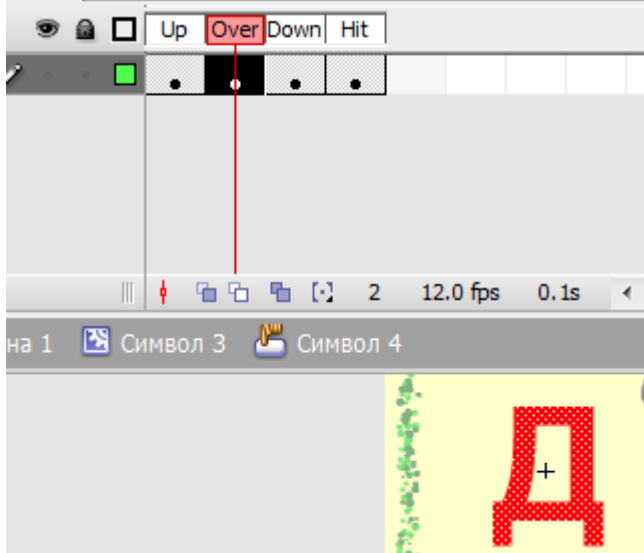
8. Заходим каждого

избавляемся от зеленого контура: выделяем его двойным щелчком и меняем цвет на белый.

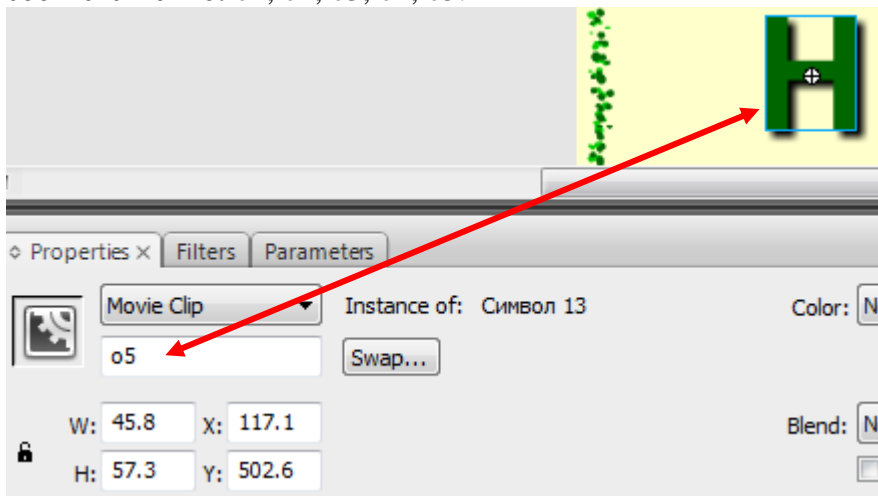
9. Добавляем новый слой с именем «Буквы» и размещаем на нем буквы.
10. Каждую букву преобразовываем в символ – тип клип. Двойным щелчком заходим внутрь клипа и преобразовываем букву еще раз в символ – тип кнопка. Добавляем следующий код для кнопки:

```
on (press) {
 startDrag ("", false);
}
on (release) {
 stopDrag ();
}
```

11. Двойным щелчком заходим внутрь кнопки и, выделив букву, нажимаем **Ctrl+B**, т.е. преобразовываем текст в графику. Затем делаем так, чтобы при наведении курсора мыши на букву она чуть-чуть сдвигалась в сторону и меняла свой цвет. Для этого копируем содержимое первого кадра с помощью кнопки F6 во второй, третий и четвертый кадры, а потом во втором кадре кнопки смещаем букву влево и вверх (по три раза нажимаем клавиши → и ↑), меняем ее цвет на другой:



12. Выходим из символа кнопка и символа клип, дважды нажав синюю стрелку.  
13. Задаем созданному объекту для перемещения - клипу - имя инстанции соответственно: о1, о2, о3, о4, о5.



14. Аналогичным образом (п. 9-12) поступаем с остальными объектами для перемещения - буквами.

## Приложение 5

### Проверка правильности перемещения объектов

А теперь уважаемые слушатели, чтобы проверить правильность перемещения объектов можно воспользоваться двумя способами: простым и более сложным ☺.

#### 1 способ (простой)

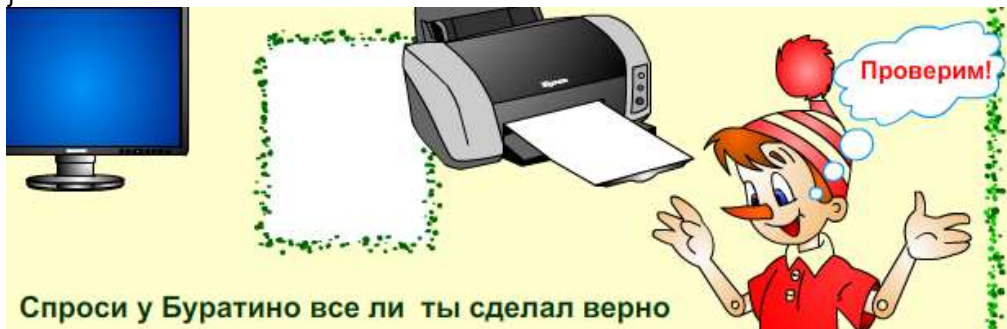
15. Добавляем новый слой с именем «Кнопка» и размещаем на нем надпись «Спроси у Буратино все ли ты сделал верно» и кнопку (изображение Буратино).

16. В слой с кнопкой Буратино вставляем следующий скрипт:

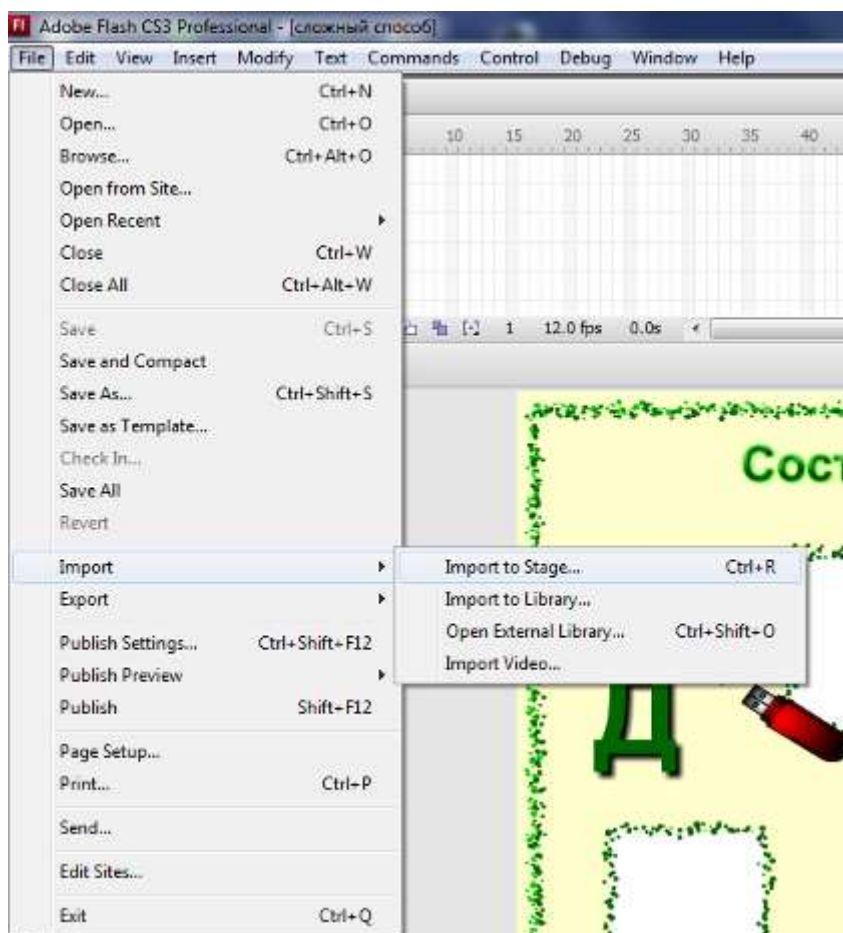
```
var verno = new Sound(); //verno – переменная для звука правильного ответа
var neverno = new Sound(); //neverno – переменная для звука неправильного
ответа
```

17. А кнопке добавляем скрипт:

```
on (release)
{
 if (_root.b1.hitTest(_root.o1) && _root.b2.hitTest(_root.o2) &&
 _root.b3.hitTest(_root.o3) && _root.b4.hitTest(_root.o4) &&
 _root.b5.hitTest(_root.o5))
 { verno.attachSound("verno"); //Связывание переменной со звуковым файлом
 verno.start(); //Старт проигрывания звука verno.mp3
 rez = "Поздравляю! Вы справились с заданием!"; //В поле с типом
динамический текст выдается сообщение «Поздравляю! Вы справились с
заданием!»
 }
 else
 { neverno.attachSound("neverno"); //Связывание переменной со звуковым
файлом
 neverno.start(); //Старт проигрывания звука neverno.mp3
 rez = "К сожалению, вы не справились с заданием!"; //В поле с типом
динамический текст выдается сообщение «К сожалению, вы не справились с
заданием!»
 } // end else if
}
```



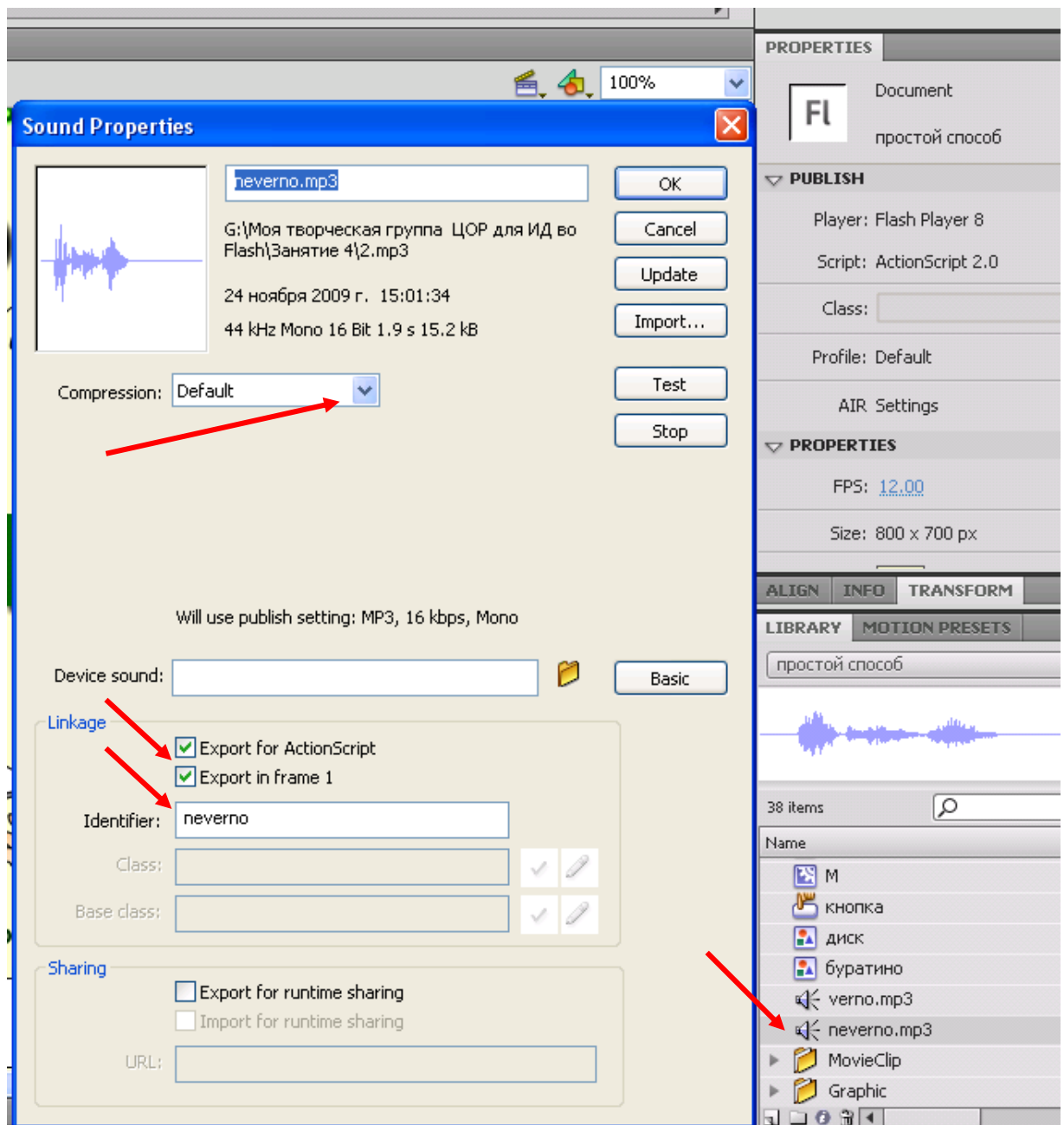
18. Звуки neverno и verno мы прописали, но их нужно добавить в библиотеку. Для этого выбираем: File (Файл) → Import (Импорт) → Import to Library... (Импорт в библиотеку...) → Имя файл → Открыть



Если вдруг Flash откажется импортировать звуки нужно будет установить плеер QuickTime.

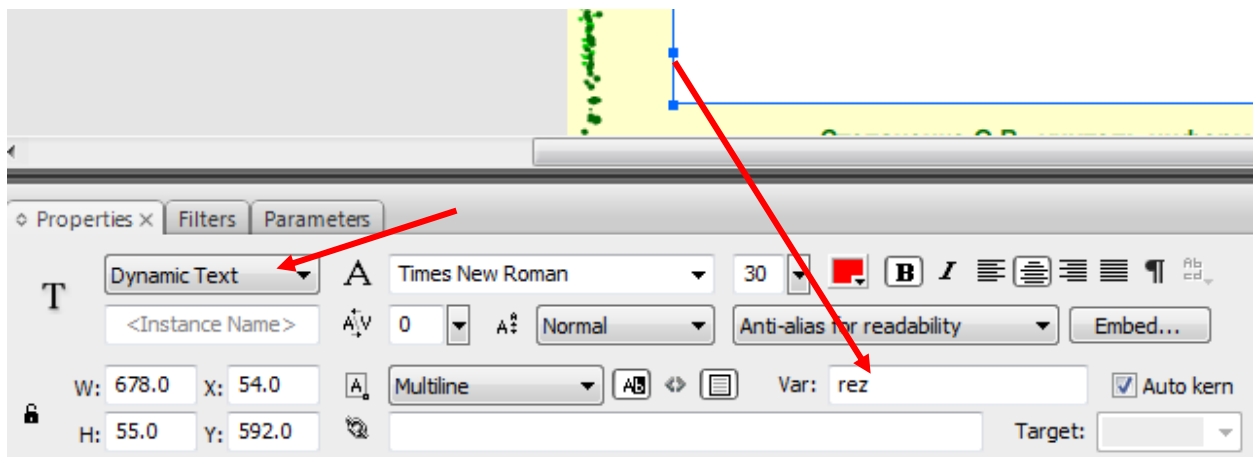
Flash сохраняет звуки в библиотеке наряду с растровыми изображениями и символами. Как и с графическими символами, нужна только одна копия звукового файла, чтобы использовать этот звук любое количество раз во Flash-клипе.

19. Теперь нам нужно настроить звуки для экспорта в ActionScript. Для этого щелкаем двойным щелчком по имени звукового файла в библиотеке и в открывшемся окне Sound Properties (свойства звука) выбираем одну из опций сжатия (Compression) - Default (по умолчанию), проставляем галочки в Linkage (Связывание) напротив пунктов: Export for ActionScript (экспортировать для ActionScript) и Export in frame 1 (экспортировать в первый кадр). В строке Identifier (Идентификатор) указываем имя звукового файла neverno для файла neverno.mp3 и verno для файла verno.mp3:



20. Создаем новый слой «Результат» и в нем размещаем поле - тип текста динамический (Dynamic Text). Советую набрать предложение для вывода (К сожалению, вы не справились с заданием! или Поздравляю! Вы справились с заданием!) и выбрать подходящий размер шрифта и цвет. А потом аккуратно все удалить, в том числе и все пробелы. Будьте внимательнее, чтобы ничего в поле не осталось! Иначе ответ постоянно будет отрицательным.

21. Присваиваем переменной имя rez:



## 2 способ (более сложный)

15. Добавляем новый слой с именем «Кнопка» и размещаем на нем надпись «Спроси у Буратино все ли ты сделал верно» и кнопку (изображение Буратино) со следующим кодом:

```
on (release)
{
 if (_root.b1.hitTest(_root.o1) && _root.b2.hitTest(_root.o2) &&
 _root.b3.hitTest(_root.o3) && _root.b4.hitTest(_root.o4) &&
 _root.b5.hitTest(_root.o5))
 {
 with (_root.rez) //Обращаемся к клипу rez
 {
 gotoAndStop(2); //Выдается сообщение «Поздравляю! Вы справились с заданием!» (2-й кадр клипа rez)
 } // End of with
 }
 else
 {
 with (_root.rez) //Обращаемся к клипу rez
 {
 gotoAndStop(3); //Выдается сообщение «К сожалению, вы не справились с заданием! Вернуться и исправить ошибки (3-й кадр клипа rez)
 } // End of with
 } // end else if
}
```

### Комментарий к представленному выше скрипту

В данном скрипте метод `hitTest` используется для того, чтобы определить перекрываются/пересекаются ли клипы `b1` и `o1` (и т.д.) или нет.

`With` выполняет некоторый код в конце указанного объекта, предоставляет сокращенный способ обращения к свойствам объекта. В конкретном скрипте осуществляется обращение к клипу `rez`.

#### Синтаксис:

```
with (object) {
 statements;
}
```

, где *object* - экземпляр *ActionScript* объекта или мувиклипа.

*statements* - команда или группа команд, заключенных в фигурные скобки.

В скрипте вместо **with** можно воспользоваться глобальной функцией **tellTarget**.

**Синтаксис:**

```
tellTarget(target) {
 statements
}
```

, где *target* - строка или ссылка, указывающие путь к фильму или экземпляру клипа (ссылки преобразуются в пути при использовании в строковом контексте).

*statements* - предложения, которые должны быть выполнены в области видимости *target*.

**TellTarget** позволяет обратиться к удаленному клипу, и в CS3 она работает (можете попробовать и заменить **with** на **tellTarget**)!

Хотя в справочнике Flash написано следующее: «Во Flash 3 и Flash 4 **tellTarget()** была главным средством связи между двумя клипами фильмов (т.е. управления одним клипом из другого). Она использовалась для вызова таких функций, как **play()**, **stop()**, и **gotoAndStop()** с удаленными клипами фильмов. Во Flash 4, где в **ActionScript** были добавлены переменные, с помощью **tellTarget()** можно было также получать и задавать значения переменных удаленных клипов. Во Flash 5 такие действия лучше осуществлять с помощью оператора точки, и оператора обращения к массиву []. Другой альтернативой функции **tellTarget()** является предложение **with**» [5].

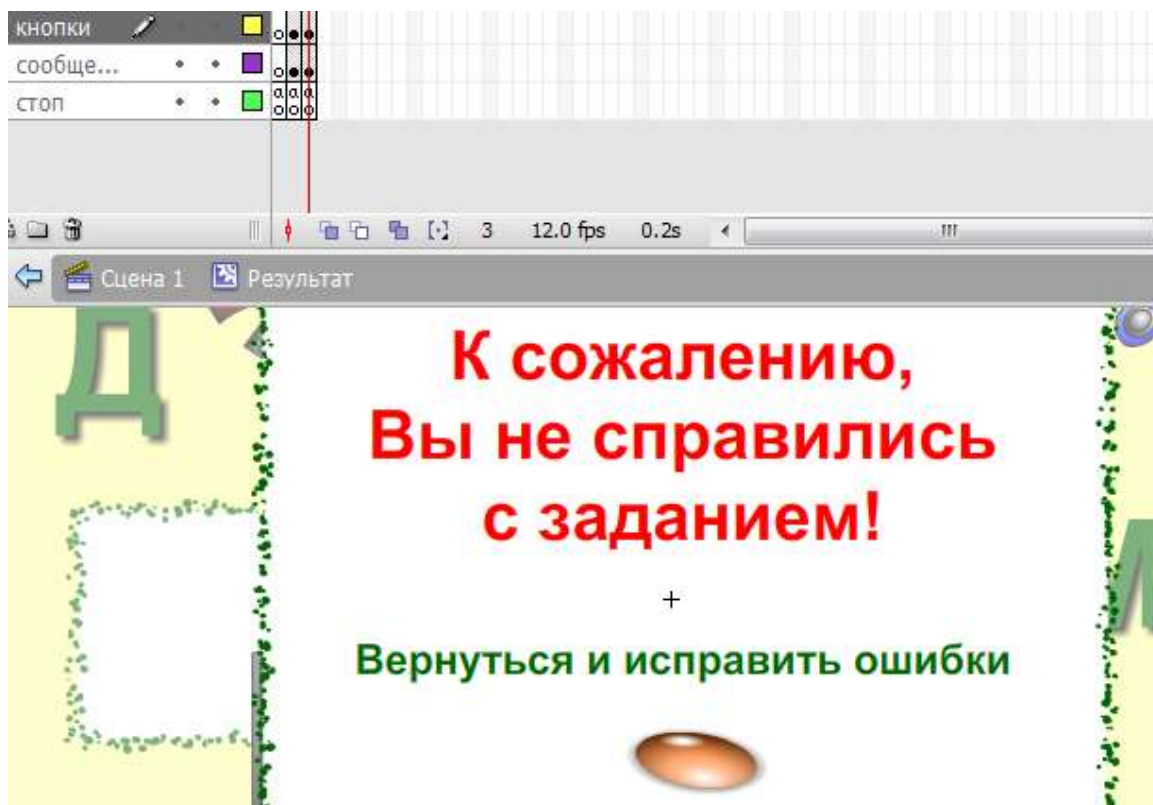
Продолжим...

- 
16. Создаем новый символ клип: *Вставить*→*Новый символ* (или нажимаем сочетание клавиш **Ctrl + F8**), тип – клип, имя – *Результат*. Выходим из символа, нажав синюю стрелку.
  17. Добавляем новый слой с именем «Результат», переносим на него созданный клип «Результат», размещаем его по центру рабочего поля.
  18. Задаем имя инстанции для нашего клипа **rez**.
  19. Двойным щелчком по точке, заходим внутрь клипа:



20. В первых трех кадрах слоя «Стоп» прописываем соответствующее действие **stop()**;
21. Добавляем новый слой с именем «Сообщение». Во 2-м кадре размещаем прямоугольник и статический текст: «Поздравляю! Вы справились с заданием!». В 3-м кадре данного слоя размещаем статический текст: «К сожалению, вы не справились с заданием! Вернуться и исправить ошибки».
22. Добавляем новый слой «Кнопки» и во 2-м кадре размещаем кнопку выхода (скрипт Вам уже знаком!), а в 3-м кадре кнопку возврата со следующим кодом:  

```
on (release)
{
 gotoAndStop(1);
}
```

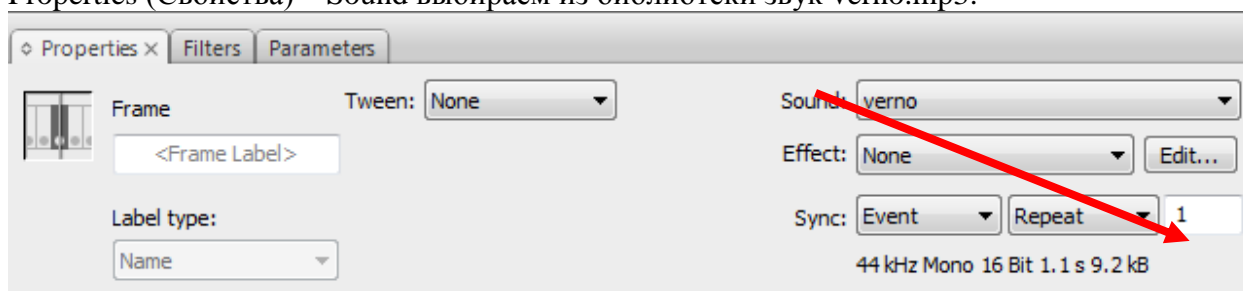


23. В данном способе также предлагаю озвучить Буратино. Для этого импортируем в библиотеку звуки verno.mp3 и neverno.mp3.

24. Двойным щелчком по точке заходим внутрь клипа rez:

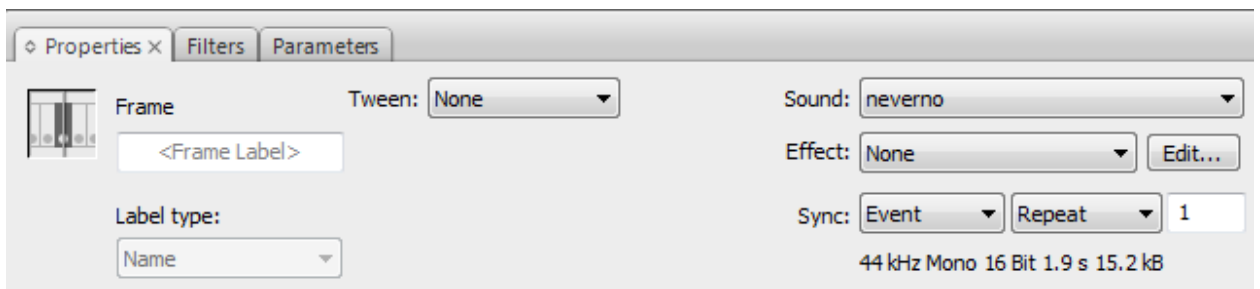


25. Во второй кадр клипа rez добавляем звук verno.mp3. Для этого сделаем активным кадр, в который будем вставлять звук, щелкнув по нему (2-й кадр слоя стоп). На вкладке Properties (Свойства) – Sound выбираем из библиотеки звук verno.mp3:

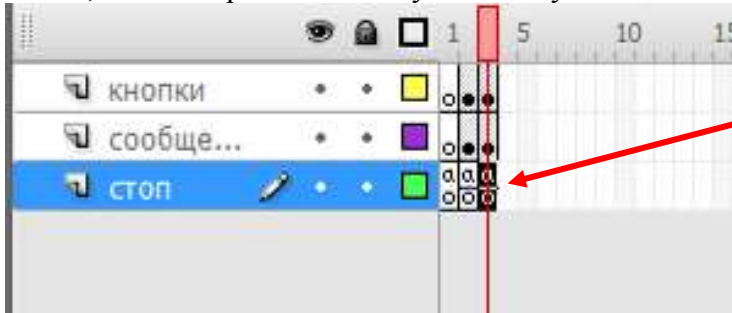


26. В третий кадр клипа rez добавляем звук neverno.mp3:

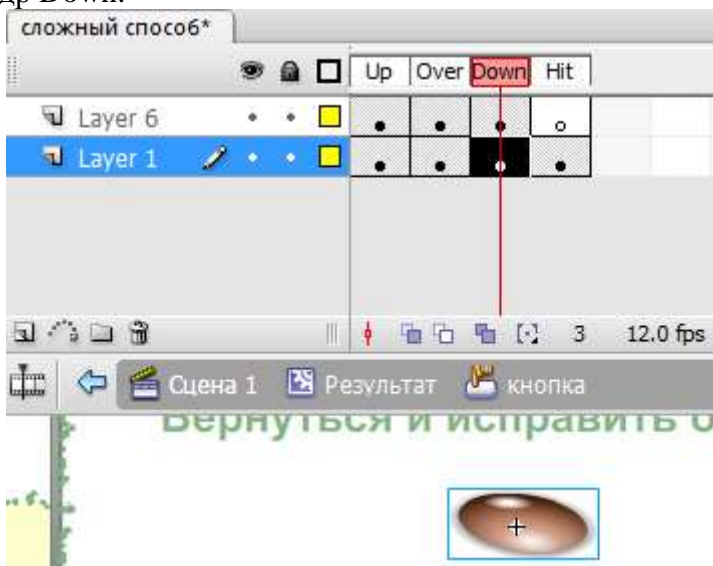




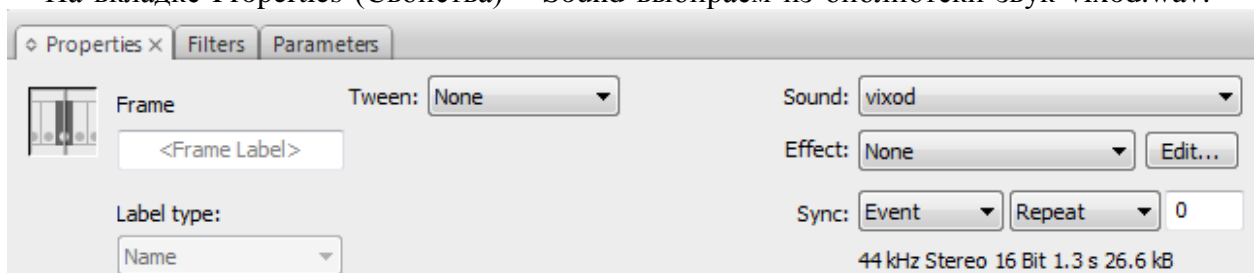
О том, что в кадр добавлен звук можно узнать по линии над кадром:



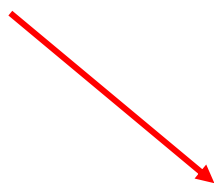
27. Чтобы озвучить кнопку, нужно зайти внутрь символа кнопка. Сделать активным кадр Down:

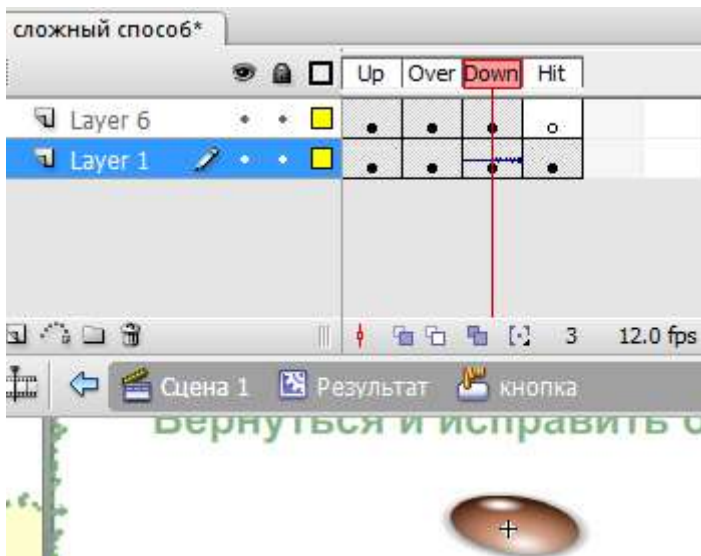


На вкладке Properties (Свойства) – Sound выбираем из библиотеки звук vixod.wav:



Звук добавлен





**Это два способа проверки перемещаемых объектов.**

Выбираем тот, который больше нравится.

Далее:

Самостоятельно добавляем код, разворачивающий клип на весь экран.

Нажимаем Ctrl+Enter - тестируем проект!

Публикуем в формате swf.

### Вариант проверки с использованием цикла с параметром

Вариант проверки с использованием цикла с параметром, при котором можно легко изменить количество объектов, не допуская ошибок: «Для тех, кто готов воспринять немного программирования, советую поступить так»:

```
flag=true; //Предположим, все объекты уже на местах
for (i=1; i<=5; i++)
{
 if (not _root["b"+i].hitTest(_root["o"+i]))
 {
 flag=false; //Один объект все-таки не на месте
 }
}
if (flag)...похвалить
```

Вариант проверки с использованием цикла с параметром действительно будет полезен при использовании большого количества объектов для перемещения.

Предлагаю воспользоваться данным скриптом, усложнив поставленную задачу в индивидуальной части занятия.

## Приложение 6

### Задания для индивидуального выполнения

1. Спланируйте ЦОР в рамках одного из изучаемых или изученных ранее предметов по специальности.
2. Подберите необходимые изображения.
3. Создайте свой ресурс с использованием рассмотренной технологии, выбрав один из способов проверки. Можно выбрать оба и сделать две сцены!

4. Обязательно укажите ФИО, курс и факультет, на котором вы учитесь.
5. Сохраните ролик с именем ФИО\_pr5, например, *StepanenkoOV\_pr5 fla*
6. Опубликуйте ролик.

## Приложение 7

### Для зачета по лабораторному занятию:

1. Оформите пояснительную записку к ролику.
2. Файлы (*StepanenkoOV\_sr3 fla*, *StepanenkoOV\_sr3.swf*, *StepanenkoOV\_zapiska.doc*) заархивируйте (*stepanenkoOV\_zad3.rar*) и сдайте преподавателю.

## Приложение 8

### Источники:

1. Лекции дистанционного мастер-класса Е.А. Томиловой «Первые шаги во Flash»: [http://it-n.ru/communities.aspx?cat\\_no=73740&lib\\_no=91030&tmpl=lib](http://it-n.ru/communities.aspx?cat_no=73740&lib_no=91030&tmpl=lib)
2. Буратино [http://www.karusel-toys.ru/images/catalog/image\\_b.php?namephoto=s\\_6216.jpg](http://www.karusel-toys.ru/images/catalog/image_b.php?namephoto=s_6216.jpg)
3. Справочник по Flash: ActionScript : Action : with <http://www.spravkaweb.ru/flash/actionscript/action/with>
4. Справочник по Flash: ActionScript : Action : tellTarget <http://www.spravkaweb.ru/flash/actionscript/action/telltarget>
5. <http://samplesland.narod.ru>
6. [http://www.sibmult.ru/sound/index.php?subaction=showfull&id=1208531084&archive=&start\\_from=&ucat=3&](http://www.sibmult.ru/sound/index.php?subaction=showfull&id=1208531084&archive=&start_from=&ucat=3&)

|                |                                                                                       |
|----------------|---------------------------------------------------------------------------------------|
| <b>Тема 19</b> | <b>Технология создания тестовых заданий с использованием текстовых полей во Flash</b> |
|----------------|---------------------------------------------------------------------------------------|

### 1.1. Технология обучения на лабораторном занятии

|                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                                                                                                                                            |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Учебное время: 2 часа.</i>                                                                                                                                                                                                                                       |                                                                                                                                                                                                                                                                                                                                            |
| <i>Форма учебного занятия: лабораторная работа</i>                                                                                                                                                                                                                  |                                                                                                                                                                                                                                                                                                                                            |
| <i>План занятия</i>                                                                                                                                                                                                                                                 | <ol style="list-style-type: none"> <li>1. Виды текста во Flash.</li> <li>2. Векторизация растровых изображений .</li> <li>3. Выполнение индивидуального задания.</li> </ol>                                                                                                                                                                |
| <i>Цель учебного занятия:</i>                                                                                                                                                                                                                                       | освоить технологию создания тестовых заданий с использованием текстовых полей во Flash: статический текст (Static Text), динамический текст (Dynamic Text) и текст, вводимый пользователем (Input Text)                                                                                                                                    |
| <i>Педагогические задачи:</i><br>Организовать работу студентов по: <ul style="list-style-type: none"> <li>• использованию различных видов текста во Flash;</li> <li>• векторизации растровых изображений;</li> <li>• выполнению индивидуального задания.</li> </ul> | <i>Результаты учебной деятельности:</i><br><i>Студент должен уметь:</i> <ul style="list-style-type: none"> <li>• использовать различные виды текста во Flash.</li> <li>• Производить векторизацию растровых изображений.</li> <li>• самостоятельно разрабатывать и выполнять образовательный ресурс по аналогии с предложенным.</li> </ul> |
| <i>Методы обучения</i>                                                                                                                                                                                                                                              | Таблица ЗХУ                                                                                                                                                                                                                                                                                                                                |
| <i>Средства обучения</i>                                                                                                                                                                                                                                            | персональные компьютеры с установленной программой Adobe Flash, печатная версия задания                                                                                                                                                                                                                                                    |
| <i>Формы организации учебной деятельности</i>                                                                                                                                                                                                                       | Фронтальная, индивидуальная.                                                                                                                                                                                                                                                                                                               |
| <i>Условия обучения</i>                                                                                                                                                                                                                                             | Кабинет вычислительной техники                                                                                                                                                                                                                                                                                                             |

### Технологическая карта лабораторного занятия

| Этап работы, время                           | Содержание деятельности                                                                   |                                    |
|----------------------------------------------|-------------------------------------------------------------------------------------------|------------------------------------|
|                                              | Преподавателя                                                                             | Студентов                          |
| 1-й этап. Введение в учебное занятие (5 мин) | 1.3. Сообщает, тему, цель учебного занятия и план его проведения ( <i>приложение 1</i> ). | Слушают, уточняют, задают вопросы. |

|                                                               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                                                                                                                                                                                                      |
|---------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2-й этап.<br>Основная<br>часть (65 мин)                       | <p>2.1. Предлагает заполнить таблицу ЗХУ (<i>приложение 2</i>).</p> <p>2.2. Напоминает виды текстовых данных, с которыми можно работать во Flash (<i>приложение 3</i>).</p> <p>2.3. Объясняет, как при подготовке изображений для тестовых заданий произвести векторизацию растровых изображений (<i>приложение 4</i>).</p> <p>2.4. Раздает задания лабораторной работы (<i>приложение 5</i>) и организует их работу.</p> <p>2.5. Раздает задания для индивидуального выполнения (<i>приложение 6</i>) и разъясняет форму отчета по лабораторной работе (<i>приложение 7</i>).</p> | <p>2.1. Заполняют таблицу ЗХУ.</p> <p>2.2. Слушают.</p> <p>2.3. Выполняют задания лабораторной работы.</p> <p>2.3. Выполняют индивидуальные задания и подготавливают отчет о проделанной работе.</p> |
| 3-й этап.<br>Заключитель-<br>но-результат-<br>ный<br>(10 мин) | <p>3.1. Просит заполнить таблицу ЗХУ.</p> <p>3.2. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> <li>• Доработать задания;</li> <li>• Изучить источники Интернет по работе с Flash (<i>приложение 8</i>).</li> </ul>                                                                                                                                                                                                                                                                                                                        | Задают уточняющие вопросы.                                                                                                                                                                           |

## Приложение 1

### Тема: Создание интерактивных заданий с проверкой правильности их решения и внедрением звукового сопровождения в среде Flash

**Цель занятия:** освоить технологию создания тестовых заданий с использованием текстовых полей во Flash: статический текст (Static Text), динамический текст (Dynamic Text) и текст, вводимый пользователем (Input Text)

#### План занятия:

1. Виды текста во Flash.
2. Векторизация растровых изображений .
3. Выполнение индивидуального задания.

## Приложение 2

| Знал(а) такие типы текстовых блоков, как | Хочу узнать про текстовые блоки... | Узнал(а) про текстовые блоки... |
|------------------------------------------|------------------------------------|---------------------------------|
|                                          |                                    |                                 |

## Приложение 3

В данном занятии мы начнем с Вами осваивать технологию работы с текстовыми полями во Flash. Текст является важной составляющей любого электронного ресурса ввиду того, что он несет наибольшую информационную нагрузку. Текст во Flash – очень простой инструмент, поскольку очень многие настройки можно установить интуитивно.

Существует несколько базовых понятий, которые необходимо знать для успешного использования текстовых полей в ваших ЦОРах. Flash поддерживает три вида текста: **статический** (static text), **динамический** (dynamic text) и **вводимый** (input text).

Из трех типов форматов текста статический текст (Static Text) используется чаще всего и с ним мы уже с Вами работали, создавая каждый ЦОР. На прошлом занятии мы использовали динамический текст (Dynamic Text), который может быть изменен Flash во время работы SWF-файла, например, вывод сообщения «Верно!» или «Неверно!» в зависимости от результата выполнения задания. В поле с динамическим текстом можно выводить количество набранных баллов и исходя из этого оценку за выполнение задания ЦОР. Ключевая разница между режимами Dynamic Text и Input Text состоит в том, что в текстовое поле пользователь может сам вводить текст. Именно работе с текстовым полем input text более подробно и будет посвящено наше занятие.

Технология работы с текстовыми полями во Flash позволяет создавать тесты с ответами, вводимыми с клавиатуры, по прохождению которых ученику выдается результат теста, содержащий количество набранных баллов, оценку за работу.

## Приложение 4

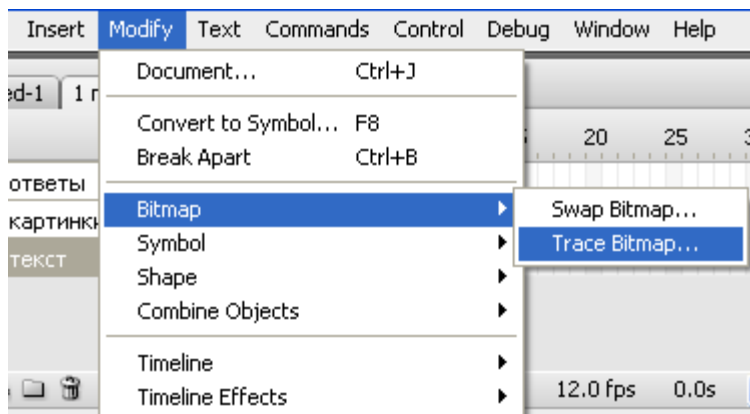
### Векторизация растровых изображений во Flash

Во время выполнения предыдущих заданий поднимался вопрос о том, как сделать прозрачный фон у изображений. Одним из вариантов является предварительная обработка изображения в программе Adobe Photoshop. Однако и во Flash можно избавиться от фона, если импортированное растровое изображение преобразовать в векторное. Эту технологию мы и освоим сейчас.

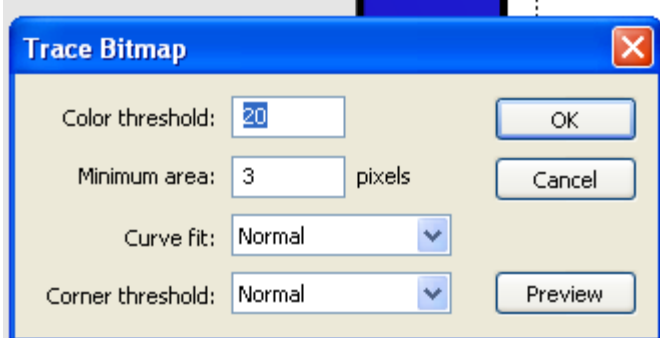
1. Импортируем растровое изображение светофора в рабочую область нового слоя «Картинки»:



2. Выделяем наше растровое изображение и выбираем в меню следующий пункт: **Modify**(Модификация) → **Bitmap** (Растр) → **Trace Bitmap** (Просмотр растра):



3. В открывшемся диалоговом окне **Trace Bitmap** мы должны в поле ввода **Color Threshold** (цветовой порог) задать числовое значение разницы между цветами, которые программа векторизации должна воспринимать как один и тот же цвет. Можно вводить числа в пределах от 1 до 500. Значение по умолчанию равно 100. В поле ввода **Minimum Area** (минимальная зона) указывается количество пикселей, которые должны быть учтены при усредненном определении цвета. Можно вводить число от 1 до 1000, значение по умолчанию равно 8. Следующие два списка **Curve Fit** (совпадение кривой) и **Corner Threshold** (угловой порог предлагают на выбор несколько степеней точности определения контуров, очерчивающих однотонные области, и углов.)
4. Выберем следующие параметры:



Вот какое векторное изображение должно получиться:



5. Теперь избавимся от фона. Для этого щелкаем мышкой по фону (о том, что он выделен, можно судить по появившимся точкам):



И нажимаем кнопку Delete на клавиатуре.

Затем выделяем все изображение и преобразовываем в символ – тип графика.



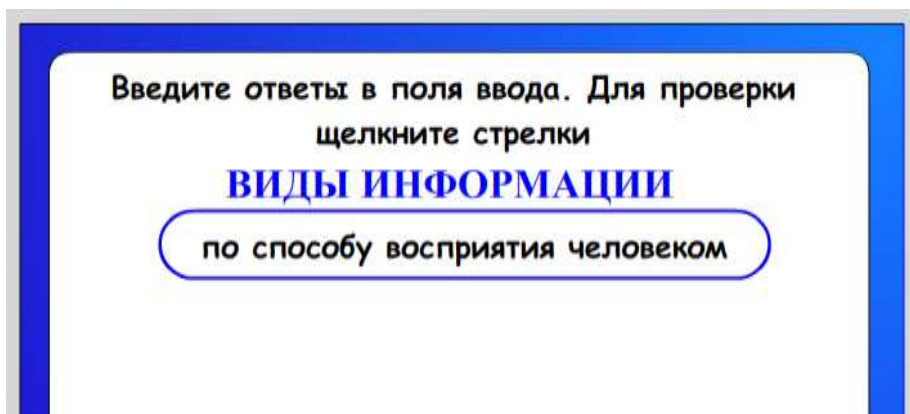
6. Удаляем исходное изображение из библиотеки. Теперь оно нам не нужно.
7. Вы можете поэкспериментировать и выбрать вариант векторизации подходящий именно Вам.

## Приложение 5

### Работа с текстовыми полями в одном кадре

#### Алгоритм выполнения работы:

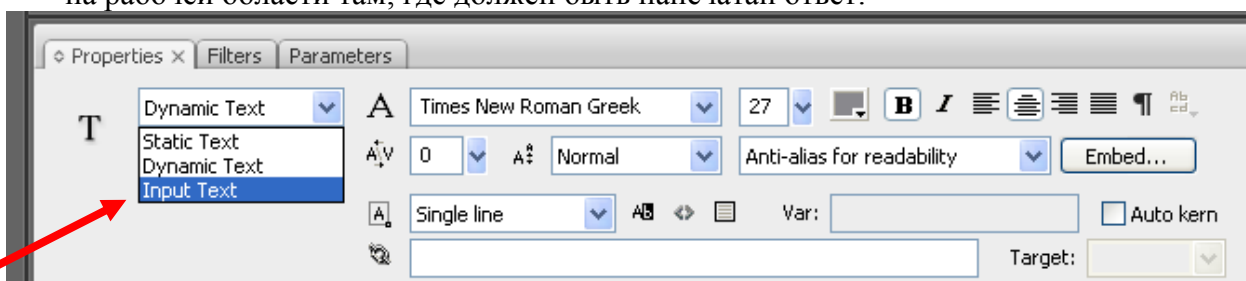
1. Откройте новый документ Flash.
2. Разместите на сцене статический текст (static text) (слой с именем «Текст»), как показано на рисунке:



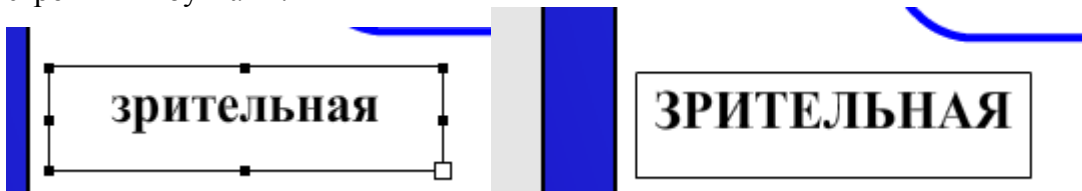
3. Согласно описанному в приложении 3 алгоритму подготовьте изображения:



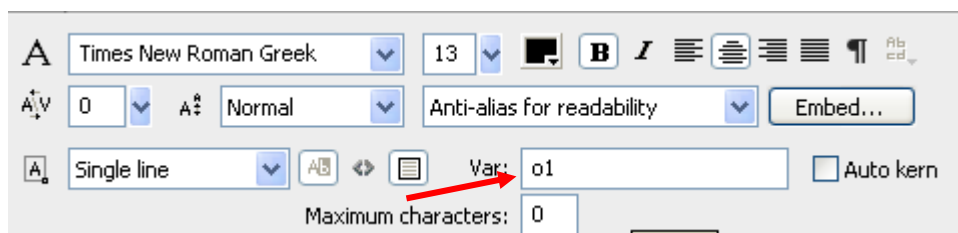
4. Теперь разместим на новом слое с именем «Ответы» поля для ввода текста (Input text). Для этого выбираем инструмент T(Text Tool) и на вкладке Свойства (Properties) выбираем тип текста Input text и располагаем *поле для ввода текста* на рабочей области там, где должен быть напечатан ответ:




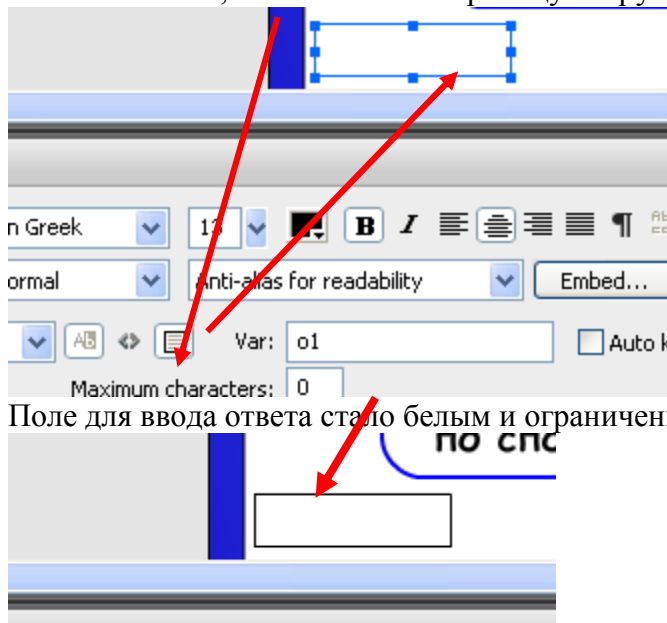
5. Устанавливаем размер шрифта, а также цвет текста, например – черный. (Замечание: Для проверки ввода стоит напечатать текст в поле, а потом его убрать, при этом расположение текста должно быть по левому краю). При вводе также учитывайте, что пользователь может напечатать текст прописными буквами, значит места нужно чуть больше чем под текст, напечатанный строчными буквами.



6. Теперь для созданного текстового поля необходимо назначить переменную (var) с именем o1:

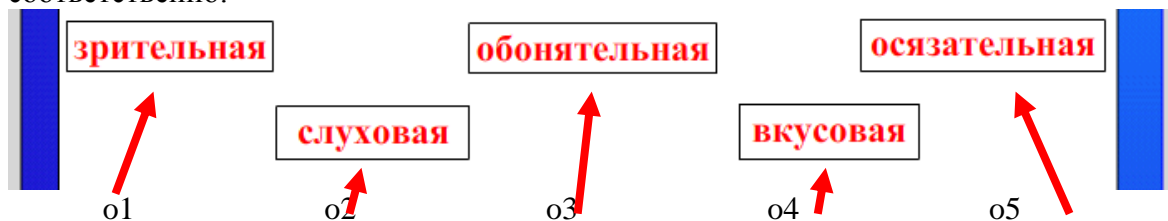


7. Для того, чтобы ученику было видно поле в которое нужно водить текст нажимаем , чтобы показать границу вокруг вводимого текста.

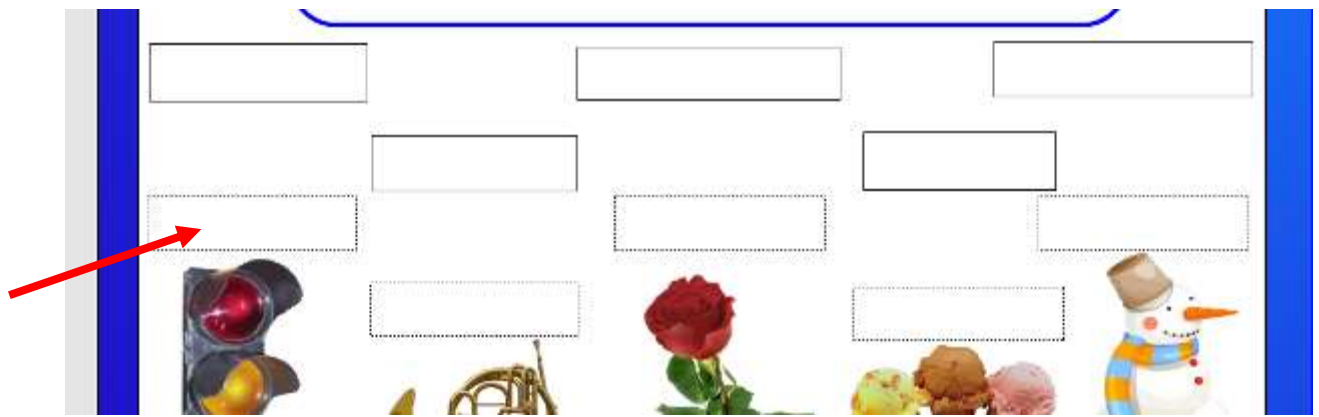


Поле для ввода ответа стало белым и ограниченным рамкой.

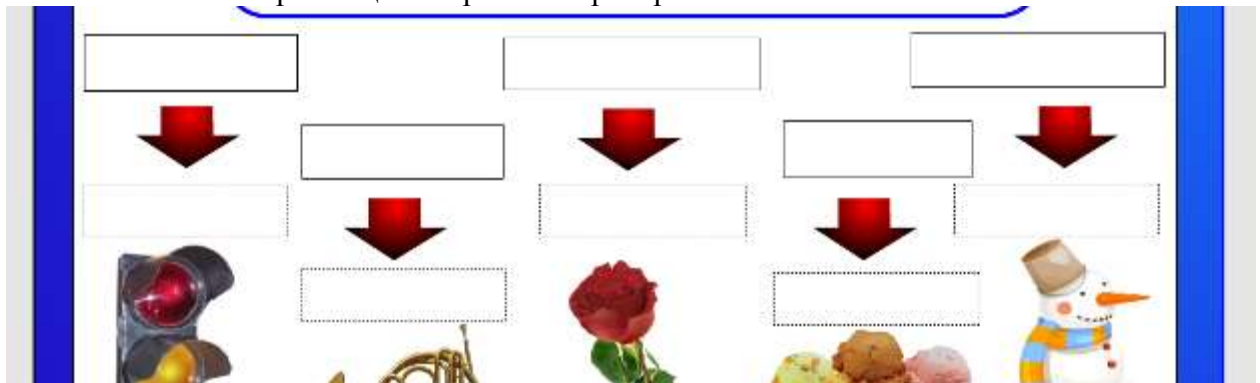
8. Аналогичным образом создаем остальные поля для ввода и назначаем для созданных текстовых полей переменную (var) с именами o2, o3, o4, o5 соответственно:



9. В этом же слое размещаем поля с динамическим текстом (Dynamic Text). Если снять выделение с поля с динамическим текстом, то можно увидеть, что оно выделено пунктирной рамкой. В данные поля будет выводиться сообщение «Верно!» или «Неверно!». Назначаем им переменную (var) с именами r1, r2, r3, r4, r5 соответственно:



10. В этом же слое размещаем стрелки и преобразовываем их в символ – тип кнопки.



11. Прописываем для каждой кнопки скрипт проверки введенного в поле ответа:

**Скрипт для первой кнопки:**

```
on (release) {
 if (o1=='зрительная' || o1=='Зрительная' || o1=='ЗРИТЕЛЬНАЯ') {
 b1 = 1; //1 балл за верный ответ
 r1 = "Верно!"; //Вывод сообщения в динамическое поле
 } else {
 b1 = 0; //0 баллов за неверный ответ
 r1 = "Неверно!"; //Вывод сообщения в динамическое поле
 }
}
```

#### Комментарий к представленному выше скрипту

Единственное, что для Вас нового в данном скрипте – это сложное логическое условие в операторе If. В ActionScript 2.0.:

&& - логическое И;

|| - логическое ИЛИ;

! - логическое НЕ;

Т.е. мы предусматриваем возможность ввода ответа в нескольких вариантах: строчными, с заглавной и прописными буквами.

Переменная b1 используется для подсчета количества верных ответов, чтобы в итоге можно было выставить оценку.

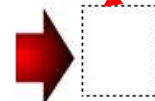
Продолжим...

12. Теперь кнопку можно скопировать столько раз, сколько нужно и в скрипте поменять только имена переменных o2, r2 и b2 и т.д. соответственно.

13. Теперь добавим еще одну кнопку и поле с динамическим текстом (имя переменной rez) для выставления итоговой оценки за работу.

Введите ответы в поля ввода. Для проверки  
щелкните стрелки

## ВИДЫ ИНФОРМАЦИИ



Скрипт для кнопки будет следующим:

```
on (release) {
 i = b1+b2+b3+b4+b5; //Подсчет количества набранных баллов
 if (i==5){ // Если набранных баллов 5
 rez = "5"; //В динамическое поле выводится оценка 5
 } else if (i==4){ // Если набранных баллов 4
 rez = "4"; //В динамическое поле выводится оценка 4
 } else if (i==3){ // Если набранных баллов 3
 rez = "3"; //В динамическое поле выводится оценка 3
 } else { // В остальных случаях
 rez = "2"; //В динамическое поле выводится оценка 2
 }
}
```

14. Самостоятельно добавляем код, разворачивающий клип на весь экран, кнопку выхода и озвучиваем кнопку выставления итоговой оценки.
15. Нажимаем Ctrl+Enter - тестируем проект!
16. Публикуем в формате swf.

### Приложение 6

#### Задания для индивидуального выполнения

1. Спланируйте ЦОР в рамках одного из изучаемых или изученных ранее предметов по специальности.
2. Подберите необходимые изображения.
3. Создайте свой ресурс с использованием рассмотренной технологии, содержащий несколько полей для ввода ответов, пошаговую проверку правильности ввода и вывод итоговой оценки за работу.
4. Обязательно укажите ФИО, курс и факультет, на котором вы учитесь.
5. Сохраните ролик с именем ФИО\_pr4, например, *StepanenkoOV\_pr4.fl*
6. Опубликуйте ролик.

### Приложение 7

#### Для зачета по лабораторному занятию:

1. Оформите пояснительную записку к ролику.
2. Файлы (*StepanenkoOV\_sr4.fl*, *StepanenkoOV\_sr4.swf*, *StepanenkoOV\_zapiska.doc*) заархивируйте (*stepanenkoOV\_zad4.rar*) и сдайте преподавателю.

### Приложение 8

#### Источники:

1. Лекции дистанционного мастер-класса Е.А. Томиловой «Первые шаги во Flash»: [http://it-n.ru/communities.aspx?cat\\_no=73740&lib\\_no=91030&tmpl=lib](http://it-n.ru/communities.aspx?cat_no=73740&lib_no=91030&tmpl=lib)
2. Матвеева Н.В. Плакат «Виды информации по способу восприятия человеком».
3. Котенок [http://www.otkritka.com/images/otkritka-koshki\\_23.jpg](http://www.otkritka.com/images/otkritka-koshki_23.jpg)
4. Светофор [http://byaki.net/uploads/posts/1187638754\\_2.jpg](http://byaki.net/uploads/posts/1187638754_2.jpg)
5. Роза <http://i82.beon.ru/78/19/841978/59/23740859/0.png>
6. Мороженое [http://free3.chechnyafree.ru/upload/iblock/2f7/ice\\_cream.jpg](http://free3.chechnyafree.ru/upload/iblock/2f7/ice_cream.jpg)
7. Снеговик <http://i029.radikal.ru/0804/1b/21dacdc22a67t.jpg>

|                |                                                             |
|----------------|-------------------------------------------------------------|
| <b>Тема 20</b> | <b>Создание дидактических игр с помощью программы Flash</b> |
|----------------|-------------------------------------------------------------|

### 1.1. Технология обучения на лабораторном занятии

|                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                                                                                                                                            |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Учебное время: 2 часа.</i>                                                                                                                                                                                                                                       |                                                                                                                                                                                                                                                                                                                                            |
| <i>Форма учебного занятия: лабораторная работа</i>                                                                                                                                                                                                                  |                                                                                                                                                                                                                                                                                                                                            |
| <i>План занятия</i>                                                                                                                                                                                                                                                 | <ol style="list-style-type: none"> <li>1. Работа с текстовыми полями в нескольких кадрах. Игра «Редактор».</li> <li>2. Вставка в ЦОР компонента «полоса прокрутки».</li> <li>3. Выполнение индивидуального задания.</li> </ol>                                                                                                             |
| <i>Цель учебного занятия:</i>                                                                                                                                                                                                                                       | освоить технологию создания дидактических игр с помощью текстовых полей с полосой прокрутки;                                                                                                                                                                                                                                               |
| <i>Педагогические задачи:</i><br>Организовать работу студентов по: <ul style="list-style-type: none"> <li>• использованию различных видов текста во Flash;</li> <li>• векторизации растровых изображений;</li> <li>• выполнению индивидуального задания.</li> </ul> | <i>Результаты учебной деятельности:</i><br><i>Студент должен уметь:</i> <ul style="list-style-type: none"> <li>• использовать различные виды текста во Flash.</li> <li>• Производить векторизацию растровых изображений.</li> <li>• самостоятельно разрабатывать и выполнять образовательный ресурс по аналогии с предложенным.</li> </ul> |
| <i>Методы обучения</i>                                                                                                                                                                                                                                              | Категориальная таблица                                                                                                                                                                                                                                                                                                                     |
| <i>Средства обучения</i>                                                                                                                                                                                                                                            | персональные компьютеры с установленной программой Adobe Flash, печатная версия задания                                                                                                                                                                                                                                                    |
| <i>Формы организации учебной деятельности</i>                                                                                                                                                                                                                       | Фронтальная, индивидуальная.                                                                                                                                                                                                                                                                                                               |
| <i>Условия обучения</i>                                                                                                                                                                                                                                             | Кабинет вычислительной техники                                                                                                                                                                                                                                                                                                             |

### Технологическая карта лабораторного занятия

| Этап работы, время                           | Содержание деятельности                                                                                                                                                                                                                                         |                                                                                            |
|----------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------|
|                                              | Преподавателя                                                                                                                                                                                                                                                   | Студентов                                                                                  |
| 1-й этап. Введение в учебное занятие (5 мин) | 1.4. Сообщает, тему, цель учебного занятия и план его проведения ( <i>приложение 1</i> ).                                                                                                                                                                       | Слушают, уточняют, задают вопросы.                                                         |
| 2-й этап. Основная часть (65 мин)            | 2.1. Предлагает выполнить работу с текстовыми полями в нескольких кадрах для создания дидактической игры «Редактор» ( <i>приложение 2</i> ).<br>2.2. Раскрывает возможности применения полосы прокрутки для текстовых полей и организует работу студентов по их | 2.1. Берут задания и выполняют за компьютерами.<br>2.2. Слушают.<br>2.3. Выполняют задания |

|                                                                         |                                                                                                                                                                                                                                                                                                                                                         |                                                                                                                       |
|-------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------|
|                                                                         | <p>добавлению в разработанной дидактической игре (<i>приложение 3</i>).</p> <p>2.3. Раздает задания для индивидуального выполнения (<i>приложение 4</i>) и разъясняет форму отчета по лабораторной работе (<i>приложение 5</i>).</p>                                                                                                                    | <p>лабораторной работы.</p> <p>2.3. Выполняют индивидуальные задания и подготавливают отчет о проделанной работе.</p> |
| <p>3-й этап.<br/>Заключитель-<br/>но-результат-<br/>ный<br/>(10 ин)</p> | <p>3.1. Просит заполнить категориальную таблицу, в которой отобразить особенности текстовых полей во Flash (<i>приложение 6</i>).</p> <p>3.2. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> <li>• Доработать задания;</li> <li>• Изучить источники Интернет по работе с Flash (<i>приложение 7</i>).</li> </ul> | <p>Задают уточняющие вопросы.</p>                                                                                     |

## Приложение 1

### Тема: Создание интерактивных заданий с проверкой правильности их решения и внедрением звукового сопровождения в среде Flash

**Цель занятия:** освоить технологию создания дидактических игр с помощью текстовых полей с полосой прокрутки.

#### План занятия

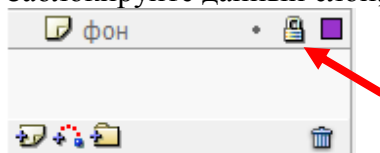
1. Работа с текстовыми полями в нескольких кадрах. Игра «Редактор».
2. Вставка в ЦОР компонента «полоса прокрутки».
3. Выполнение индивидуального задания.

## Приложение 2

### Работа с текстовыми полями в нескольких кадрах. Игра «Редактор»

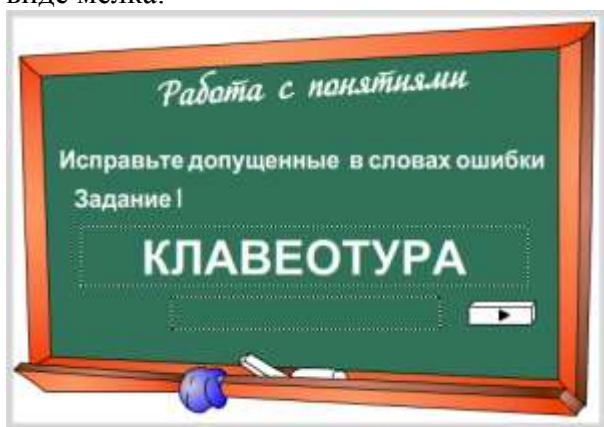
#### Алгоритм выполнения работы:

1. Откройте новый документ Flash.
2. Разместите на сцене «Фон» отрисованное изображение школьной доски. Заблокируйте данный слой, установив замочек.

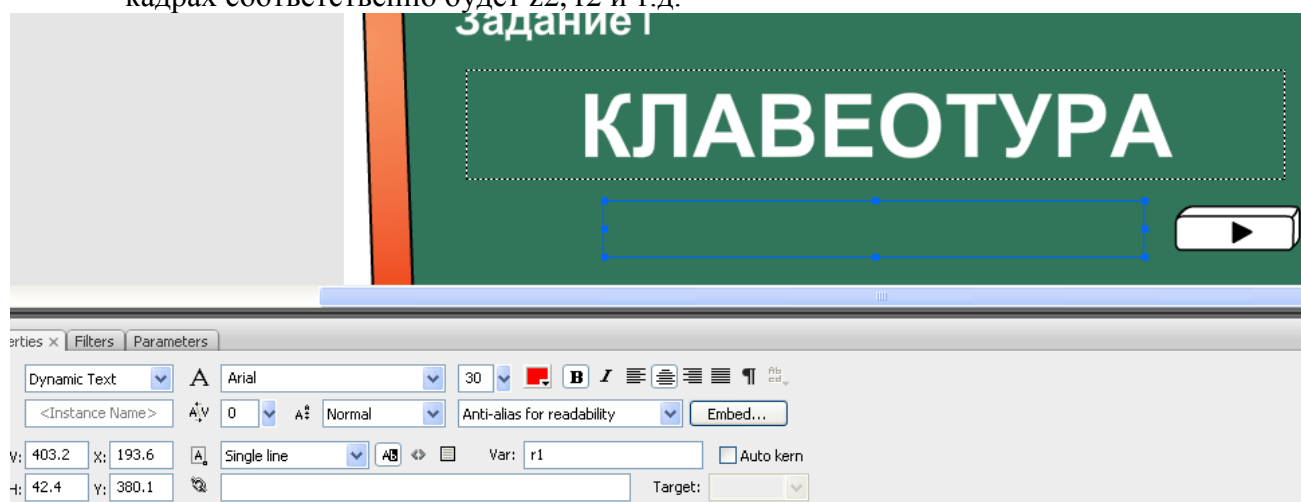


3. Создайте новый слой с именем «Задания» и разместите на нем статический текст: «Работа над понятиями», «Исправьте допущенные в словах ошибки», «Задание 1».
4. В 1-м кадре слоя «Задания» разместите поле для ввода текста (Input Text) «Клавеотура» (здесь неправильный текст *клавеотура* необходимо напечатать в поле и не удалять), поле с динамическим текстом (Dynamic Text) под сообщение

«Вы не исправили ошибку!» и нарисуйте кнопку перехода на следующий кадр в виде мелка:

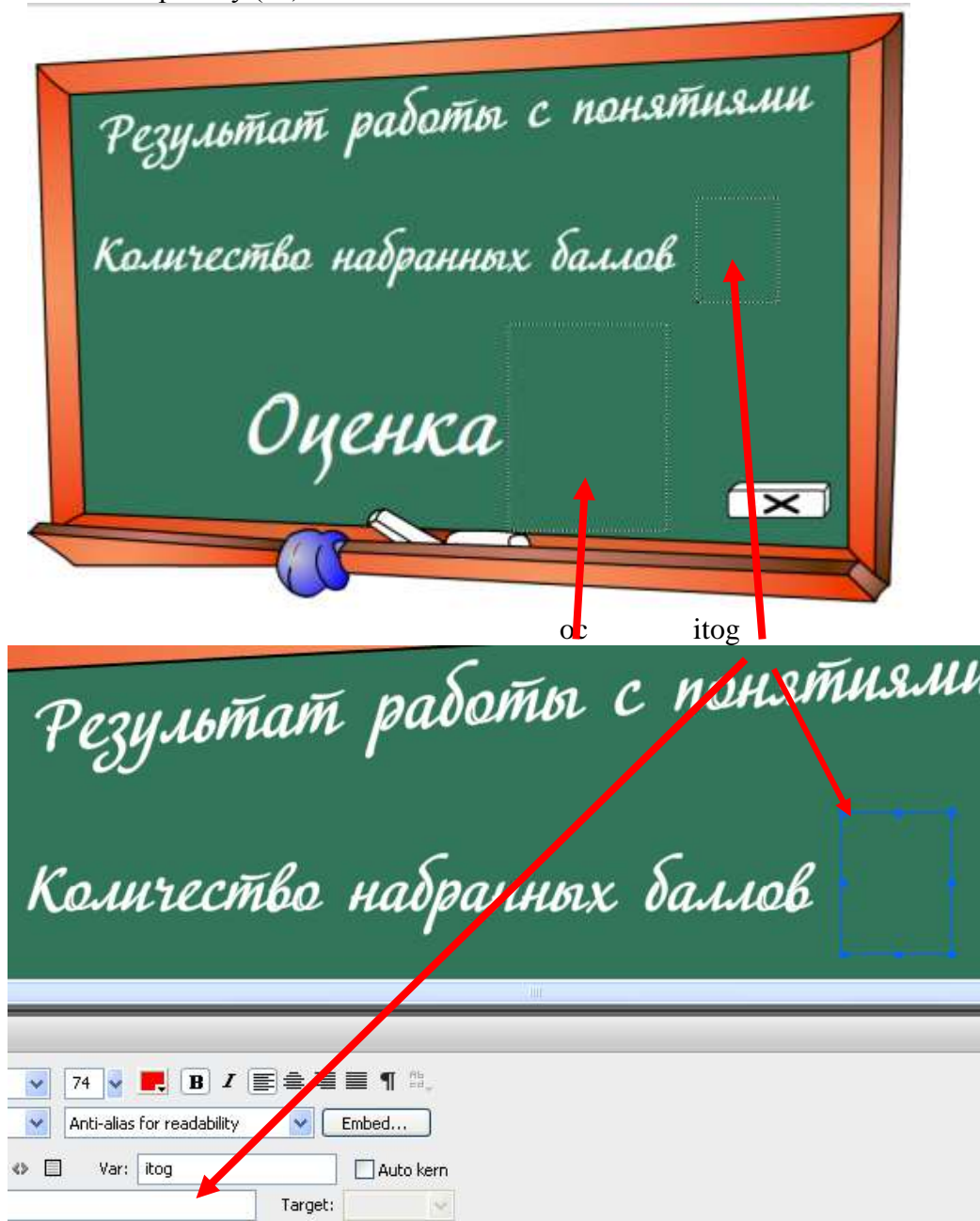


5. Теперь для созданных текстовых полей необходимо назначить переменную (var). Поле для ввода – z1, поле с динамическим текстом – r1. Во 2-м и последующих кадрах соответственно будет z2, r2 и т.д.



6. Преобразовываем мелок в символ – тип кнопка и прописываем следующий код:
- ```
on (release) {
    if (z1=='КЛАВИАТУРА' || z1=='клавиатура' || z1==Клавиатура) { //Если набран
        текст в одном из трех вариантов
        i1 = 1;
        gotoAndPlay(2); //Переход на второй кадр
    } else {
        i1 = 0;
        r1 = "Вы не исправили ошибку!"; //Выдается сообщение в
        динамическое поле
    }
}
```
7. Прописываем в 1-кадре слоя «Фон» действие: stop();
8. Ставим курсор во 2-й кадр слоя «Задания» и нажимаем F6.
9. Вносим необходимые изменения в тексте, именах переменных и в коде кнопки.
10. Аналогичным образом (пункт 7-9) добавляем еще несколько кадров с заданиями.

11. На последнем кадре слоя «Задания» размещаем два поля с динамическим текстом: 1) для вывода количества набранных баллов (itog); 2) поле для вывода оценки за работу (oc).



12. В последний кадр слоя «Задания» добавляем следующий код:

```
itog = i1+i2+i3+i4+i5+i6+i7
if (itog==7){ // если набранных баллов 7
oc = "5";
} else if (itog>=5){ // если набранных баллов от 5 до 6
oc = "4";
} else if (itog==4){ // если набранных баллов 4
oc = "3";
} else { // в остальных случаях
oc = "2";
}
```

13. Самостоятельно добавляем код, разворачивающий клип на весь экран, кнопку выхода.
14. В 1 кадре обнуляем переменные, отвечающие за подсчет баллов:
i1=0; i2=0; i3=0; i4=0; i5=0; i6=0; i7=0;
15. Нажимаем Ctrl+Enter - тестируем проект!

16. Вы, наверное, уже обратили внимание на то, что переход к следующему заданию возможен только после того, как исправлены все ошибки, т.е. задание выполнено правильно. Предлагаю скрипт для кнопки, которая будет запускать анимированный ролик, если задание выполнено верно и выдавать сообщение, если задание выполнено неверно (можно выводить и другую анимацию – по желанию). А как и куда – решаем сами!

```
on (release) {
    if (z1=='КЛАВИАТУРА' || z1=='клавиатура' || z1=='Клавиатура') { //если в
поле для ввода текст

        {
            with (_root.nout)
            {
                verno.attachSound("verno");
                verno.start();
                r1=""; //обнуляем динамическое поле, если ранее было
выведено сообщение
                i1=1; //1 балл за верный ответ
                gotoAndPlay(2); //проигрываем клип с именем nout со 2 кадра
            } // End of with
        }

    } else {
        neverno.attachSound("neverno");
        neverno.start();
        i1 = 0; //0 баллов за неверный ответ
        r1 = "Вы не исправили ошибку!"; //в динамическое поле выводим
сообщение
    }
}
```

17. Самостоятельно исправляем скрипт кнопки-мелка, чтобы была возможность переходить к выполнению следующего задания, даже если предыдущее выполнено неверно.
18. Публикуем в формате swf.

Разрабатывая задания по данной технологии для редактирования учащимся можно предлагать не только отдельные слова, но и целые предложения.

Приложение 3

Работа с динамическим полем, содержащим компонент «полоса прокрутки»

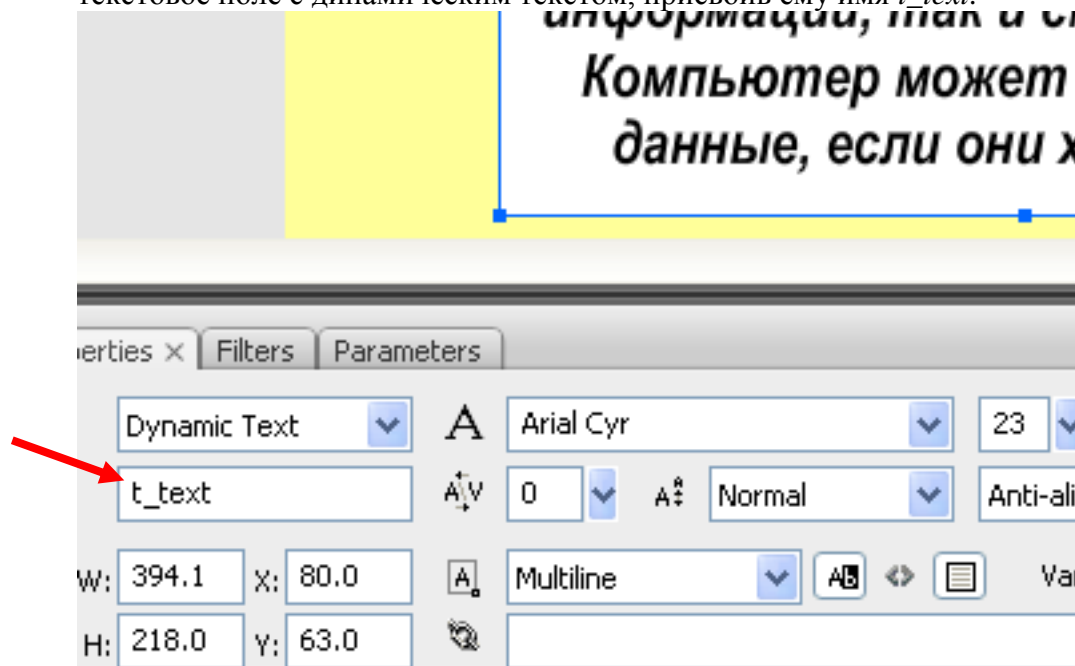
В рамках данного занятия предлагаю освоить возможность вставки в ЦОР компонента «полоса прокрутки».

Компонент ScrollBar обеспечивает функциональность на уровне drag-and-drop при добавлении вертикальных и горизонтальных полос прокрутки для динамических

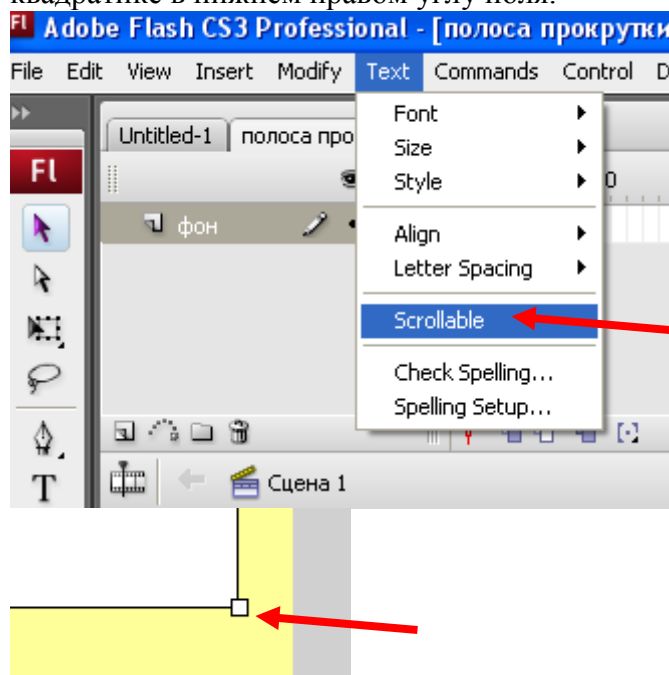
текстовых полей, а также для полей ввода. Добавление скроллбаров к текстовым полям позволяет данным полям принимать большие количества текста, не требуя при этом просмотра всего текста сразу.

Алгоритм выполнения работы:

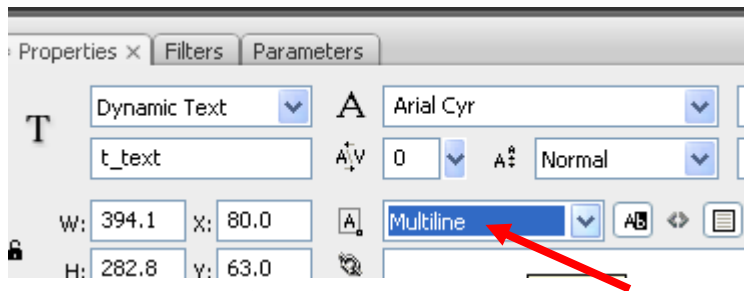
1. Откройте новый документ Flash.
2. Переименуйте существующий слой в «Фон», установите цвет фона по желанию или добавьте фоновую картинку.
3. Выберите инструмент Текст (Т) и создайте в нужном месте и нужного размера текстовое поле с динамическим текстом, присвоив ему имя *t_text*:



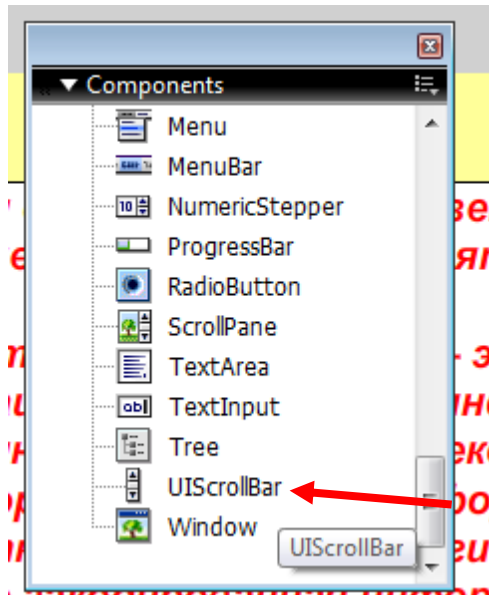
4. Для того, чтобы сделать поле прокручиваемым, нужно поставить галочку в главном меню *Текст (Text) → С прокруткой (Scrollable)* или *Shift+двойной клик* на квадратике в нижнем правом углу поля.



5. Для созданного поля устанавливаем свойство *многострочный (Multiline)*:

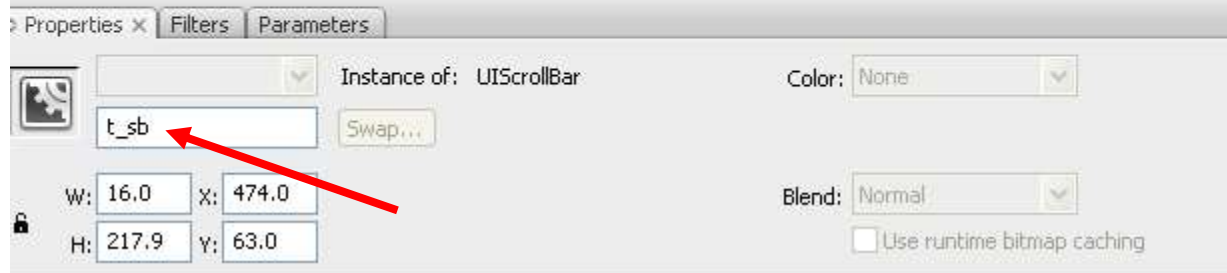


6. Теперь в поле набираем или вставляем текст.
7. Нажимаем Ctrl+Enter - тестируем проект!
Обратите внимание, что текст уже прокручиваемый, но у него нет скроллбара и прокручивать можно только колесиком мыши или стрелками. Чтобы это исправить и используется `UIScrollBar`.
8. Для того, чтобы разместить полосу прокрутки нужно выбрать пункт главного меню *Окно (Window) → Компоненты (Components)* или нажать сочетание клавиш *Ctrl+F7* и в появившемся окне найти `UIScrollBar`:



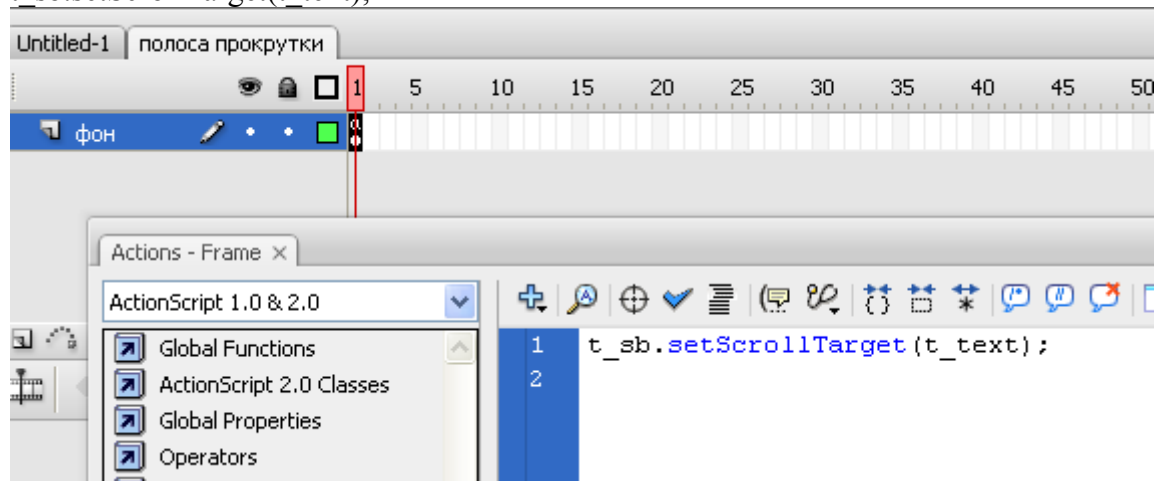
9. Полосу прокрутки перетаскиваем из меню, размещаем рядом с текстом (размер и размещение не имеют принципиального значения) и задаем ей имя `t_sb`:

**Человек может обрабатывать
числовую, текстовую, графическую
информацию.
При обработке информации может
изменяться как форма представления
информации, так и смысл сообщений.
Компьютер может обрабатывать
данные, если они хранятся в его**



10. Для кадра, в котором размещена полоса прокрутки, прописываем следующее действие:

```
t_sb.setScrollTarget(t_text);
```



11. Нажимаем Ctrl+Enter - тестируем проект!

12. Иногда бывает необходимо подгружать текстовую информацию из внешнего файла. Для этого в кадре, в котором размещена полоса прокрутки, дописываем следующий код:

```
System.useCodepage = true  
t_sb.setScrollTarget(t_text);  
var my_lv:LoadVars = new LoadVars();  
my_lv.onData = function(src:String) {
```

```

if (src != undefined) {
t_text.text = src;
} else {
t_text.text = "Ошибка при загрузке текста"; }
};
my_lv.load("1.txt");

```

Комментарий к представленному выше скрипту

В строке `System.useCodepage = true` байт 0xDE перекодирован в 0x42E, т.е. будет переведен в Unicode формат. Если эту строчку не использовать, то вместо русских букв во Flash будут выведены «крякозябли» ☺

`t_sb.setScrollTarget(t_text);` //полоса прокрутки содержащая поле с динамическим текстом с именем `t_text`

`var my_lv:LoadVars = new LoadVars();` //описывается новая переменная для загрузки

`my_lv.onData = function(src:String) {` // функция (переменная `src` с типом строка)
`if (src != undefined) {` //если глобальная переменная содержит элементарное неопределенное значение - то, которое имеет переменная, значение которой не указано

`t_text.text = src;` //в текстовое поле `t_text` загрузится текст

`} else {` //иначе

`t_text.text = "Ошибка при загрузке текста";` //выведется сообщение об ошибке загрузки файла

`my_lv.load("1.txt");` //загрузка содержимого текстового файла `1.txt`, расположенного в той же директории, что и ролик.

Продолжим...

13. Удаляем текст из поля `t_text` и создаем в той же папке файл `1.txt` с необходимым нам текстом (этот текст и будет загружаться в поле с полосой прокрутки).

14. Нажимаем `Ctrl+Enter` - тестируем проект!

Приложение 4

Задания для индивидуального выполнения

1. Спланируйте ЦОР в рамках одного из изучаемых или изученных ранее предметов по специальности.
2. Создайте свой ресурс с использованием рассмотренных технологий, содержащий 5 заданий на редактирование текста и динамическое поле с полосой прокрутки; внесите необходимые изменения в код подсчета оценки за работу.
3. Обязательно укажите ФИО, курс и факультет, на котором вы учитесь.
4. Сохраните ролик с именем ФИО_pr5, например, *StepanenkoOV_pr5 fla*
5. Опубликуйте ролик.

Приложение 5

Для зачета по лабораторному занятию:

1. Оформите пояснительную записку к ролику.
2. Файлы (*StepanenkoOV_pr5 fla*, *StepanenkoOV_pr5.swf*, *StepanenkoOV_zapiska.doc*) заархивируйте (*stepanenkoOV_zan5.rar*) и сдайте преподавателю.

Приложение 6

Категориальная таблица «Виды текста во Flash»

статический (static text)	динамический (dynamic text)	вводимый (input text)

Приложение 7

Источники

1. Лекции дистанционного мастер-класса Е.А. Томиловой «Первые шаги во Flash»: http://it-n.ru/communities.aspx?cat_no=73740&lib_no=91030&tmpl=lib
2. Школьная доска <http://i053.radikal.ru/0806/e4/708fb8d049ed.jpg>

Тема 21	Создание электронного пера и теста на установление соответствия при помощи цвета во Flash
----------------	--

1.1. Технология обучения на лабораторном занятии

<i>Учебное время:</i> 2 часа.	
<i>Форма учебного занятия:</i> лабораторная работа	
<i>План занятия</i>	<ol style="list-style-type: none"> 1. Создание электронного пера в виде инструмента «Карандаш». 2. Создание теста на установление соответствия при помощи цвета. 3. Кнопка с параметром Alpha 4. Выполнение индивидуального задания.
<i>Цель учебного занятия:</i>	освоить технологию работы с цветом, создание авторского инструмента «карандаш», разработки теста на установление соответствия при помощи цвета;
<i>Педагогические задачи:</i> Организовать работу студентов по: <ul style="list-style-type: none"> • Созданию электронного пера в виде инструмента «Карандаш»; • созданию теста на установление соответствия при помощи цвета; • разработке кнопки с параметром Alpha; • выполнению индивидуального задания. 	<i>Результаты учебной деятельности:</i> <i>Студент должен уметь:</i> <ul style="list-style-type: none"> • Создавать электронного пера в виде инструмента «Карандаш». • Создавать тест на установление соответствия при помощи цвета. • Разрабатывать кнопку с параметром Alpha. • самостоятельно разрабатывать и выполнять образовательный ресурс по аналогии с предложенным.
<i>Методы обучения</i>	Частично-поисковый метод обучения
<i>Средства обучения</i>	персональные компьютеры с установленной программой Adobe Flash, печатная версия задания
<i>Формы организации учебной деятельности</i>	Фронтальная, индивидуальная.
<i>Условия обучения</i>	Кабинет вычислительной техники

Технологическая карта лабораторного занятия

Этап работы, время	Содержание деятельности	
	Преподавателя	Студентов
1-й этап. Введение в учебное занятие (5 мин)	1.5. Сообщает, тему, цель учебного занятия и план его проведения (<i>приложение 1</i>).	Слушают, уточняют, задают вопросы.

2-й этап. Основная часть (65 мин)	<p>2.1. Предлагает выполнить работу по созданию инструмента «Карандаш» (<i>приложение 2</i>).</p> <p>2.2. Организует работу студентов по созданию тестов на установление соответствия при помощи цвета (<i>приложение 3</i>).</p> <p>2.3. Описывает этапы разработки кнопки с параметром Alpha (<i>приложение 4</i>).</p> <p>2.4. Раздает задания для индивидуального выполнения (<i>приложение 5</i>) и разъясняет форму отчета по лабораторной работе (<i>приложение 6</i>).</p>	<p>2.1. Берут задания и выполняют за компьютерами.</p> <p>2.2. Слушают.</p> <p>2.3. Выполняют задания лабораторной работы.</p> <p>2.3. Выполняют индивидуальные задания и подготавливают отчет о проделанной работе.</p>
3-й этап. Заключитель- но-результат- ный (11 мин)	<p>3.3. Дает и разъясняет задания для домашней работы:</p> <ul style="list-style-type: none"> • Доработать задания; • Изучить источники Интернет по работе с Flash (<i>приложение 7</i>). 	Задают уточняющие вопросы.

Приложение 1

Тема: Создание электронного пера и теста на установление соответствия при помощи цвета во Flash

Цель занятия: освоить технологию работы с цветом, создание авторского инструмента «карандаш», разработки теста на установление соответствия при помощи цвета.

План занятия

1. Создание электронного пера в виде инструмента «Карандаш».
2. Создание теста на установление соответствия при помощи цвета.
3. Кнопка с параметром Alpha
4. Выполнение индивидуального задания.

Приложение 2

Используя электронное перо (функция граффити), на интерактивной доске можно написать, начертить, дорисовать, нарисовать, подчеркнуть, зачеркнуть т.п. В разрабатываемом во Flash авторском цифровом образовательном ресурсе также можно создать свое электронное перо. В данном занятии предлагаем освоить технологию работы с цветом во Flash, познакомиться с созданием инструмента «карандаш» и возможностью разработки теста на установление соответствия при помощи цвета. А

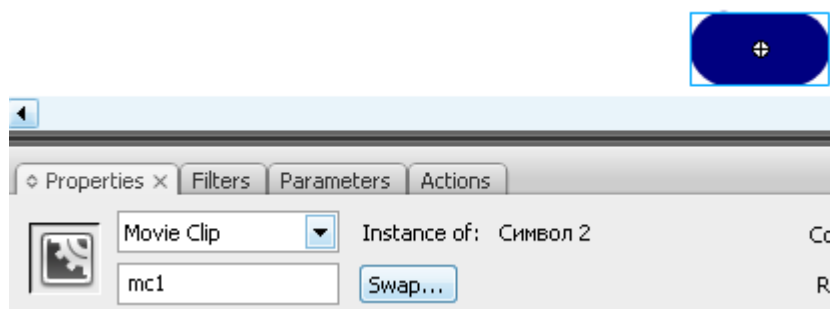
также мы научимся загружать графические изображения в формате gif и использовать их для создания клипа и создавать кнопку с параметром Alpha.

1. Создание инструмента «Карандаш»

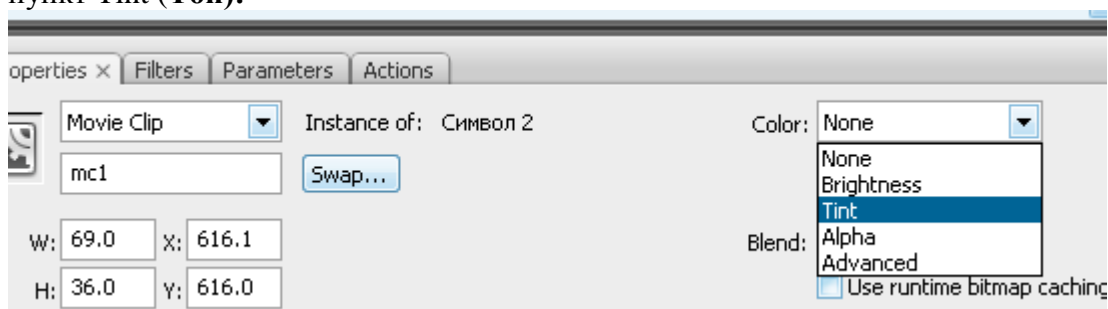
1. Откройте новый документ Flash.
2. Разместите в 1-м кадре слоя «Фон» статический текст *Придумайте по картинке нужные суждения* и картинку так, как показано на рисунке:



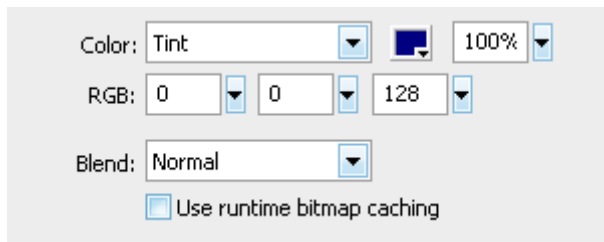
3. Добавьте новый слой с именем «Карандаш».
4. В 1-м кадре слоя «Карандаш» добавьте прямоугольник с радиусом углов 20 точек и синим цветом заливки.
5. Преобразовываем добавленный прямоугольник в символ – тип клип и присваиваем ему имя mc1:



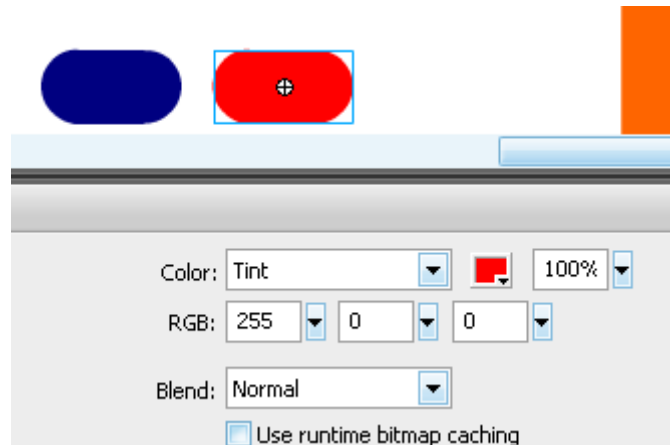
6. На вкладке **Свойства** выбираем из выпадающего списка свойства Color (**Цвет**) пункт Tint (**Тон**):



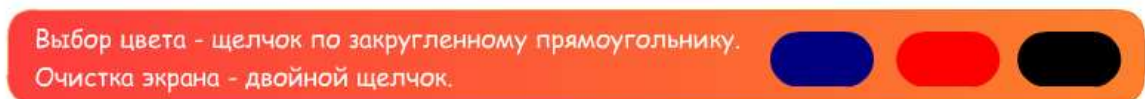
7. Устанавливаем следующие параметры RGB – 0, 0, 128:



8. Перетаскиваем из библиотеки наш символ – синий закругленный прямоугольник, присваиваем ему имя mc2 и вновь вносим изменения на вкладке Свойства – Цвет – Тон – красный:



9. Аналогичным образом (п. 7-8) добавляем еще один цвет – черный.
10. Добавляем инструкцию по работе с инструментом «карандаш»: *Выбор цвета - щелчок по закругленному прямоугольнику. Очистка экрана - двойной щелчок.*



11. В 1-й кадр слоя «Карандаш» добавляем следующий код:

```
var color = 0xFFFFFF; // Белый цвет по умолчанию
// Начинаем рисование при нажатии левой кнопки мыши
this.onMouseDown = function() {
    penUp = false; // Перо опущено
    this.moveTo(_xmouse, _ymouse);
    this.lineStyle(2, color);
};
// Если кнопка мыши нажата, при перемещении курсора мыши продолжаем рисование
this.onMouseMove = function() {
    if (!penUp) { // Если перо опущено...
        this.lineTo(_xmouse, _ymouse);
        updateAfterEvent();
    }
};
// Прекращаем рисование при отпускании кнопки мыши
this.onMouseUp = function() {
    penUp = true; // Перо поднято
};
```

12. Ниже под предыдущим кодом добавляем код выбора цвета палитры карандаша:
- ```
for (i = 0; i < 4; i++) {
 this["mc" + i].onPress = function() {
```

```

switch (Number(this._name.substr(-1))) {
case 1 :
 color = 0x000080; // Темно-синий
 break;
case 2 :
 color = 0xff0000; //Красный
 break;
case 3 :
 color = 0x000000; //Черный
 break;
}
};
}

```

Чтобы число считалось шестнадцатеричным, в языке ActionScript необходимо поставить перед ним префикс **0x**. Таким образом, число 0xFF0000 соответствует чистому красному цвету, 0x00FF00 - зеленому и 0x0000FF - синему. Аналогично, значение 0xFF6600 означает оранжевый цвет, 0xFFFFFFFF - белый, 0xFFFF00 - желтый и так далее. Узнать код нужного цвета всегда можно на вкладке Color.

13. Для того, чтобы экран можно было очистить, используем двойной щелчок мыши и под предыдущим кодом добавляем следующий:

// ---ОБНАРУЖЕНИЕ ДВОЙНОГО ЩЕЛЧКА-----

```

var delay:Number = 300;
var myBroadcaster:Object = new Object();
AsBroadcaster.initialize(myBroadcaster);
var myListener:Object = new Object();
myBroadcaster.addListener(myListener);
Mouse.addListener(myListener);
myListener.onMouseDown = function() {
 clickInterval = getTimer() - clickTime;
 clickTime = getTimer();
 if (clickInterval < delay) {
 myBroadcaster.broadcastMessage("onDoubleClick");
 clickTime = -delay;
 }
}

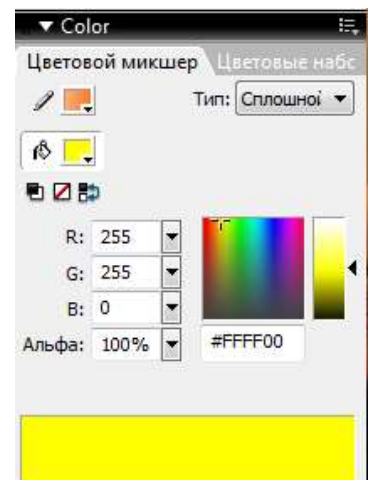
```

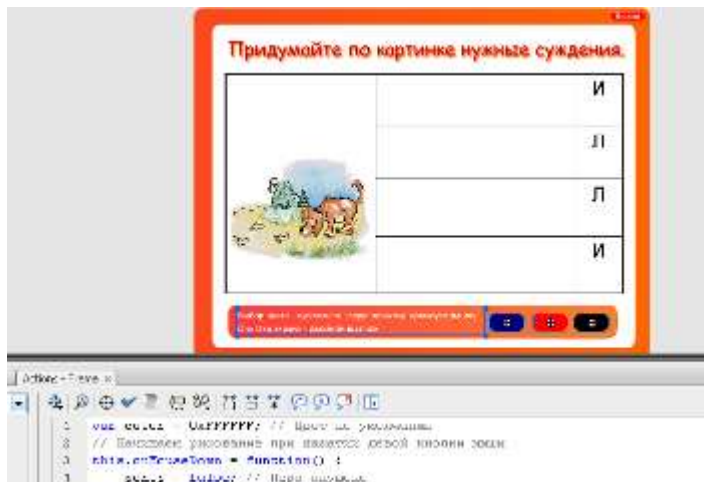
//-----ОЧИСТКА ЭКРАНА ПРИ ДВОЙНОМ ЩЕЛЧКЕ-----

```

myListener.onDoubleClick = function() { //Функция двойной щелчок
 clear();
};

```





### Комментарий к представленному выше скрипту

Метод `lineStyle` (<толщина линии в пикселах (0-255)>, <цвет в формате 0xRRGGBB>, <непрозрачность (0-100)>) задает параметры стиля линии программного рисования. Стиль линии необходимо задавать перед ее рисованием.

Метод `moveTo(x, y)` размещает виртуальное «перо» в начальной точке рисования. В качестве параметров выступают координаты этой начальной точки. Координаты отсчитываются от точки фиксации клипа.

Метод `lineTo(x, y)` соединяет прямой текущую точку и точку с координатами x, y.

Продолжим...

14. Самостоятельно добавляем код, разворачивающий клип на весь экран, и кнопку выхода.
15. Нажимаем Ctrl+Enter - тестируем проект!
16. Публикуем в формате swf.

## Приложение 3

### 2. Создание теста на установление соответствия при помощи цвета

1. Откройте новый документ Flash.
2. Установите цвет фона по желанию, размер рабочего поля. Добавьте статический текст: «Установите соответствие при помощи цвета», «Обонятельная информация», «Вкусовая информация», «Зрительная информация».
3. В 1-м кадре разместите три кнопки с такими скриптами:

**Красная**

```
on (release) {
 _global.cvet = 0xff0000;
}
```

**Синяя**

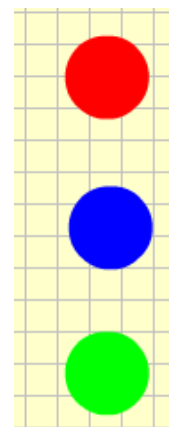
```
on (release) {
 _global.cvet = 0x0000FF;
}
```

**Зеленая**

```
on (release) {
 _global.cvet = 0x00FF00;
}
```

4. Создаем еще три кнопки (заливка белая, контур коричневый)

**Кнопка с именем b1**



```

on (release) {
 myColor = new Color(b1);
 myColor.setRGB(_global.cvet);
}

```

**Кнопка с именем b2**

```

on (release) {
 myColor = new Color(b2);
 myColor.setRGB(_global.cvet);
}

```

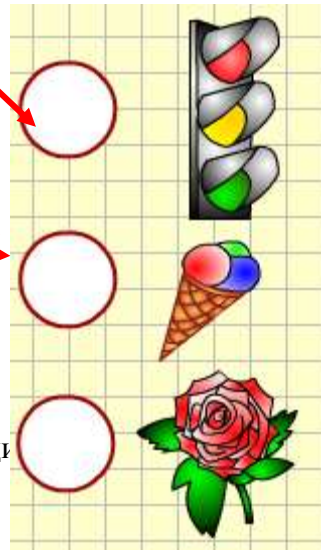
**Кнопка с именем b3**

```

on (release) {
 myColor = new Color(b3);
 myColor.setRGB(_global.cvet);
}

```

Добавляем соответствующие изображения.



5. Добавляем кнопку проверки со следующим кодом:
 

```

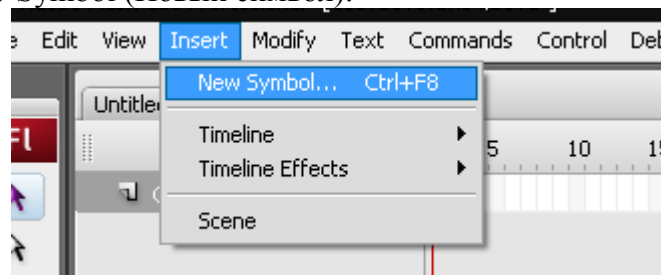
on (press) {
 c1 = new Color(b1);
 c2 = new Color(b2);
 c3 = new Color(b3);
 if ((c1.getRGB()== 0x00FF00) && (c2.getRGB()== 0x0000FF) && (c3.getRGB()== 0xff0000)) {
 t = "Верно!";
 } else {
 t = "Неверно!";
 }
}

```
6. Размещаем динамическое поле с именем **t**.
7. Самостоятельно добавляем код, разворачивающий клип на весь экран, и кнопку выхода.
8. Нажимаем Ctrl+Enter - тестируем проект!

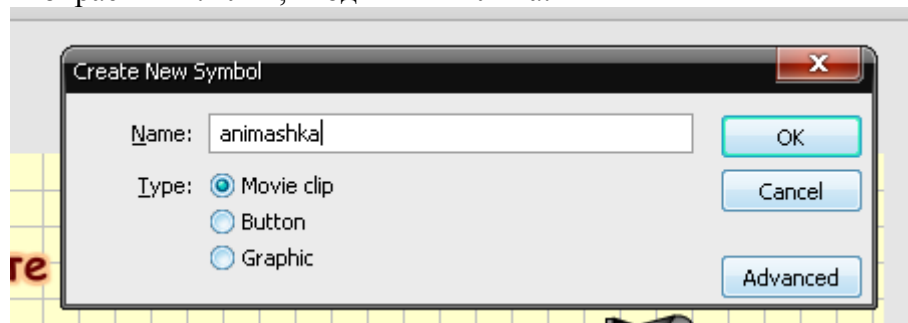


9. Для некоторых слушателей сложным оказалось в предыдущем занятии создание собственной анимации, поэтому я предлагаю научиться вставлять в ЦОР чужую (разумеется, с соблюдением авторских прав) «анимашку» в формате gif. Итак, для

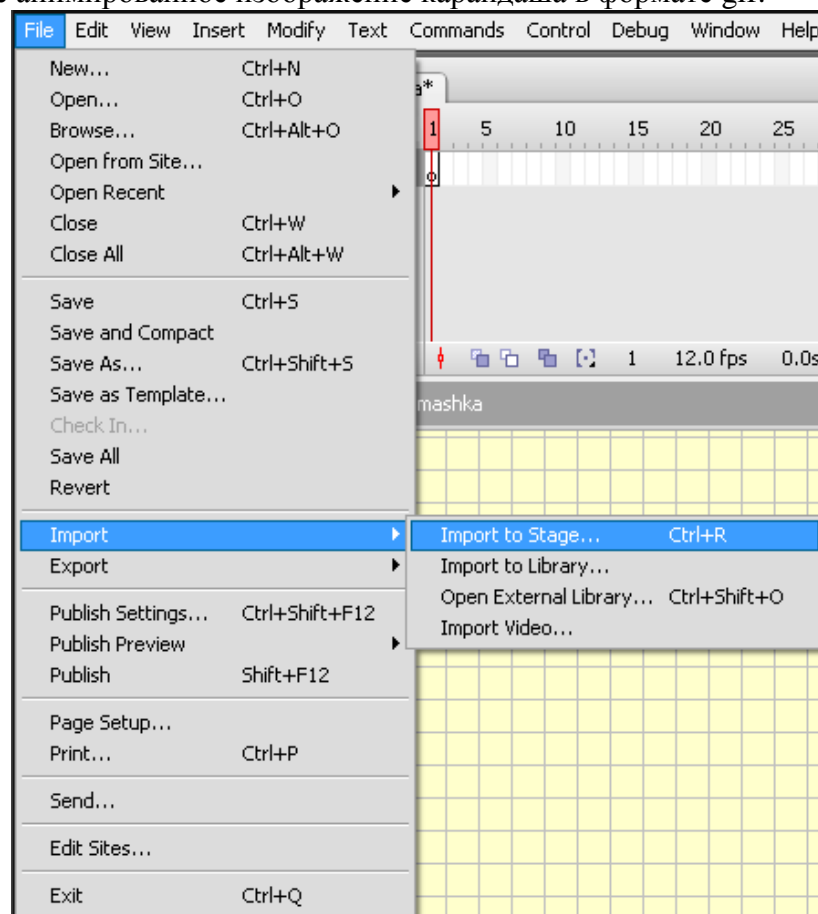
этого нам нужно создать новый символ. Выбираем пункт меню Insert (Вставка)→New Symbol (Новый символ):



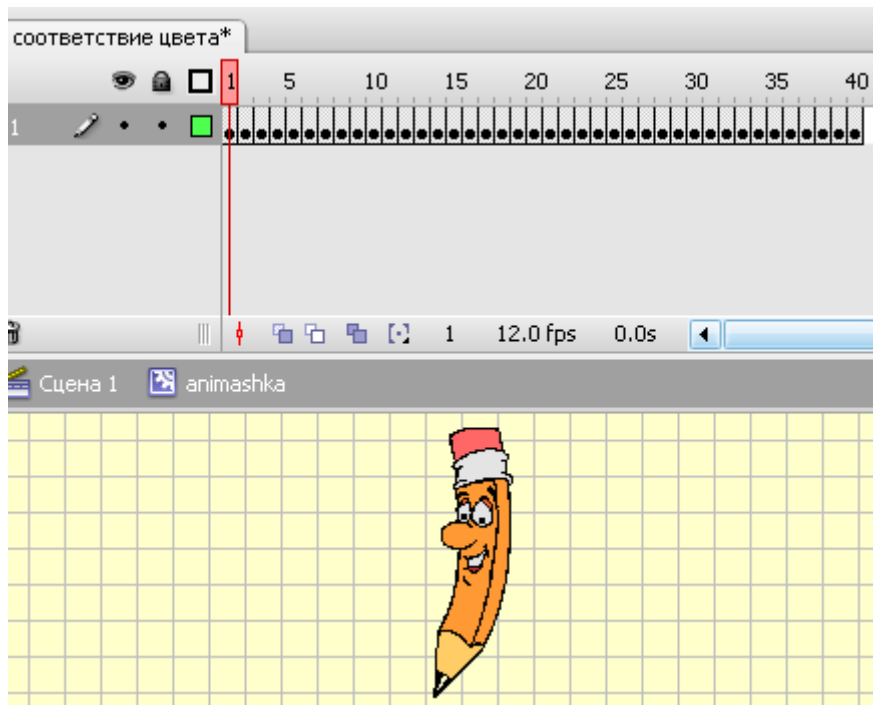
10. Выбираем тип: клип, вводим имя клипа:



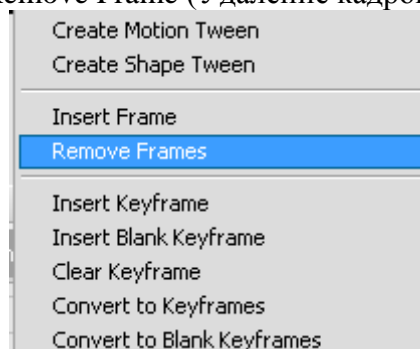
10. Мы оказываемся внутри клипа. Теперь импортируем заимствованное в Интернете анимированное изображение карандаша в формате gif:



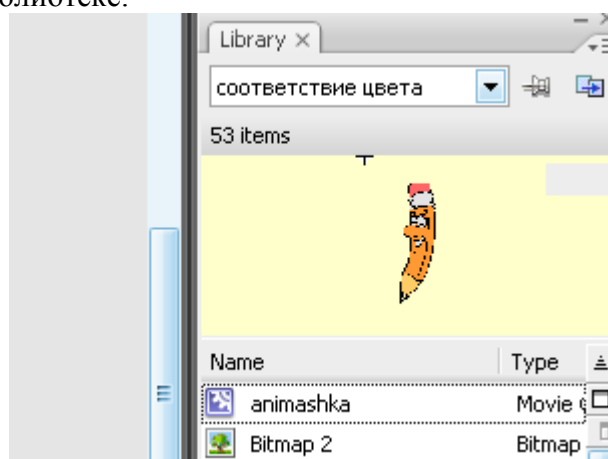
11. Анимация добавлена, и каждое действие карандаша распределилось по кадрам: у нас получилась покадовая анимация:



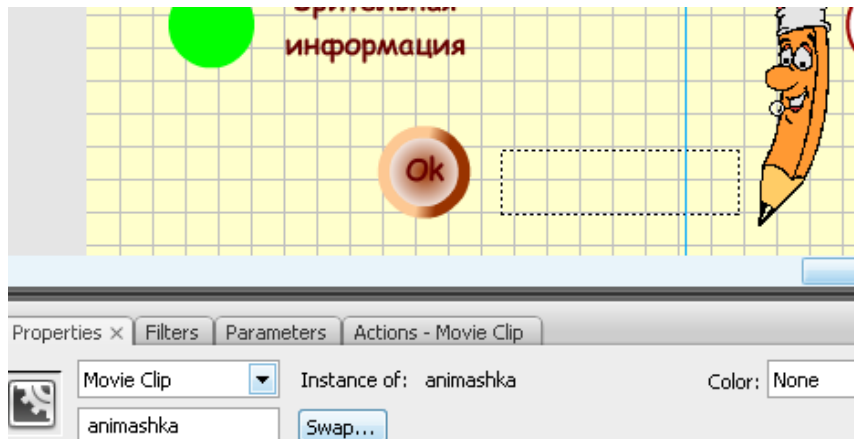
12. Далее удаляем лишние кадры (когда карандаш стирает), для этого выделяем ненужные кадры и щелкаем левой клавишей мыши и в открывшемся диалоговом меню выбираем пункт Remove Frame (Удаление кадров):



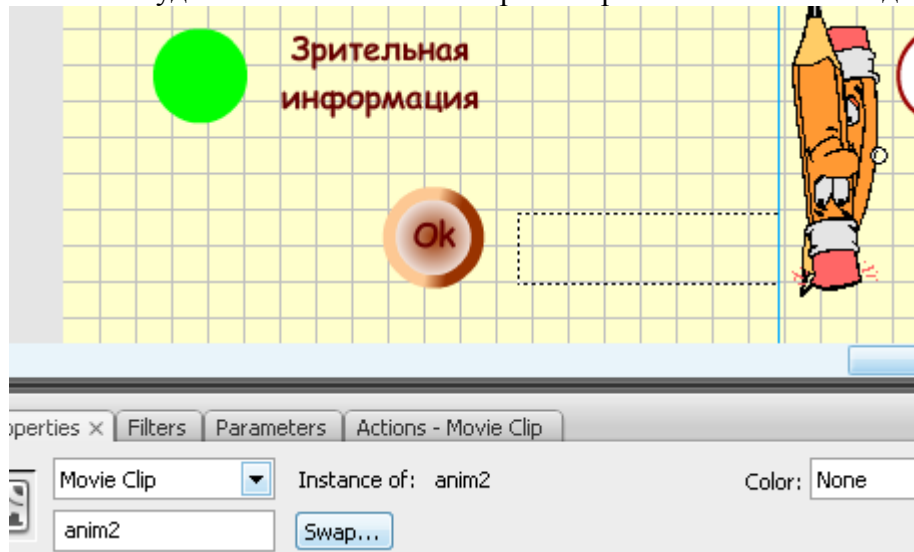
13. Выходим из символа, нажав синюю стрелку. Созданный таким образом символ размещается в библиотеке.



14. Перетаскиваем созданный символ на рабочее поле, присваиваем ему имя animashka. Данный клип будет появляться только при правильном выполнении задания:

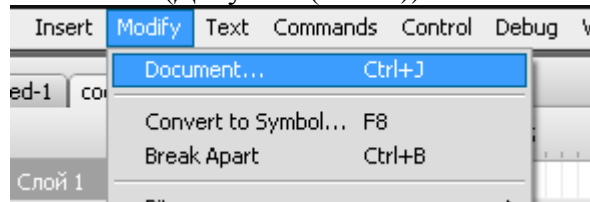


15. Аналогичным образом (п. 9-14) создаем клип с карандашом, который только стирает anim2 и будет появляться только при неверном выполнении задания.

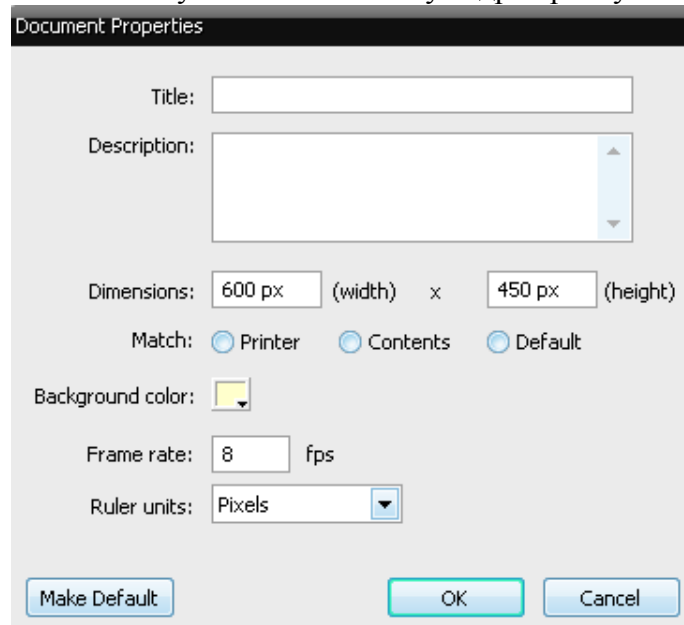


16. В кадр добавляем скрипт, позволяющий скрыть наши клипы:  
`animashka._visible = false; //карандаш невиден`  
`anim2._visible = false; //карандаш2 невиден`
17. Для того, чтобы через несколько секунд клип исчезал добавляем в кадр скрипт:  
`onEnterFrame = function()`  
`{ if ((getTimer() - start_time) / 1000 > 3)`  
`{animashka._visible = false;`  
`anim2._visible = false;`  
`}`  
`}`
18. Теперь мы должны отредактировать скрипт кнопки, чтобы клипы появлялись только при определенном условии  
 если верно:  
`animashka._visible = true;`  
`anim2._visible = false;`  
 если неверно:  
`anim2._visible = true;`  
`animashka._visible = false;`
19. Нажимаем Ctrl+Enter - тестируем проект!

20. Слишком «шустрый» получился карандаш. Чтобы уменьшить его скорость нужно выбрать клип (щелчок по клипу), открыть пункт меню Modify (Модификация)→Document...(Документ(Ctrl+J)).



21. Далее в открывшемся окне установить частоту кадров равную 8:



Чем больше число, тем частота кадров выше, а, следовательно, анимация «проигрывается» быстрее.

22. Публикуем в формате swf.

## Приложение 4

### 3. Кнопка с параметром Alpha

#### Алгоритм выполнения работы:

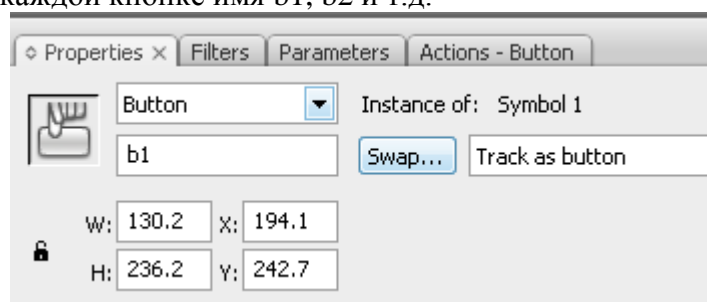
1. Открываем новый документ Flash.
2. Называем слой именем «Фон» и устанавливаем цвет фона по желанию. Я использовала в качестве заливки Bitmap (растровое изображение).
3. Размещаем на рабочем поле все необходимые объекты.



4. Выделяем каждое графическое изображение и преобразовываем в символ – тип кнопка.



5. Присваиваем каждой кнопке имя b1, b2 и т.д.



6. Для кнопки b1 прописываем следующее действие:  
on (release)

```
{
 if (pr == 0)
 {
 b1._alpha = 100;
 pr = 1;
 }
 else
```

```

 {
 b1._alpha = 50;
 pr = 0;
 } // end else if
}

```

И т.д.

7. Для кнопки Проверки скрипт будет таким:

```

on (release)
{
 if ((b2._alpha == 50) && (b3._alpha == 50) && (b5._alpha == 50) &&
 (b10._alpha == 50) && (b1._alpha == 100) && (b4._alpha == 100) &&
 (b6._alpha == 100) && (b7._alpha == 100) && (b9._alpha == 100))
 {
 res1 = "Верно!";
 z1=1;
 }
 else
 {
 res1 = "Неверно!";
 z1=0;
 } // end else if
}

```

8. Добавляем необходимое число кадров.  
 9. Самостоятельно добавляем код разворачивания клипа на весь экран и кнопку выхода.  
 10. Публикуем в формате swf.



## Приложение 5

### Задания для индивидуального выполнения

1. Спланируйте ЦОР в рамках одного из изучаемых или изученных ранее предметов по специальности.
2. Создайте свой ресурс с использованием рассмотренных технологий.
3. Обязательно укажите ФИО, курс и факультет, на котором вы учитесь.
4. Сохраните ролик с именем ФИО\_pr6, например, *StepanenkoOV\_pr6.fl*
5. Опубликуйте ролик.

## Приложение 6

### Для зачета по практическому занятию:

1. Оформите пояснительную записку к ролику.
2. Файлы (*StepanenkoOV\_pr6 fla*, *StepanenkoOV\_pr6 swf*, *StepanenkoOV\_zapiska.doc*) заархивируйте (*stepanenkoOV\_zan6.rar*) и сдайте преподавателю.

## Приложение 7

### Источники

1. Анимация «Карандаш» <http://aceweb.narod.ru/gif/gif2/gif/u/21.gif>
2. Альберт Д., Альберт Е. "ActionScript 2.0. Наиболее полное руководство (+ CD-ROM)". - БХВ-Петербург, 2005. – 1120 с.
3. Компьютерная графика  
<http://canon.osu.cctpu.edu.ru/diplom/flash/4.6.asp?ThemeID=4>
4. Ларина Э.С., Горностаева А.М. Обучающий семинар «Разработка дизайна приложения» <http://www.it-n.ru/Attachment.aspx?Id=47579>
5. Лекции дистанционного мастер-класса Е.А. Томиловой «Первые шаги во Flash»: [http://it-n.ru/communities.aspx?cat\\_no=73740&lib\\_no=91030&tmpl=lib](http://it-n.ru/communities.aspx?cat_no=73740&lib_no=91030&tmpl=lib)
6. Macromedia Flash MX и программирование на ActionScript  
<http://www.intuit.ru/departement/internet/flashmxascrip/14/5.html>
7. Монитор <http://www.materiel.net/live/93816.jpg>
8. Графический планшет <http://demiart.ru/forum/uploads/post-38031-1203548696.jpg>

|                |                                                                                                                                      |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------|
| <b>Тема 22</b> | <b>Технология разработки ЦОР на установление хронологической последовательности с использованием технологии перемещения объектов</b> |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------|

### 1.1. Технология обучения на лабораторном занятии

|                                                                                                                                                                                                                                          |                                                                                                                                                                                                                                                                                                          |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Учебное время: 2 часа.</i>                                                                                                                                                                                                            |                                                                                                                                                                                                                                                                                                          |
| <i>Форма учебного занятия: лабораторная работа</i>                                                                                                                                                                                       |                                                                                                                                                                                                                                                                                                          |
| <i>План занятия</i>                                                                                                                                                                                                                      | 1. Разработка ЦОР на установление хронологической последовательности.<br>2. Выполнение индивидуального задания.                                                                                                                                                                                          |
| <i>Цель учебного занятия:</i>                                                                                                                                                                                                            | освоить технологию разработки ЦОР на установление хронологической последовательности с использованием технологии перемещения объектов;                                                                                                                                                                   |
| <i>Педагогические задачи:</i><br>Организовать работу студентов по: <ul style="list-style-type: none"> <li>• разработке ЦОР на установление хронологической последовательности;</li> <li>• выполнению индивидуального задания.</li> </ul> | <i>Результаты учебной деятельности:</i><br><i>Студент должен уметь:</i> <ul style="list-style-type: none"> <li>• Разрабатывать ЦОР на установление хронологической последовательности.</li> <li>• Самостоятельно разрабатывать и выполнять образовательный ресурс по аналогии с предложенным.</li> </ul> |
| <i>Методы обучения</i>                                                                                                                                                                                                                   | Репродуктивный метод обучения                                                                                                                                                                                                                                                                            |
| <i>Средства обучения</i>                                                                                                                                                                                                                 | персональные компьютеры с установленной программой Adobe Flash, печатная версия задания                                                                                                                                                                                                                  |
| <i>Формы организации учебной деятельности</i>                                                                                                                                                                                            | Фронтальная, индивидуальная.                                                                                                                                                                                                                                                                             |
| <i>Условия обучения</i>                                                                                                                                                                                                                  | Кабинет вычислительной техники                                                                                                                                                                                                                                                                           |

### Технологическая карта лабораторного занятия

| Этап работы, время                              | Содержание деятельности                                                                                                                                                                                                                       |                                                          |
|-------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------|
|                                                 | Преподавателя                                                                                                                                                                                                                                 | Студентов                                                |
| 1-й этап.<br>Введение в учебное занятие (5 мин) | 1.6. Сообщает, тему, цель учебного занятия и план его проведения ( <i>приложение 1</i> ).                                                                                                                                                     | Слушают, уточняют, задают вопросы.                       |
| 2-й этап.<br>Основная часть (65 мин)            | 2.1. Дает описание ЦОР ( <i>приложение 2</i> ).<br>2.2. Организует работу студентов по созданию образовательного ресурса ( <i>приложение 3</i> ).<br>2.3. Раздает задания для индивидуального выполнения ( <i>приложение 4</i> ) и разъясняет | 2.1. слушают, уточняют.<br>2.2. выполняют задания на ПК. |

|                                                              |                                                                                                                                                                                                     |                                                                                    |
|--------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------|
|                                                              | форму отчета по лабораторной работе<br>(приложение 5).                                                                                                                                              | 2.3. Выполняют индивидуальные задания и подготавливают отчет о проделанной работе. |
| 3-й этап.<br>Заключитель-<br>но-результат-<br>ный<br>(12 ин) | 3.4. Дает и разъясняет задания для домашней работы: <ul style="list-style-type: none"> <li>• Доработать задания;</li> <li>• Изучить источники Интернет по работе с Flash (приложение 6).</li> </ul> | Задают уточняющие вопросы.                                                         |

## Приложение 1

### Тема: Технология разработки ЦОР на установление хронологической последовательности с использованием технологии перемещения объектов

**Цель занятия:** освоить технологию разработки ЦОР на установление хронологической последовательности с использованием технологии перемещения объектов.

#### План занятия

1. Разработка ЦОР на установление хронологической последовательности.
2. Выполнение индивидуального задания

## Приложение 2

### Технология разработки ЦОР на установление хронологической последовательности. Прием использования технологии перемещения объектов

При разработке данного урока использовался метод hitTest. Новым в данном уроке является:

1. Изменение цвета текста в зависимости от правильности выполнения задания.

### История ЭВМ.

#### Вехи истории вычислительной техники

Установите правильные соответствия между датами и их краткими описаниями.

1770 г.

Создается одно из первых механических вычислительных устройств - машина Якобсона

1956 г.

В СССР создана первая в стране серийная универсальная полупроводниковая управляемая ЭВМ шестого поколения "Днепр-1" (В.М.Глушков, Б.Н.Малиновский). Выпускалась на протяжении 10 лет

1961 г.

Создание и выпуск машин ЭВМ - СМ-1, СМ-2, СМ-3 и СМ-4 с широким диапазоном применений: в научных работах, для управления технологическими процессами, обработка экспериментальных данных в реальном масштабе времени и т.д.

1975 г.

Разработана ЭВМ БЭСМ-2. Руководитель разработки - С.А.Лебедев

1987 г.

Государственная комиссия приняла окончательный образцы нового поколения вычислительной техники УКБЦ ("Электроника МС 8202, 88115")

Проверить

### История ЭВМ.

#### Вехи истории вычислительной техники

Установите правильные соответствия между датами и их краткими описаниями.

1770 г.

Создается одно из первых механических вычислительных устройств - машина Якобсона

1956 г.

Государственная комиссия приняла окончательный образцы нового поколения вычислительной техники УКБЦ ("Электроника МС 8202, 88115")

1961 г.

В СССР создана первая в стране серийная универсальная полупроводниковая управляемая ЭВМ шестого поколения "Днепр-1" (В.М.Глушков, Б.Н.Малиновский). Выпускалась на протяжении 10 лет

1975 г.

Создание и выпуск машин ЭВМ - СМ-1, СМ-2, СМ-3 и СМ-4 с широким диапазоном применений: в научных работах, для управления технологическими процессами, обработка экспериментальных данных в реальном масштабе времени и т.д.

1987 г.

Разработана ЭВМ БЭСМ-2. Руководитель разработки - С.А.Лебедев

Проверить

Ошибок: 2  
Ваша оценка: 3

2. Вывод результатов выполнения интерактивного задания несколькими строками в одно текстовое поле.

Поздравляем!  
Ошибок нет.  
Ваша оценка 5

3. При перемещении одного объекта на место другого объекта происходит соответствующая замена их расположения.

### История ЭВМ.

#### Вехи истории вычислительной техники

Установите правильные соответствия между датами и их краткими описаниями.

1770 г.

Создается одно из первых механических вычислительных устройств - машина Якобсона

1956 г.

Разработана ЭВМ БЭСМ-2. Руководитель разработки - С.А.Лебедев

1961 г.

Государственная комиссия приняла окончательный образцы нового поколения вычислительной техники УКБЦ ("Электроника МС 8202, 88115")

1975 г.

Создание и выпуск машин ЭВМ - СМ-1, СМ-2, СМ-3 и СМ-4 с широким диапазоном применений: в научных работах, для управления технологическими процессами, обработка экспериментальных данных в реальном масштабе времени и т.д.

1987 г.

В СССР создана первая в стране серийная универсальная полупроводниковая управляемая ЭВМ шестого поколения "Днепр-1" (В.М.Глушков, Б.Н.Малиновский). Выпускалась на протяжении 10 лет

Проверить

Ошибок: 2  
Ваша оценка: 3

### История ЭВМ.

#### Вехи истории вычислительной техники

Установите правильные соответствия между датами и их краткими описаниями.

1770 г.

Создается одно из первых механических вычислительных устройств - машина Якобсона

1956 г.

Разработана ЭВМ БЭСМ-2. Руководитель разработки - С.А.Лебедев

1961 г.

В СССР создана первая в стране серийная универсальная полупроводниковая управляемая ЭВМ шестого поколения "Днепр-1" (В.М.Глушков, Б.Н.Малиновский). Выпускалась на протяжении 10 лет

1975 г.

Создание и выпуск машин ЭВМ - СМ-1, СМ-2, СМ-3 и СМ-4 с широким диапазоном применений: в научных работах, для управления технологическими процессами, обработка экспериментальных данных в реальном масштабе времени и т.д.

1987 г.

Государственная комиссия приняла окончательный образцы нового поколения вычислительной техники УКБЦ ("Электроника МС 8202, 88115")

Проверить

Ошибок: 2  
Ваша оценка: 3



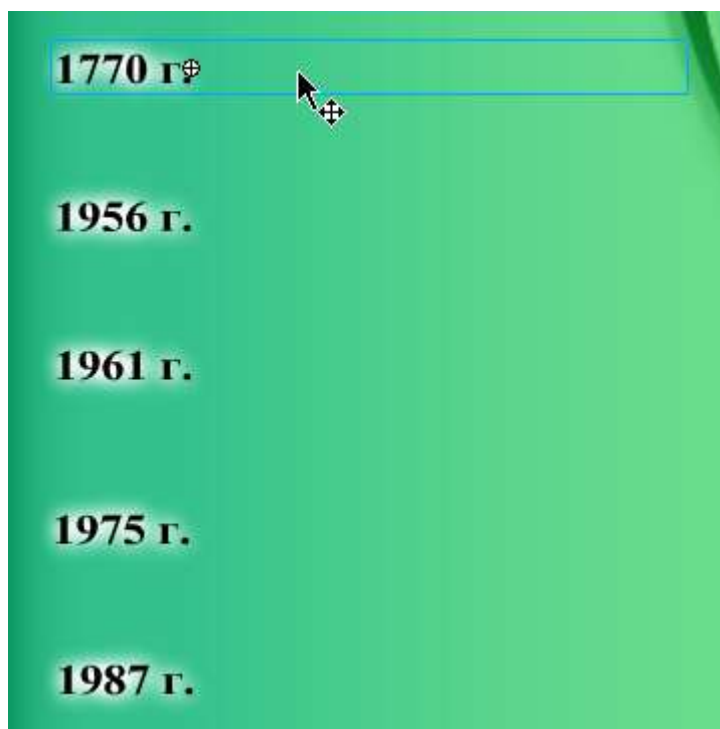
### Приложение 3

#### Алгоритм разработки ЦОРа

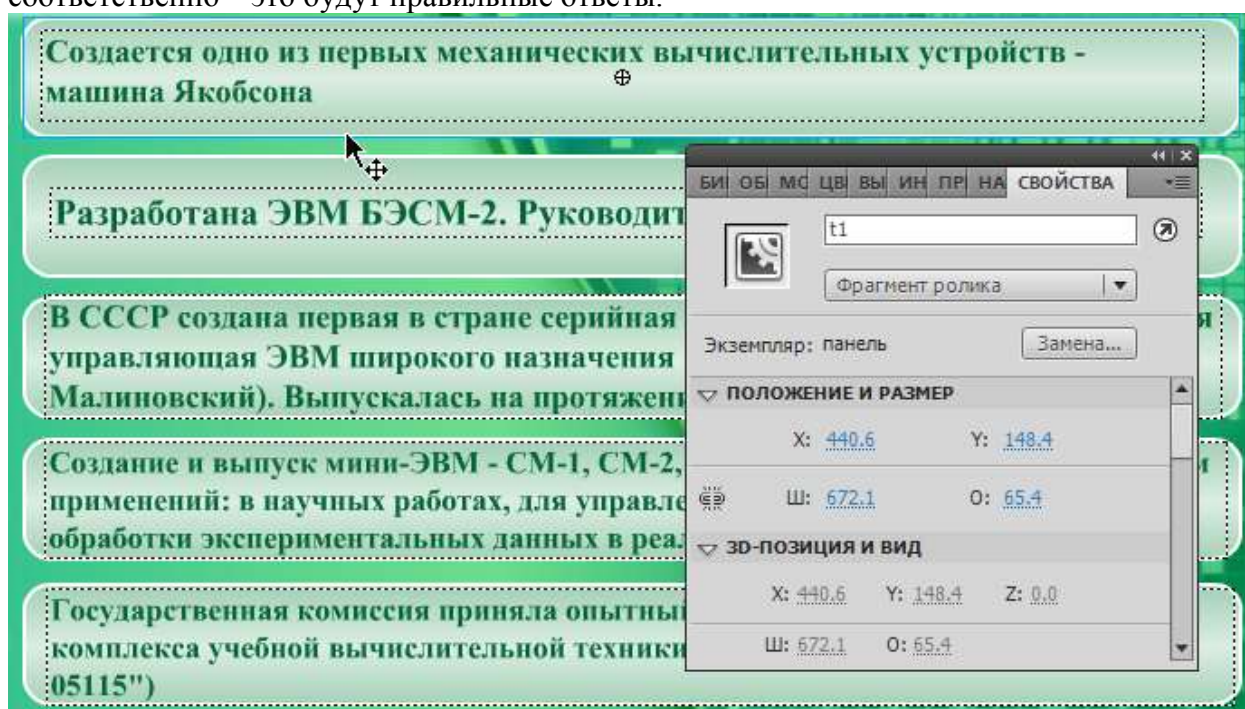
17. Откройте новый документ Flash.
18. Импортируйте графическое изображение для фона.
19. Разместите на 2-м слое статический текст с заданием:



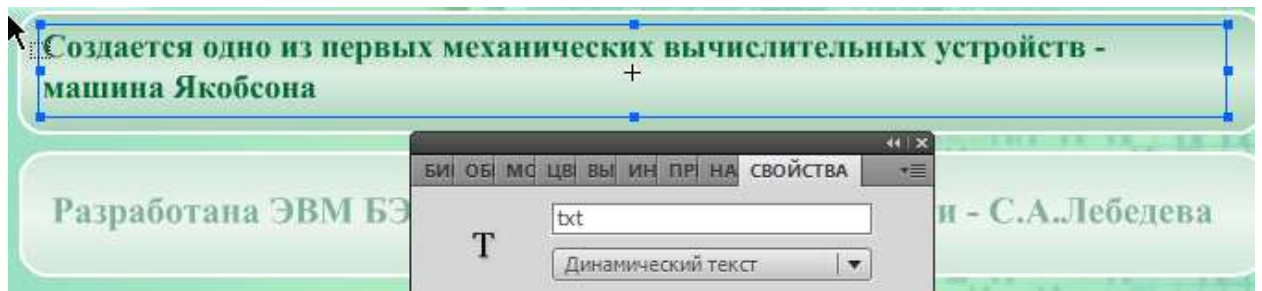
20. Разместите пять дат. Преобразовываем каждую отдельно в символ — тип клип и задаем имена клипам p1, p2, p3, p4, p5 соответственно:



21. Напротив каждой даты размещаем символы – тип клип с имена t1, t2, t3, t4, t5 соответственно – это будут правильные ответы.



22. Для того, чтобы создать клип, необходимо разместить прямоугольник, преобразовать его в символ – тип клип, зайти внутрь символа двойным щелчком, выделить прямоугольник и вновь преобразовать в символ – тип кнопка. Добавить новый слой и разместить на нём поле с динамическим текстом и именем txt:



Выйти из символа, нажав синюю стрелку.

23. Аналогичным образом создаем клипы t2, t3, t4, t5.

24. В новый слой с именем Actions добавляем скрипт:

```
function init()
{
 pop._visible = 0; //поле с выводом результата невидимо
 pop.swapDepths(_root.t1.getDepth);
 for (_root.col_ask = 1; _root["t" + _root.col_ask] != undefined;
 _root.col_ask++)
 {
 } // end of for
 --_root.col_ask;
 for (var _loc2 = 1; _loc2 < _root.col_ask + 1; ++_loc2)
 {
 var _loc3 = Math.round(Math.random() * (_root.col_ask - 1)) + 1;
//генерация случайного расположения объектов
 tmp_x = _root["t" + _loc2]._x;
 tmp_y = _root["t" + _loc2]._y;
 _root["t" + _loc2]._x = _root["t" + _loc3]._x;
 _root["t" + _loc2]._y = _root["t" + _loc3]._y;
 _root["t" + _loc3]._x = tmp_x;
 _root["t" + _loc3]._y = tmp_y;
 _root["t" + _loc2].onPress = f1;
 _root["t" + _loc2].onRelease = f2;
 _root["t" + _loc2].onReleaseOutside = f2; //Событие происходит, если
кнопка отпускается в тот момент, когда указатель мыши располагается вне
активной зоной
 myC = new Color(_root["t" + _loc2].txt);
 myC.setRGB(26163);
 } // end of for
} // End of the function

function test()
{
 _root.oc = 5;
 var i = 1;
 while (i < _root.col_ask + 1)
 {
 if (!_root["t" + i].hitTest(eval("_root.p" + i)))
 {
 myC = new Color(_root["t" + i].txt);
 myC.setRGB(16724787);
 }
 }
}
```

```

 --_root.oc;
 }
 else
 {
 myC = new Color(_root["t" + i].txt);
 myC.setRGB(26163);
 } // end else if
 ++i;
} // end while
} // End of the function
mov = function (obj, x, y, step)
{
 var _loc3 = 1;
 var _loc9 = 1;
 var _loc7 = 1;
 var _loc6 = 1;
 if (Math.abs(y - obj._y) > step)
 {
 _loc3 = Math.abs(x - obj._x) / Math.abs(y - obj._y);
 }
 else
 {
 _loc9 = 0;
 } // end else if
 if (x - obj._x < 0)
 {
 _loc7 = -1;
 } // end if
 if (y - obj._y < 0)
 {
 _loc6 = -1;
 } // end if
 obj._x = obj._x + _loc7 * _loc3 * step;
 obj._y = obj._y + _loc6 * _loc9 * step;
 if (Math.abs(y - obj._y) < 3 * step && Math.abs(x - obj._x) < 3 * _loc3 *
step)
 {
 clearInterval(_root.iii);
 obj._x = x;
 obj._y = y;
 } // end if
 updateAfterEvent();
};
f1 = function ()
{
 this.swapDepths(_root.getNextHighestDepth());
 _root.xt = this._x;
 _root.yt = this._y;

```

```

this._alpha = 70;
this._xscale = 100;
this._yscale = 100;
this.startDrag(false);
};
f2 = function ()
{
 stopDrag (); //закончить перемещение
 var col_ask = 16;
 this._alpha = 100;
 this._xscale = 100;
 this._yscale = 100;
 hit = 0;
 var i = 1;
 while (i < _root.col_ask + 1)
 {
 if (this.hitTest(eval("_root.t" + i)) && this._name != "t" + i)
 {
 hit = i;
 } // end if
 ++i;
 } // end while
 if (hit > 0)
 {
 this._x = _root["t" + hit]._x;
 this._y = _root["t" + hit]._y;
 clearInterval(_root.iii);
 var tdepth = _root["t" + hit].getDepth;
 _root["t" + hit].swapDepths(_root.getNextHighestDepth());
 _root.iii = setInterval(_root.mov, 5, _root["t" + hit], _root.xt, _root.yt,
5);
 _root["t" + hit].swapDepths(tdepth);
 }
 else
 {
 this._x = _root.xt;
 this._y = _root.yt;
 } // end else if
};
init();

```

25. Размещаем кнопку проверки



и добавляем следующий скрипт, нажав F9:

```

on (press)
{

```

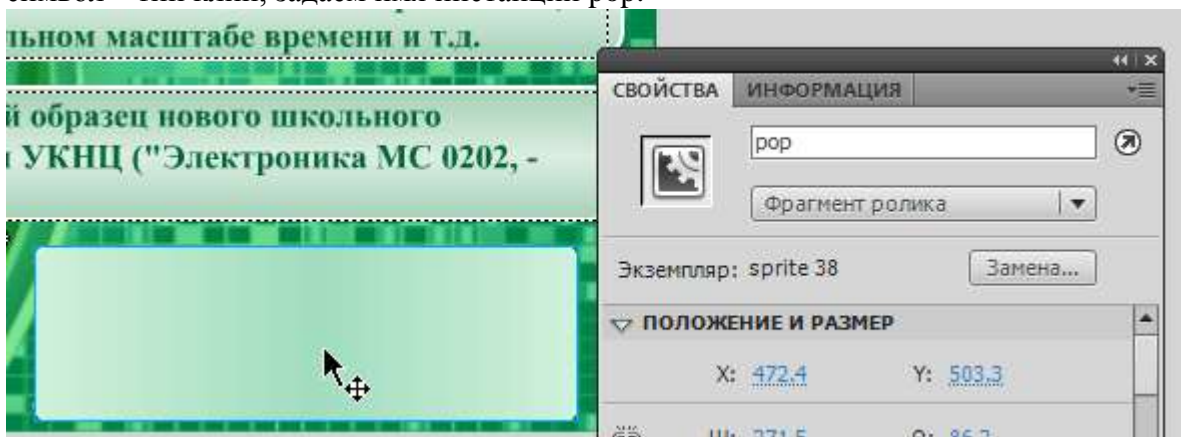
```

pop._visible = 1;
pop.swapDepths(_root.getNextHighestDepth());
_root.got._visible = 0;
_root.osh._visible = 0;
test(); //Вызов функции
if (_root.oc == 5) //если выполнено все правильно
{
 _root.pop.texts = "Поздравляем!\nОшибок нет.\nВаша оценка 5";
}
else if (_root.oc < 5 && _root.oc > 2) //если меньше 5, но больше 2
{
 _root.pop.texts = "Ошибок: " + (5 - _root.oc) + "\nВаша оценка " +
_root.oc;
}
else
{
 _root.pop.texts = "Ошибок: " + (5 - _root.oc) + "\nВаша оценка ";
 if (_root.oc < 1) //если меньше 1
 {
 _root.oc = 1;
 } // end if
 _root.pop.texts = _root.pop.texts + (_root.oc + "\nНеобходимо
повторить!");
} // end else if
}

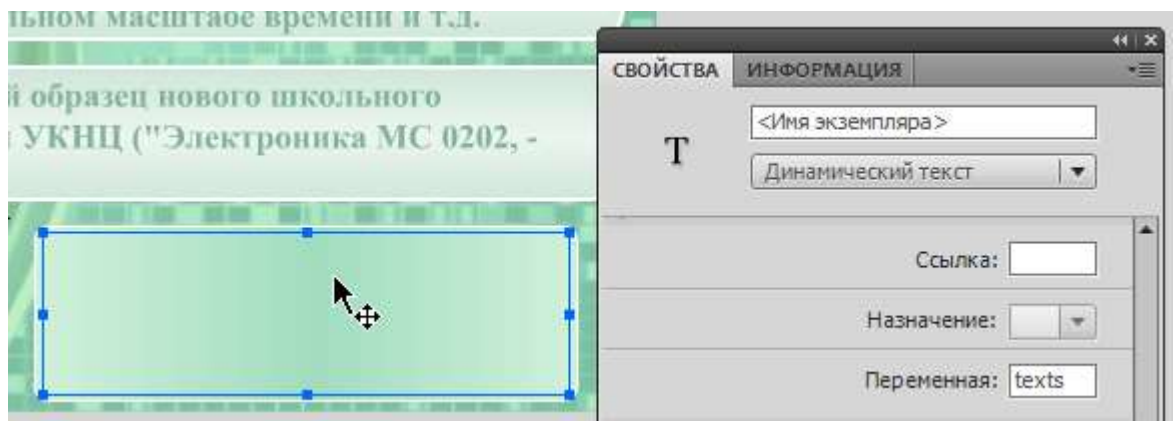
```

**\n** в данном скрипте используется для того, чтобы в текстовом поле выполнялся переход на новую строку.

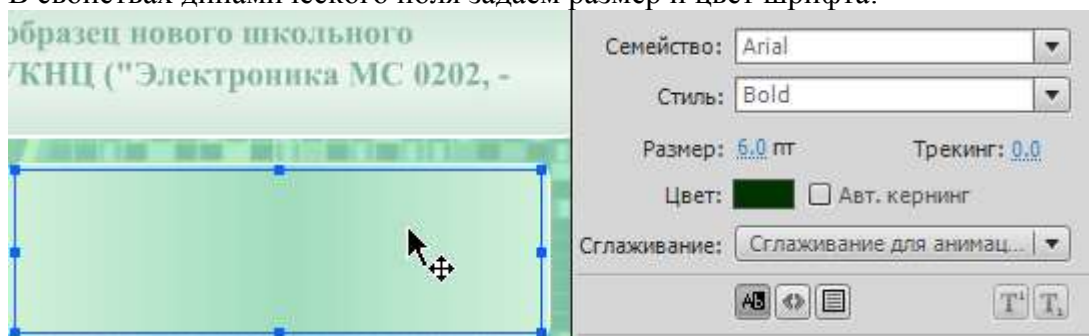
26. Добавляем прямоугольное поле под вывод результата, преобразовываем его в символ – тип клип, задаем имя инстанции pop.



27. Двойным щелчком заходим внутрь символа pop и размещаем динамический текст с именем переменной texts:



28. В свойствах динамического поля задаем размер и цвет шрифта:



29. Выходим из символа, нажав синюю стрелку.

30. Добавляем код, разворачивающий клип на весь экран, и кнопку выхода.

31. Нажимаем Ctrl+Enter - тестируем проект!

32. Публикуем в формате swf.

## Приложение 4

### Задания для индивидуального выполнения

6. Спланируйте ЦОР в рамках одного из изучаемых или изученных ранее предметов по специальности.
7. Создайте свой ресурс с использованием рассмотренных технологий.
8. Обязательно укажите ФИО, курс и факультет, на котором вы учитесь.
9. Сохраните ролик с именем ФИО\_pr7, например, *StepanenkoOV\_pr7.fla*
10. Опубликуйте ролик.

## Приложение 5

### Для зачета по лабораторному занятию:

1. Оформите пояснительную записку к ролику.
2. Файлы (*StepanenkoOV\_pr7.fla*, *StepanenkoOV\_pr7.swf*, *StepanenkoOV\_zapiska.doc*) заархивируйте (*stepanenkoOV\_zan7.rar*) и сдайте преподавателю.

## Приложение 6

### Источники:

Фон <http://www.pixforge.net/vectors/5820-pixel-vector-blue-backgrounds.html>  
<http://www.flash-soft.ru/fg17/Index04.htm>

|                |                                                                                  |
|----------------|----------------------------------------------------------------------------------|
| <b>Тема 23</b> | <b>Использование приема создания маски в качестве курсора при разработке ЦОР</b> |
|----------------|----------------------------------------------------------------------------------|

### 1.1. Технология обучения на лабораторном занятии

|                                                                                                                                                                                                                              |                                                                                                                                                                                                                                                                                              |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Учебное время: 2 часа.</i>                                                                                                                                                                                                |                                                                                                                                                                                                                                                                                              |
| <i>Форма учебного занятия: лабораторная работа</i>                                                                                                                                                                           |                                                                                                                                                                                                                                                                                              |
| <i>План занятия</i>                                                                                                                                                                                                          | 1. Разработка ЦОР с использованием приема создания маски.<br>2. Выполнение индивидуального задания.                                                                                                                                                                                          |
| <i>Цель учебного занятия:</i>                                                                                                                                                                                                | освоить технологию разработки ЦОР с использованием приема создания маски;                                                                                                                                                                                                                    |
| <i>Педагогические задачи:</i><br>Организовать работу студентов по: <ul style="list-style-type: none"> <li>• разработке ЦОР с использованием приема создания маски;</li> <li>• выполнению индивидуального задания.</li> </ul> | <i>Результаты учебной деятельности:</i><br><i>Студент должен уметь:</i> <ul style="list-style-type: none"> <li>• Разрабатывать ЦОР с использованием приема создания маски.</li> <li>• Самостоятельно разрабатывать и выполнять образовательный ресурс по аналогии с предложенным.</li> </ul> |
| <i>Методы обучения</i>                                                                                                                                                                                                       | Репродуктивный метод обучения                                                                                                                                                                                                                                                                |
| <i>Средства обучения</i>                                                                                                                                                                                                     | персональные компьютеры с установленной программой Adobe Flash, печатная версия задания                                                                                                                                                                                                      |
| <i>Формы организации учебной деятельности</i>                                                                                                                                                                                | Фронтальная, индивидуальная.                                                                                                                                                                                                                                                                 |
| <i>Условия обучения</i>                                                                                                                                                                                                      | Кабинет вычислительной техники                                                                                                                                                                                                                                                               |

### Технологическая карта лабораторного занятия

| Этап работы, время                                 | Содержание деятельности                                                                                                                                                                                                                       |                                    |
|----------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------|
|                                                    | Преподавателя                                                                                                                                                                                                                                 | Студентов                          |
| 1-й этап.<br>Введение в учебное занятие<br>(5 мин) | 1.7. Сообщает, тему, цель учебного занятия и план его проведения ( <i>приложение 1</i> ).                                                                                                                                                     | Слушают, уточняют, задают вопросы. |
| 2-й этап.<br>Основная часть (65 мин)               | 2.1. Дает описание ЦОР ( <i>приложение 2</i> ).<br>2.2. Организует работу студентов по созданию образовательного ресурса ( <i>приложение 3</i> ).<br>2.3. Раздает задания для индивидуального выполнения ( <i>приложение 4</i> ) и разъясняет | 2.1. Слушают, уточняют.            |

|                                                              |                                                                                                                                                                                                        |                                                                                                                        |
|--------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------|
|                                                              | форму отчета по лабораторной работе<br>(приложение 5).                                                                                                                                                 | 2.2.Выполняют задания на ПК.<br><br>2.3. Выполняют индивидуальные задания и подготавливают отчет о проделанной работе. |
| 3-й этап.<br>Заключитель-<br>но-результат-<br>ный<br>(13 ин) | 3.5. Дает и разъясняет задания для домашней работы:<br><ul style="list-style-type: none"> <li>• Доработать задания;</li> <li>• Изучить источники Интернет по работе с Flash (приложение 6).</li> </ul> | Задают уточняющие вопросы.                                                                                             |

### Приложение 1

#### Тема: Использование приема создания маски в качестве курсора при разработке ЦОР

**Цель занятия:** освоить технологию разработки ЦОР с использованием приема создания маски.

#### План занятия

1. Разработка ЦОР с использованием приема создания маски.
2. Выполнение индивидуального задания.

### Приложение 2

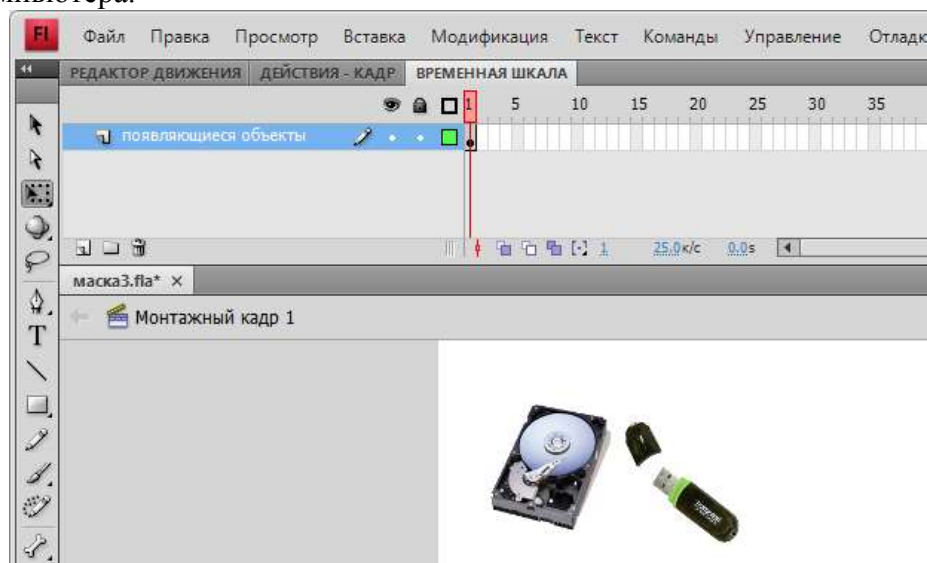
#### Технология создания волшебного ластика. Прием использования маски

Уважаемые студенты, представляю вашему вниманию новый урок. Я назвала этот прием «Волшебный ластик». Стоит только провести по экрану карандашом (пером или пальчиком ... :) и в нужном месте появляется фрагмент изображения (это может быть также текст, и все что угодно, в общем все, что мы разместим с Вами на соответствующем слое). Можно предложить учащимся по фрагменту отгадать, что это за картинка. А может это будет испорченное сообщение, скрывающее определение какого-то понятия, или закодированный текст? Как еще можно использовать данный прием, я думаю, фантазия и творческий замысел вам подскажут! Основой при разработке данного урока является использование приема создания маски в качестве курсора.

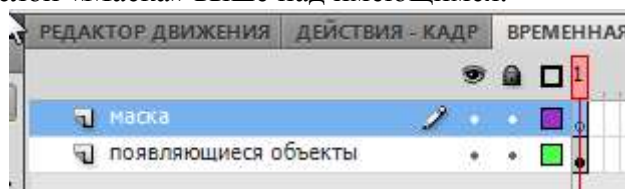
### Приложение 3

## Алгоритм разработки ЦОР

1. Откройте новый документ Flash.
2. Разместите в 1-м кадре слоя «Появляющиеся объекты» изображения устройств компьютера.



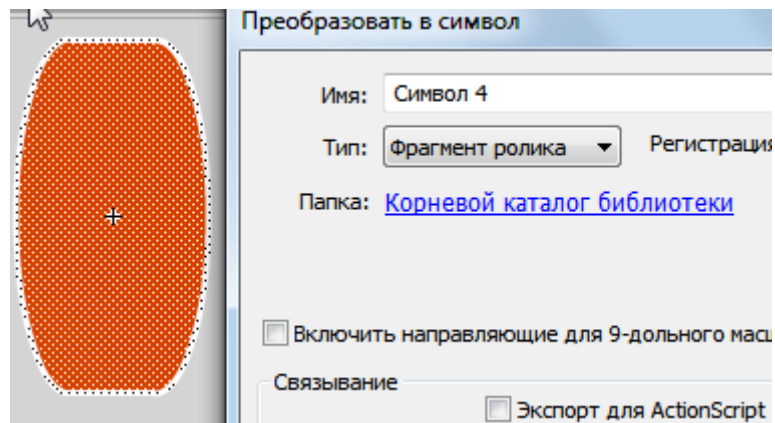
3. Создайте новый слой «Маска» выше над имеющимся.



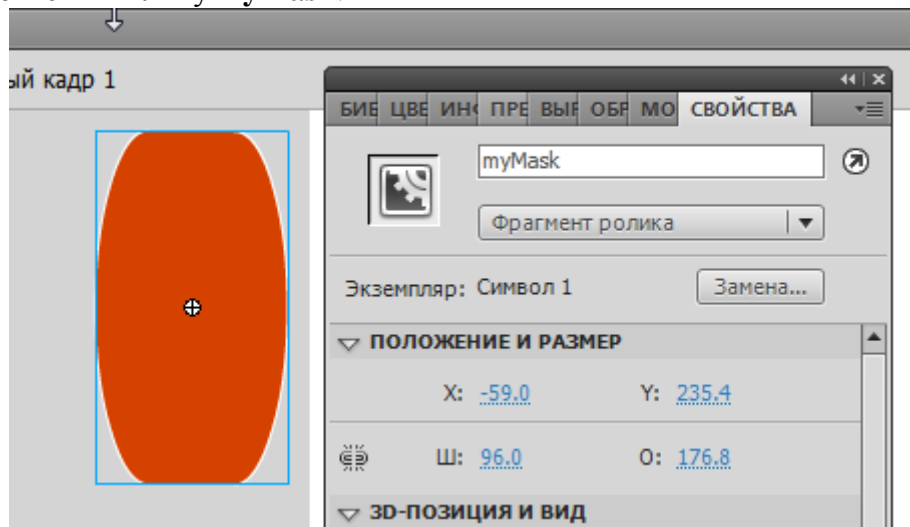
4. В кадр слоя «Маска» добавьте следующий скрипт:

```
Mouse.addListener(myMask);
myMask.onMouseMove = function() {
 var mx = this._xmouse;
 var my = this._ymouse;
 var eWidth = 40;
 this.moveTo(mx-eWidth, my-eWidth);
 this.beginFill(0, 400);
 this.lineTo(mx+eWidth, my-eWidth);
 this.lineTo(mx+eWidth, my+eWidth);
 this.lineTo(mx-eWidth, my+eWidth);
 this.lineTo(mx-eWidth, my-eWidth);
 this.endFill();
};
```

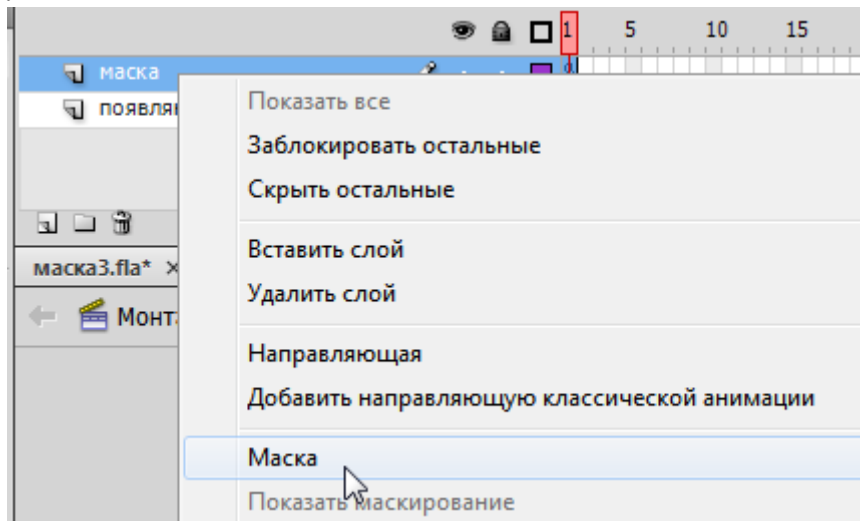
5. На слое «Маска» за границами рабочего поля разместите скругленный прямоугольник (это может быть и любая другая фигура ☺), преобразуйте его в символ – тип клип:



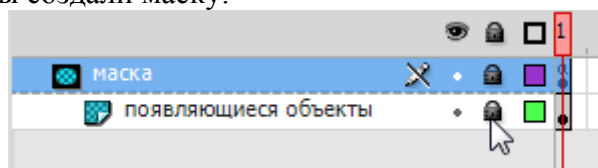
6. Присвойте имя клипу **myMask**:



7. Щелкните правой кнопкой мыши по слою с именем «Маска» и выберите пункт «Маска»:

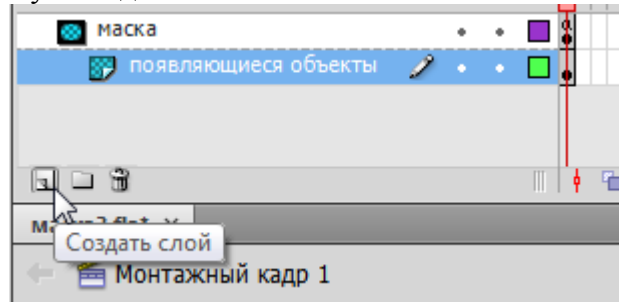


8. Таким образом, мы создали маску:

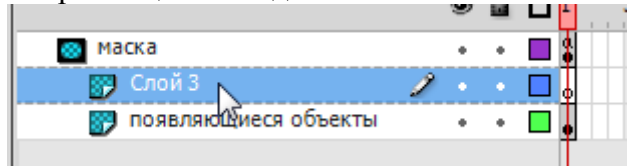


Замочки можно будет снять для последующей работы и корректировки расположения графических объектов – изображений устройств компьютера.

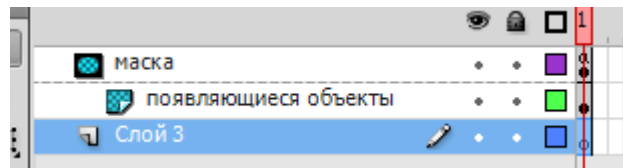
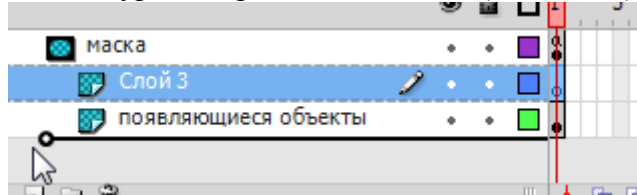
9. Щелкните по значку «Создать слой»



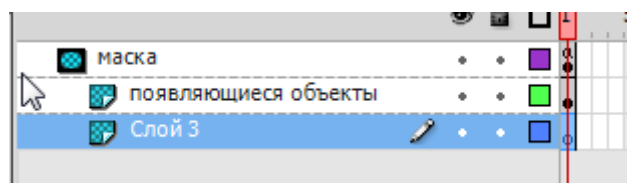
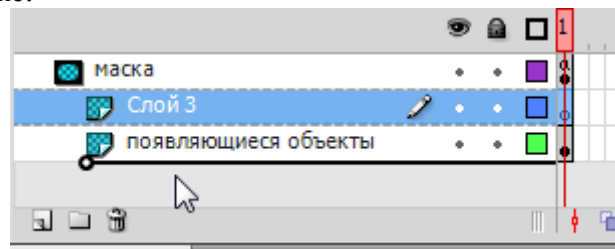
10. Автоматически слой размещается под слоем «Маска»:



11. Однако нам это не нужно... Поэтому мы переносим созданный слой ниже слоя «Появляющиеся объекты», разместив курсор так, чтобы появилась линия с кругом, указывающая на уровень размещения слоя (вне маски):

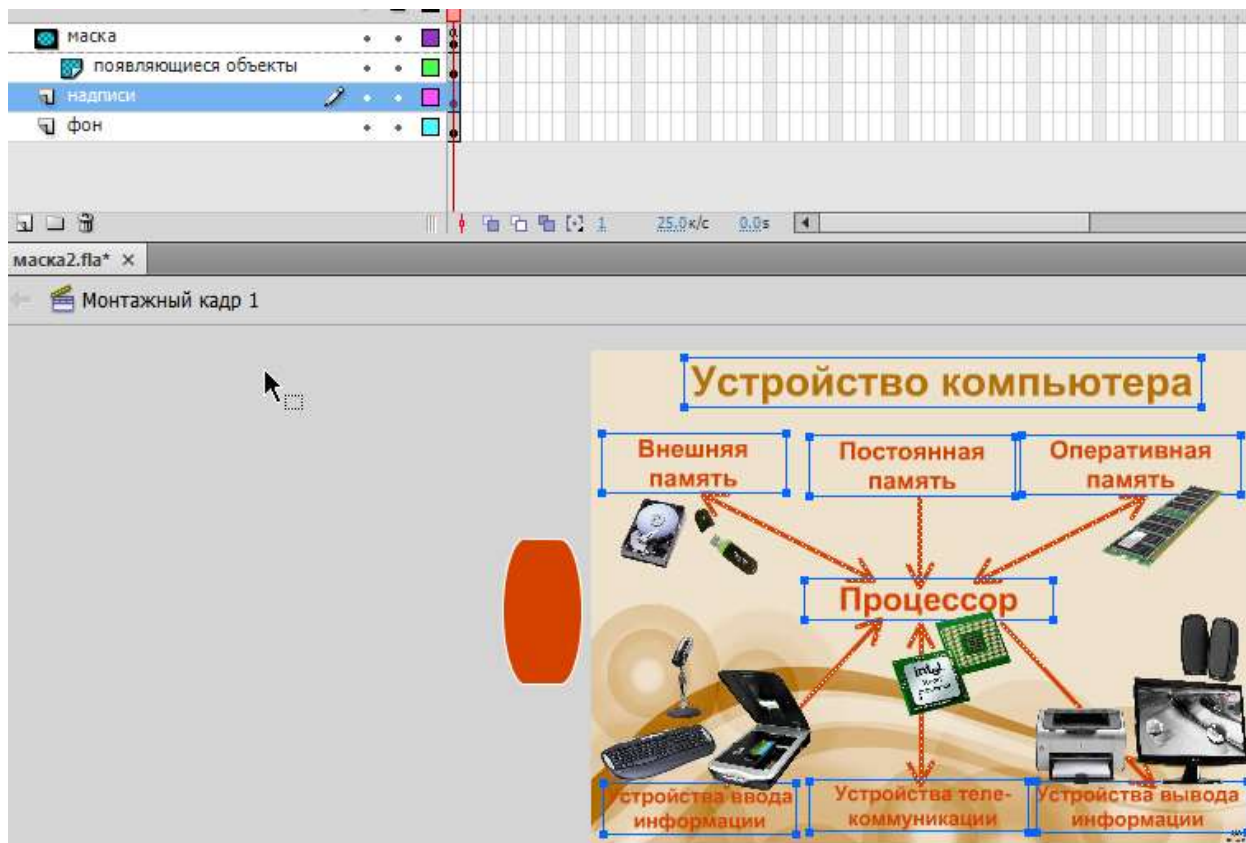


Если выбрать вот такое расположение направляющей, то слой все равно будет относиться к маске:



12. Переименовываем слой 3, дав ему имя «Фон» и размещаем на нем фоновое изображение.

13. Добавляем новый слой с именем «Надписи» и размещаем на нем все надписи, которые будут видны при загрузке ролика:



14. Добавляем код, разворачивающий клип на весь экран, и кнопку выхода.
15. Нажимаем Ctrl+Enter - тестируем проект!
16. Публикуем в формате swf.

#### Приложение 4

##### Задания для индивидуального выполнения

1. Спланируйте ЦОР в рамках одного из изучаемых или изученных ранее предметов по специальности.
2. Создайте свой ресурс с использованием рассмотренных технологий.
3. Обязательно укажите ФИО, курс и факультет, на котором вы учитесь.
4. Сохраните ролик с именем ФИО\_pr8, например, *StepanenkoOV\_pr8 fla*
5. Опубликуйте ролик.

#### Приложение 5

##### Для зачета по лабораторному занятию:

1. Оформите пояснительную записку к ролику.
2. Файлы (*StepanenkoOV\_pr8 fla*, *StepanenkoOV\_pr8 swf*, *StepanenkoOV\_zapiska.doc*) заархивируйте (*stepanenkoOV\_zan8.rar*) и сдайте преподавателю.

#### Приложение 6

##### Источники:

1. Фон [http://i.allday.ru/uploads/posts/2009-04/1239286784\\_light-backgrounds-15.jpg](http://i.allday.ru/uploads/posts/2009-04/1239286784_light-backgrounds-15.jpg)
2. Монитор <http://www.materiel.net/live/93816.jpg>
3. Микрофон <http://ancheiopossoallenare.com/contenuti/wp-content/uploads/2008/06/microfono1.gif>
4. Принтер <http://tonerdirect.bg/images/hpljp1006w.jpg>
5. Флешка <http://desc.allshops.ru/images/207779.jpg>

6. Жесткий диск <http://www.ferra.ru/images/144/144686.jpg>
7. Процессор [http://www.microchoix.com/M\\_images/trait110485.jpg](http://www.microchoix.com/M_images/trait110485.jpg)
8. Сканер <http://regmedia.co.uk/2007/08/16/v500.jpg>
9. Клавиатура [http://www.pc.deal.pl/foto/klawiatura\\_chicony\\_kbp\\_0401\\_black\\_silver.jpg](http://www.pc.deal.pl/foto/klawiatura_chicony_kbp_0401_black_silver.jpg)
10. Монитор [http://alt-tab.com.ar/wp-content/uploads/LG\\_22.jpg](http://alt-tab.com.ar/wp-content/uploads/LG_22.jpg)
11. Колонки <http://www.vobis.pl/foto.ashx?w=350&s=881678>

|                |                                                                      |
|----------------|----------------------------------------------------------------------|
| <b>Тема 24</b> | <b>Технология объединения нескольких ЦОР в один авторский ресурс</b> |
|----------------|----------------------------------------------------------------------|

### 1.1. Технология обучения на лабораторном занятии

|                                                                                                                                                                                                                                           |                                                                                                                                                                                                                                                                                   |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Учебное время: 2 часа.</i>                                                                                                                                                                                                             |                                                                                                                                                                                                                                                                                   |
| <i>Форма учебного занятия: лабораторная работа</i>                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                   |
| <i>План занятия</i>                                                                                                                                                                                                                       | 1. Технология объединения нескольких ЦОР в один авторский ресурс.<br>2. Выполнение индивидуального задания.                                                                                                                                                                       |
| <i>Цель учебного занятия:</i>                                                                                                                                                                                                             | освоить технологию объединения нескольких ЦОР в один авторский ресурс;                                                                                                                                                                                                            |
| <i>Педагогические задачи:</i><br>Организовать работу студентов по: <ul style="list-style-type: none"> <li>Освоению технологии объединения нескольких ЦОР в один авторский ресурс;</li> <li>выполнению индивидуального задания.</li> </ul> | <i>Результаты учебной деятельности:</i><br><i>Студент должен уметь:</i> <ul style="list-style-type: none"> <li>объединять нескольких ЦОР в один авторский ресурс;</li> <li>самостоятельно разрабатывать и выполнять образовательный ресурс по аналогии с предложенным.</li> </ul> |
| <i>Методы обучения</i>                                                                                                                                                                                                                    | Репродуктивный метод обучения                                                                                                                                                                                                                                                     |
| <i>Средства обучения</i>                                                                                                                                                                                                                  | персональные компьютеры с установленной программой Adobe Flash, печатная версия задания                                                                                                                                                                                           |
| <i>Формы организации учебной деятельности</i>                                                                                                                                                                                             | Фронтальная, индивидуальная.                                                                                                                                                                                                                                                      |
| <i>Условия обучения</i>                                                                                                                                                                                                                   | Кабинет вычислительной техники                                                                                                                                                                                                                                                    |

### Технологическая карта лабораторного занятия

| Этап работы, время                           | Содержание деятельности                                                                                                                           |                                    |
|----------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------|
|                                              | Преподавателя                                                                                                                                     | Студентов                          |
| 1-й этап. Введение в учебное занятие (5 мин) | 1.8. Сообщает, тему, цель учебного занятия и план его проведения ( <i>приложение 1</i> ).                                                         | Слушают, уточняют, задают вопросы. |
| 2-й этап.                                    | 2.4. Дает описание ЦОР ( <i>приложение 2</i> ).<br>2.5. Организует работу студентов по созданию образовательного ресурса ( <i>приложение 3</i> ). | 2.1. Слушают, уточняют.            |

|                                                      |                                                                                                                                                       |                                                                                                                        |
|------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------|
| Основная часть (65 мин)                              | 2.6. Раздает задания для индивидуального выполнения ( <i>приложение 4</i> ) и разъясняет форму отчета по лабораторной работе ( <i>приложение 5</i> ). | 2.2.Выполняют задания на ПК.<br><br>2.3. Выполняют индивидуальные задания и подготавливают отчет о проделанной работе. |
| 3-й этап.<br>Заключительно-результативный<br>(14 ин) | 3.6. Дает и разъясняет задания для домашней работы:<br><ul style="list-style-type: none"><li>• Доработать задания.</li></ul>                          | Задают уточняющие вопросы.                                                                                             |

### Приложение 1

#### Тема: Технология объединения нескольких ЦОР в один авторский ресурс

**Цель занятия:** освоить технологию объединения нескольких ЦОР в один авторский ресурс.

#### План занятия

1. Технология объединения нескольких ЦОР в один авторский ресурс.
2. Выполнение индивидуального задания.

### Приложение 2

#### Технология объединения нескольких ЦОРов в один авторский ресурс

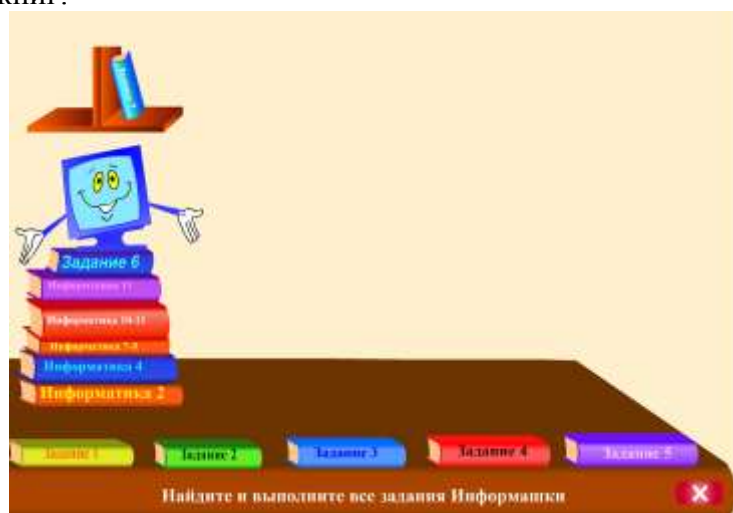
Если у Вас имеется несколько небольших ЦОРов, то их можно объединить в один полноценный ресурс. Для этого нужно:

#### Алгоритм выполнения работы:

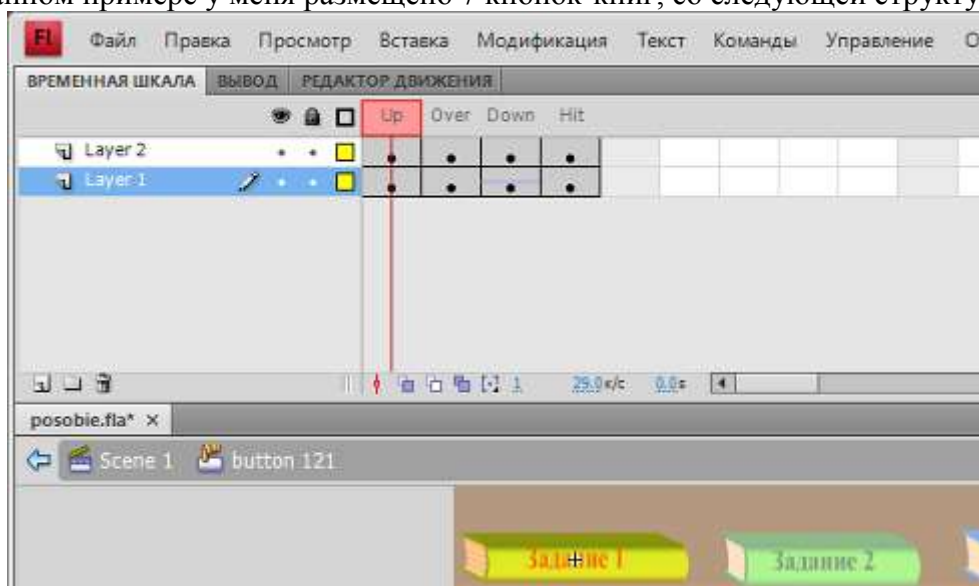
1. Открываем новый документ Adobe Flash.
2. Создаем титульный лист ресурса



3. Оформляем второй кадр, на котором у нас будет находиться меню, например в виде стопки книг:

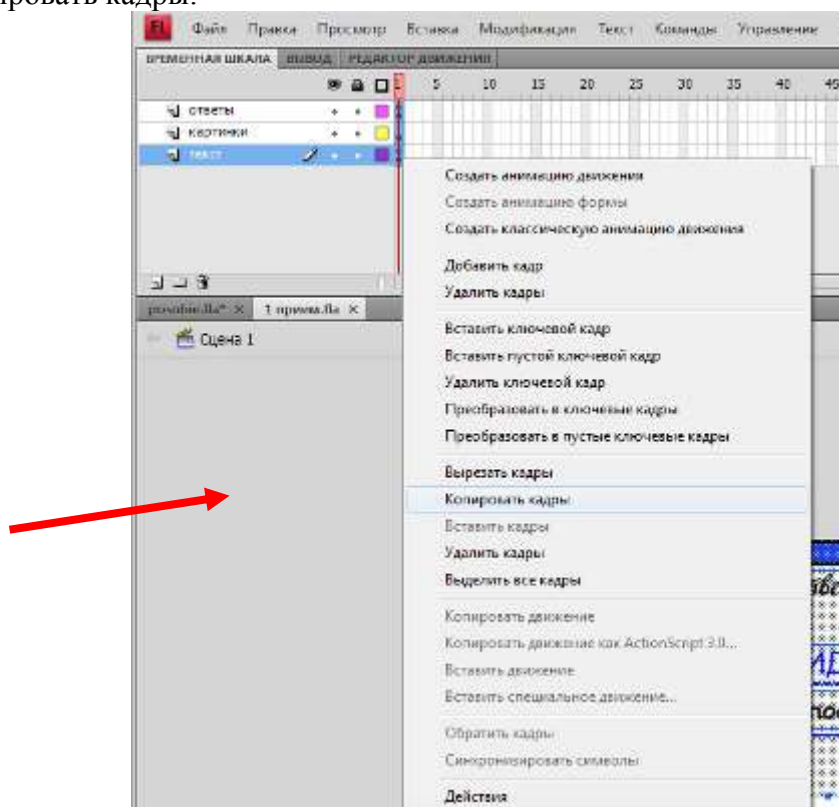


В данном примере у меня размещено 7 кнопок-книг, со следующей структурой:

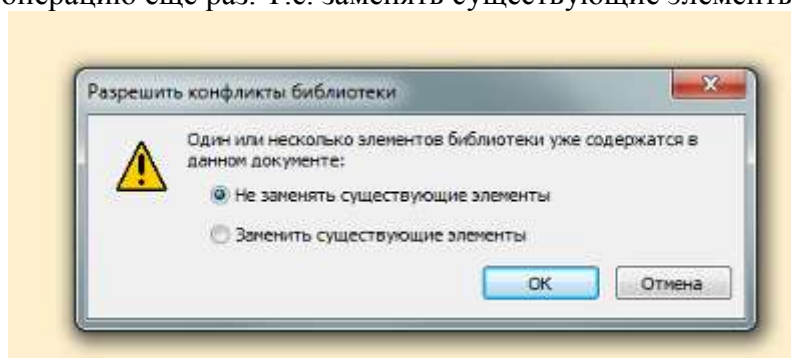


4. Теперь все созданные ранее ЦОРы мы должны перенести в данный ресурс. Это делается следующим образом:

5. Открываем исходник ЦОРа, который будет у нас запускаться кнопкой Задание 3.
6. Выделяем все кадры, щелкаем левой кнопкой мыши и выбираем пункт Копировать кадры:

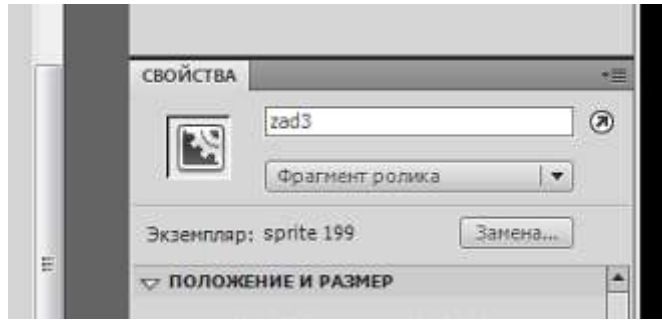


7. Переходим в рабочий исходник и создаем новый пустой символ, нажав сочетание клавиш Ctrl+F8.
8. Щелкаем левой кнопкой мыши по 1 кадру и выбираем пункт Вставить кадры.
9. Если появляется сообщение «Один или несколько элементов библиотеки уже содержатся в данном документе», то необходимо вернуться назад к исходнику ЦОРа, переименовать все символы библиотеки уникальными именами и повторить операцию еще раз. Т.е. заменять существующие элементы нельзя!!!



*По умолчанию, когда мы создаем символы, им присваиваются имена Символ1, Символ 2 и т.п. Лучше всего использовать уникальные имена, тогда проблем не будет.*

10. Как только кадры вставлены, необходимо выйти из символа, открыть библиотеку и переместить созданный символ в центр рабочего поля. Далее присваиваем созданному символу новое имя (InstanceName), то по которому мы будем к нему обращаться, например zad3



11. Аналогичным образом поступаем с остальными ЦОРами (п.5-9). *Примечание: удобнее размещать каждый символ-клип на отдельный слой.*

12. Для кнопок-книг, которые будут запускать наши клипы, вставляем следующий скрипт:

on (press)

{

    tellTarget("\_root.mon")

    {

        gotoAndStop(1); *//закрывается клип mon*

        tellTarget("")

    {

        tellTarget("\_root.zad4")

        {

            gotoAndStop(1); *//закрывается клип zad4*

            tellTarget("")

        {

            tellTarget("\_root.zad6")

            {

                gotoAndStop(1); *//закрывается клип zad6*

                tellTarget("")

            {

                tellTarget("\_root.zad5")

                {

                    gotoAndStop(1); *//закрывается клип zad5*

                    tellTarget("")

                {

                    tellTarget("\_root.zad7")

                    {

                        gotoAndStop(1); *//закрывается клип zad7*

                        tellTarget("")

                    {

                        tellTarget("\_root.rez")

                        {

                            gotoAndStop(1); *//закрывается клип rez*

                            tellTarget("")

                        {

                            tellTarget("\_root.zad3")

                            {

                                gotoAndStop(2); *//открывается клип zad3*

                                tellTarget("")

                            {

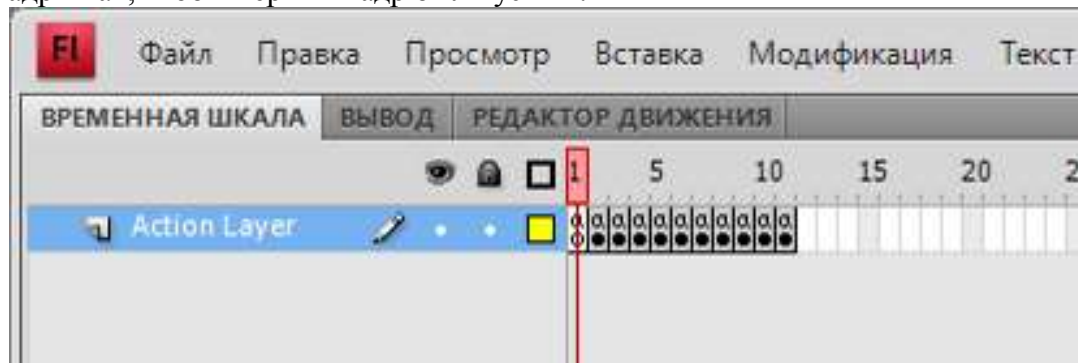
```

} // End of TellTarget
} // End of TellTarget
} // End of TellTarget
} // End of TellTarget
} // End of TellTarget
} // End of TellTarget
} // End of TellTarget
} // End of TellTarget
} // End of TellTarget
} // End of TellTarget
} // End of TellTarget
} // End of TellTarget
} // End of TellTarget
}

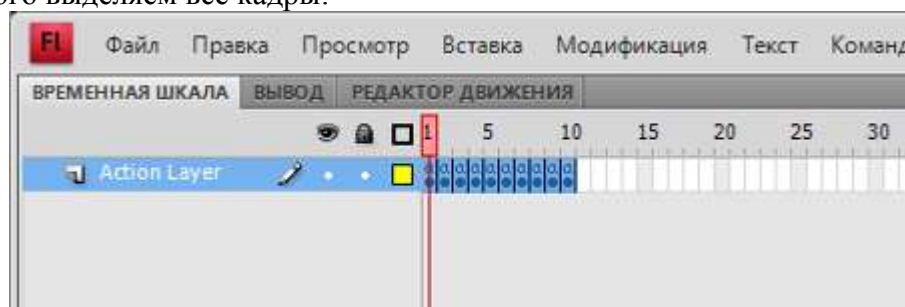
```

Таким образом, кнопка **Задание 3**, которая нужный клип **zad3** запускает, должна все остальные клипы закрывать.

13. Последнее, что нужно сделать – это в каждом созданном клипе переместить все кадры так, чтобы первый кадр был пустым:



Для этого выделяем все кадры:



Когда у основания стрелки мышки появился прямоугольник - просто перетаскиваем и бросаем все кадры на один кадр назад.

В 1 пустом кадре прописываем скрипт остановки:

```
stop();
```

Это мы сделали для того, чтобы при переходе на первый кадр клипа – он закрывался `gotoAndStop(1);`, а при переходе на второй –открывался `gotoAndStop(2);`.

14. Добавляем в 1 кадр нашего ресурса скрипты разворачивания клипа на весь экран, отключения меню, во 2 кадр - кнопку выхода.
15. Нажимаем Ctrl+Enter - тестируем проект!
16. Публикуем в формате swf.

### **Задания для индивидуального выполнения**

1. Объедините созданные на лабораторных занятиях авторские ЦОР в единый образовательный ресурс с использованием рассмотренной технологии.
2. Обязательно укажите ФИО, курс и факультет, на котором вы учитесь.
3. Сохраните ролик с именем ФИО\_pr9, например, *StepanenkoOV\_pr9.flv*
4. Опубликуйте ролик.

### **Приложение 5**

#### **Для зачета по лабораторному занятию:**

1. Оформите пояснительную записку к ролику.
2. Файлы (*StepanenkoOV\_pr9.flv*, *StepanenkoOV\_pr9.swf*, *StepanenkoOV\_zapiska.doc*) заархивируйте (*stepanenkoOV\_zan9.rar*) и сдайте преподавателю.

## Раздел V. Технологии создания мультимедийных обучающих видеокурсов (Camtasia Studio).

### Модель технологии обучения

|                |                                                                                                         |
|----------------|---------------------------------------------------------------------------------------------------------|
| <b>Тема 25</b> | <b>Выполнение предварительных настроек и запись видео с экрана монитора в программе Camtasia Studio</b> |
|----------------|---------------------------------------------------------------------------------------------------------|

#### 1.1. Технология обучения на лекции

|                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                  |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Учебное время:</i> 2 часа.                                                                                                                                                                                                          |                                                                                                                                                                                                                                                  |
| <i>Форма учебного занятия:</i> лабораторная работа                                                                                                                                                                                     |                                                                                                                                                                                                                                                  |
| <i>План занятия</i>                                                                                                                                                                                                                    | <ol style="list-style-type: none"> <li>1. Настройка Панели записи.</li> <li>2. Использование скрин маркера ScreenDraw.</li> <li>3. Запись действий с экрана.</li> </ol>                                                                          |
| <i>Цель учебного занятия:</i>                                                                                                                                                                                                          | Сформировать умения по предварительной настройке и последующей записи видеоурока с помощью программы Camtasia Studio                                                                                                                             |
| <i>Педагогические задачи:</i><br>Организовать работу студентов по <ul style="list-style-type: none"> <li>• настройке Панели записи;</li> <li>• использование скрин маркера ScreenDraw;</li> <li>• записи действий с экрана.</li> </ul> | <i>Результаты учебной деятельности:</i><br><i>Студент должен уметь:</i> <ul style="list-style-type: none"> <li>• Настраивать Панель записи.</li> <li>• Использовать скрин маркер ScreenDraw.</li> <li>• Записывать действия с экрана.</li> </ul> |
| <i>Методы обучения</i>                                                                                                                                                                                                                 | Частично-поисковый метод                                                                                                                                                                                                                         |
| <i>Средства обучения</i>                                                                                                                                                                                                               | Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение                                                                                                                                                                |
| <i>Формы организации учебной деятельности</i>                                                                                                                                                                                          | Фронтальная, индивидуальная.                                                                                                                                                                                                                     |
| <i>Условия обучения</i>                                                                                                                                                                                                                | Кабинет вычислительной техники                                                                                                                                                                                                                   |

#### Технологическая карта лабораторного занятия

| Этап работы, время | Содержание деятельности |           |
|--------------------|-------------------------|-----------|
|                    | Преподавателя           | Студентов |

|                                                          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |                                                                                                                                                                                                                                                                            |
|----------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1-й этап.<br>Введение в учебное занятие<br>(5 мин)       | 1.9. Сообщает, тему, цель, планируемые результаты учебного занятия и план его проведения. ( <i>приложение 1</i> ).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | Слушают, уточняют, задают вопросы.                                                                                                                                                                                                                                         |
| 2-й этап.<br>Основная часть<br>(72 мин)                  | <p>2.1. Распределяет темы для создания видеоуроков (<i>приложение 2</i>).</p> <p>2.2. Просит выполнить настройку согласно пунктам лабораторного задания (<i>приложение 3</i>).</p> <p>2.3. Напоминает студентам как использовать во время аписи видеоурока скрин маркером ScreenDraw (<i>приложение 4</i>).</p> <p>2.4. Предлагает студентам составить письменный сценарий действий, производимых во время записи с указанием в какой момент необходимо воспользоваться скрин маркером ScreenDraw.</p> <p>2.5. Напоминает, каким образом производится запись (<i>приложение 5</i>) и просит произвести запись видеоурока и сохранить видеофайл для дальнейшей обработки.</p> | <p>2.1. Выбирают тему для создания видеоурока.</p> <p>2.2. Выполняют настройку в Camtasia Studio.</p> <p>2.3. Слушают, уточняют.</p> <p>2.4. Составляют сценарий видеоурока.</p> <p>2.5. Производят запись видеоурока и сохраняют видеофайлы для дальнейшей обработки.</p> |
| 3-й этап.<br>Заключитель но- результирую щий<br>(10 мин) | 3.1. Дает и разъясняет задания для домашней работы: Повторив тему практической работы по озвучиванию видеоурока добавить речевые комментарии к отснятому на занятии материалу.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | Задают уточняющие вопросы.                                                                                                                                                                                                                                                 |

## Приложение 1

**Тема: Выполнение предварительных настроек и запись видео с экрана монитора в программе Camtasia Studio**

**Цель занятия:** сформировать умения по предварительной настройке и последующей записи видеоурока с помощью программы Camtasia Studio.

**План занятия:**

1. Настройка Панели записи.
2. Использование скрин маркера ScreenDraw.
3. Запись действий с экрана.

## Приложение 2

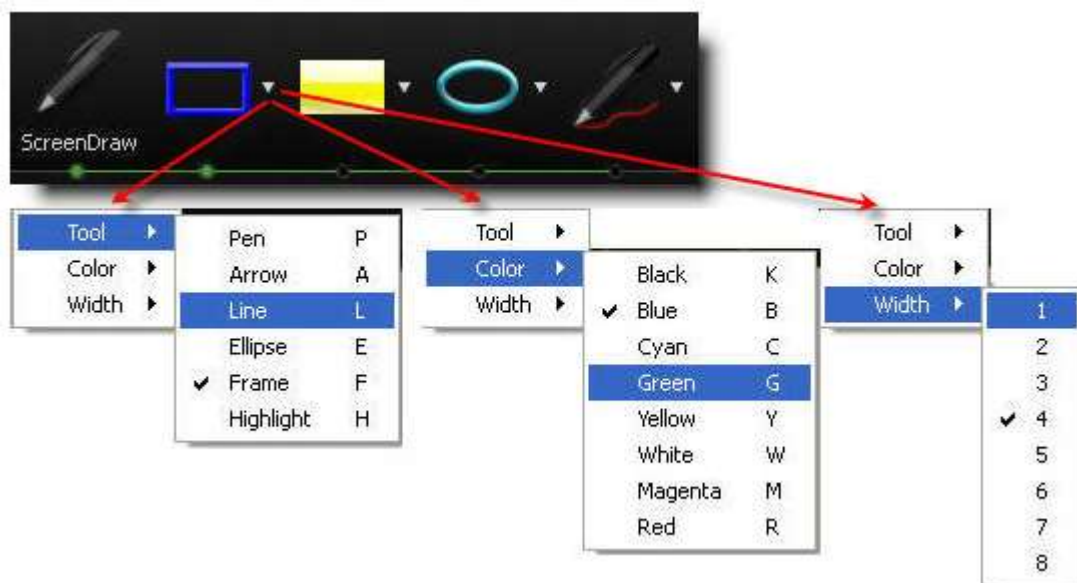
| №   | Тема                                                                         | ФИО студента |
|-----|------------------------------------------------------------------------------|--------------|
| 1.  | Интерфейс программы Microsoft Word 2010                                      |              |
| 2.  | Способы создания различных видов списков с программе Microsoft Word 2010     |              |
| 3.  | Работа со стилями и создание оглавления в программе Microsoft Word 2010      |              |
| 4.  | Создание и редактирование формул в программе Microsoft Word 2010             |              |
| 5.  | Создание, редактирование и форматирование таблиц в Microsoft Word 2010       |              |
| 6.  | Оформление колонок и вставка номеров страниц в программе Microsoft Word 2010 |              |
| 7.  | Способы вставки сносок в документ в программе Microsoft Word 2010            |              |
| 8.  | Создание списка литературы в программе Microsoft Word 2010                   |              |
| 9.  | Вставка и редактирование изображений в программе Microsoft Word 2010         |              |
| 10. | Этапы создания указателя терминов в программе Microsoft Word 2010            |              |
| 11. | Использование инструмента Рецензирование в Microsoft Word 2010               |              |
| 12. | Интерфейс программы Microsoft Excel 2010                                     |              |
| 13. | Библиотека функций в программе Microsoft Excel 2010                          |              |
| 14. | Диалоговое окно Формат ячеек в программе Microsoft Excel 2010                |              |
| 15. | Вставка и редактирование формул в программе Microsoft Excel 2010             |              |
| 16. | Сортировка и фильтр в программе Microsoft Excel 2010                         |              |

|     |                                                                                             |  |
|-----|---------------------------------------------------------------------------------------------|--|
| 17. | Применение различных форматов в программе Microsoft Excel 2010                              |  |
| 18. | Условное форматирование в программе Microsoft Excel 2010                                    |  |
| 19. | Использование абсолютных и относительных ссылок в формулах в программе Microsoft Excel 2010 |  |
| 20. | Имена ячеек, определяемые пользователем в программе Microsoft Excel 2010                    |  |
| 21. | Работа с диаграммами в программе Microsoft Excel 2010                                       |  |
| 22. | Интерфейс программы Microsoft Power Point 2010                                              |  |
| 23. | Использование объектов WordArt в пакете MS Office 2010                                      |  |
| 24. | Настройка анимации в программе Microsoft Power Point 2010                                   |  |
| 25. | Настройка переходов слайдов в программе Microsoft Power Point 2010                          |  |
| 26. | Вставка таблиц и диаграмм на слайды в программе Microsoft Power Point 2010                  |  |
| 27. | Создание фотоальбома в программе Microsoft Power Point 2010                                 |  |
| 28. | Интерфейс программы Corel Draw CS5                                                          |  |
| 29. | Панель инструментов программы Corel Draw CS5                                                |  |
| 30. | Инструменты создания геометрических фигур в программе Corel Draw CS5                        |  |

### Приложение 3

- 1) Запустите программу, о которой необходимо создать видеоурок (MS Word, MS Excel, MS Power Point или CorelDraw) и уменьшите размеры окна приложения.
- 2) Запустите программу Camtasia Studio и нажмите на кнопке Record the Screen.
- 3) В открывшейся Панели записи выполните следующие настройки:
  - a) В группе Select Area выберите пункт Custom и установите размеры экрана, с которого будет производиться запись, равным окну соответствующего вашему варианту приложения (приложения, открытого в пункте 1).
  - b) Отключите использование web-камеры и запись звука.
  - c) С помощью меню Effects установите подсвечивание указателя и щелчки мыши (Effects – Cursor – Highlight cursor & clicks).
  - d) Откройте меню Tools – Options. В открывшемся окне Tools Options в разделе Saving вкладки General, выбираем формат файла .camrec .
  - e) Настройте панель записи так, чтобы отображались группы Statistics, Duration, Effects (в меню Tools выбрать пункт Recording toolbars и установить соответствующие пункты).

### Приложение 4



## Приложение 5

- 1) Для выполнения записи действий с экрана монитора нажмите на кнопку Rec на Панели записи



- 2) Появится предупреждающее окошко с надписью: Press F10 to stop the recording (Нажмите F10 для остановки записи). После этого пойдет обратный отсчет: 3, 2, 1, запись... Ограничивающая рамка области записи в виде зеленой пунктирной линии будет преобразована в зеленые мигающие уголки.
- 3) Для того чтобы сделать паузу во время записи необходимо нажать кнопку Pause.



- 4) Для того чтобы завершить запись необходимо нажать на клавиатуре кнопку F10 либо развернуть Панель записи и нажать на ней кнопку Stop.



- 5) Если мы хотим отказаться от сохранения сделанной записи и удалить ее, необходимо нажать на кнопку Delete.



- 6) После того как мы сделали необходимую запись, нажимаем кнопку Stop. Сразу открывается окно предварительного просмотра Preview.
- 7) Если записанное видео нам не понравилось – нажимаем на кнопку Delete.



- 8) Поскольку наша запись требует доработки, нажимаем в окне предварительного просмотра кнопку Save and Edit (Сохранить и редактировать).



|                |                                                                                                                          |
|----------------|--------------------------------------------------------------------------------------------------------------------------|
| <b>Тема 26</b> | <b>Редактирование видеоклипов, вставка выносок и использование эффекта увеличения (Zoom) в программе Camtasia Studio</b> |
|----------------|--------------------------------------------------------------------------------------------------------------------------|

### 1.1. Технология обучения на лекции

|                                                                                                                                                                                                                                                                             |                                                                                                                                                                                                                                                                                   |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Учебное время: 2 часа.</i>                                                                                                                                                                                                                                               |                                                                                                                                                                                                                                                                                   |
| <i>Форма учебного занятия: лабораторная работа</i>                                                                                                                                                                                                                          |                                                                                                                                                                                                                                                                                   |
| <i>План занятия</i>                                                                                                                                                                                                                                                         | <ol style="list-style-type: none"> <li>1. Редактирование видео (выделение, удаление или перемещение фрагментов ролика).</li> <li>2. Вставка выносок.</li> <li>3. Использование увеличения (Zoom).</li> </ol>                                                                      |
| <i>Цель учебного занятия:</i>                                                                                                                                                                                                                                               | Сформировать умения по редактированию видеоклипа, вставки выносок и использованию инструмента Zoom в программе Camtasia Studio                                                                                                                                                    |
| <i>Педагогические задачи:</i><br>Организовать работу студентов по <ul style="list-style-type: none"> <li>• редактированию видео (выделение, удаление или перемещение фрагментов ролика);</li> <li>• вставке выносок;</li> <li>• использованию увеличения (Zoom).</li> </ul> | <i>Результаты учебной деятельности:</i><br><i>Студент должен уметь:</i> <ul style="list-style-type: none"> <li>• Редактировать видео (выделение, удаление или перемещение фрагментов ролика).</li> <li>• Вставлять выноски.</li> <li>• Использовать увеличение (Zoom).</li> </ul> |
| <i>Методы обучения</i>                                                                                                                                                                                                                                                      | Обучение сообща                                                                                                                                                                                                                                                                   |
| <i>Средства обучения</i>                                                                                                                                                                                                                                                    | Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение                                                                                                                                                                                                 |
| <i>Формы организации учебной деятельности</i>                                                                                                                                                                                                                               | индивидуальная.                                                                                                                                                                                                                                                                   |
| <i>Условия обучения</i>                                                                                                                                                                                                                                                     | Кабинет вычислительной техники                                                                                                                                                                                                                                                    |

### Технологическая карта лабораторного занятия

| Этап работы, время | Содержание деятельности |           |
|--------------------|-------------------------|-----------|
|                    | Преподавателя           | Студентов |

|                                                          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                                                                                                                                                                                                                                                                                |
|----------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1-й этап.<br>Введение в учебное занятие<br>(5 мин)       | 1.10. Сообщает, тему, цель, планируемые результаты учебного занятия и план его проведения. ( <i>приложение 1</i> ).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | Слушают, уточняют, задают вопросы.                                                                                                                                                                                                                                             |
| 2-й этап.<br>Основная часть<br>(73 мин)                  | <p>2.1. Напоминает способы редактирования видеоклипов в среде Camtasia Studio и предлагает выполнить необходимые операции редактирования над видеофайлами, записанными на предыдущем занятии (<i>приложение 2</i>).</p> <p>2.2. «Обучение сообща». Делит студентов на группы и предлагает ознакомиться со способами вставки выносок в клип (<i>приложение 3</i>).</p> <p>2.3. Предлагает оформить выноски в своих видеоуроках.</p> <p>2.4. Знакомит с основными моментами по использованию эффекта увеличения (Zoom) (<i>приложение 4</i>).</p> <p>2.5. Предлагает применить эффект увеличения в своих видеоуроках и сохранить результат.</p> | <p>2.1. Выполняют редактирование видеоклипа.</p> <p>2.2. В группах разбирают виды выносок и способы их внедрения в клип.</p> <p>2.3. Оформляют различные виды выносок в своих видеоуроках.</p> <p>2.4. Слушают.</p> <p>2.5. Применяют инструмент Zoom в своих видеоуроках.</p> |
| 3-й этап.<br>Заключительная-результатирующая<br>(10 мин) | 3.1. Дает и разъясняет задания для домашней работы:<br>Доработать видеоуроки.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | Задают уточняющие вопросы.                                                                                                                                                                                                                                                     |

## Приложение 1

**Тема: Редактирование видеоклипов, вставка выносок и использование эффекта увеличения (Zoom) в программе Camtasia Studio**

**Цель занятия:** сформировать умения по редактированию видеоклипа, вставки выносок и использованию инструмента Zoom в программе Camtasia Studio.

**План занятия:**

1. Редактирование видео (выделение, удаление или перемещение фрагментов ролика).
2. Вставка выносок.
3. Использование увеличения (Zoom).

## Приложение 2

1. Откройте окно программы Camtasia Studio.
2. Дайте команду Файл – Открыть проект и в открывшемся диалоговом окне выберите видеофайл, записанный вами на предыдущем уроке.
3. Просмотрите видео в области предварительного просмотра и отметьте фрагменты, которые необходимо удалить или переместить.
4. С помощью временной шкалы выполните операции удаления или перемещения фрагментов видео.

Рассмотрим некоторые операции редактирования, которые можно совершать с клипами, добавленными на временную шкалу.

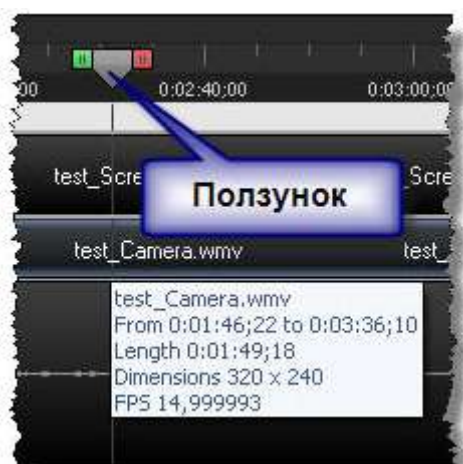



Рис. 1. Ползунок воспроизведения.

- а. Для того чтобы разрезать (разделить) видео в определенном месте необходимо сначала поместить туда бегунок, а затем на панели инструментов нажать на кнопку Split (Разделить) . Также можно зайти в меню Edit (Правка) и нажать на пункт Split (Разделить). Либо на клавиатуре нажать на клавишу S. Разделить видео на части нам может понадобиться для того, чтобы удалить неблагоприятные участки.
- б. Мы уже сказали о том, как разделить (разрезать наше видео). Теперь рассмотрим, как выделить отдельный участок клипа. Для этого нужно воспользоваться красным или зеленым ярлыком, который имеется по сторонам от ползунка. Для того чтобы выделить участок необходимо потянуть указателем мыши за эти ярлыки.

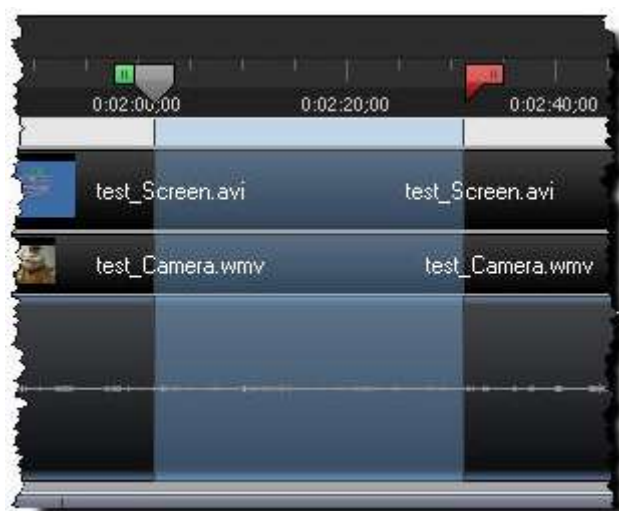





Рис. 2. Выделенный участок клипа.

- с. Теперь мы можем совершать операции с этим выделенным участком. Для этого можно воспользоваться меню Edit (Правка), горячими клавишами и кнопками на панели инструментов Временной шкалы.



| Команда                  | Кнопка на панели инструментов Временной шкалы                                       | Сочетание клавиш |
|--------------------------|-------------------------------------------------------------------------------------|------------------|
| <b>Cut</b> (Вырезать)    |  | Ctrl + X         |
| <b>Copy</b> (Копировать) |  | Ctrl + C         |
| <b>Paste</b> (Вставить)  |  | Ctrl + V         |

- d. После того, как мы вырезали или скопировали участок клипа, мы можем его вставить в любом месте нашего видео. В месте вставки клип будет разрезан и в месте разреза будет вставлен скопированный участок.
- e. Для того чтобы вернуть ползунку прежний вид необходимо сделать двойной щелчок левой кнопкой мыши на нем.
- f. Для того чтобы отменить сделанные операции нужно воспользоваться кнопками на панели инструментов Временной шкалы . Либо воспользоваться горячими клавишами: Ctrl + Z – отмена последней сделанной операции, Ctrl + Y – повторение последней сделанной операции. Либо воспользоваться в меню Edit командами Undo (Отменить) и Redo (Повторить).

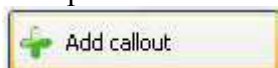
### Вставка выносок.

Программа Camtasia Studio представляет широкие возможности по вставке различных выносок в обучающий фильм.

Для того чтобы вставить выноску в определенный кадр необходимо нажать на панели вкладок кнопку Callouts (Выноски).



В сервисной панели необходимо нажать на кнопку Add callout (Добавить выноску).



Также можно нажать горячую клавишу на клавиатуре С.

В окне предварительного просмотра мы можем видеть добавленную выноску. По умолчанию это стрелка.



Рис. 7.1. Добавленная выноска в виде стрелки в окне предварительного просмотра.

На временной шкале появится дополнительная дорожка Callout и мы увидим клип, обозначающий выноску в виде черного прямоугольника с закругленными углами.



Рис. 7.2. Выноска на временной шкале.

Для того чтобы выделить выноску необходимо щелкнуть один раз левой кнопкой мыши по выноске в окне предварительного просмотра либо на временной шкале. На временной шкале выноска появляется в момент добавления в том месте, в котором находился в данный момент ползунок.

Выноску на временной шкале можно перемещать, перетаскивая ее с помощью мыши. Длина прямоугольника обозначающего выноску говорит о продолжительности появления выноски в кадре. Для того чтобы увеличить продолжительность появления выноски в фильме необходимо подвести указатель мыши к краю прямоугольника и потянуть левой кнопкой мыши, тем самым, растягивая прямоугольник.



Рис. 7.3. Изменяем форму выноски с помощью маркеров.

Для того чтобы изменить форму выноски необходимо воспользоваться круглыми маркерами по углам и краям выноски. Для того чтобы изменить расположение выноски в кадре необходимо просто перетянуть ее с помощью мыши. Для того чтобы изменить размеры и пропорции выноски необходимо воспользоваться белыми маркерами. Для того чтобы изменить форму конкретной фигуры можно воспользоваться желтым маркером. Для того чтобы повернуть выноску (в данном

случае изменить направление стрелки) необходимо воспользоваться зеленым маркером.

Теперь обратим свое внимание на сервисную панель.

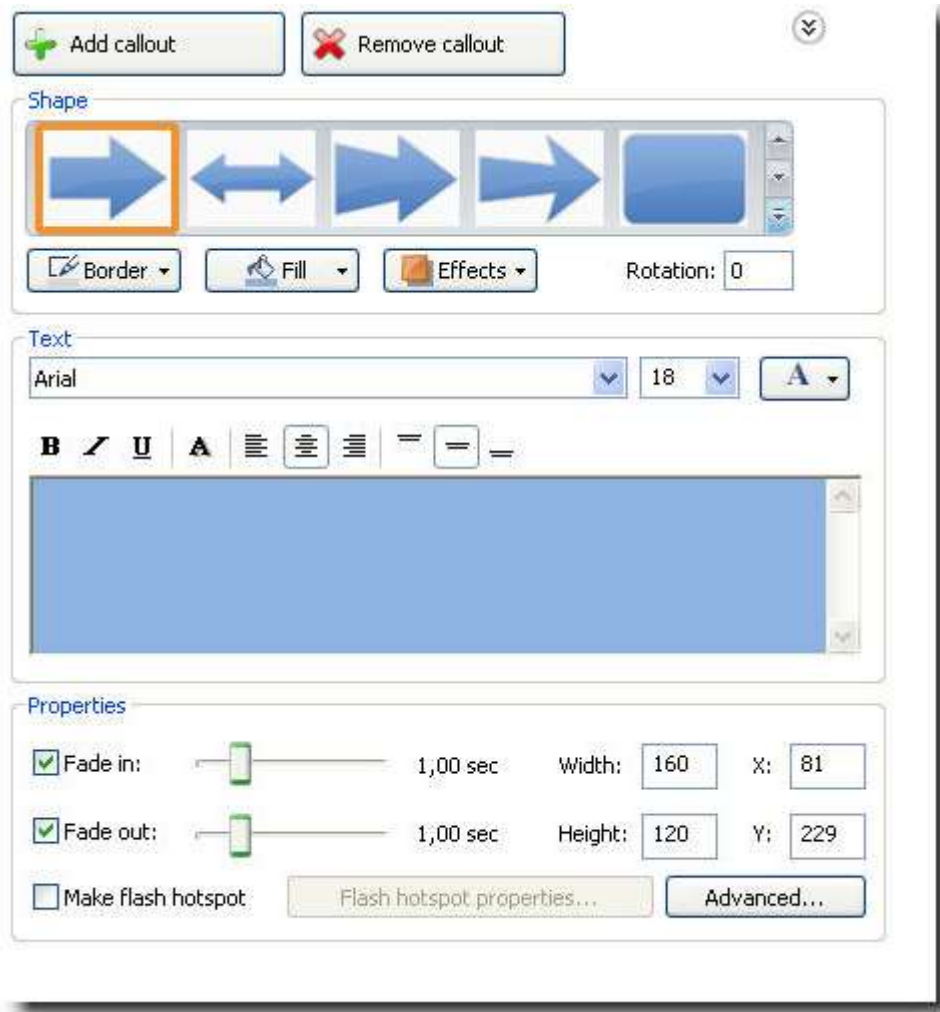


Рис. 7.4. Параметры выносок в сервисной панели.

Если выноска в данный момент выделена, тогда нам доступны дополнительные настройки выносок. В разделе Shape (Форма или Фигура) можно поменять фигуру выноски. Для этого выберите в выпадающем списке подходящую форму.

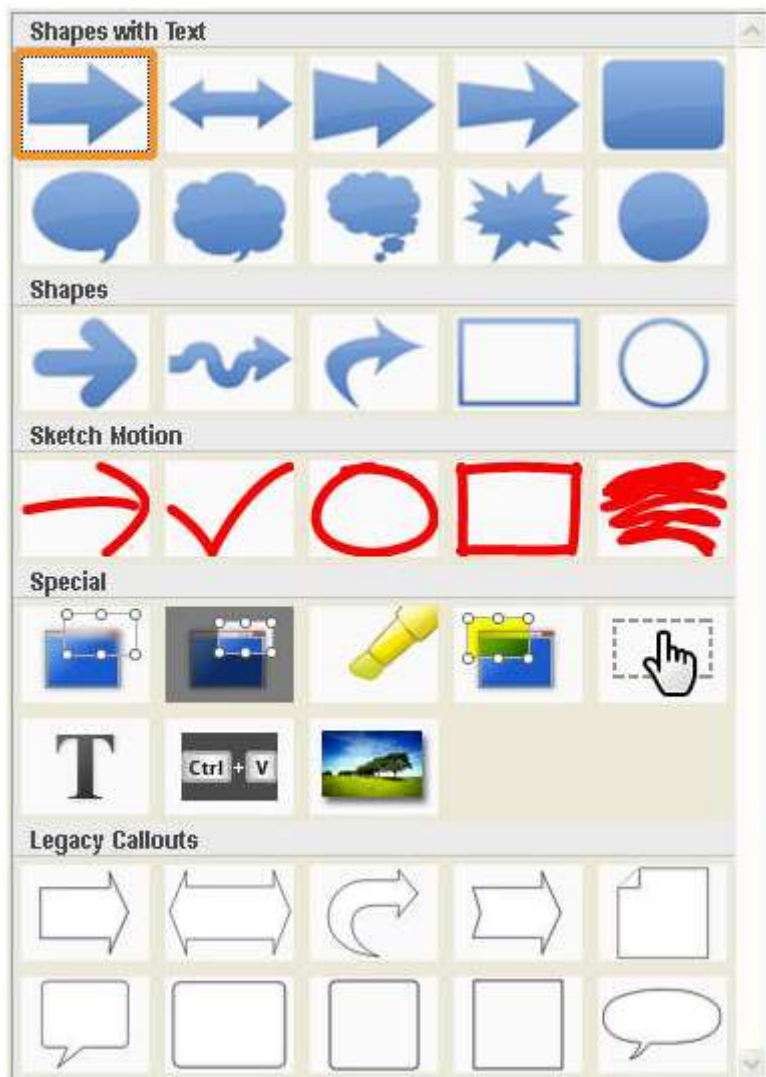



Рис. 7.5. Различные формы выносок.

Если вы сделаете щелчок на какой-либо иконке левой кнопкой мыши, тогда активная выноска (выделенная) в кадре изменит свой вид.

В сервисной панели доступны следующие параметры.

Если нажать на кнопку Border (Граница) , то в выпадающем списке вы сможете выбрать цвет и толщину линии, обрамляющую выноску.

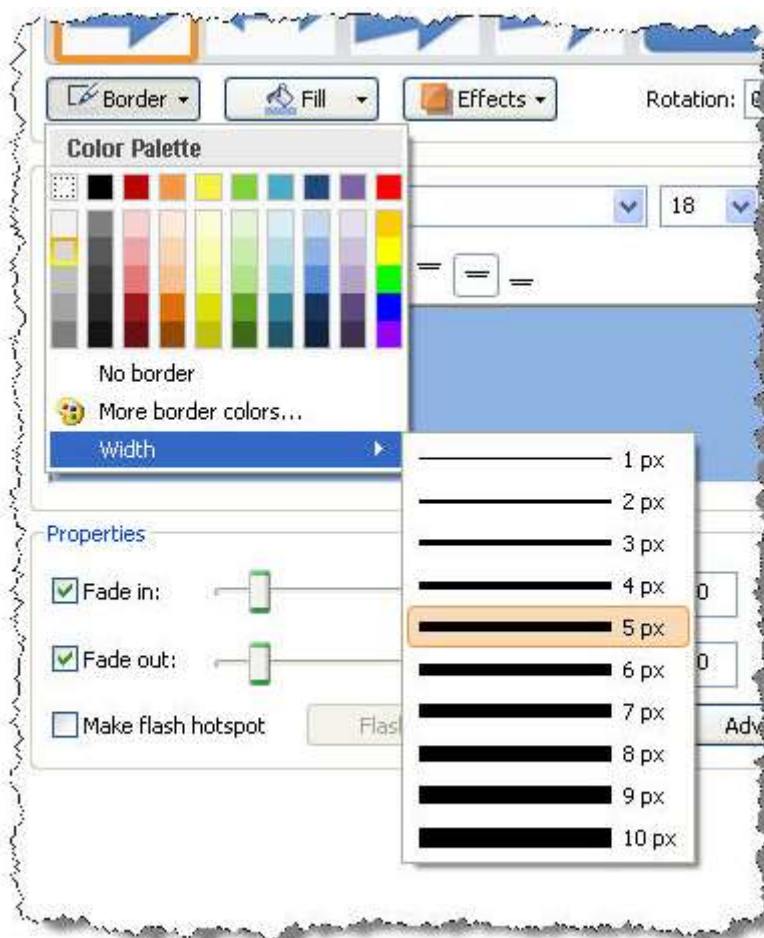

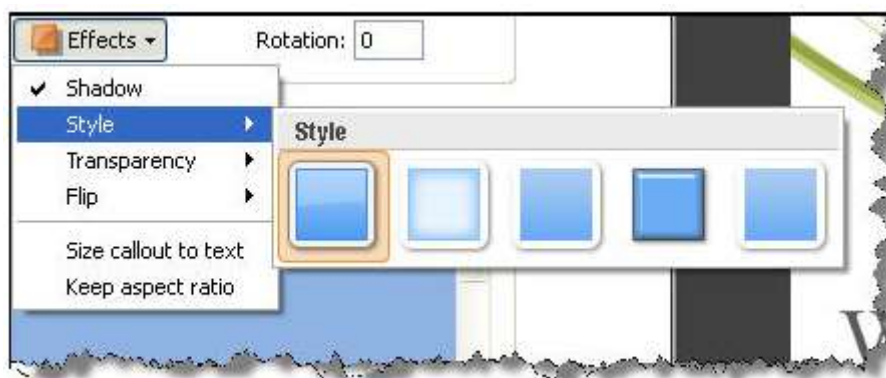


Рис. 7.6. Изменяем границу выноски.

Если выбрать пункт No border (Без границы), тогда линии обрамляющей выноску не будет. В пункте Width (Ширина) в выпадающем списке можно выбрать толщину рамки. Если набор предложенных цветов вас не удовлетворяет, тогда нужно нажать на пункт More border colors (Больше цветов границы) и выбрать подходящий.

Если нажать на кнопку Fill (Заливка) , тогда можно выбрать цвет заливки. Если в выпадающем списке выбрать пункт No fill (Без заливки), тогда наша выноска будет прозрачной.

Если нажать на кнопку Effects (Эффекты), тогда мы сможем изменить вид нашей выноски.



Если отмечен пункт Shadow, тогда выноска имеет тень. В пункте Style изменяем графический стиль выноски, выбирая подходящий из предложенного набора. В пункте Transparency изменяем прозрачность выноски. Имеется готовый набор значений прозрачности выноски. Если нажать на пункт More, то тогда в следующем окошке мы можем произвольно изменить прозрачность с помощью ползунка или введя цифровое значение.

Если отметить "галочкой" пункт Opaque Text, тогда изменения прозрачности не коснутся текста, который был введен в выноску.

В пункте Flip мы можем отразить нашу выноску, выбрав соответствующий пункт Horizontal (Горизонтально) или Vertical (Вертикально).

Если выбрать пункт Size callout to text, тогда размер выноски будет подогнан под размер текста. В этом случае вы не сможете изменять произвольно размер выноски, так как будут отсутствовать белые маркеры изменения размера. Если выбрать пункт Keep aspect ratio тогда один из размеров выноски (ширина или высота) будет изменяться в зависимости от другого размера.

В поле Rotation можно ввести значение угла поворота выноски. При этом выноска поворачивается против часовой стрелки. Текст в выноске при этом ориентируется правильно.

В следующем разделе Text можно ввести текст, который будет отображаться на выноске. Здесь же можно определить гарнитуру, размер, цвет шрифта и выравнивание текста.

В разделе Properties можно определить размеры выноски, введя соответствующие значения в поля Width (Ширина) и Height (Высота).

Положение выноски в кадре можно определить, введя значения в соответствующие поля X и Y. Напомним, что оси X и Y начинаются в левом верхнем углу кадра.

В этом же разделе Properties можно определить плавное появление (Fade in) и/или плавное исчезновение (Fade out) выноски в кадре, отметив "галочкой" соответствующий пункт. Продолжительность появления или исчезновения выноски в кадре определяем в секундах с помощью ползунков.

На временной шкале эффект плавного появления и плавного исчезновения выноски обозначен серыми треугольниками по сторонам клипа (прямоугольника), обозначающего выноску.

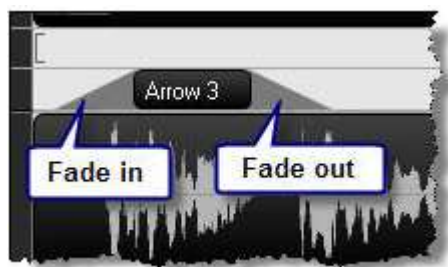
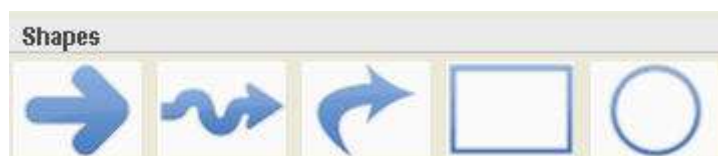


Рис. 7.7. Обозначение плавного появления и плавного исчезновения выноски на временной шкале.

Только что мы с вами рассмотрели выноски, которые позволяют включать в себя текст. В выпадающем списке это группа под названием Shapes with text.



Также в разделе Shape можно выбрать выноски, в которые нельзя вводить текст. В выпадающем списке эта группа носит название просто Shapes.



Свойства этих выносок, отображаемые в сервисной панели нам уже знакомы.

Следующая группа выносок, которая доступна в Camtasia Studio – это Sketch motion



Они представляют собой анимационный эффект рисования маркером по экрану.

В свойствах этой выноски можно указать продолжительность времени рисования в секундах с помощью ползунка в пункте Draw time.

Теперь рассмотрим группу специальных выносок (Special), в которые включены также выделение, подсветка, размытие и др.



Первая выноска Blur callout, позволяет добавлять участки с размытием.

Это необходимо когда мы хотим скрыть какой-либо участок в кадре нашего видео. Например, адрес сайта, электронной почты, лицензионный номер и т.д.

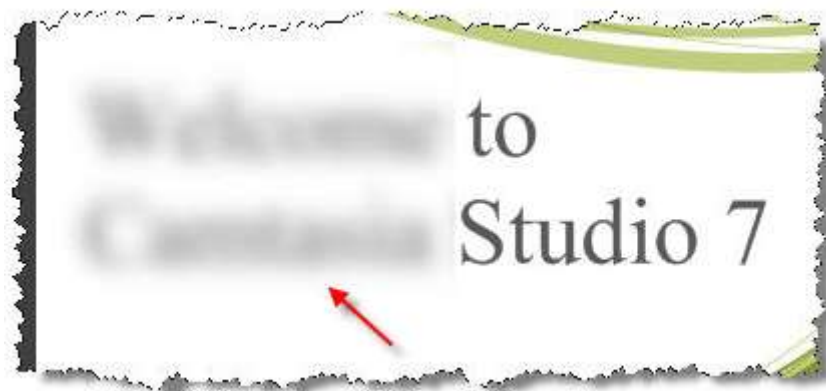
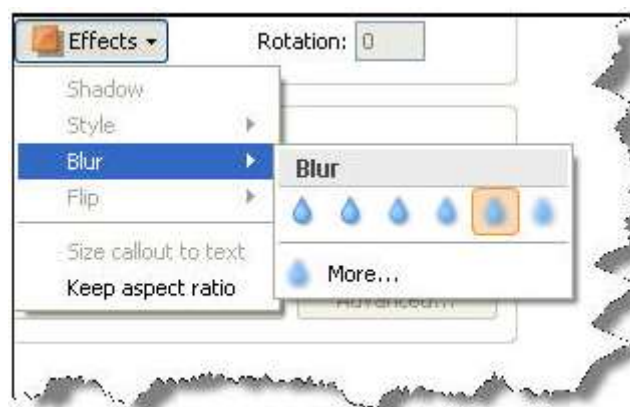


Рис. 7.8. Применение эффекта Blur callout – размытие.

В настройках этой выноски нет ничего для нас нового, кроме параметра Blur, который можно вызвать, нажав на кнопку Effects.



Следующая выноска это Spotlight – Затенение.

При ее применении затеняется вся область экрана, а область самой выноски остается прозрачной.

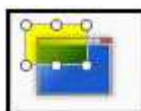


Рис. 7.9. Применение эффекта Spotlight.

Следующие две выноски ничем друг от друга не отличаются, кроме того, что они предназначены для подсветки темных и светлых областей в кадре.



Highlight callout – предназначена для подсветки темных областей в Кадре



Highlight rectangle – предназначена для подсветки светлых областей в Кадре.

Следующая выноска - это Hotspot (Горячая область). Предназначена для добавления



интерактивных элементов в ваш фильм.

По сути дела эта выноска является кнопкой. Естественно, работать будет в том случае, если ваш фильм экспортирован в формате Flash.

С помощью этой выноски можно осуществить навигацию по фильму с помощью нажатий мышкой по определенным участкам кадра. Единственное что нужно отметить, эта выноска является прозрачной, поэтому чтобы она была видна на экране, необходимо подложить под нее, например другую выноску, допустим стрелку, чтобы пользователю было понятно на что нажимать. В эту выноску можно добавлять текст. Кроме того, с помощью этого элемента Hotspot можно симитировать работу в какой-либо программе, например, нажатие на пункты меню.

Для того чтобы настроить интерактивность этой выноски необходимо в сервисной панели нажать на кнопку Flash hotspot properties.

В следующем окне настраиваем интерактивность.

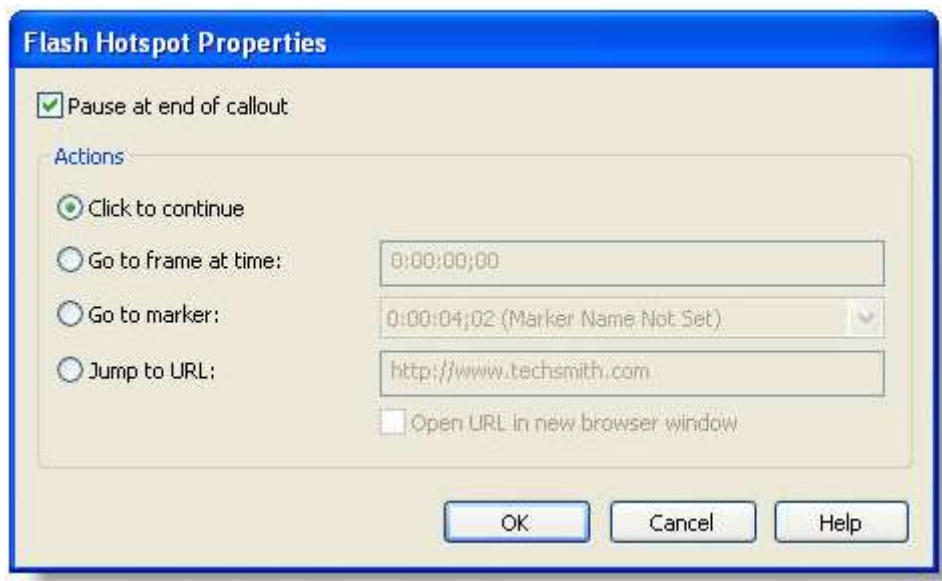


Рис. 7.10. Настройка интерактивности Hotspot (Горячей области).

Пункт **Pause at end of callout** (Пауза в конце выноски) отмечен по умолчанию, его нельзя отключить. Зритель должен щелкнуть по выноске, чтобы продолжить просмотр.

**Click to continue** (Щелчок для продолжения) – для продолжения просмотра зритель должен щелкнуть по области.

**Go to frame at time** (Перейти в кадр в определенный момент времени) – после щелчка зритель продолжает просматривать запись с определенного момента времени, которое можно ввести в поле.

**Go to marker** (Перейти в точку с меткой) – после щелчка зритель просматривает запись с места в видео, в котором до этого была установлена метка. Напротив этого пункта можно выбрать метку из выпадающего списка

**Jump to URL** (Перейти по ссылке) – в этом случае при нажатии на область зритель будет перенаправлен по указанному Интернет-адресу.

Пункт **Open URL in new browser window** означает, что страница откроется в новом окне браузера.

Нужно добавить, что свойства интерактивности можно применить и к другим категориям выносок, но не ко всем.

Для того чтобы применить интерактивность к конкретной выноске нужно в ее свойствах отметить "галочкой" пункт **Make flash hotspot**, а затем нажать на кнопку **Flash hotspot properties**, появится окно **Flash hotspot properties**, которое мы только что рассмотрели.

К выноскам из группы **Sketch motion**, к выноскам **Blur callout**, **Spotlight**, **Highlight callout**, **Highlight rectangle**, свойства интерактивности применить нельзя.

Следующий тип выноски, который можно вставлять в Camtasia Studio, - это текстовая



Ничего особенного из себя она не представляет. В выноске можно просто ввести необходимый нам текст. В сервисной панели со всеми настройками этой выноски мы уже знакомы.

Следующий интересный тип выноски – это Traditional keystrokes.



Позволяет визуализировать нажатие сочетаний клавиш.

Для того чтобы вставить такую выноску в сервисной панели в разделе Text нужно щелкнуть левой кнопкой мыши в поле Enter keystroke or combination (Нажмите клавишу или комбинацию). После этого на клавиатуре необходимо нажать сочетание клавиш.

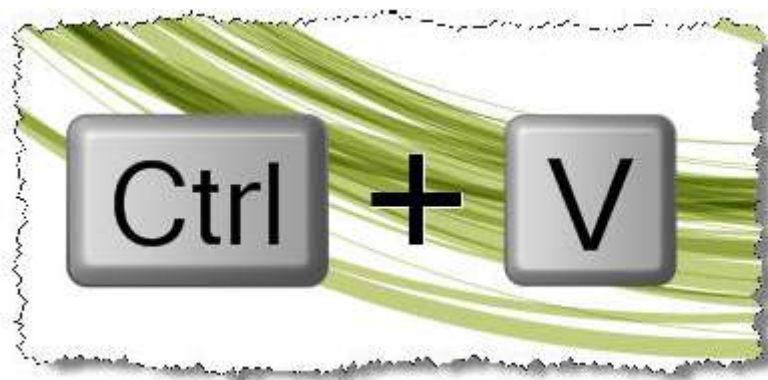


Рис. 7.11. Пример использования выноски Traditional keystrokes.

Если мы в процессе записи действий с экрана нажимали сочетание клавиш на клавиатуре, то данный тип выносок мы можем вставить автоматически. Для этого в сервисной панели нужно нажать на кнопку Generate from camrec.

Появится предупреждающее окно, в котором вы соглашаетесь или нет с тем, что предыдущие выноски, визуализирующие нажатие клавиш на клавиатуре будут заменены на новые.



Рис. 7.12. Все существующие выноски Traditional keystrokes будут заменены на вновь созданные. Хотите продолжить?

Далее появится еще одно окно, в котором вы определяете некоторые параметры добавляемых выносок.

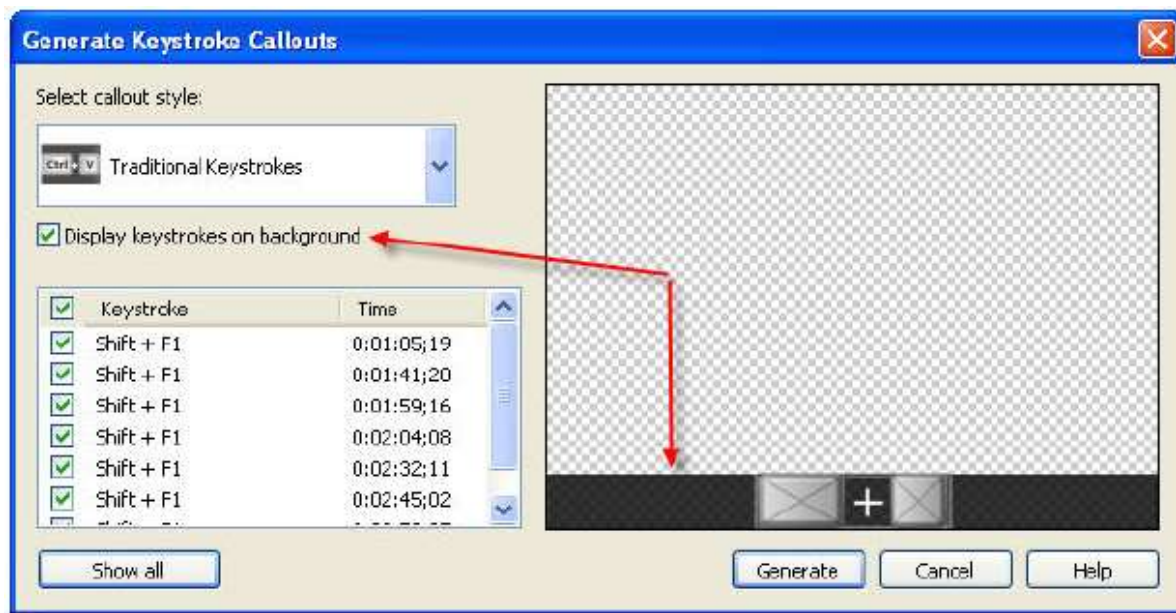
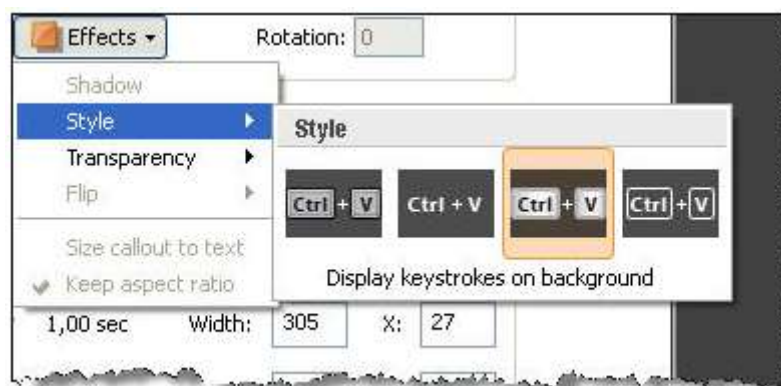


Рис. 7.13. Автоматическое добавление выносок Traditional keystrokes

В выпадающем списке Select callout style выбираем нужный нам стиль выносок. Если отметить пункт Display keystrokes on background, то в этом случае выноски будут отображены на фоне. Ниже в таблице отмечаем необходимые нам сочетания клавиш, после чего можно нажать на кнопку Generate.

Нужно отметить, что такая операция будет возможна, если перед записью в настройках Панели записи (Рекордера), (меню Tools, пункт Options, окно Tools Options, вкладка General, раздел Capture), был отмечен пункт Capture keyboard input (Захватывать нажатия клавиш на клавиатуре). Впрочем, этот пункт отмечен "галочкой" по умолчанию.

Стиль такого типа выносок можно поменять также, если нажать на кнопку Effects в сервисной панели.



Также в качестве выноски можно использовать простой рисунок. После нажатия на следующую иконку выноски открывается окно, в котором можно выбрать файл рисунка.




Доступны форматы файлов рисунков BMP, PNG и JPG.

Настройки данной выноски аналогичны рассмотренным до этого.

Интересно, что для этого типа выносок можно применять заливку. Также поверх вставленной картинки можно ввести текст.

Также нужно отметить, что если в корзине клипов (Clip Bin) была загружена картинка, то ее тоже можно использовать как выноску. Для этого по ней нужно щелкнуть правой кнопкой мыши и в контекстном меню выбрать пункт Add as callout (Добавить как выноску).

Для того чтобы удалить выноску необходимо сначала выделить ее одинарным щелчком левой кнопки мыши, а затем на клавиатуре нажать клавишу Delete, либо нажать правую кнопку мыши и в контекстном меню выбрать пункт Remove from timeline.

Если в сервисной панели нажать на кнопку , то тогда появится список всех выносок, которые мы вставили в свой фильм. Отмечая в списке определенный пункт, можно легко перемещаться по временной шкале, находя ту или иную нужную нам выноску.

## Приложение 4

### Использование увеличения (Zoom).

Программа Camtasia Studio предоставляет очень интересный методический инструмент, который можно и нужно использовать в своих видео уроках – это инструмент приближения или увеличения отдельных участков кадра (Zoom).

Для того чтобы перейти к использованию этого инструмента необходимо в Панели вкладок нажать на кнопку Zoom-n-Pan.



После этого на временной шкале появится дополнительная дорожка Zoom. Для добавления увеличения в сервисной панели необходимо нажать на кнопку Add

keyframe (Добавить ключевой кадр).



На временной шкале, на дорожке Zoom появится ключевой кадр в виде синего ромбика. Это означает, что если ползунок дойдет до этого кадра, тогда на экране будет произведено увеличение определенного участка кадра. Только перед этим этот

участок кадра нам необходимо указать. Делается это в сервисной панели в небольшом окошке предварительного просмотра.

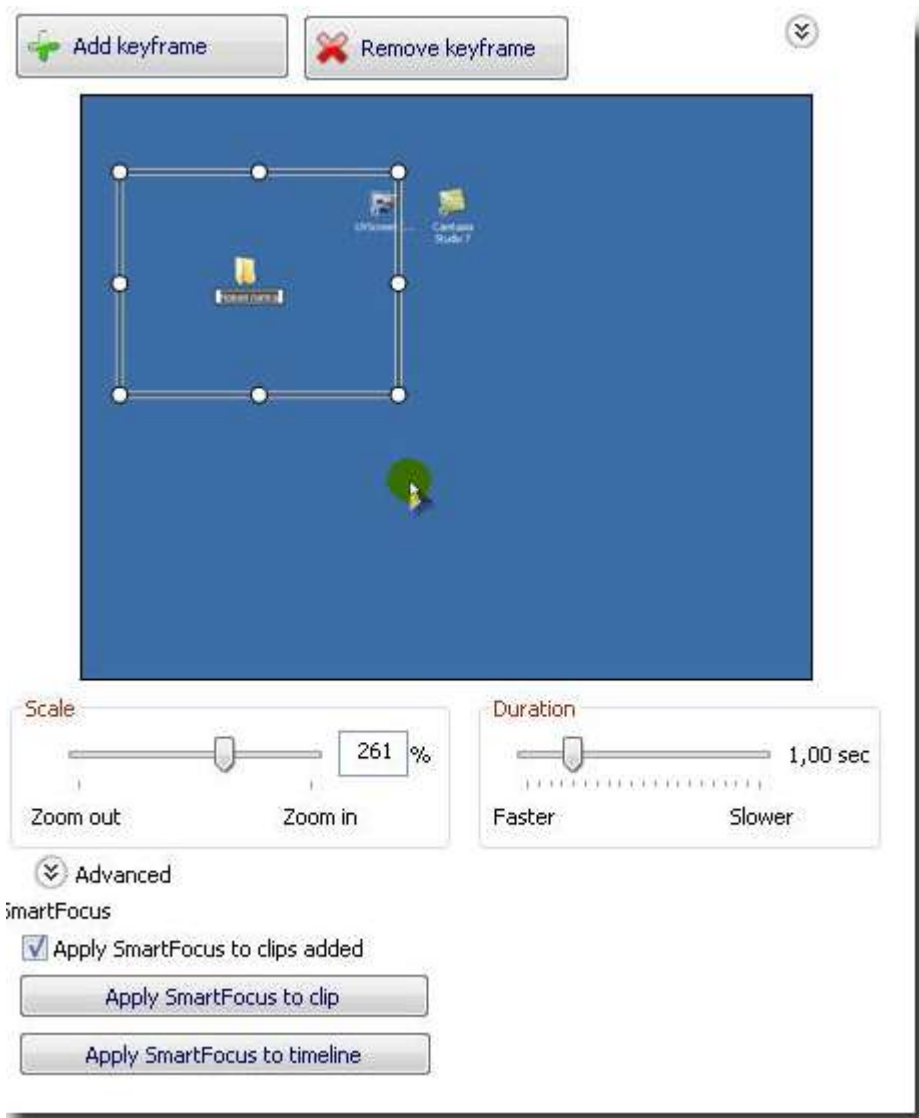


Рис. 8.1. Параметры в сервисной панели инструмента Zoom-n-Pan.


В этом окошке имеется рамка увеличения с круглыми белыми маркерами. После того как вы добавили ключевой кадр на временной шкале, эта рамка занимает все пространство кадра. В этом случае увеличения не происходит. Далее с помощью маркеров ограничивающей рамки мы можем уменьшить размеры этой рамки. Это означает, что когда ползунок дойдет до ключевого кадра будет произведено увеличение до пределов этой рамки. Это будет видно в окне предварительного просмотра справа. Т.е. зритель увидит только часть кадра, увеличенную до размеров всего кадра.

В разделе Scale (Увеличение) мы можем указать значение масштабирования с помощью ползунка или числа. В разделе Duration (Продолжительность) мы указываем время увеличения в секундах. На временной шкале время, в течение

которого происходит увеличение, отмечено синим треугольником слева от ключевого кадра.

После того как мы произвели увеличение определенного участка кадра,

нам нужно это увеличение ликвидировать, т.е. произвести обратную операцию. Для этого добавляем еще один ключевой кадр, нажатием на кнопку Add keyframe. В настройках этого ключевого кадра указываем увеличение (Scale) равное 100%. Т.е. дойдя до этого кадра, зритель опять увидит весь кадр полностью.

Если нажать на кнопку  напротив пункта Advanced, раскроются дополнительные опции для настройки.

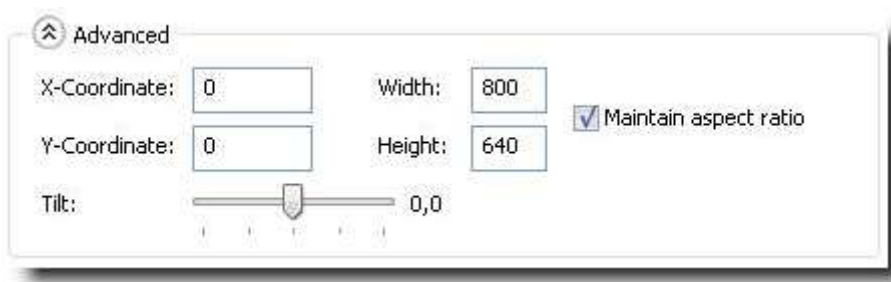


Рис. 8.2. Дополнительные настройки инструмента Zoom-n-Pan.



В полях X-Coordinate и Y-Coordinate указываем значения координаты X и координаты Y соответственно левого верхнего угла рамки увеличения. В полях Width (Ширина) и Height (Высота) указываем ширину и высоту рамки увеличения. Если отмечен пункт Maintain aspect ratio (Сохранять пропорции), то тогда высота рамки увеличения будет изменяться в зависимости от ширины и наоборот.

Очень интересен параметр Tilt (Наклон), изменяя который с помощью ползунка мы изменяем наклон плоскости кадра относительно оси X.




Рис. 8.3. Пример применения эффекта Tilt (Наклон).

После добавления ключевого кадра с эффектом Tilt необходимо добавить еще один кадр с нулевым значением наклона, чтобы кадр принял прежний вид.

Если нажать вверху сервисной панели на кнопку , то тогда появится список всех ключевых кадров, которые мы вставили на временную шкалу. Справа от этого списка имеется кнопка  Expand the selected keyframe(s) to full-screen (Развернуть выбранные ключевые кадры на полный экран), при нажатии на которую увеличивающая рамка занимает все пространство кадра, т.е. увеличения не происходит.

Кстати в списке можно выделять сразу несколько ключевых кадров. Для того чтобы выделить диапазон смежных ключевых кадров, необходимо зажать клавишу Shift на клавиатуре, для того чтобы выделить отдельные не смежные ключевые кадры, необходимо зажать на клавиатуре клавишу Ctrl и выделять указателем мыши.

Ниже находится еще одна кнопка  Open zoom options (Открыть настройки увеличения). Если мы ее нажмем, тогда откроется окно Options вкладка Zoom. Это же окно можно открыть из меню Tools пункт Options.

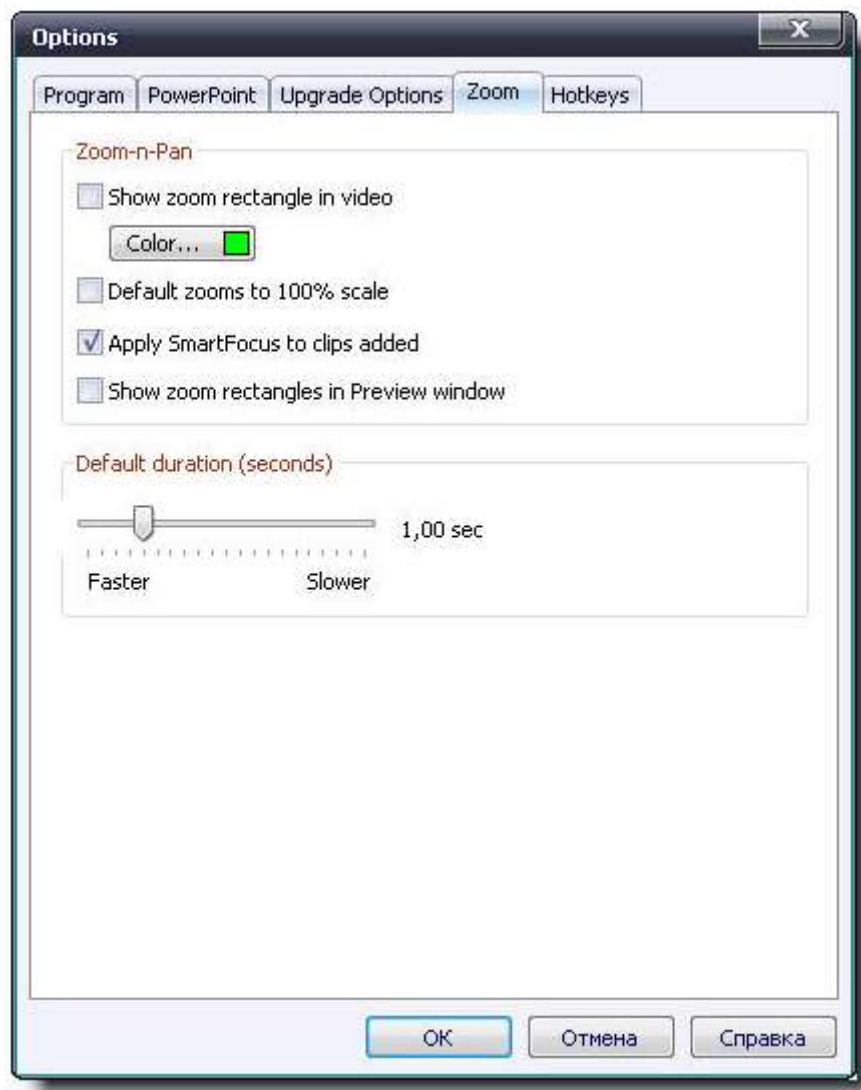


Рис. 8.4. Окно настроек инструмента Zoom.

В разделе Zoom-n-Pan имеются следующие пункты.

Show zoom rectangle in video (Показать прямоугольник увеличения в видео) – при включении этой опции в видео будет показан прямоугольник, ограничивающий область, до которой будет произведено увеличение. Ниже можно сделать настройку цвета этого прямоугольника.

Default zooms to 100% scale (Увеличение по умолчанию масштабом 100%) – если отмечена эта опция, тогда при добавлении ключевых кадров увеличение всегда будет установлено в масштабе 100%, т.е. по сути дела увеличение кадра отсутствует. Это удобно. Тогда после добавления ключевого кадра, в котором мы сделали увеличение, нужно добавить еще один кадр, в котором необходимо убрать увеличение вручную. Если же отмечен этот пункт, тогда этого делать нам нет необходимости.

Apply SmartFocus to clips added (Применить инструмент SmartFocus к добавляемому клипу) – будет применен инструмент SmartFocus в процессе добавления видео на временную шкалу. Об этом инструменте мы поговорим отдельно. Этот пункт отмечен по умолчанию.

Show zoom rectangle in Preview window – показать прямоугольник увеличения в окне предварительного просмотра.

В разделе Default duration (seconds) мы указываем продолжительность увеличения в секундах, которое будет установлено по умолчанию.

В сервисной панели инструмента Zoom-n-Pan имеется также раздел SmartFocus. Суть инструмента SmartFocus заключается в том, что в процессе записи действий с экрана, рекордер Camtasia Studio (Панель записи) автоматически добавляет ключевые кадры увеличения Zoom-n-Pan, на основании движений и положений указателя мыши на экране. После добавления видео на временную шкалу автору удобнее затем редактировать ключевые кадры.

Если вы планируете применять SmartFocus в процессе редактирования, тогда используйте следующие советы, чтобы помочь инструменту SmartFocus предсказать, к чему применить эффект увеличения.

Двигайте указателем мыши по экрану помедленнее. Не нажимайте кнопку мыши слишком часто.

Записывайте действия на всем экране, т.е. записывайте полный экран, а экспортируйте видео при меньших размерах. SmartFocus идеально подходит для случаев, когда вам нужно записать весь экран при текущих разрешениях или более высоких, а экспортировать окончательное видео при небольшом разрешении (рекомендуется 640x480 или меньше).

Например, если вы записываете полноэкранное приложение, а затем экспортируете окончательное видео для распространения на iPod, запись всего экрана невозможно просмотреть при разрешении iPod (320x240) без масштабирования.

Записывайте видео более 30 секунд. SmartFocus оптимизирован для видео продолжительностью более 30 секунд. Для более коротких клипов, вы можете вручную добавить ключевые кадры Zoom-n-Pan.

Наводите указатель мыши туда, куда вы хотите, чтобы смотрел зритель. Держите указатель мыши в том месте, где производится запись. Например, если вы хотите, чтобы зритель сосредоточился на гиперссылке, поместите указатель мыши над гиперссылкой или вблизи нее.

Не "разговаривайте" указателем мыши. Не дергайте, не водите указателем мыши по экрану, когда вы говорите, если этого не требуют ваши комментарии.

Держите указатель мыши неподвижным при вводе текста. SmartFocus не может предсказать, правильное действие, если вы щелкните в текстовом поле, переместите указатель на другую сторону экрана, а затем начнете вводить текст. Если вы хотите, чтобы зритель увидел то, что вы ввели в текстовое поле, держите указатель мыши в текстовом поле.

Используйте колесо прокрутки мыши (если это возможно).

Попробуйте использовать колесо прокрутки мыши, а не перетаскивание на полосе прокрутки. Может случиться так, что инструмент SmartFocus будет увеличивать полосу прокрутки, а не содержание прокручиваемого окна.

Например, колесо мыши можно применять при прокрутке длинных страниц сайта.

Теперь поговорим об оставшихся функциях в сервисной панели, которые касаются инструмента SmartFocus.

В разделе SmartFocus есть опция Apply SmartFocus to clips added (Применить инструмент SmartFocus к добавляемому клипу), о которой мы уже говорили, когда рассматривали окно Options.

Ниже находится кнопка Apply SmartFocus to clip (Применить SmartFocus к клипу). Если нажать на эту кнопку, тогда к выделенному клипу будет добавлены ключевые кадры увеличения. Если нажать на кнопку Apply SmartFocus to timeline (Применить SmartFocus к временной шкале), тогда ключевые кадры увеличения будут добавлены по всей продолжительности фильма. Необходимо помнить, что в этом случае все ключевые кадры, которые были добавлены до этого, будут переопределены.

|                |                                                                                              |
|----------------|----------------------------------------------------------------------------------------------|
| <b>Тема 27</b> | <b>Вставка эффектов переходов и титульного клипа в видеоурок в программе Camtasia Studio</b> |
|----------------|----------------------------------------------------------------------------------------------|

### 1.1. Технология обучения на лекции

|                                                                                                                                                                                                 |                                                                                                                                                                                                  |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Учебное время:</i> 2 часа.                                                                                                                                                                   |                                                                                                                                                                                                  |
| <i>Форма учебного занятия:</i> лабораторная работа                                                                                                                                              |                                                                                                                                                                                                  |
| <i>План занятия</i>                                                                                                                                                                             | <ol style="list-style-type: none"> <li>1. Использование эффектов переходов.</li> <li>2. Добавление титульного клипа.</li> </ol>                                                                  |
| <i>Цель учебного занятия:</i>                                                                                                                                                                   | Сформировать умения по использованию эффектов переходов и добавлению титульного клипа в видеоурок в программе Camtasia Studio                                                                    |
| <i>Педагогические задачи:</i><br>Организовать работу студентов по <ul style="list-style-type: none"> <li>• использованию эффектов переходов;</li> <li>• добавлению титульного клипа.</li> </ul> | <i>Результаты учебной деятельности:</i><br><i>Студент должен уметь:</i> <ul style="list-style-type: none"> <li>• Использовать эффекты переходов.</li> <li>• Добавлять титульный клип.</li> </ul> |
| <i>Методы обучения</i>                                                                                                                                                                          | Частично-поисковый метод                                                                                                                                                                         |
| <i>Средства обучения</i>                                                                                                                                                                        | Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение                                                                                                                |
| <i>Формы организации учебной деятельности</i>                                                                                                                                                   | индивидуальная.                                                                                                                                                                                  |
| <i>Условия обучения</i>                                                                                                                                                                         | Кабинет вычислительной техники                                                                                                                                                                   |

### Технологическая карта лабораторного занятия

| Этап работы, время                                 | Содержание деятельности                                                                                      |                                         |
|----------------------------------------------------|--------------------------------------------------------------------------------------------------------------|-----------------------------------------|
|                                                    | Преподавателя                                                                                                | Студентов                               |
| 1-й этап.<br>Введение в учебное занятие<br>(5 мин) | 1.1. Сообщает, тему, цель, планируемые результаты учебного занятия и план его проведения.<br>(приложение 1). | Слушают, уточняют, задают вопросы.      |
| 2-й этап.<br>Основная часть                        | 2.1. Рассказывает о возможностях использования эффекты переходов в видеоуроках<br>(приложение 2).            | 2.1. Слушают, задают вопросы, уточняют. |

|                                                                     |                                                                                                                                                                                                                                       |                                                                                                                                      |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| (74 ин)                                                             | 2.2. Предлагает оформить эффекты переходов в своих видеоуроках.<br>2.3. Знакомит со способом добавления титульного клипа ( <i>приложение</i> ).<br>2.4. Предлагает вставить титульный клип в своих видеоуроках и сохранить результат. | 2.2. Оформляют эффекты переходов в своих видеоуроках.<br><br>2.3. Слушают.<br><br>2.4. Вставляют титульный клип в своих видеоуроках. |
| 3-й этап.<br>Заключитель-<br>но-<br>результирую-<br>щий<br>(10 мин) | 3.1. Дает и разъясняет задания для домашней работы:<br>Доработать видеоуроки.                                                                                                                                                         | Задают уточняющие вопросы.                                                                                                           |

#### Приложение 1

#### Тема: Вставка эффектов переходов и титульного клипа в видеоурок в программе Camtasia Studio

**Цель занятия:** сформировать умения по использованию эффектов переходов и добавлению титульного клипа в видеоурок в программе Camtasia Studio.

#### План занятия:

1. Использование эффектов переходов.
2. Добавление титульного клипа.

#### Приложение 2

### Использование эффектов переходов.

В этом уроке мы с вами поговорим об использовании эффектов переходов. Эффекты переходов обычно используются при монтаже видео и предоставляются практически всеми редакторами видео монтажа. Предоставляет эту функцию и программа Camtasia Studio.

Для того чтобы перейти к использованию эффектов переходов необходимо на Панели вкладок нажать на кнопку Transitions.



Временная шкала автоматически перейдет в режим раскадровки.

Напомним, что в этот режим можно также перейти из меню View (Вид) пункт Storyboard (Раскадровка), либо с помощью сочетания клавиш Ctrl+G.



Рис. 1. Временная шкала в режиме раскадровки.

Каждый кадр, по сути, здесь представляет собой отдельный клип. Если мы перейдем в режим раскадровки сразу после начала редактирования, т.е. после того, как только мы закончили запись с экрана, то тогда здесь будет присутствовать только один кадр (клип). Отдельные кадры (клипы) можно сортировать, просто перетаскивая их указателем мыши.

В сервисной панели мы видим иконки, которые обозначают тот или иной вид эффекта переходов. Для того чтобы просмотреть конкретный переход необходимо сделать двойной щелчок указателем мыши на определенной иконке. Пример применения эффекта перехода мы можем видеть в окне предварительного просмотра.

На временной шкале в режиме раскадровки между клипами присутствует иконка со стрелкой. Для того чтобы применить эффект перехода необходимо перетянуть указателем мыши иконку эффекта на стрелку между клипами.



Рис. 2. Вид эффекта перехода на временной шкале в режиме Timeline.

По умолчанию длительность эффекта перехода 3 секунды. Для того чтобы изменить эту длительность, установленную по умолчанию, необходимо в меню Tools выбрать пункт Options, и в окне Options, на вкладке Program, в разделе Default duration (seconds), в пункте Transitions изменить значение.

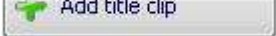
Для того чтобы изменить длительность эффекта перехода на временной шкале необходимо щелкнуть по нему правой кнопкой мыши и в контекстном меню выбрать пункт Transition duration (Продолжительность эффекта перехода). В появившемся окошке указать необходимую продолжительность в секундах.

## Добавление титульного клипа.

В Camtasia Studio имеется возможность вставить в наш фильм титульный или заголовочный клип.

Для того чтобы добавить титульный клип необходимо на панели вкладок нажать на кнопку Title Clips.



А затем в сервисной панели нажать на кнопку Add title clip . На временной шкале будет добавлен титульный клип.

В сервисной панели мы можем ввести текст, который будет показан на экране.

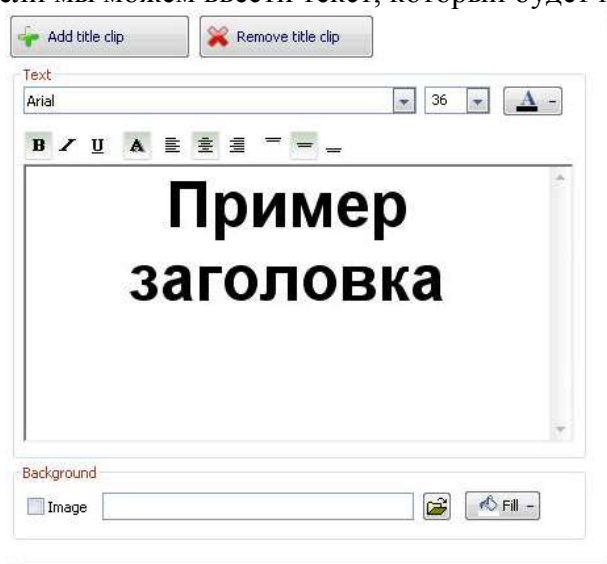



Рис.1. Настройки титульного клипа в сервисной панели.

Также здесь мы можем изменить гарнитуру, размер, цвет и начертание шрифта. Можно также указать выравнивание текста по горизонтали и по вертикали. Изменить расположение текста в пределах кадра также можно в окне предварительного просмотра. Там имеется рамка с белыми маркерами. С помощью этих маркеров мы можем изменить размеры рамки и переместить ее, располагая текст в любом месте кадра.




Рис. 2. Рамка в окне предварительного просмотра, спомощью которой можно перемещать текст титульного клипа по кадру.

Фоновый цвет можно выбрать ниже, если нажать на кнопку Fill (Заливка).

Также можно определить фоновое изображение, указав расположение файла, нажав на кнопку  .

Если убрать «галочку» напротив пункта Image, тогда определенное изображение не будет показано.

Титульные клипы можно вставлять в любое место нашего видео. Предварительно для этого может понадобится разрезать видео.

Напоминаем, что делается это с помощью нажатия на кнопку  на панели инструментов временной шкалы. Либо нажать на клавиатуре горячую клавишу S.

Для того чтобы изменить продолжительность клипа необходимо потянуть его за край на временной шкале с помощью указателя мыши. Либо можно щелкнуть правой кнопкой мыши по клипу и в контекстном меню выбрать пункт Title duration и в появившемся окне указать значение продолжительности в секундах.

Для того чтобы перейти к редактированию конкретного титульного клипа необходимо сделать двойной щелчок по нему на временной шкале.

|                |                                                                                     |
|----------------|-------------------------------------------------------------------------------------|
| <b>Тема 28</b> | <b>Вставка субтитров и тестовых заданий в видеоурок в программе Camtasia Studio</b> |
|----------------|-------------------------------------------------------------------------------------|

### 1.1. Технология обучения на лекции

|                                                                                                                                                                                    |                                                                                                                                                                                           |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Учебное время:</i> 2 часа.                                                                                                                                                      |                                                                                                                                                                                           |
| <i>Форма учебного занятия:</i> лабораторная работа                                                                                                                                 |                                                                                                                                                                                           |
| <i>План занятия</i>                                                                                                                                                                | 1. Работа с субтитрами.<br>2. Добавление теста.                                                                                                                                           |
| <i>Цель учебного занятия:</i>                                                                                                                                                      | Сформировать умения по внедрению субтитров и тестовых заданий в видеоурок в программе Camtasia Studio                                                                                     |
| <i>Педагогические задачи:</i><br>Организовать работу студентов по <ul style="list-style-type: none"> <li>• Работе с субтитрами.</li> <li>• Добавлению тестовых заданий.</li> </ul> | <i>Результаты учебной деятельности:</i><br><i>Студент должен уметь:</i> <ul style="list-style-type: none"> <li>• Работать с субтитрами.</li> <li>• Добавлять тестовые задания.</li> </ul> |
| <i>Методы обучения</i>                                                                                                                                                             | Техника «Зиг-заг»                                                                                                                                                                         |
| <i>Средства обучения</i>                                                                                                                                                           | Лазерный проектор, визуальные и раздаточные материалы, информационное обеспечение                                                                                                         |
| <i>Формы организации учебной деятельности</i>                                                                                                                                      | индивидуальная.                                                                                                                                                                           |
| <i>Условия обучения</i>                                                                                                                                                            | Кабинет вычислительной техники                                                                                                                                                            |

### Технологическая карта лабораторного занятия

| Этап работы, время                                 | Содержание деятельности                                                                                                                                                                                                                                                                                                        |                                    |
|----------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------|
|                                                    | Преподавателя                                                                                                                                                                                                                                                                                                                  | Студентов                          |
| 1-й этап.<br>Введение в учебное занятие<br>(5 мин) | 1.2. Сообщает, тему, цель, планируемые результаты учебного занятия и план его проведения. ( <i>приложение 1</i> ).                                                                                                                                                                                                             | Слушают, уточняют, задают вопросы. |
| 2-й этап.<br>Основная часть<br>(75 ин)             | 2.1. Техника «Зиг-заг». Делит студентов на пары и предлагает каждой из пар ознакомиться с одним из экспертных листов ( <i>приложение 2</i> ) и выполнить соответствующие задания на компьютере.<br>2.2. После того, как каждая пара выполняет свою часть работы, студенты меняются своими партнерами, имеющими на руках разные | 2.1.                               |

|                                                                           |                                                                            |                            |
|---------------------------------------------------------------------------|----------------------------------------------------------------------------|----------------------------|
|                                                                           | экспертные листы и методом взаимообучения учатся выполнять другие функции. |                            |
| 3-й этап.<br>Заключитель-<br>но-<br>результатиру-<br>ющий<br><br>(10 мин) | 3.1. Дает и разъясняет задания для домашней работы:                        | Задают уточняющие вопросы. |

## Приложение 1

### Тема: Вставка субтитров и тестовых заданий в видеоурок в программе Camtasia Studio

**Цель занятия:** сформировать умения внедрению субтитров и тестовых заданий в видеоурок в программе Camtasia Studio.

**План занятия:**

1. Работа с субтитрами.
2. Добавление теста.

## Приложение 2

### Экспертный лист 1

Программа Camtasia Studio позволяет добавлять в наше обучающее видео субтитры (Captions - Подписи). Это может понадобиться, если предполагаемая аудитория – это люди с ограниченными физическими возможностями, например, слабослышащие.

Для того чтобы перейти к добавлению субтитров, на панели вкладок нажимаем на кнопку Captions.



Если эта вкладка отсутствует, тогда необходимо нажать на кнопку More и в выпадающем меню выбрать пункт Captions



Сразу появляется справочное окно Camtasia Studio Tip.

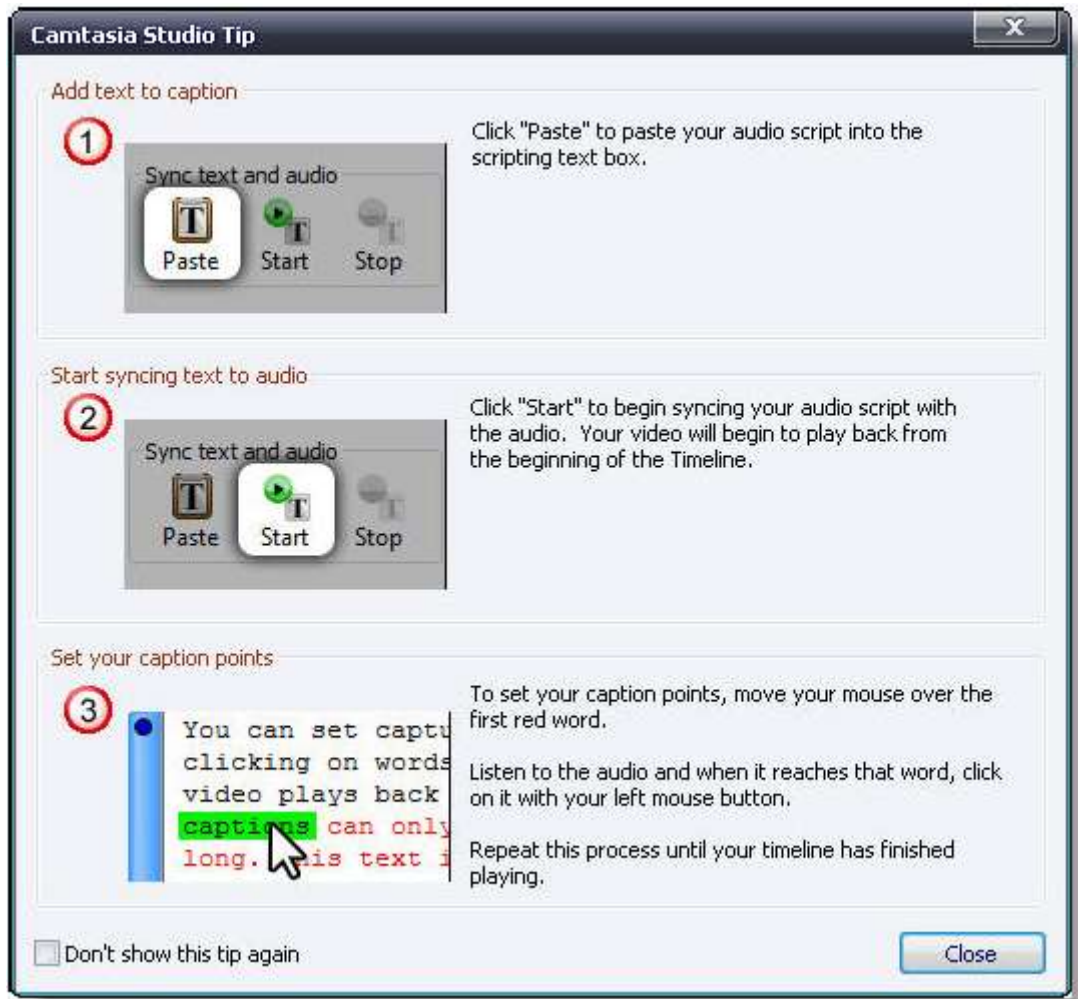


Рис. 13.1. Совет от Camtasia Studio по субтитрам.

В этом окне дается совет о том, как синхронизировать субтитры со звуком в нашем видео. В Camtasia Studio имеется два способа синхронизации субтитров со звуком: это автоматический и ручной способ. В данном случае рассматривается автоматический способ добавления субтитров. Все операции производятся в сервисной панели, которая имеет следующий вид.

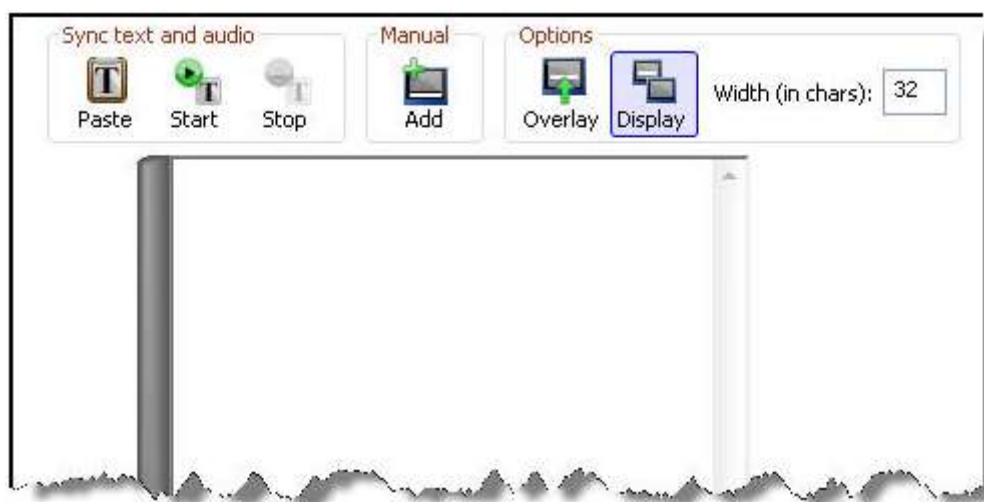


Рис. 13.2. Сервисная панель в режиме добавления субтитров.

### 1) Add text to captions - Добавьте текст субтитров

Click «Paste» to paste your audio script into the scripting text box - Нажмите кнопку «Вставить», чтобы вставить звуковой сценарий в сценарное текстовое поле. Необходимо нажать на кнопку Paste, чтобы вставить текст субтитров из буфера обмена. Естественно перед этим текст нужно скопировать. Текст вставляется в текстовое поле, которое имеется в сервисной панели. Цвет текста будет красный.

### 2) Start syncing text to audio - Начните синхронизацию текста со звуком

Click “Start” to begin syncing your audio script with the audio. Your video will begin to play back from the beginning of the Timeline. Нажмите кнопку "Старт", чтобы начать синхронизацию сценария со звуком. Ваше видео будет воспроизводиться с начала временной шкалы. Вы сможете следить за действиями, которые вы записали в окне предварительного просмотра.

### 3) Set your caption points - Установите точки субтитров

To set your caption points, move your mouse over the first red word. Для установки точки субтитров, наведите указатель мыши на первое красное слово

Listen to the audio and when it reaches that word, click on it with your left mouse button.

Слушайте звук, и когда звуковое сопровождение достигнет этого слова, щелкните на нем левой кнопкой мыши.

Repeat this process until your timeline has finished playing. Повторяйте эту процедуру, пока звук на временной шкале не перестанет воспроизводиться.

В процессе нажатия на красные слова, отдельные участки текста будут появляться в окне предварительного просмотра в виде субтитров внизу.

Слева от текстового поля будут появляться синие точки, обозначающие ключевой кадр, в котором и появится текст субтитров, а также время появления субтитров на временной шкале. Нажимая на синие точки, ползунок перемещается в соответствующее место на временной шкале. При добавлении субтитров на

временной шкале появляется дополнительная дорожка Caption. Ключевые кадры на ней отображаются в виде синих кружочков.



Рис. 13.3. Дорожка Caption на временной шкале и ключевые кадры на ней.

Также субтитры можно добавлять в не автоматическом режиме. Для этого необходимо переместить ползунок в необходимое место на временной шкале и в сервисной панели в разделе Manual нажать на кнопку Add (Добавить). После этого добавить необходимый текст в текстовое поле. При необходимости повторить эту операцию.

Также в сервисной панели имеются некоторые настройки субтитров, которые доступны в разделе Options.

Субтитры в окне предварительного просмотра отображаются внизу кадра на фоне черной полосы, не затрагивая тем самым изображение кадра. Если нажать на кнопку Overlay, тогда субтитры будут наложены прямо на изображение кадра.

Для отключения субтитров в нашем видео нужно нажать на кнопку Display.

Может случиться так, что при автоматическом добавлении текста субтитров в наше видео, отдельные слова могут не помещаться в отведенное для них место на кадре. Эти слова в текстовом поле в сервисной панели отмечаются красным цветом. Эти слова можно просто перенести с помощью операции вырезания-копирования в другой ключевой кадр.

Для текста субтитров отводится определенное поле на кадре. Это поле измеряется по ширине в литерках. По умолчанию ширина этого поля 32 литеры. Можно изменить эту ширину в сервисной панели, введя в поле Width (in chars) числовое значение от 30 до 100. Тогда ширина текстового поля субтитров в сервисной панели измениться. Если мы увеличиваем ширину текстового поля, тогда в окне предварительного просмотра можно видеть, что размер шрифта текста субтитров уменьшается.

Если перед автоматической синхронизацией субтитров со звуком мы уже вставляли ключевые кадры с текстом, то может появиться предупреждающее окно Sync Text and Audio.



Рис. 13.4. Начало записи титров.

Where would you like to begin syncing your text and audio?

Где бы вы хотели начать, чтобы синхронизировать текст и звук?

Start at beginning of Timeline and remove all existing captions points

Начните в начале Временной шкалы и удалите все существующие точки субтитров. Будут удалены все существующие ключевые кадры с субтитрами.

Start at current captions and remove subsequent caption points

Начните с текущего текста и удалите последующие точки субтитров.

## Приложение 3

### Добавление теста.

Программа Camtasia Studio предоставляет возможность добавить интерактивный тест в видео урок. Тест естественно сохраняется, только если видео урок будет экспортирован в Flash формат (MPEG-4, FLV или SWF).

Для того чтобы перейти к добавлению теста необходимо на панели вкладок нажать на кнопку Quizzing.



Если эта вкладка отсутствует, тогда необходимо нажать на кнопку More и в выпадающем меню выбрать пункт Quizzing.

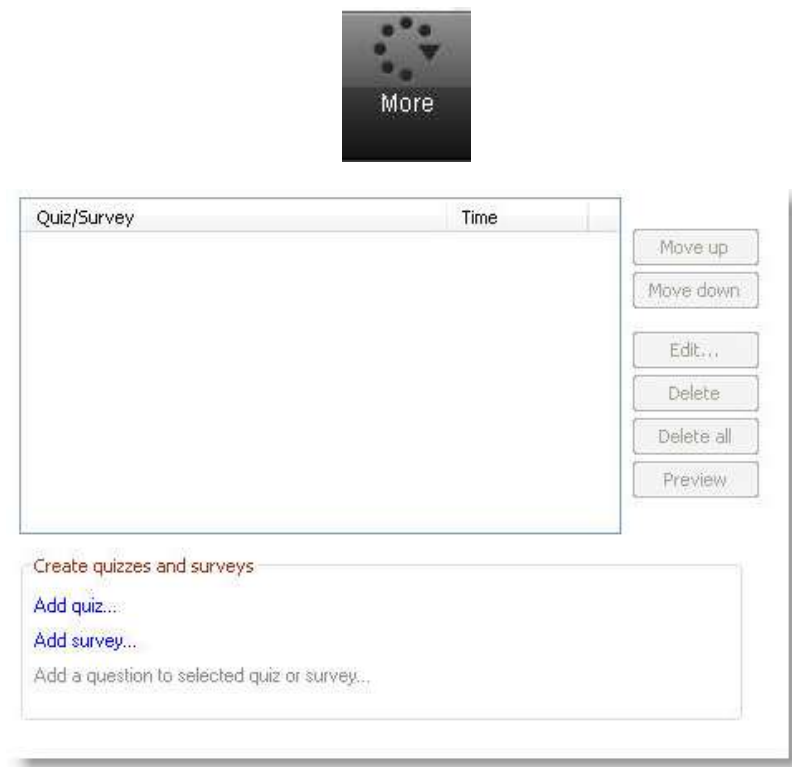


Рис. 14.1. Сервисная панель в режиме добавления тестов.

Для того чтобы добавить тест необходимо для начала на временной шкале установить ползунок на необходимую позицию. Это может быть в принципе любое место на временной шкале, в начале, в конце, в середине нашей записи. Затем в сервисной панели нажимаем на гиперссылку [Add quiz...](#) (Добавить тест)

После этого появится окно Quiz Appearance And Feedback (Викторина Внешний вид и обратная связь).

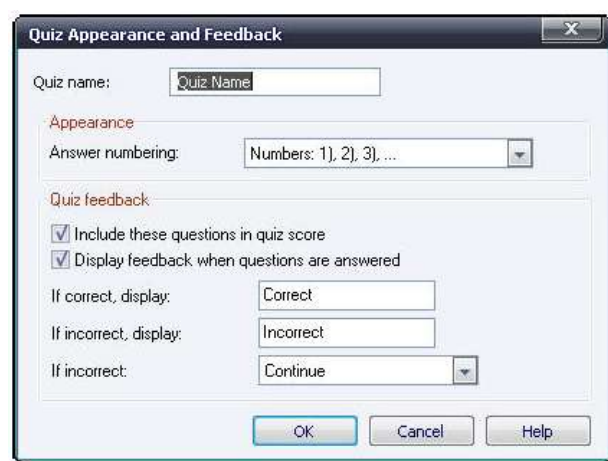


Рис. 14.2. Задаем параметры теста.

В поле Quiz name вводим имя теста. В разделе Appearance (Внешний вид) выбираем нумерацию ответов (Answer numbering). Возможны следующие варианты: Letters – нумерация ответов в виде букв a), b), c), Numbers – нумерация ответов цифрами, No numbering – ответы без нумерации.

В разделе Quiz feedback (Обратная связь в тесте) настраиваем следующие параметры.

Include these questions in quiz score - Включить эти вопросы в тесте для подсчета.

Display feedback when questions are answered – Показать обратную связь когда вопросы пройдены. Если выбрана эта опция, тогда доступны ниже поля для ввода текста.

В поле If correct, display: мы вводим текст, который будет показан, если ответ на вопрос правильный. В поле If incorrect, display: мы вводим текст, который будет показан, если ответ на вопрос не правильный.

Также в пункте If incorrect мы выбираем из выпадающего списка такие варианты. Continue – если ответ на вопрос был не правильным, тогда тест будет продолжаться дальше. Jump to time – в этом случае, если ответ на вопрос не правильный, тогда воспроизведение видео урока начнется с той временной позиции, которая будет указана здесь же, правее в поле.

После сделанных настроек нажимаем кнопку ОК.

Далее появляется окно Question Types, в котором мы можем выбрать тип вопроса, а вернее вариант выбора ответов.

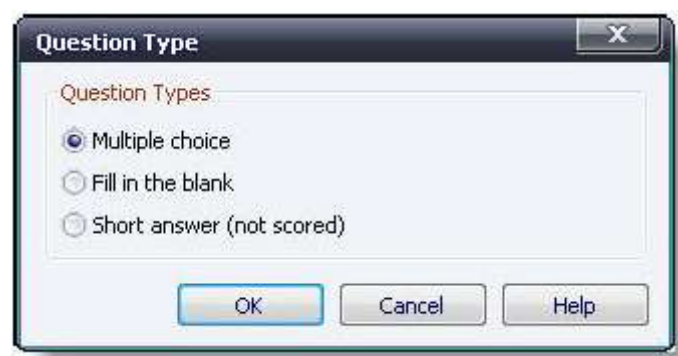


Рис. 14.3. Выбор типа ответа.

Multiply choice – множественный выбор. Если выбрана эта опция, тогда пользователь имеет возможность выбрать правильный ответ из нескольких вариантов.

Fill in the blank – заполнить пустое поле. При выборе этого варианта пользователь должен будет ввести в поле необходимый ответ.

Short answer (not scored) – короткий ответ (подсчет не ведется). В данном случае пользователь может просто ввести необходимый ответ в предоставленное поле ввода.

После выбора необходимого варианта нажимаем кнопку ОК.

Сервисная панель примет другой вид.

The screenshot shows a web-based interface for adding a question to a quiz. At the top right are two buttons: 'Add question' (highlighted with a blue border) and 'Return to manager'. Below these are two input fields: 'Quiz name:' with a text box containing 'Quiz Name', and 'Question:' with a text box containing '<Type a question here>'. Below the 'Question' field is a section titled 'Answers'. Inside this section, there is a dropdown menu labeled 'Select the correct answer' with a green arrow icon. Below the dropdown is a table with one row. The first column of the table contains a checked checkbox, the second column contains the text '<Type an answer choice here>', and the third column contains the word 'Correct'. To the right of the table are five buttons: 'Add', 'Delete', 'Move up', 'Move down', and 'Edit...'. At the bottom of the panel are two buttons: 'Appearance and feedback...' and 'Preview'.

Рис. 14.4. Сервисная панель в режиме добавления вопросов в тест.

В поле Question: вводим вопрос. В раздел Answers мы должны ввести несколько ответов, если на предыдущем шаге мы выбрали пункт Multiply choice (Множественный выбор). Для добавления следующего ответа нажимаем на кнопку Add. Для выбора правильного ответа просто отмечаем его «галочкой». В окне предварительного просмотра вы можете наблюдать, как будет выглядеть визуально ваш тест.

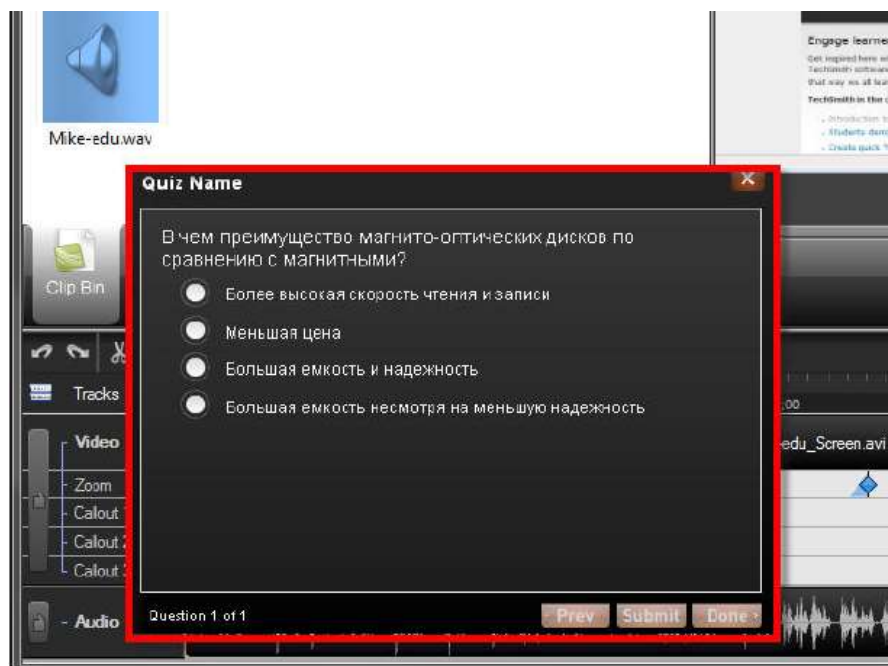


Рис. 14.5. Тест в окне предварительного просмотра.

Для того чтобы добавить следующий вопрос необходимо нажать в сервисной панели сверху кнопку Add question (Добавить вопрос).

Опять появится окошко Question Types, в котором можно выбрать тип ответа на вопрос.

После добавления теста на временной шкале появляется дополнительная дорожка Quiz, на которой отмечена позиция теста в виде розового ромбика.



Рис. 14.6. Ключевой кадр теста на временной шкале.

При воспроизведении экспортированного видео, когда ползунок дойдет до позиции теста, воспроизведение закончится и пользователю необходимо будет ответить на вопросы, выбрав соответствующие ответы либо введя их в текстовое поле. После этого, в зависимости от настроек, воспроизведение будет продолжаться далее либо продолжаться с указанного временного момента.

Если после ввода вопросов нажать на кнопку Preview (Предпросмотр), тогда будет создана HTML-страница с Flash-тестом и открыта в браузере.

Если нажать на кнопку Appearance and feedback (внешний вид и обратная связь), тогда откроется окно Appearance and feedback.

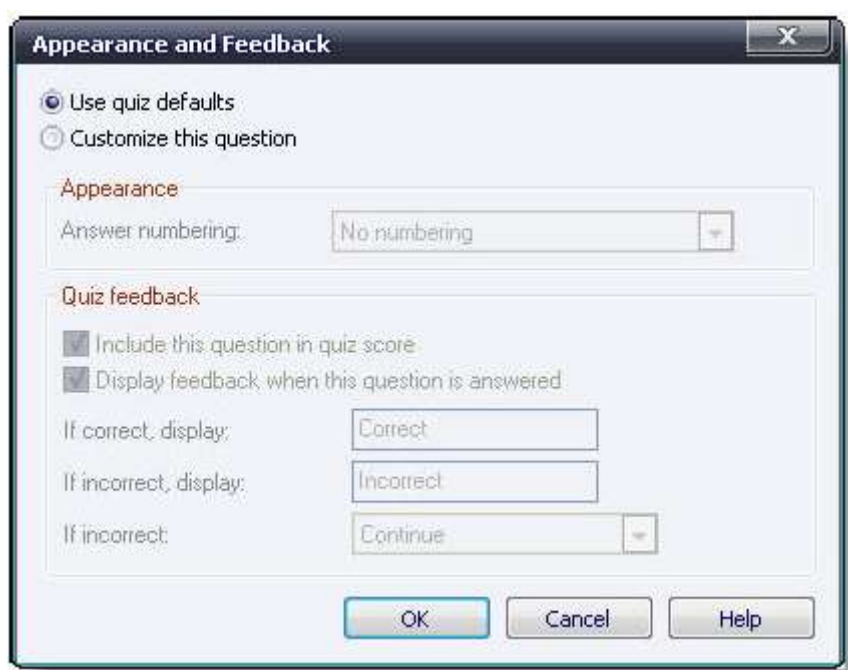


Рис. 14.7. Определяем внешний вид и обратную связь для конкретного вопроса.

В этом окне можно настроить внешний вид и обратную связь для конкретного вопроса.

Если в сервисной панели сверху нажать на кнопку Return to manager, тогда можно вернуться к списку тестов и вопросов.

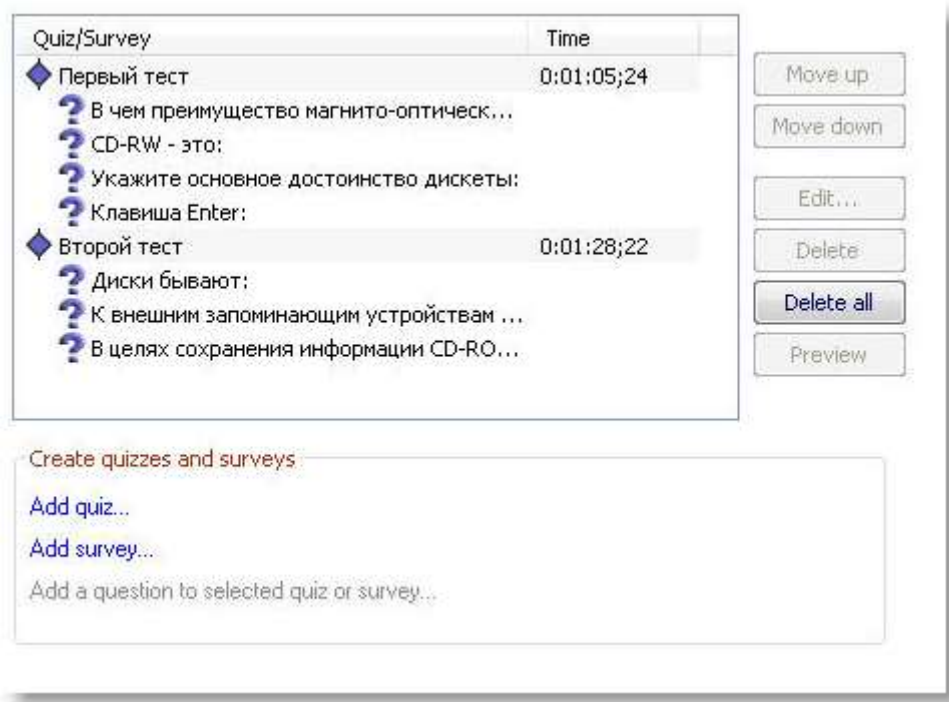


Рис. 14.8. Список тестов в сервисной панели.

Для того чтобы отредактировать настройки конкретного теста или вопроса, нужно в списке выделить название теста или вопроса и нажать на кнопку Edit... справа.

Для того чтобы добавить вопрос к конкретному тесту нужно выделить необходимый тест и нажать ниже на гиперссылку Add a question to selected quiz or survey... (Добавить вопрос к выделенному тесту или исследованию).

Также в сервисной панели имеется еще один тип теста - это Survey (Обследование). В отличие от Quiz (Викторина), тип теста, который служит для проверки усвоенных знаний, Survey служит для сбора информации о пользователе: e-mail адрес, ФИО и т.п.

Для добавления этого типа теста в сервисной панели нажимаем на гиперссылку Add survey.