

ОГЛАВЛЕНИЕ

Введение	4
Лабораторная работа 1. Верстка веб-страницы с использованием таблиц стилей	5
Лабораторная работа 2. Создание скриптов на языке Java Script	17
Лабораторная работа 3. Работа со звуком в программе LMMS	33
Лабораторная работа 4. Flash-анимация	47
Лабораторная работа 5. Работа в видеоредакторе для нелинейного монтажа Lightworks	53
Лабораторная работа 6. Настройка служб Windows Media	63
Заключение	74
Список использованных источников	75

Введение

Современный человек применяет мультимедиа технологии во всех сферах жизни и деятельности: деловой сфере, обучении и сфере развлечений.

Цель данного лабораторного практикума «Мультимедиа технологии» - формирование умений и навыков работы с мультимедиа технологиями, использование технологий мультимедиа для создания, обработки и компоновки стандартных форматов файлов текстовой, графической, звуковой и видеоинформации.

Лабораторный практикум содержит лабораторные работы по дисциплине «Мультимедиа технологии». Каждая лабораторная работа посвящена специфике обработки разных типов данных и включает задания для самостоятельной работы студентов.

Материал лабораторного практикума успешно апробирован в учебном процессе со студентами специальности 201203 «Информационные системы и технологии», и студентами направлений подготовки 230400 «Информационные системы» и 230700 «Прикладная информатика».

Лабораторная работа 1. Верстка веб-страницы с использованием таблиц стилей

Задание 1.1. Создать главную веб-страницу для сайта кафедры информатики. Макет страницы представлен на рисунке 1.1.

Порядок работы:

1. Запустить Блокнот (Notepad).

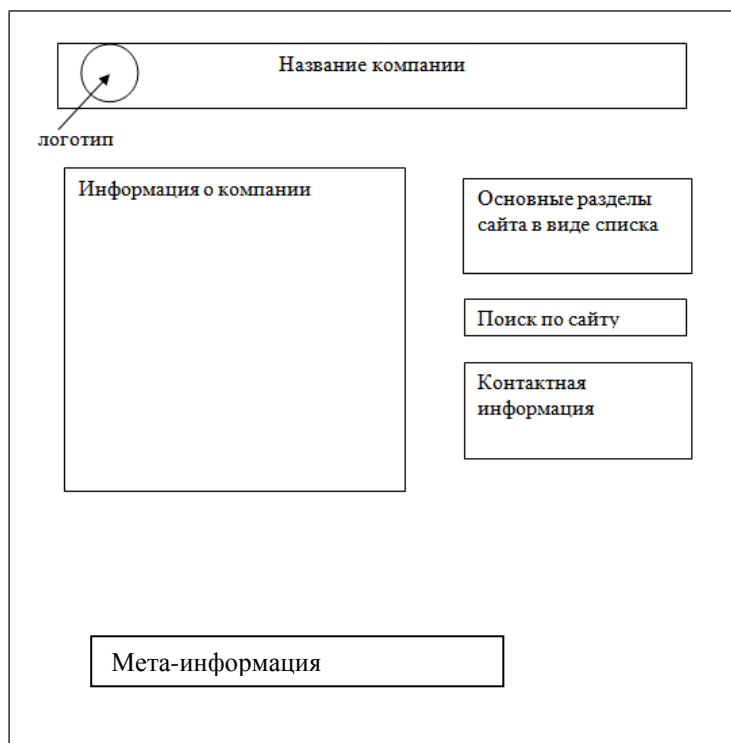


Рисунок 1.1 Макет главной веб-страницы сайта

2. Набрать HTML-код для главной страницы сайта. Поскольку оформление будет выполняться с использованием CSS, то необходимо в коде учесть состав и размещение объектов. Для формулирования CSS-правил необходимо задать идентификаторы или классы для отдельных элементов:

- для заголовка – title;
- для основной информации – content;
- для перечня разделов – sections;
- для поиска – search;
- для мета-информации – meta;

- для блока контактной информации – contacts.
- для дополнительного форматирования абзацев в блоках content и contacts – text.

Исходный код HTML-документа:

```
<HTML>
<HEAD>
<TITLE>
О кафедре
</TITLE>
<link rel="stylesheet" href="style.css">
</HEAD>
<BODY>
<div id="title"><h1>Кафедра информатики</h1></div>
<div id="content">
<h2>О кафедре</h2>
<div class="text">
<p>Кафедра информатики рада приветствовать Вас!</p>
<p>Кафедра информатики – это самая первая кафедра факультета
прикладной информатики</p>
<p>В 2015 г. кафедре информатики исполняется 45 лет!</p>
</div>
</div>
<div id="sections">
<ul>
<li><a href="predmety">Дисциплины кафедры</a>
<li><a href="rabota">Учебно-методическая и научная работа</a>
<li><a href="vacans">Вакансии</a>
<li><a href="photo">Фотогалерея</a>
</ul>
<form id="search" action="search/" method="get">
<p>Поиск по сайту
```

```

        <p><input type="text"> <button type="submit">Искать</button>
    </form>
</div>
<div id="contacts">
<h2>Контакты</h2>
<div class="text">
<br>кафедра информатики
<br>Наш адрес: 614000, Пермь, ул. Луначарского д.3
<br>Телефон:+7 (342) 2912-811,
<br>E-mail:informatics@pgsha.ru
</div>
</div>
</div>
<div id="meta">
    <p>Сайт изготовлен на кафедре информатики, &copy; 2014
    </div>
</BODY>
</HTML>

```

3. Сохранить в файле с именем: D:\ФИО\glav.html

4. Запустить созданный файл и проверить его работу (рис.1.2)

Кафедра информатики

О кафедре

Кафедра информатики рада приветствовать Вас!

Кафедра информатики – это самая первая кафедра факультета прикладной информатики

В 2015 г. кафедре информатики исполняется 45 лет!

- [Дисциплины кафедры](#)
- [Учебно-методическая и научная работа](#)
- [Вакансии](#)
- [Фотогалерея](#)

Поиск по сайту

Рисунок 1.2 Результат просмотра веб-страницы

Задание 1.2. Описать правила, определяющие расположение элементов. Заголовок, основную информацию и мета-информацию разместить в нормальном потоке со статическим позиционированием. Остальные блоки разместить справа с помощью абсолютного позиционирования и точных координат.

Порядок работы:

1. Запустить Блокнот (Notepad).
2. Определить размещение для элемента body:

```
body{  
    position: relative; width:900px;  
    margin: 0 40px; padding:20px;  
}
```

Все остальные элементы будут располагаться внутри блока body.

3. Сохранить в файле D:\ФИО\style.css. В дальнейшем сохранение выполнять после каждого пункта.

4. Описать правило размещения заголовка. Заголовок расположен первым и занимает свое место в верхней части страницы. Для него необходимо задать высоту:

```
#title{  
    margin-bottom:0px;  
    height:80px;  
}
```

5. Описать правило размещения блока основной информации, который должен следовать сразу после заголовка. Необходимо задать правую границу так, чтобы потом можно было расположить остальные элементы.

```
#content {  
    margin:10px 350px 10px 0;  
}
```

6. Определить правила расположения на странице остальных элементов:

```

#sections {
    position:absolute;
    top:140px; right:0;
    width:250px;
    padding-right:70px;
}
#search {
    position:absolute; top:170px; right:0;
    width:280px;
    margin-right:20px;
    padding:10px;
}
#contacts{
    position:absolute; top:450px; right:0;
    width:300px;
    margin-right:20px;
}

```

7. Определить правила размещения заголовка второго уровня:

```

h2{
    margin:0;
    height:40px;
    line-height:40px;
    padding-left:10px;
}

```

8. Определить правила размещения мета-информации. Хотя в HTML-документе она описана последней, с учетом того, что предшествующим ей элементам установлено абсолютное позиционирование и, следовательно, они извлекаются из нормального потока, мета-информация будет выведена сразу же после последнего статического блока — блока основной информации.

```

#meta{

```

```
padding-top:10px;
margin:0
}
```

9. Запустить файл glav.html и посмотреть результат. Обратить внимание на блок основных разделов. В HTML-документе он был описан как маркированный список (рис. 1.3).

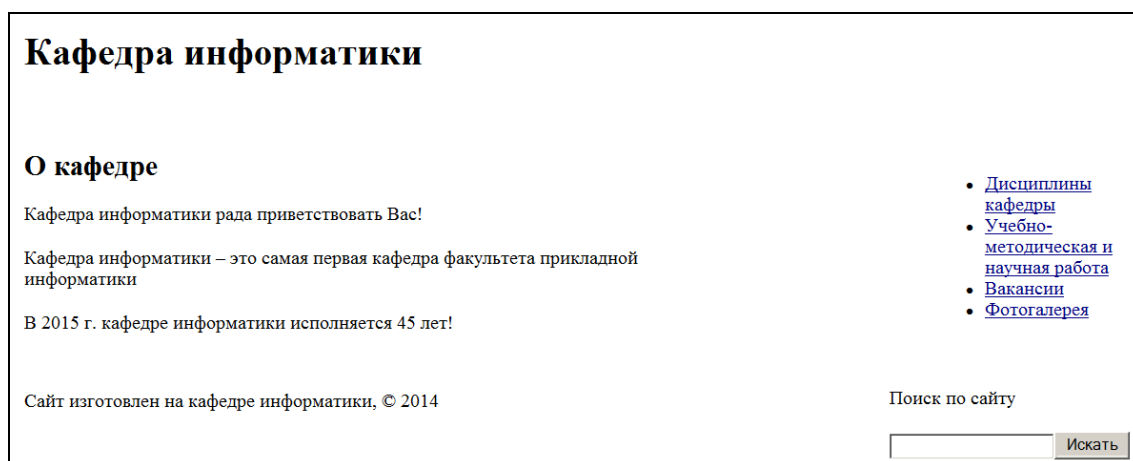


Рисунок 1.3 Результат просмотра веб-страницы

10. Отключить маркеры и разместить блок основных разделов справа на странице с помощью абсолютного позиционирования и точных координат. С помощью объявления list-style:none отключить маркеры. Установить отступы по 20 пикселей сверху и снизу списка. Элементу li задать границы в 10 пикселей, чтобы увеличить межстрочный интервал между элементами списка (рис. 1.4).

```
ul {
    list-style: none;
    margin:20px 0; padding:0;
}
li {
    margin: 10px 0; padding: 0;
}
```


Дисциплины кафедры
Учебно-методическая
и научная работа
Вакансии
Фотогалерея

Рисунок 1.4 Отформатированный блок основных разделов

11. Сохранить изменения в файле D:\ФИО\style.css.

12. Запустить файл glav.html и посмотреть результат. Если необходимо, выполнить корректировку.

Задание 1.3. Описать правила, определяющие шрифт и цвет элементов.

Порядок работы:

1. Запустить Блокнот (Notepad).

2. Открыть для редактирования файл D:\ФИО\style.css.

3. Определить фон страницы:

```
html{  
    background-color:#505050;  
}
```

4. Установить единый для всего документа фон, шрифт и цвет текста.

При необходимости можно будет изменить эти свойства для отдельных элементов.

```
body{  
    font:2ex/20px Tamoha, Arial;  
    background-color:white;  
    color:#505050;  
}
```

5. Добавить логотип компании рядом с заголовком, а фон заголовка задать близким к цветовой гамме логотипа оттенком.

```
#title {  
    background: url(logo1.png) #1767ab no-repeat;
```

```
}
```

6. Шрифт заголовка увеличить в 2,5 раза относительно заданного шрифта в правиле к блоку body.

```
h1{  
    font:250% Tamoha, Arial;  
}
```

7. Для оформления подзаголовков в блоках основной информации и контактной информации использовать градиент размером 10x50 и скопировать его по ширине. Кроме этого, изменить цвет самого текста.

```
h2 {  
    background:url(title.png) repeat-x;  
}  
h1, h2{  
    color:#fff79f;  
}
```

8. Запустить файл glav.html и посмотреть результат. Если необходимо, выполнить коррективку (рис.1.5).

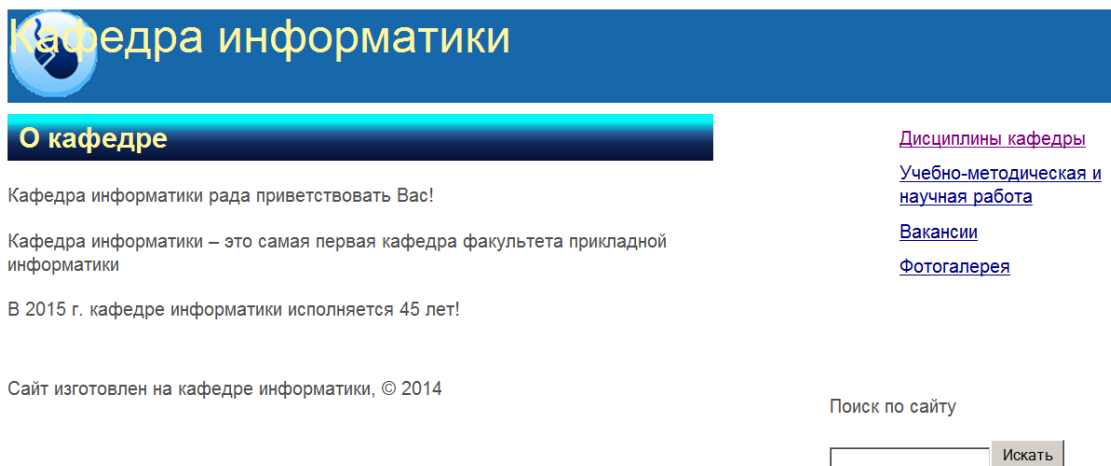


Рисунок 1.5 Результат просмотра веб-страницы

9. Ограничить блоки content, search и contacts тонкими рамками.

```
#content, #search, #contacts {  
    border-style: solid;  
    border-width:thin;
```

```
border-color:#505050;
}
```

10. Запустить файл glav.html и посмотреть результат (рис.1.6).

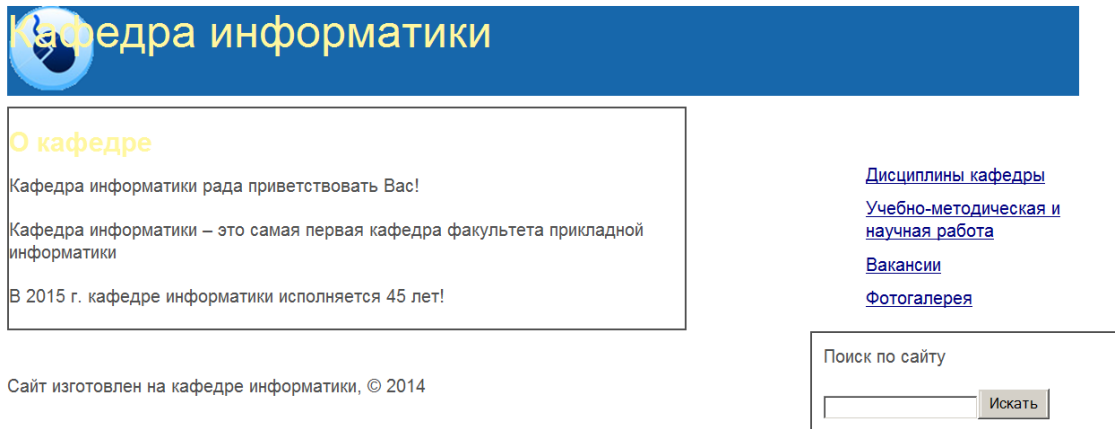


Рисунок 1.6 Блоки content, search и contacts ограничены тонкими рамками

11. Отредактировать цвет и толщину блока основных разделов.

```
a {
color:#1767ab;
text-decoration: none;
font:140% Tamoha, Arial;
}
```

Объявление `text-decoration: none` позволяет избавиться от подчеркивания элементов раздела, которые являются гиперссылками.

12. Запустить файл glav.html и посмотреть результат (рис.1.7).

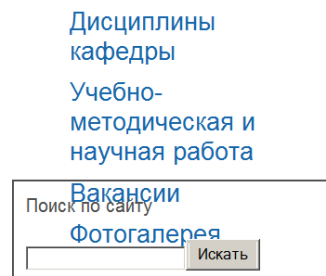


Рисунок 1.7 Блоки content и search пересекаются

13. Блок search необходимо подвинуть вниз, чтобы избавиться от пересечения с блоком content. Для этого достаточно изменить значение верхней границы: прежнее значение 170px заменить на значение 240px.

14. Запустить файл glav.html и посмотреть результат. Если необходимо, выполнить корректировку.

15. Добавить к блоку основных разделов эффект изменения цвета при наведении мыши:

```
a:hover {  
    color:#4ee335;  
}
```

16. Для блока мета-информация выбрать более светлый оттенок, нежели имеет основной текст, и уменьшить размер шрифта. Добавить небольшую картинку, которая будет входить в блок мета-информации.

```
#meta{  
    font:90% Tahoma, Arial;  
    color:#8e8e8e;  
}  
#meta a{  
    text-decoration:none;  
    color:#8e8e8e;  
}  
#meta{  
    background:url(z.png) right bottom no-repeat;  
    height:500px;  
}
```

17. Запустить файл glav.html и посмотреть результат (рис.1.8).

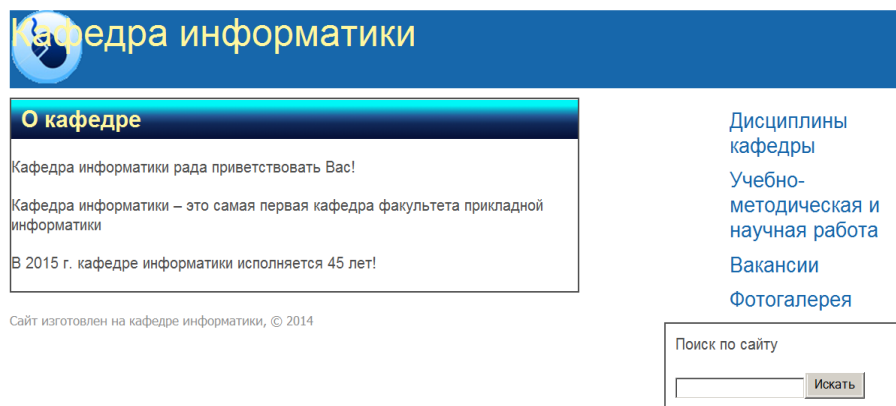


Рисунок 1.8 Применение правил для блока мета-информации

18. Установить фиксированную высоту для блока мета, чтобы оставить немного свободного места внизу веб-страницы. Блок мета следует последним и расположен в нормальном потоке, а значит, увеличение его высоты не нарушит расположение остальных элементов. Добавить небольшую картинку в конце страницы. Она будет входить в блок мета-информации.

```
#meta{
    background:url(z.png) right bottom no-repeat;
    height:500px;
}
```

19. Сдвинуть текст заголовка, чтобы он не закрывал логотип и был расположен у верхнего края шапки.

```
h1{
    padding-left:90px;
    margin:0;height:80px;
    line-height:80px;
}
```

20. Запустить файл glav.html и посмотреть результат (рис.1.9).

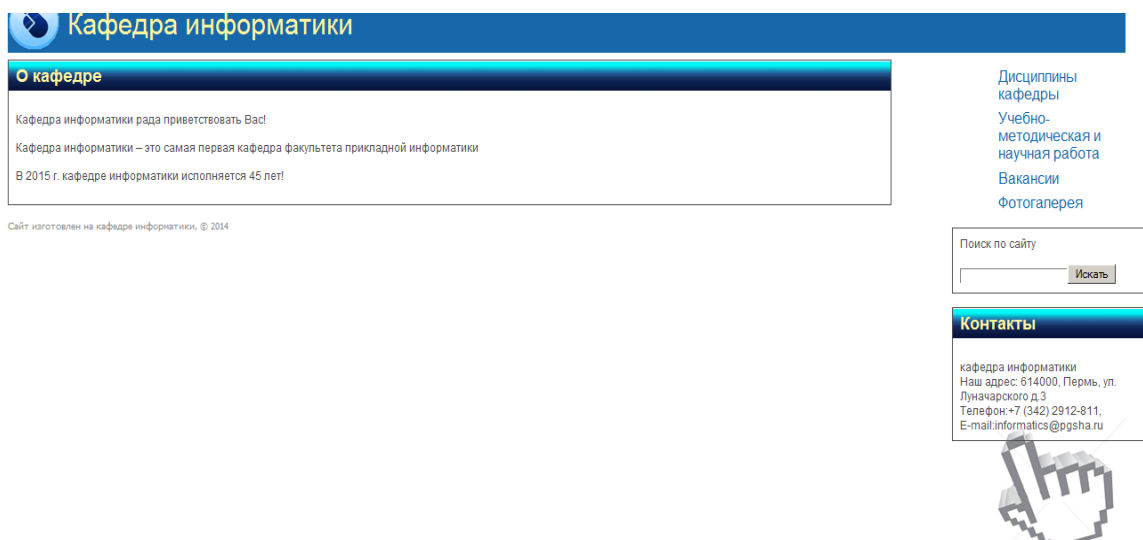


Рисунок 1.9 Результат просмотра готовой веб-страницы

Самостоятельная работа 1.1. Создать HTML-документ табличной структуры (табл.1.1). Применить к созданному документу таблицу стилей. Обязательные элементы: заголовки трех уровней, различный цвет шрифта на разных уровнях, цвет фона блоков таблицы. Все остальные параметры выбрать самостоятельно.

Таблица 1.1 – Структура HTML-страницы

Логотип компании	Название компании	Баннер												
Список основных разделов Сайта	Общая информация о компании	<div>Руководство компании</div> <table><tr><th>ФИО</th><th>Должность</th><th>Телефон</th></tr><tr><td>Иванов</td><td>Дир к ор</td><td>1234</td></tr><tr><td>Петров</td><td>Зам.директора</td><td>5678</td></tr><tr><td>Сидоров</td><td>Гл. бухгалтер</td><td>8765</td></tr></table>	ФИО	Должность	Телефон	Иванов	Дир к ор	1234	Петров	Зам.директора	5678	Сидоров	Гл. бухгалтер	8765
	ФИО	Должность	Телефон											
Иванов	Дир к ор	1234												
Петров	Зам.директора	5678												
Сидоров	Гл. бухгалтер	8765												
	Продукция компании	Адрес												
Счетчик сайта	Часы аналоговые	Сведения о разработчике												

Лабораторная работа 2. Создание скриптов на языке Java Script

Задание 2.1. Набрать и проверить работу скрипта. Результат показан на рисунке 2.1.

```
<SCRIPT LANGUAGE="JavaScript">  
//Выводится дата и время посещения страницы  
Now = new Date();  
document.write("Сегодня " + Now.getDate()+  
"-" + (Now.getMonth()+1) + "-" + Now.getFullYear()  
+ ". Вы зашли на Web-страницу ровно в: " + Now.getHours() +  
":" + Now.getMinutes() + " и " + Now.getSeconds() +  
" секунд.")  
</SCRIPT>
```

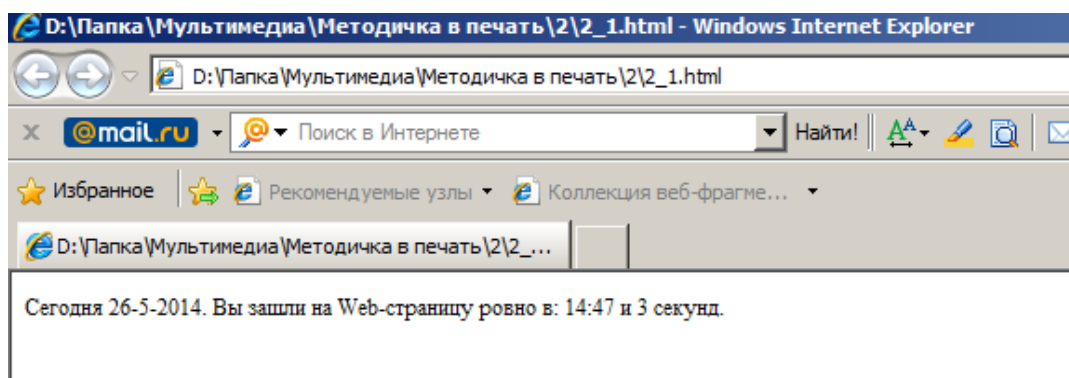


Рисунок 2.1 Результат работы скрипта 2.1

Задание 2.2. Набрать и проверить работу скрипта, который выводит на Web-странице дату, разделенную косой чертой и приветствие. Результат работы скрипта представлен на рисунке 2.2.

```
<SCRIPT LANGUAGE="JavaScript">  
// Автор скрипта: ФИО  
Now = new Date();  
var mpo = Now.getMonth();  
var mpo1 = mpo + 1;
```

```

document.write("<FONT COLOR='green'>Доброго времени суток!
Сегодня у нас:</FONT>"
+ Now.getDate() + "/"
+ mpo1 + "/"
+ Now.getFullYear() + ".");
</SCRIPT>

```

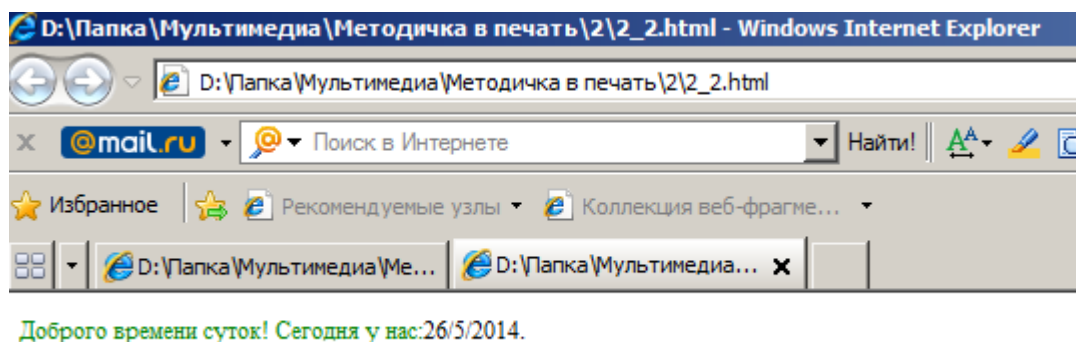


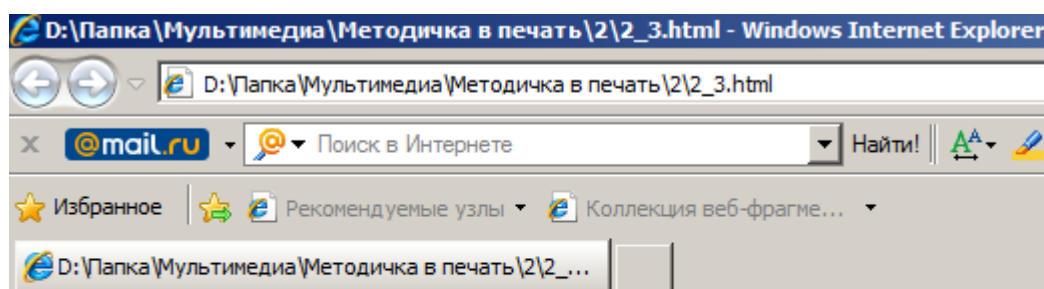
Рисунок 2.2 Результат работы скрипта 2.2

Задание 2.3. Набрать и проверить работу скрипта. Результат работы скрипта показан на рисунке 2.3. При наведении курсора мыши на ссылку строка состояния в окне браузера изменится на «Почтовая служба».

```

<A HREF="http://www.mail.ru"
onMouseOver="window.status='Почтовая служба';
return true">Ссылка</A>

```



[Ссылка](#)



Рисунок 2.3 Результат работы скрипта 2.3

Задание 2.4. Набрать и проверить работу скрипта. Результат показан на рисунке 2.4. При наведении указателя мыши меняется цвет фона.

```
<A HREF="http://www.mail.ru"  
onMouseOver="document.bgColor='pink';  
return true">Щелкните здесь</A>
```

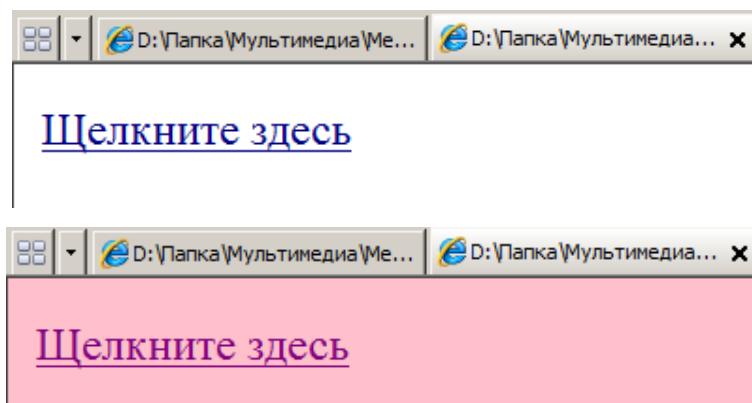


Рисунок 2.4 Результат работы скрипта 2.4

Задание 2.5. Набрать и проверить работу скрипта. В данном сценарии объединены события из заданий 2.3 и 2.4. Для их разделения в скрипте используется запятая. При наведении курсора мыши на ссылку изменится строка состояния в окне браузера и цвет фона.

```
<A HREF="http://www.mail.ru"  
onMouseOver="document.bgColor='pink',  
onMouseOver=window.status='Посетите MAIL.RU';  
return true">Щелкните здесь</A>
```

Самостоятельная работа 2.1. Воспользуйтесь методом `alert()` (предупредить), который выводит небольшое диалоговое окно с текстом и кнопкой ОК. Окно предупреждения должно появляться при наведении курсора на ссылку. Формат команды:

```
alert('выводимый в окне текст')
```

Результат работы сценария показан на рисунке 2.5.

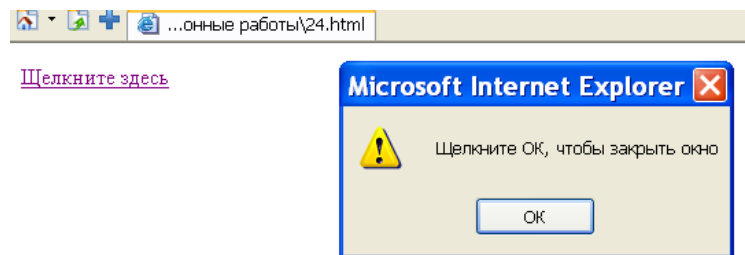


Рисунок 2.5 Результат выполнения самостоятельной работы 1

Задание 2.6. Набрать и проверить работу скрипта. Результат показан на рисунке 2.6. Команда onClick запускает событие щелчком по ссылке.

```
<A HREF="http://www.mail.ru"  
onClick="alert('Посмотреть почту!');">  
Щелкните здесь</A>
```

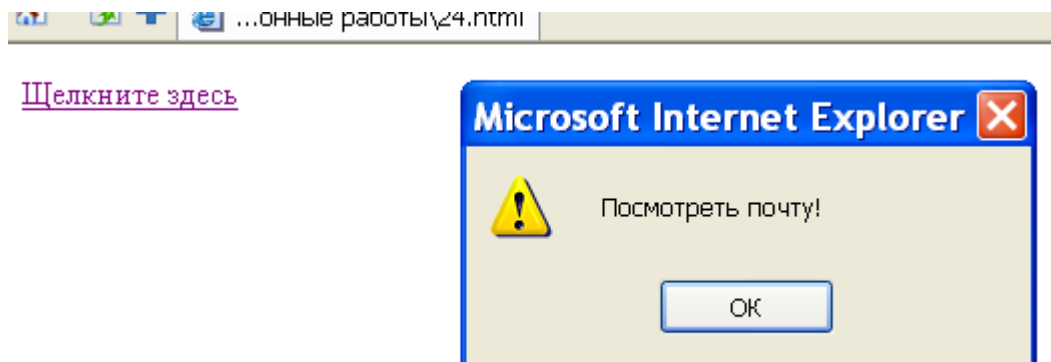


Рисунок 2.6 При щелчке на ссылке появляется окно с сообщением "Посмотреть почту!"

Задание 2.7. Набрать и проверить работу скрипта. Результат показан на рисунке 2.7. При использовании этого сценария выводится текстовое поле, при щелчке по которому в строке состояния выводится строка 'Текст в строке состояния'. Используется команда onFocus – это обработчик событий, который вызывает действие, когда пользователь "фокусируется" на элементе страницы. Он будет работать для элементов формы: флажков, текстовых полей, текстовых областей и др.

```
<FORM>  
<INPUT TYPE="text" SIZE="30"
```

```
onFocus="window.status='Вы щелкнули в текстовом поле';">
</FORM>
```

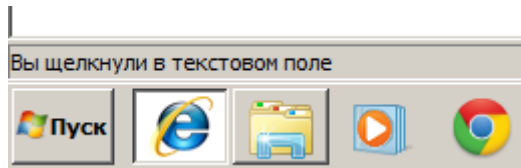


Рисунок 2.7 Результат работы скрипта 2.7

Задание 2.8. Набрать и проверить работу скрипта с использованием команды `onChange`. Команда `onChange` - позволяет сообщить пользователю о том, что он изменил свой ответ. Сообщение должно быть выдано в строке состояния. Результат показан на рисунке 2.8.

```
<FORM>
<INPUT TYPE="text" SIZE="60" VALUE="Впишите свое имя и
щелкните вне текстового поля"
onChange="window.status='Вы изменили текст в текстовом поле';">
</FORM>
```

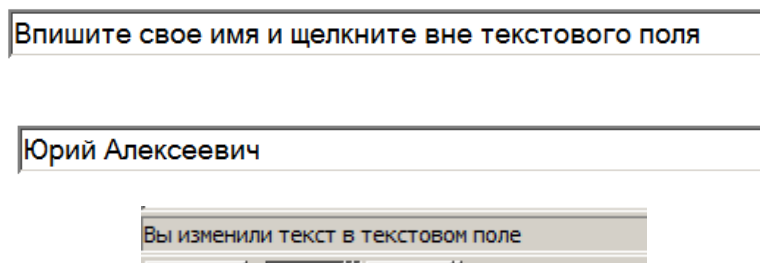


Рисунок 2.8 Результаты работы скрипта 2.8

Задание 2.9. Набрать и проверить работу скрипта с использованием команды `onSubmit`. Команда `onSubmit` позволяет вызвать какое-либо действие при нажатии кнопки Submit. Результаты работы показаны на рисунке 2.9.

Предварительно нужно создать файл `receive.html`, содержимое которого:

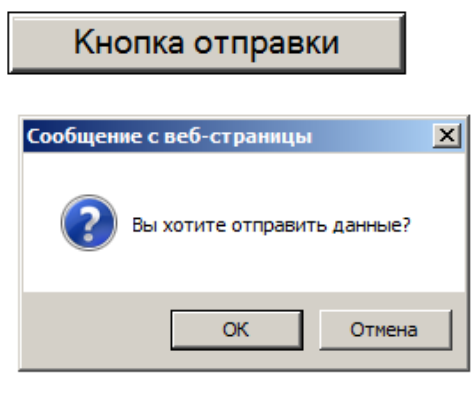
```
<HTML><BODY>Данные приняты!</BODY></HTML>
```

Содержимое скрипта:

```

<FORM NAME=f ACTION="receive.html"
onSubmit="return confirm('Вы хотите отправить данные?')">
<INPUT TYPE=submit VALUE="Кнопка отправки" NAME=s>
</FORM>

```



Данные приняты!

Рисунок 2.9 Результаты работы скрипта 2.9

Задание 2.10. Создать форму, которая будет взаимодействовать с пользователем. Форма должна иметь три элемента: поле ввода с просьбой ввести имя; два поля для флажков с вопросом о том, что предпочитает пользователь — лето или зиму; кнопку отправки данных (submit).

С каждым элементом должно произойти следующее.

- При вводе имени в строке состояния должны появиться слова: Введите свое имя.
- Два поля с флажками должны отослать в строку состояния слова: Вы выбрали... и выбор пользователя.
- При нажатии на кнопку отправки должно появиться окно с благодарностью пользователю за участие в опросе.

Результаты работы показаны на рисунке 2.10.

```
<FORM action="">
```

```
Name: <INPUT TYPE="text" SIZE="30"
```

```
onFocus="window.status='Введите свое имя';">
```

Что вы предпочитаете:

```

<INPUT TYPE="checkbox"
  onClick="window.status='Вы выбрали лето';">
  лето
<INPUT TYPE="checkbox"
  onClick="window.status='Вы выбрали зиму';">
  зима
<INPUT TYPE="submit"
  onClick="alert('Спасибо за участие в опросе');">
</FORM>

```

Name: Что вы предпочитаете: ☐ лето ☐ зима

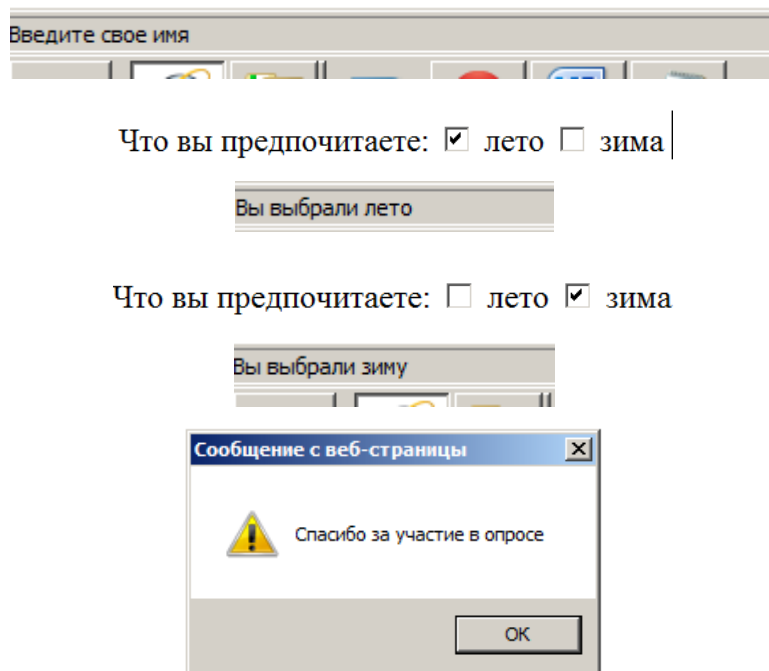


Рисунок 2.10 Результаты работы скрипта 2.10

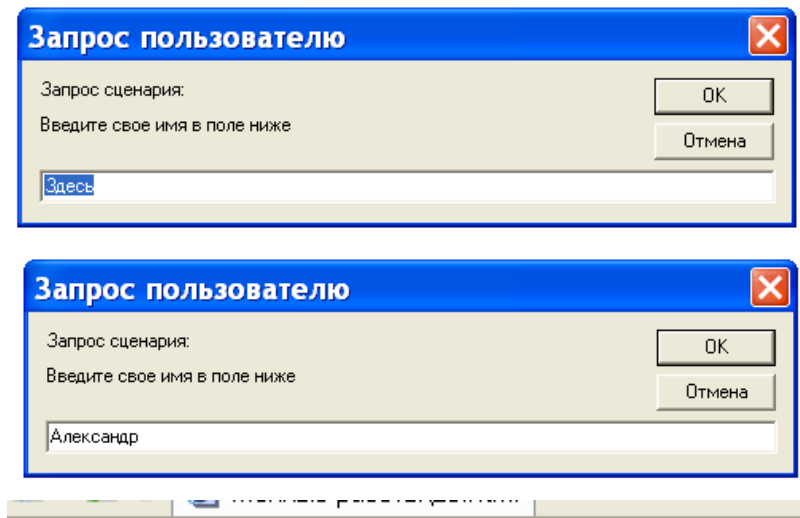
Задание 2.11. Набрать и проверить работу скрипта. Результат работы скрипта показан на рисунке 2.11. Текст в скобках должен располагаться в одной строке.

```

<SCRIPT type="text/javascript">
/* Этот скрипт предназначен для получения информации
от пользователя и вывода ее на странице */
var user_name = prompt ("Введите свое имя в поле ниже", "Здесь");

```

```
document.write("Привет, " + user_name + "! Добро пожаловать на мою  
страницу!");  
</SCRIPT>
```



Привет, Александр! Добро пожаловать на мою страницу!

Рисунок 2.11 Результаты работы скрипта 2.11

Задание 2.12. Набрать и проверить работу скрипта. Результат работы скрипта показан на рисунке 2.12.

```
<SCRIPT type="text/javascript">  
var name = prompt("Пожалуйста, напишите свое имя", "");  
var d = new Date();  
var y = d.getFullYear();  
var m = d.getMonth() + 1;  
var d = d.getDate();  
var t = m + '/' + d + '/' + y + ' ';  
document.write("<TITLE>")  
document.write ("Привет "+name+ ". Сегодня " +t+ ". Спасибо, что  
зашли.");  
document.write("</TITLE>")  
</SCRIPT>
```

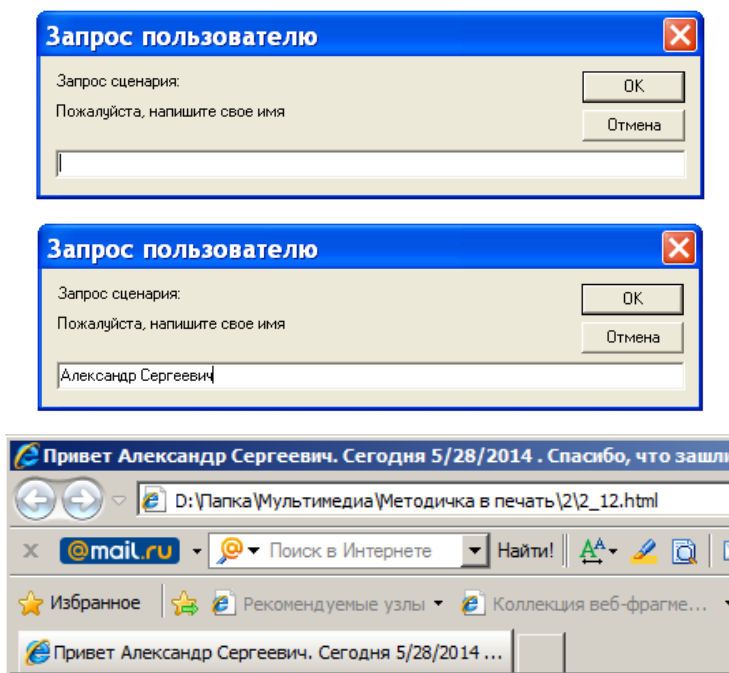


Рисунок 2.12 Результаты работы скрипта 2.12

Задание 2.13. Набрать и проверить работу скрипта. Результат работы скрипта показан на рисунке 2.13. Текст в скобках должен располагаться в одной строке.

```
<SCRIPT LANGUAGE="javascript">
var an = navigator.appName;
var av = navigator.appVersion;
var acn = navigator.appCodeName;
var ua = navigator.userAgent;
document.write("Вы пользуетесь <B>" +an+ "</B>, версия " +av+
".<BR>Кодовое название " +acn+ ", заголовок " +ua+ ".");
</SCRIPT>
```

Вы пользуетесь **Microsoft Internet Explorer**, версия 4.0 (compatible; MSIE 8.0; Windows NT 6.1; WOW64; Trident/4.0; MRSPUTNIK 2, 4, 1, 162; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; InfoPath.3).
Кодовое название Mozilla, заголовок Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; WOW64; Trident/4.0; MRSPUTNIK 2, 4, 1, 162; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; InfoPath.3).

Рисунок 2.13 Результат работы скрипта 2.13

Задание 2.14. Набрать и проверить работу скрипта. Результат работы скрипта показан на рисунке 2.14. Текст в скобках должен располагаться в одной строке.

```
<SCRIPT LANGUAGE="javascript">
var bgc = document.bgColor;
var fgc = document.fgColor;
var lc = document.linkColor;
var al = document.alinkColor;
var vlc = document.vlinkColor;
var url = document.location;
var ref = document.referrer;
var t = document.title;
var lm = document.lastModified;
document.write("Цвет фона этой страницы <B>"
+bgc+ "</B>.")
document.write("<BR>Цвет текста этой страницы <B>" +fgc+ "</B>.")
document.write("<BR>Цвет ссылок этой страницы <B>" +lc+ "</B>.")
document.write("<BR>Цвет активной ссылки этой страницы
<B>" +al+ "</B>.")
document.write("<BR>Цвет посещенной ссылки этой страницы
<B>" +vlc+ "</B>.")
document.write("<BR>URL этой страницы <B>" +url+ "</B>.")
document.write("<BR>До этого вы были на странице <B>" + ref+
"</B>.")
document.write("<BR>Заголовок этой страницы (TITLE) <B>" +t+
"</B>.")
document.write("<BR>Последние изменения в документ внесены: <B>"
+lm+ "</B>.")
</SCRIPT>
```


Цвет фона этой страницы #ffff.
Цвет текста этой страницы #000000.
Цвет ссылок этой страницы #0000ff.
Цвет активной ссылки этой страницы #0000ff.
Цвет посещенной ссылки этой страницы #800080.
URL этой страницы file:///D:/Папка/Мультимедиа/Методичка%20в%20печать/2/2_14.html.
До этого вы были на странице .
Заголовок этой страницы (TITLE) .
Последние изменения в документ внесены: 05/28/2014 14:53:44.

Рисунок 2.14 Результат работы скрипта 2.14

Задание 2.15. Набрать и проверить работу скрипта. Результат работы скрипта показан на рисунке 2.15.

```
<SCRIPT LANGUAGE="javascript">  
var h = history.length;  
document.write("До этой страницы вы посетили" +h+ " страниц.")  
</SCRIPT>
```

До этой страницы вы посетили0 страниц.

Рисунок 2.15 Результат работы скрипта 2.15

Задание 2.16. Набрать и проверить работу скрипта, который состоит из двух частей: описания функции и команды onLoad, которая запускает функцию.

```
<HTML>  
<HEAD>  
<SCRIPT LANGUAGE="javascript">  
function dateinbar()  
{  
var d = new Date();  
var y = d.getFullYear();  
var da = d.getDate();  
var m = d.getMonth() + 1;  
var t = da + '/' + m + '/' + y;  
defaultStatus = "Вы прибыли на страницу " + t + ".";
```

```

}
</SCRIPT>
</HEAD>
<BODY BGCOLOR="" onLoad="dateinbar()">
тело документа
</body>
</HTML>

```

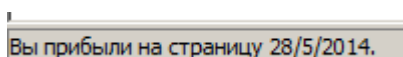


Рисунок 2.16. Результат работы выводится в строке состояния

Самостоятельная работа 2.2. Необходимо создать функцию, которая создает два запроса (prompt). Один запрос должен следовать за другим с новой строки. Первый запрос предлагает пользователю ввести свое имя, второй — фамилию. Затем та же функция должна вызвать окно сообщения (alert) с текстом:

Привет, "имя фамилия", добро пожаловать на " адрес страницы ", мою замечательную страницу!

Создать и проверить работу скрипта, результат работы которого показан на рисунке 2.17.

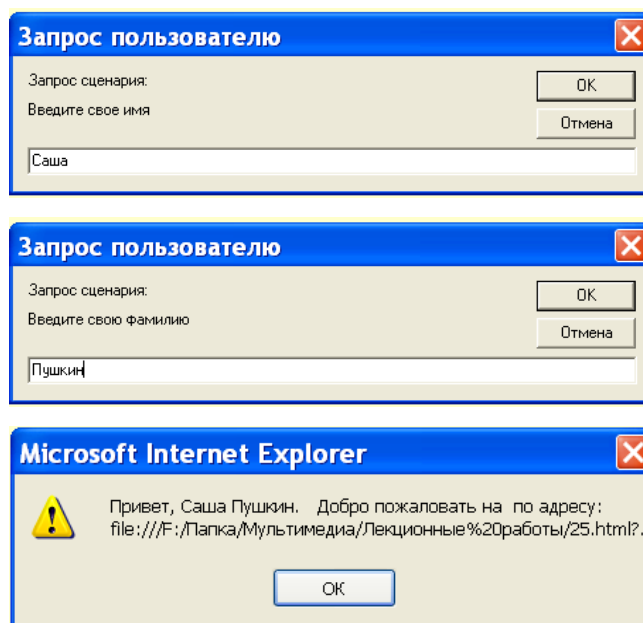
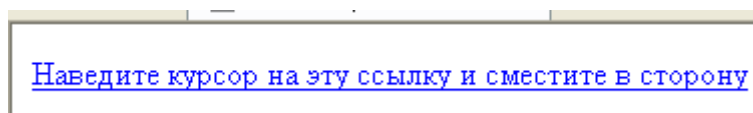


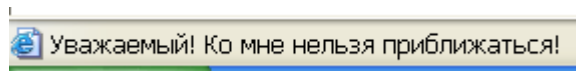
Рисунок 2.17 Результат выполнения самостоятельной работы 2

Задание 2.17. Набрать и проверить работу скрипта с использованием команды последствия. Результат работы показан на рисунке 2.18.

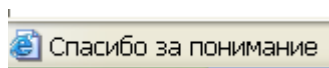
```
<HTML>
<HEAD>
</HEAD>
<BODY onUnload="alert('Уже уходите?')">
  <A HREF="index.htm" onMouseOver="window.status='Уважаемый! Ко
мне нельзя приближаться!';
return true"
onMouseOut="window.status='Спасибо за понимание'; return true">
  Наведите курсор на эту ссылку и сместите в сторону</A>
</body>
</HTML>
```



Результат загрузки сценария



Содержимое строки состояния при наведении курсора



Содержимое строки состояния при неактивной ссылке

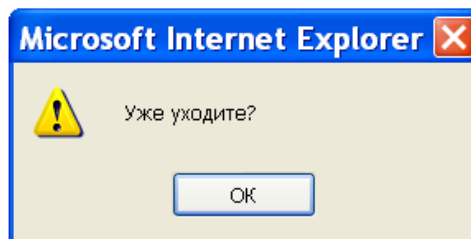


Рисунок 2.18 Сообщение, выдаваемое при попытке уйти со страницы

Самостоятельная работа 2.3. В этом задании нужно использовать функции onUnload, onMouseOver, и onMouseOut. Необходимо сделать следующее:

- Создать страницу с гипертекстовой ссылкой.

- Когда курсор указывает на ссылку, в строке состояния должны появляться слова: "Привет, пользователь 'название браузера'!".
- Когда курсор уходит со ссылки, в строке состояния должен появляться текст: "Не скучаете у нас на 'URL страницы'?"
- Если щелкнуть на ссылке, должно появиться окно со словами: "Уже уходите? Сейчас всего только 'текущее время'";
- Время должно определяться с помощью функции.
- Для вывода окна сообщения использовать команду onUnload.
- Не пользоваться командой onClick.

Результат работы программы показан на рисунке 2.19.

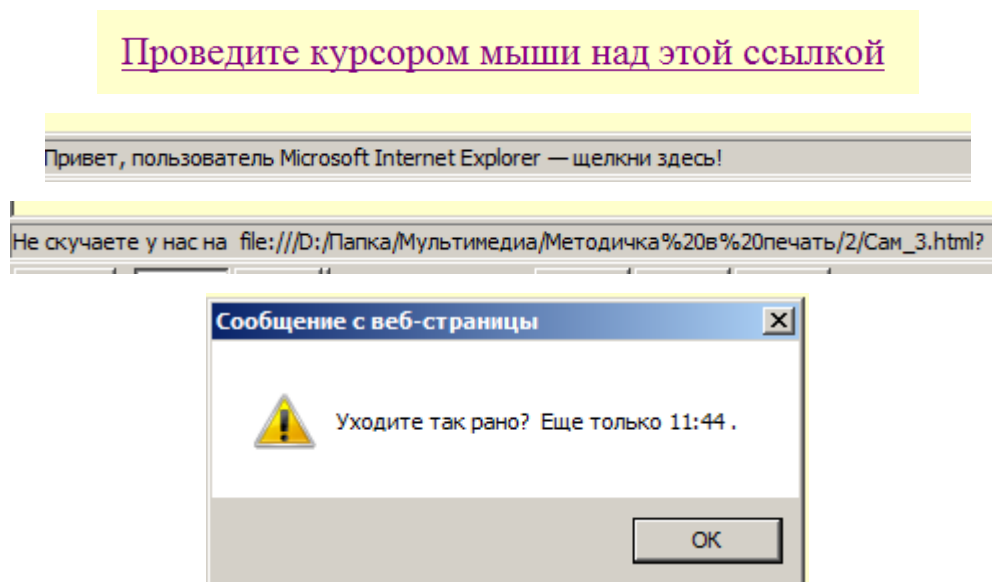


Рисунок 2.19 Результат выполнения самостоятельной работы 3

Задание 2.19. Набрать и проверить работу скрипта, который должен открывать окно. В этом окне должен выводиться документ doc.html, который нужно предварительно создать с любым текстом в теле документа.

```
<SCRIPT type="text/javascript" >
window.open('doc.html', 'joe', config='height=300,width=300')
</SCRIPT>
```

Результат работы показан на рисунке 2.20.

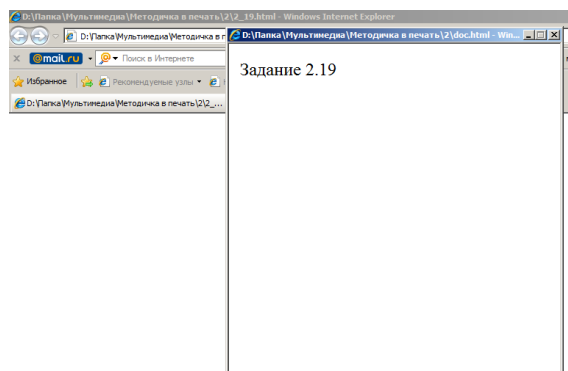


Рисунок 2.20 Результат работы скрипта 2.19

Добавить к предыдущему скрипту ссылку, которая будет закрывать окно:

`Щелкните, чтобы закрыть`

Результат работы ссылки показан на рисунке 2.21.

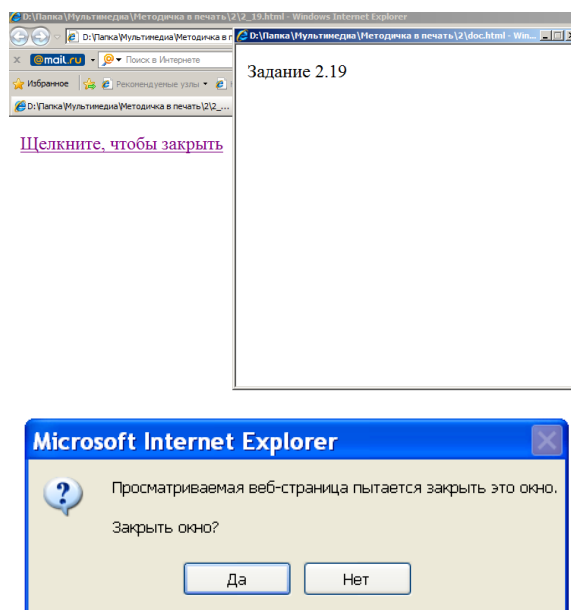


Рисунок 2.21 Результат работы программы

Задание 2.20. Набрать и проверить работу сценария, который содержит функцию, открывающую новое окно. Новое окно и его содержимое будет содержаться в том же документе HTML (рис.2.22).

`<HTML>`

`<SCRIPT type="text/javascript">`

`function openindex()`

```

{
var OpenWindow=window.open("", "newwin", "height=300,width=300");
OpenWindow.document.write("<HTML>")
OpenWindow.document.write("<TITLE>Новое окно</TITLE>")
OpenWindow.document.write("<BODY BGCOLOR='00ff00'>")
OpenWindow.document.write("<CENTER>")
OpenWindow.document.write("<font size=+1>Новое окно</font><P>")
OpenWindow.document.write("<a href='http://www.pgsha.ru'
target='main'>Эта ссылка<BR> откроется в основном окне</a><p>")
OpenWindow.document.write("<P><HR WIDTH='60%'><P>")
OpenWindow.document.write("<a href="      onClick='self.close()'>Эта
ссылка закроет окно</a><p>")
OpenWindow.document.write("</CENTER>")
OpenWindow.document.write("</HTML>")
}
</SCRIPT>
<BODY onLoad="openindex()">

</BODY>
</HTML>

```

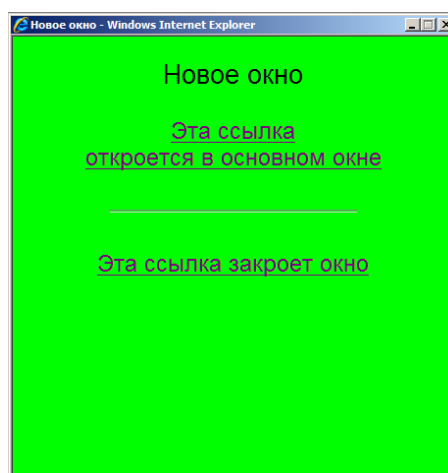


Рисунок 2.22. Результат работы скрипта 2.20

Самостоятельная работа 2.4. Написать сценарий, который откроет новое окно размером 300 на 500 пикселей, которое содержит две ссылки:

одна откроет новую страницу в главном окне; вторая откроет новую страницу в том же окне.

Страница, которая откроется в том же маленьком окне, должна содержать ссылку, закрывающую окно.

Фон страницы желтый (ffff00).

Самостоятельная работа 2.5. Написать функцию, которая открывает окно. Документ, который появится в окне, должен иметь зеленый фон и заголовок TITLE: "Привет, "имя пользователя", вот Ваше зеленое окно!" Имя пользователя можно узнать с помощью запроса (prompt). Добавьте ссылку, которая закрывает окно.

Лабораторная работа 3. Работа со звуком в программе LMMS

Задание 3.1. Запустить программу.

Порядок работы:

1. Выполнить цепочку действий: *Пуск – Все программы – LMMS 1.1.0 – LMMS* (рис.3.1).

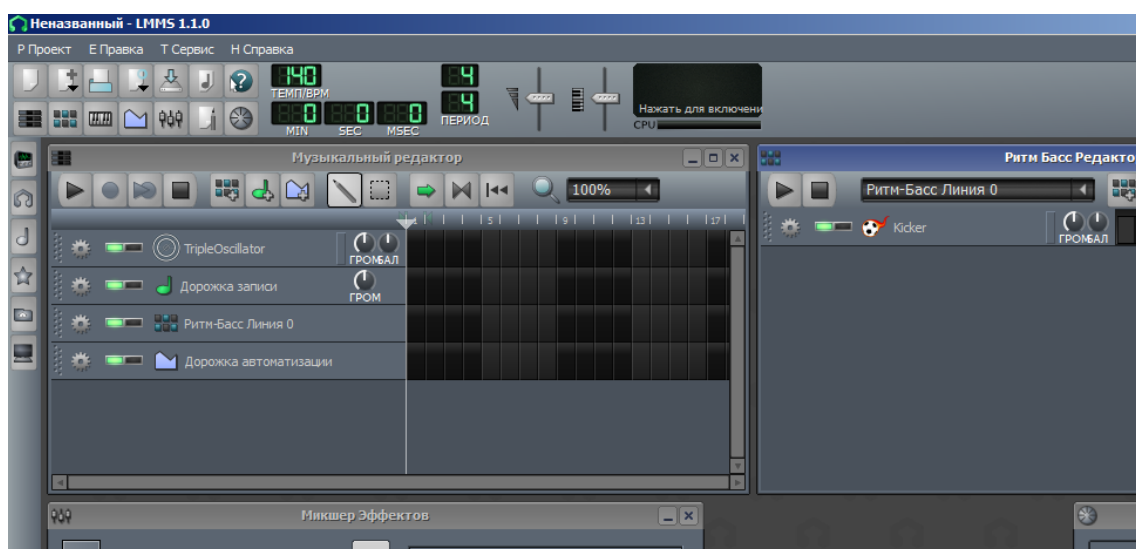


Рисунок 3.1 Окно LMMS

Задание 3.2. Загрузить вокальную партию, которая послужит основой трека.

Порядок работы:

1. Выполнить цепочку действий: Проект – Открыть – в строке *Перейти к* указать: D:\ФИО\LMMS\golos.mmpz. Результат показан на рисунке 3.2.



Рисунок 3.2 Окно программы с загруженной вокальной партией

2. Установить в главной панели темп (скорость):

Для основной мелодии необходимо установить темп равный 97. Нажать левую клавишу мыши на окне с темпом (скоростью) и, передвигая ее вверх или вниз, задать необходимый темп (рис. 3.3).



Рисунок 3.3 Окно установки темпа


Задание 3.3. Подключить к каждой дорожке файл с вокальной партией.

Порядок работы:

1. Щелчком мыши по названию дорожки *golos_№* вызвать окно основных свойств дорожки (рис.3.4).



Рисунок 3.4 Окно основных свойств дорожки

2. Кнопкой *Открыть другую запись*  вызвать окно *Открыть звуковой файл* и выбрать файл D:\ФИО\LMMS\vokal.wav.
3. Повторить пп.1-2 для всех четырех дорожек.
4. Прослушать вокальную партию.
5. Сохранить проект как новую версию: D:\ФИО\LMMS\golos-01.mmpz.

Задание 3.4. Добавить новую ритм-басс линию.

Порядок работы:


1. Щелчком мыши по кнопке *Добавить ритм/басс*  в окне музыкального редактора добавить *Ритм-Басс Линию 0* (рис.3.5).



Рисунок 3.5 Добавление ритм-басс линии

2. Щелчком мыши по названию нового паттерна вызвать Ритм-Басс редактор (рис. 3.6). Окно редактора пока пустое.

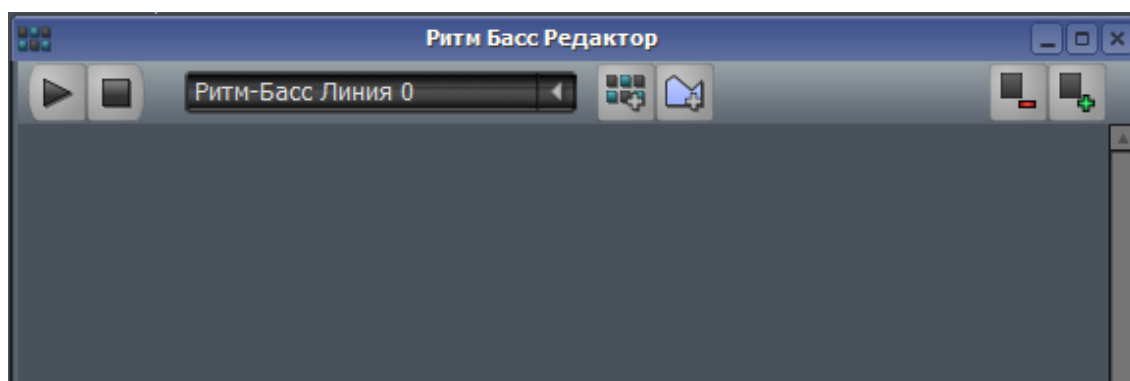


Рисунок 3.6 Окно Ритм Басс редактора

3. Добавить в окно Ритм Басс Редактора три инструмента из папки Мои записи\drums (рис.3.7). Для вставки в окно редактора нужного инструмента достаточно дважды щелкнуть по нему мышью.

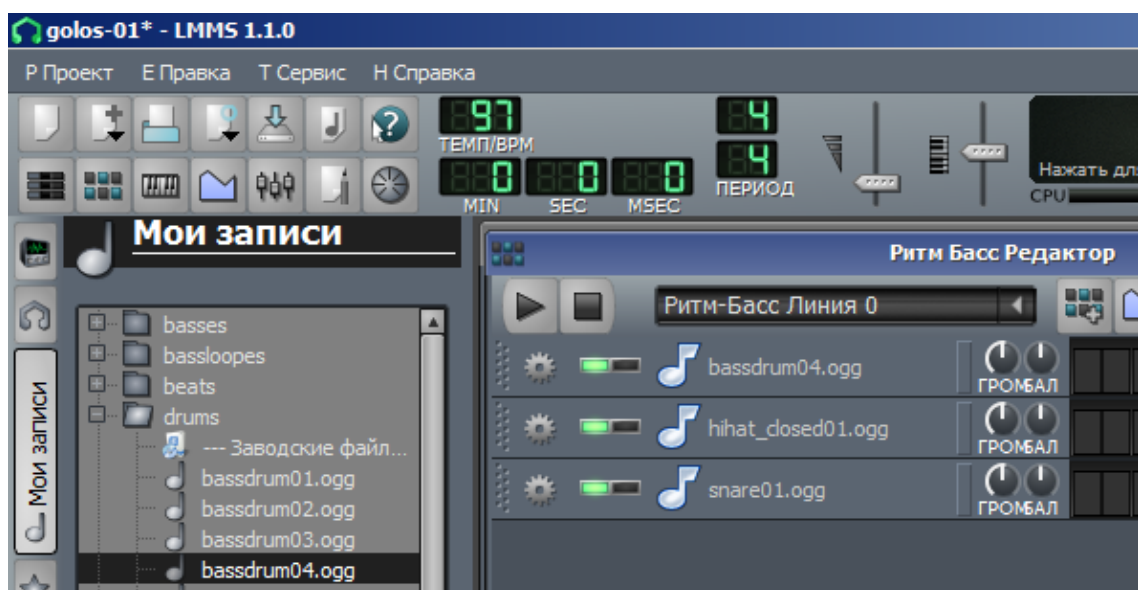


Рисунок 3.7 Инструменты ритм-басс линии

6. Настроить порядок звучания инструментов в соответствии с рисунком 3.8. Щелчок мыши включает/выключает инструмент.

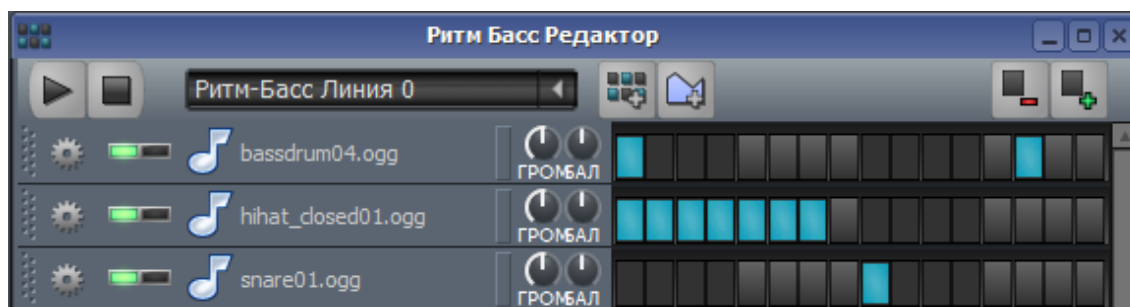


Рисунок 3.8 Порядок звучания инструментов ритм-басс линии 0

7. Закрыть окно Ритм Басс Редактора.

8. Настроить порядок звучания Ритм-Басс Линии 0 в соответствии с рисунком 3.9.



Рисунок 3.9 Настроенная ритм-басс линия 0

9. Прослушать проект.

10. Сохранить проект: D:\ФИО\LMMS\golos-01.mmpz.

Задание 3.5. Добавить новую ритм-басс линию. Инструменты и порядок их звучания представлены на рисунке 3.10. Порядок звучания Ритм-Басс Линии 1 показан на рисунке 3.11. Выполнить самостоятельно.

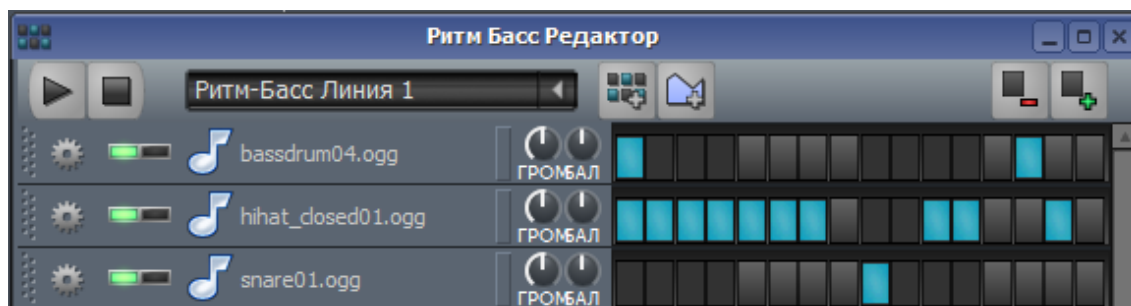


Рисунок 3.10 Инструменты и порядок их звучания в ритм-басс линии 1

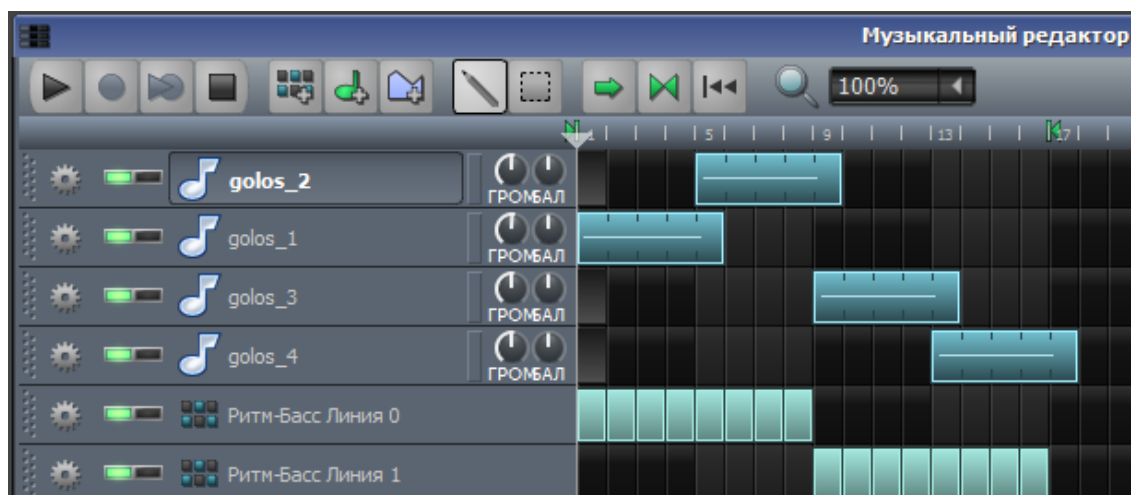


Рисунок 3.10 Добавлены две ритм-басс линии

Задание 3.6. Добавить и настроить нотную линию.

Порядок работы:

1. Найти инструмент LB302: Instrument plugins\LB302 и переместить его с помощью мыши в окно музыкального редактора (рис.3.11).

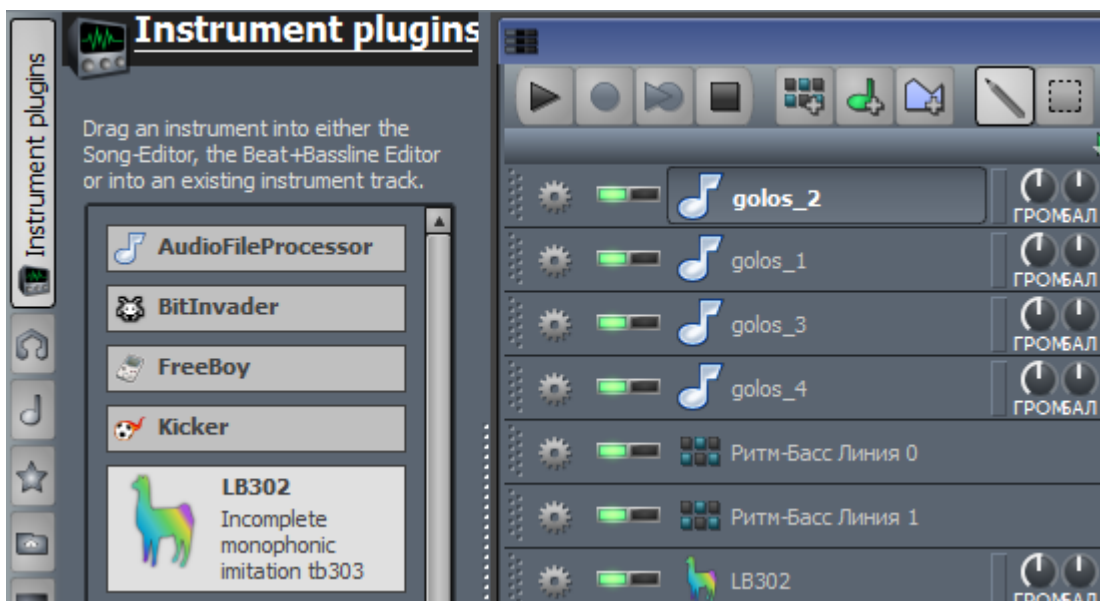


Рисунок 3.11 Добавлен новый инструмент LB302

2. Настроить новый инструмент в окне нотного редактора, для перехода в который нужно дважды щелкнуть мышью в рабочем окне музыкального редактора по пустой дорожке инструмента LB302. Окно настройки показано на рисунке 3.12.

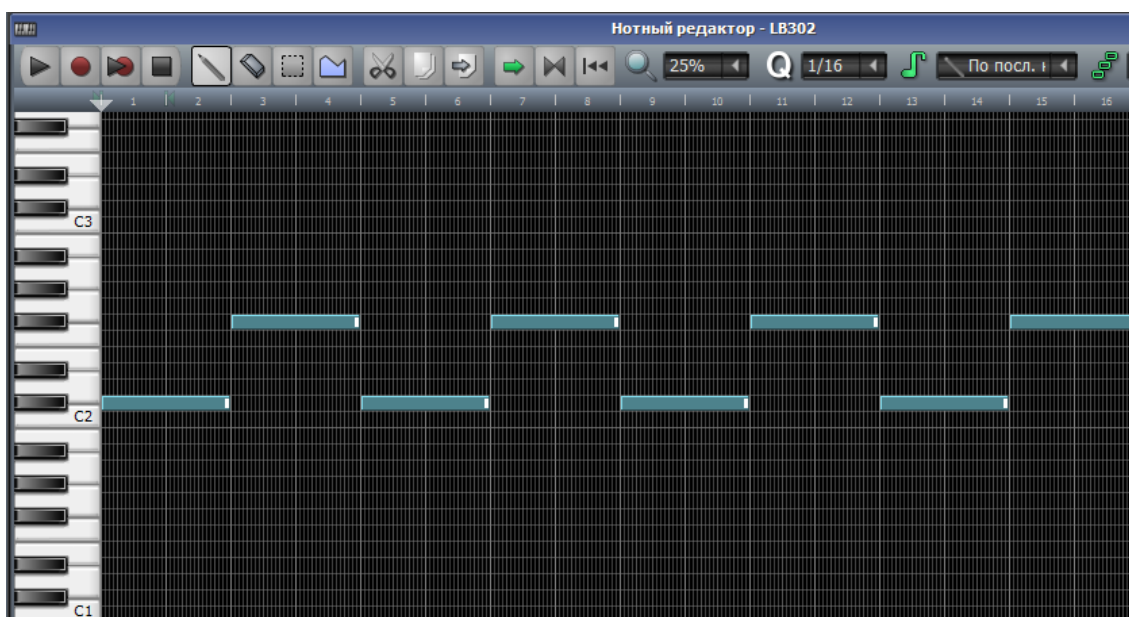


Рисунок 3.12 Окно настройки инструмента LB302

3. Окно музыкального редактора после настройки должно выглядеть следующим образом (рис. 3.13):

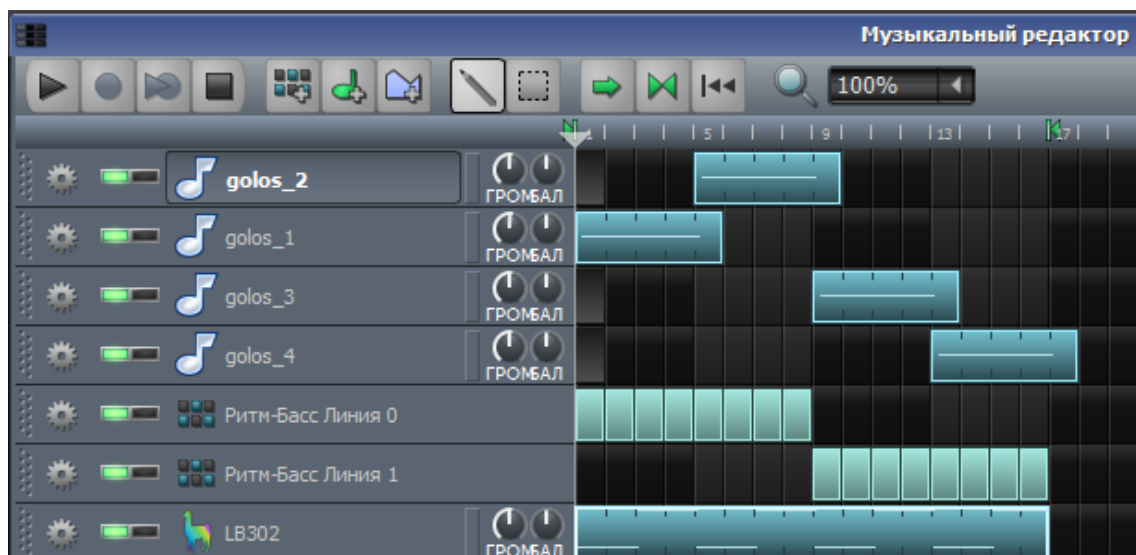


Рисунок 3.13 Окно музыкального редактора с инструментом LB302

4. Прослушать и сохранить проект.

5. Установить основные настройки инструмента LB302 в соответствии с таблицей 1.

Таблица 1 Настройки инструмента LB302

№ п/п	Группа	Параметр	Значение
1	Основные настройки	Громкость	29,9
2	Плагины	CUT	0,67
3	Плагины	RES	0,425
4	Плагины	ENV MOD	0,61
5	Плагины	DEC	0,375
6	Плагины	DIST	0,1
7	Плагины	SLIDE	0,6
8	Функции	ДИАП	1,0
9	Функции	RANGE	1,0
10	Функции	TIME	463
11	Функции	GATE	38

6. Включить параметр ARPEGGIO и проверить значения других параметров группы *ФУНКЦИИ* в соответствии с рисунком 3.14.

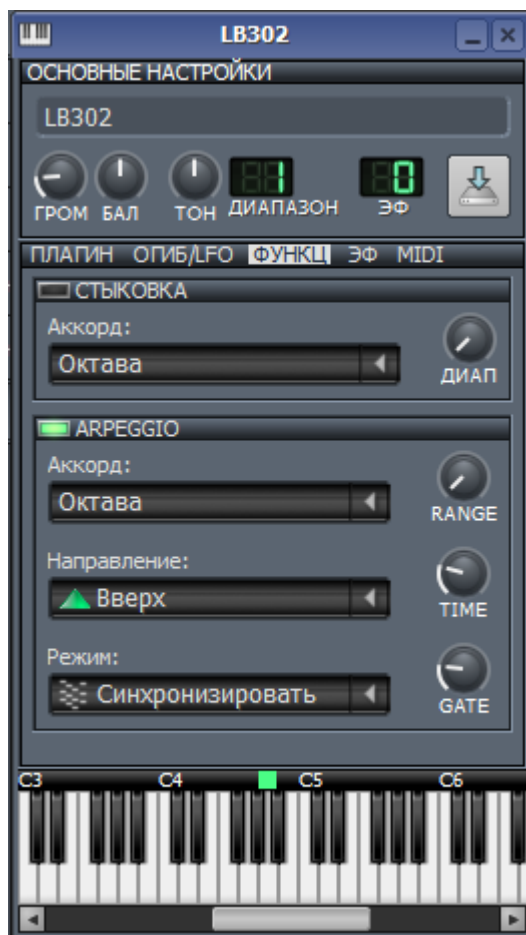


Рисунок 3.14 Окно настройки параметров группы *Функции*

7. Прослушать и сохранить проект.

Задание 3.7. Добавить и настроить следующую нотную линию.

Порядок работы:

1. Найти инструмент `pad_ethereal.xpf`: Мои предустановки\Organic\`pad_ethereal.xpf` и переместить его с помощью мыши в окно музыкального редактора (рис.3.15).

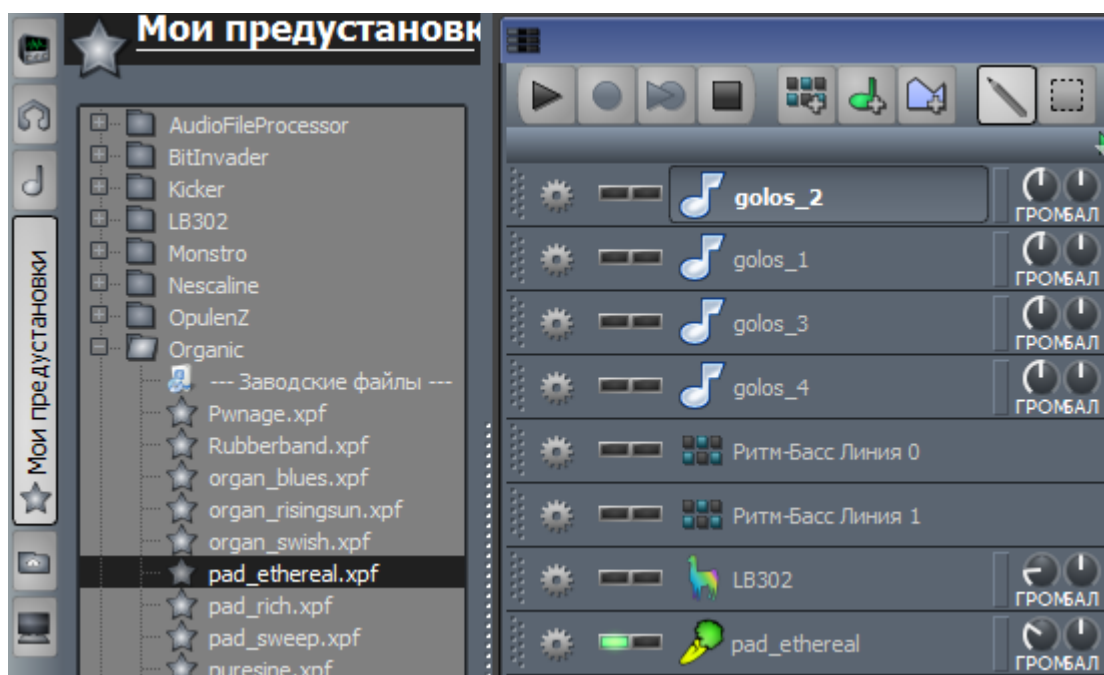


Рисунок 3.15 Добавлен новый инструмент pad_ethereal.xpf

2. Настроить нотную линию для нового инструмента в окне нотного редактора. Окно нотного редактора показано на рисунке 3.16. Выполнить самостоятельно.

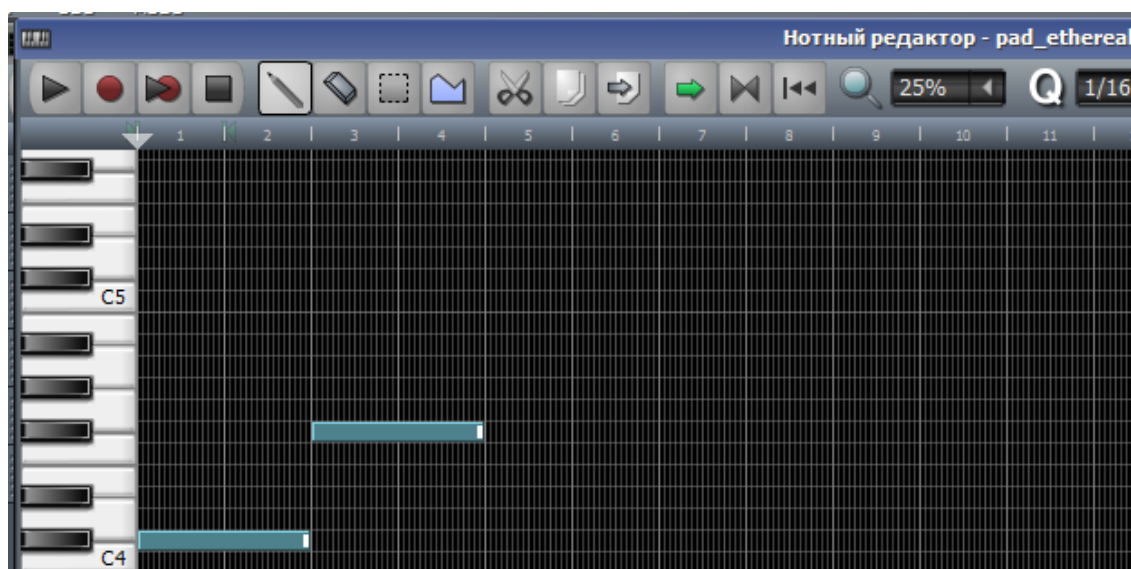


Рисунок 3.16 Нотная линия инструмента pad_ethereal.xpf

В окне музыкального редактора дорожка инструмента pad_ethereal.xpf должна выглядеть следующим образом (рис.3.17):

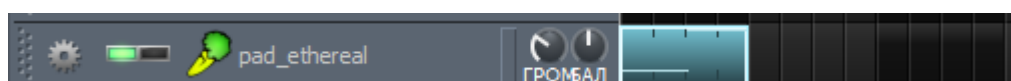


Рисунок 3.17 Инструмент pad_ethereal.xpf

3. Чтобы инструмент звучал весь необходимый отрезок времени, нужно скопировать заполненный фрагмент (рис.3.18).

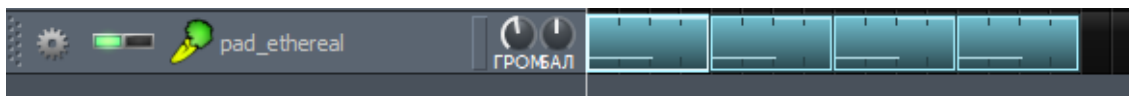


Рисунок 3.18 Дорожка инструмента pad_ethereal.xpf

4. В окне основных настроек инструмента pad_ethereal.xpf выполнить настройки в соответствии с таблицей 2.

Таблица 2 – Настройки инструмента pad_ethereal.xpf

№ п/п	Группа	Параметр	Значение
1	Основные настройки	Громкость	92,6
2	ОГИБ/LFO	AMT	1,0
3	ОГИБ/LFO	DEL	0,0
4	ОГИБ/LFO	ATT	0,678
5	ОГИБ/LFO	HOLD	0,329
6	ОГИБ/LFO	DEC	0,702
7	ОГИБ/LFO	SUST	0,446
8	ОГИБ/LFO	REL	2,0
9	ОГИБ/LFO	DEL	1,0
11	ОГИБ/LFO	SPD	0,1
12	ОГИБ/LFO	AMT	0

Результат настроек показан на рисунке 3.19.



Рисунок 3.19 Настройки инструмента pad_ethereal.xpf

5. Прослушать и сохранить проект.

Задание 3.8. Добавить и настроить инструмент ScaryBell.xpf.

Порядок работы:

1. Найти инструмент ScaryBell.xpf: Мои предустановки\Monstro\ScaryBell.xpf и добавить его в окно музыкального редактора(рис.3.20).

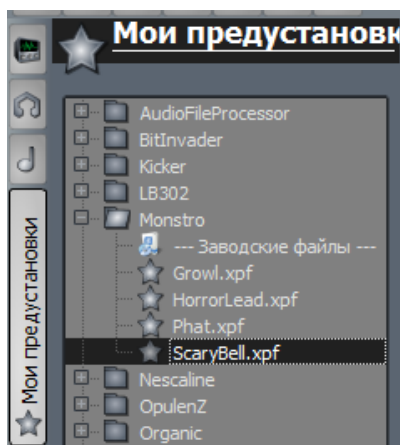


Рисунок 3.20 Расположение инструмента ScaryBell.xpf

2. Настроить нотную линию для нового инструмента в окне нотного редактора. Окно нотного редактора показано на рисунке 3.21. Выполнить самостоятельно.

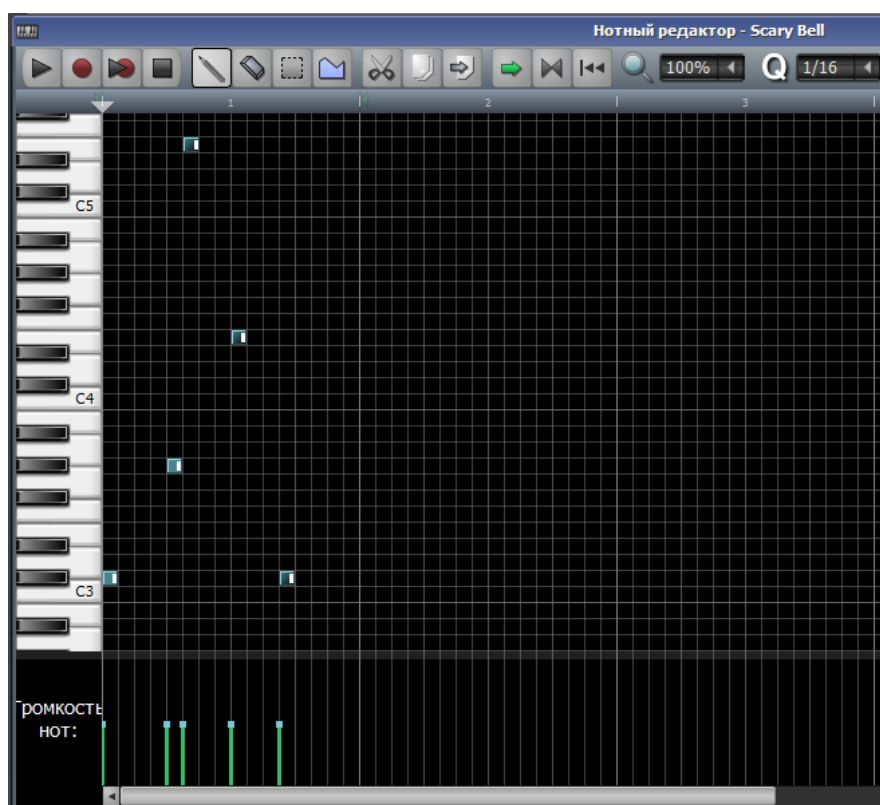


Рисунок 3.21 Нотная линия инструмента ScaryBell.xpf

3. В окне музыкального редактора скопировать нужное количество раз (рис. 3.22).




Рисунок 3.22 Окно музыкального редактора после добавления всех инструментов

4. Прослушать и сохранить проект.

Задание 3.9. Настроить микшер эффектов для более объемного звучания.

Порядок работы:

1. В панели инструментов вызвать микшер эффектов  (рис.3.23).

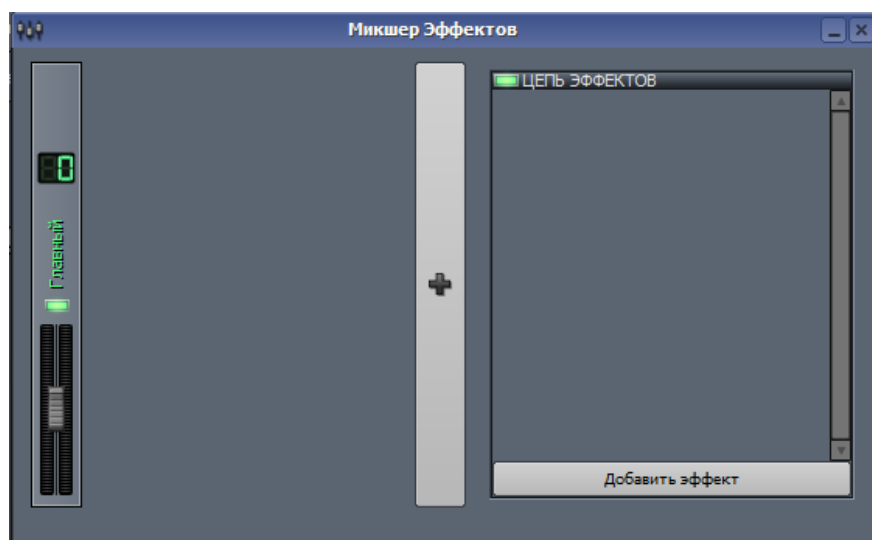


Рисунок 3.23 Микшер эффектов

2. Кнопкой *Добавить эффект* вызвать список доступных эффектов и выбрать эффект LADSPA: CalfReverb LADSPA (рис.3.24).

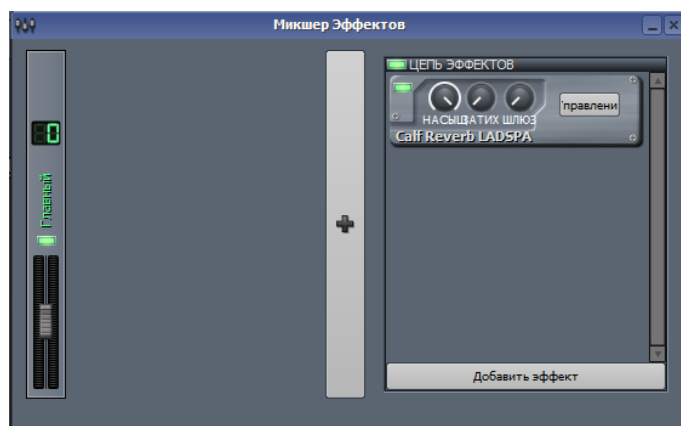


Рисунок 3.24 Микшер эффектов с добавленным эффектом

3. Нажать кнопку управление и произвести настройку (рис.3.25).



Рисунок 3.25 Окно настройки параметров эффекта


4. Добавить настроенный эффект в список эффектов кнопкой . В микшере эффектов новый эффект запишется как эффект 1 (рис.3.26).



Рисунок 3.26 Окно микшера эффектов с добавленным эффектом

5. Прослушать и сохранить проект.

Лабораторная работа 4. Flash-анимация

Задание 4.1. Создать покадровую анимацию «Качели». Покадровая анимация – это анимация, полностью составленная из ключевых кадров. Автор сам должен определить содержимое кадра и его "длительность", т.е. сколько статических кадров будет занимать изображение.

Порядок работы:

5. Запустить Adobe Flash Professional CS6.

В окне запуска выбрать: Создать – ActionScript 3.0 (рис. 4.1).

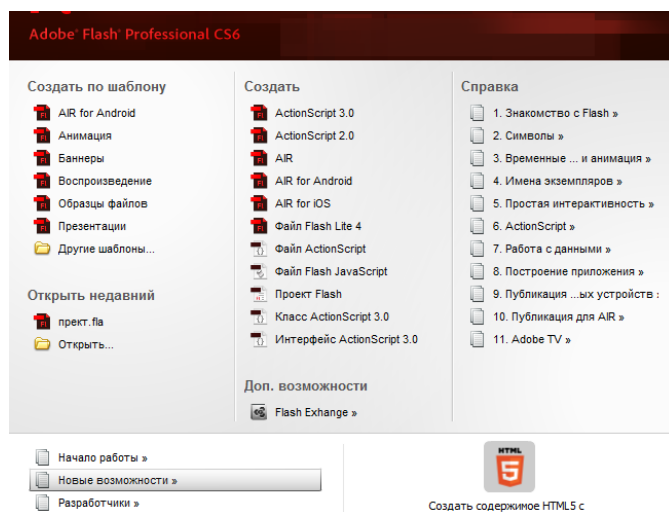


Рисунок 4.1 Окно запуска программы

Результат загрузки программы показан на рисунке 4.2.

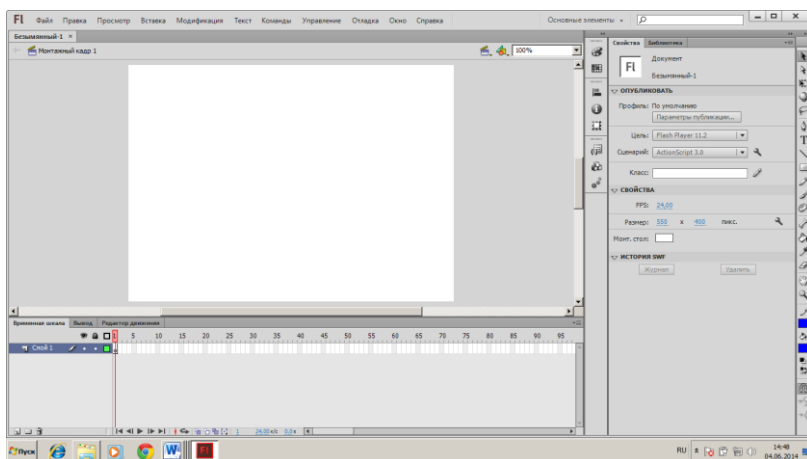


Рисунок 4.2 Интерфейс программы

6. Нарисовать объекты: кубик, гиря, доска, подставка доски. Каждый объект сохранить как символ (меню правой кнопки на выделенном объекте – Преобразовать в символ... - Задать имя - Ок). Вся работа выполняется в

окне Монтажный кадр1 с использованием инструментов из панели инструментов, расположенной справа. Для каждого нарисованного объекта необходимо выполнить преобразование в символ (рис. 4.3).

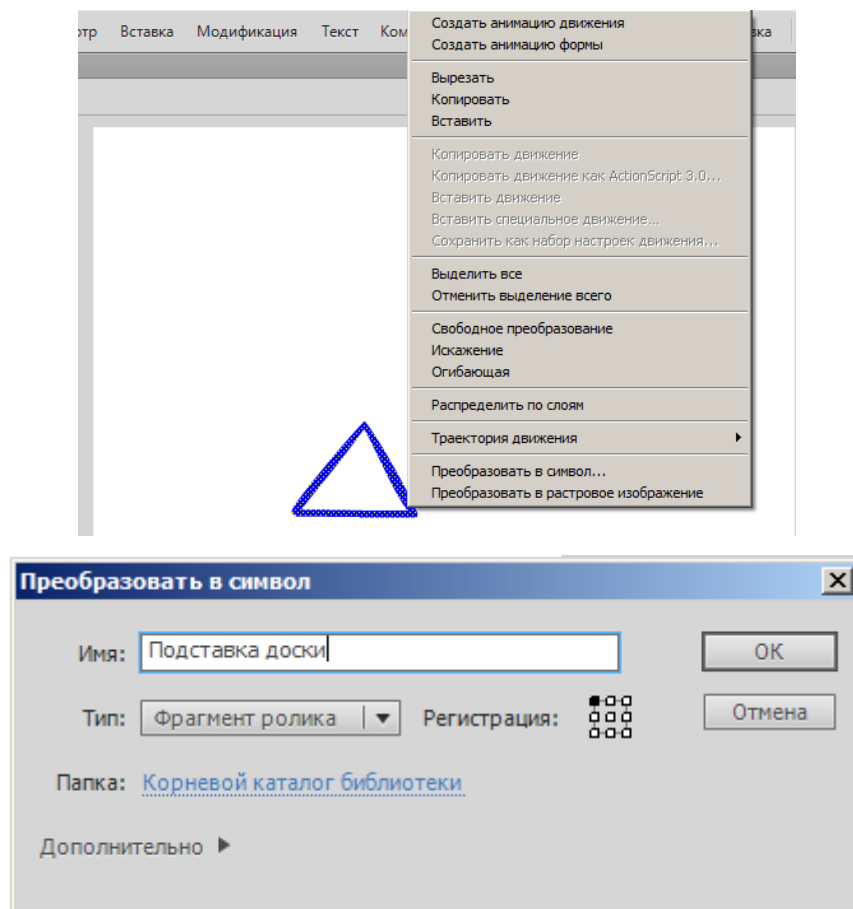


Рисунок 4.3 Преобразование нарисованного объекта в символ

7. Объекты расположить, как показано на рисунке 4.4.

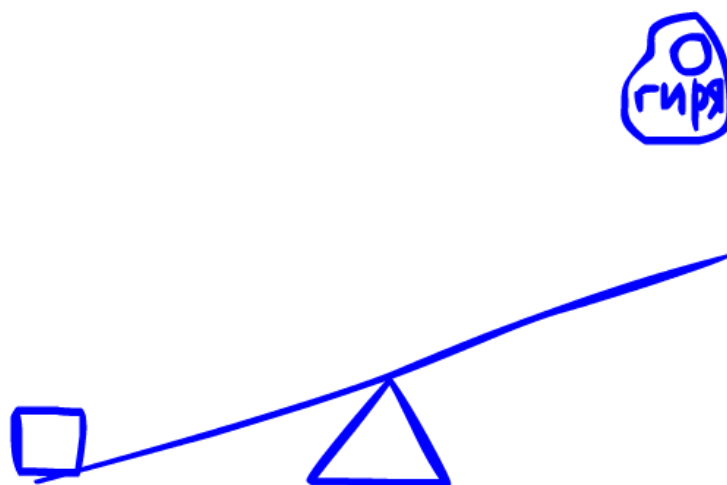


Рисунок 4.4 Первоначальное расположение объектов

8. Следующие два кадра заполнить содержимым первого кадра. Для этого необходимо нажать дважды F6. В результате первые три кадра оказались заполнены первоначальным рисунком.

9. Для прорисовки нового кадра нужно нажать F7 и выполнить преобразование объектов, как показано на рисунке 4.5. Заполнить новым содержимым еще два кадра.

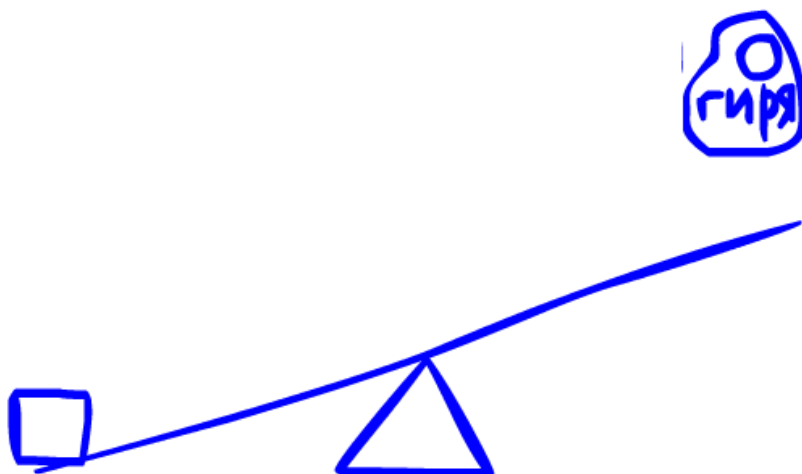


Рисунок 4.5 Второе состояние объектов

10. Следующие два состояния объектов показаны на рисунках 4.6 и 4.7. Добавление кадров выполнить самостоятельно.

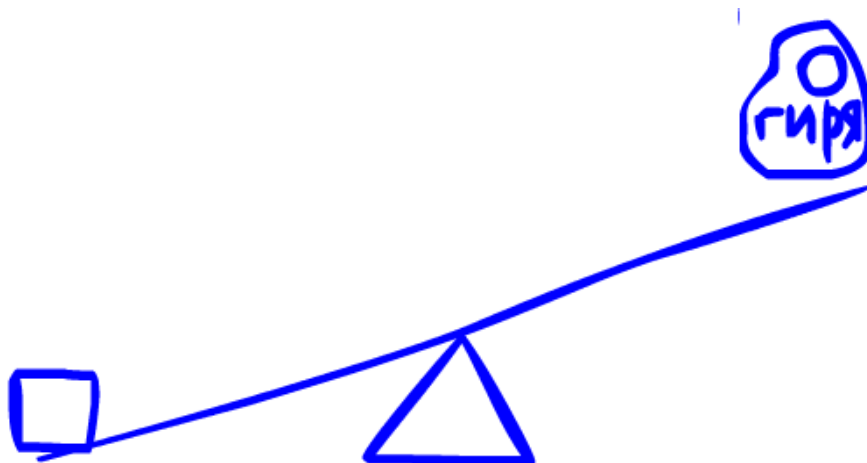


Рисунок 4.6 Третье состояние объектов

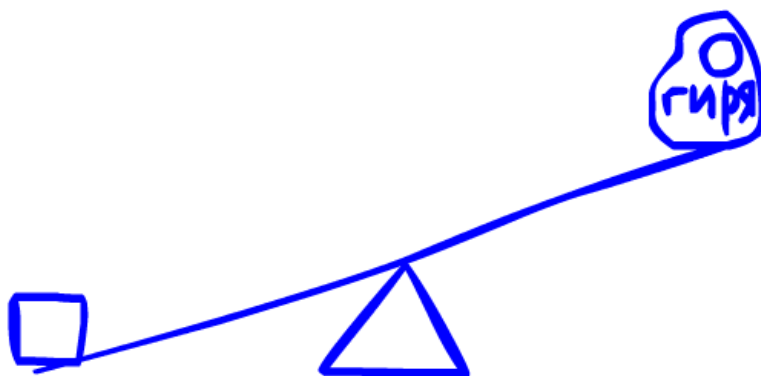


Рисунок 4.7 Четвертое состояние объектов

7. Для выполнения следующего рисунка можно включить Свободное преобразование через меню правой кнопки, вызванное на объекте «доска» (рис. 4.8).

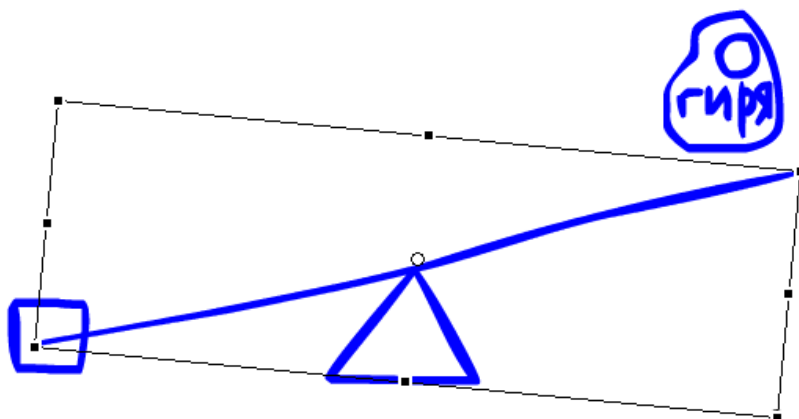


Рисунок 4.8 Включено Свободное преобразование объекта «доска»

8. Все дальнейшие преобразования выполнить самостоятельно. Конечное состояние объектов показано на рисунке 4.9.

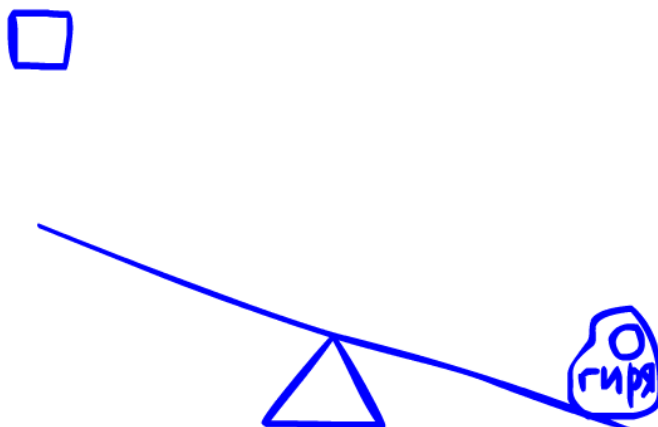


Рисунок 4.9 Конечное состояние объектов.

9. Протестировать ролик в программе, выполнив последовательность действий, показанных на рисунке 4.10:

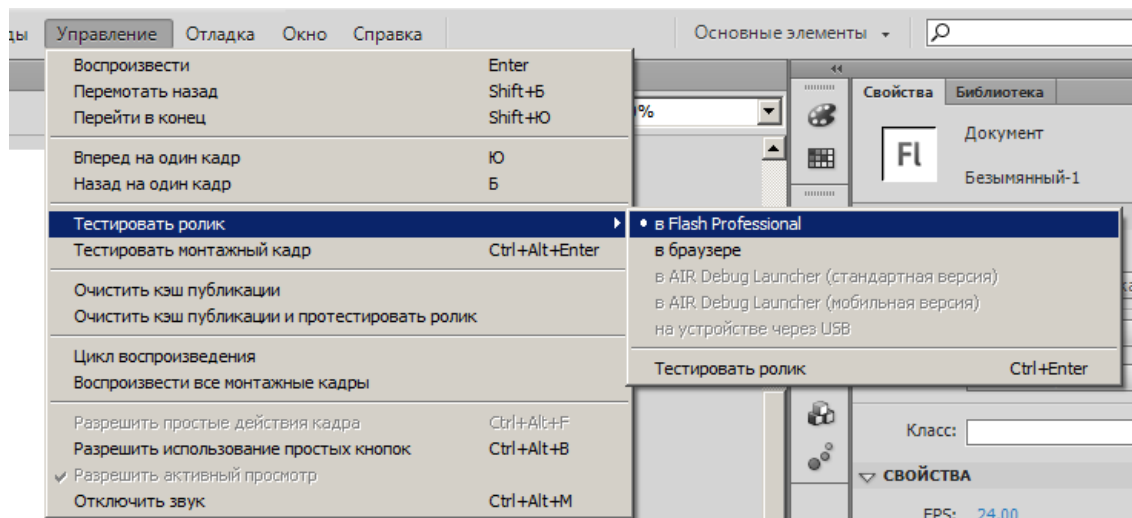


Рисунок 4.10 Тестирование ролика в программе

10. Сохранить ролик в своей папке с именем Качели.gif, выполнив последовательность действий: Файл – Экспорт – Экспортировать ролик (рис.4.11).

11. Проверить работу ролика, запустив файл Качели.gif.

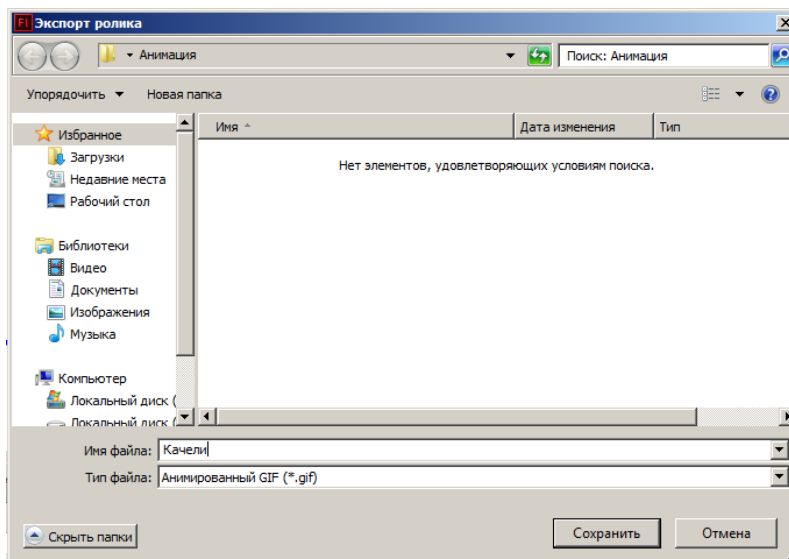


Рисунок 4.11 Экспорт ролика в формате GIF

12. Добавить фон к рисунку. Для этого нужно добавить новый слой (рис. 4.12).

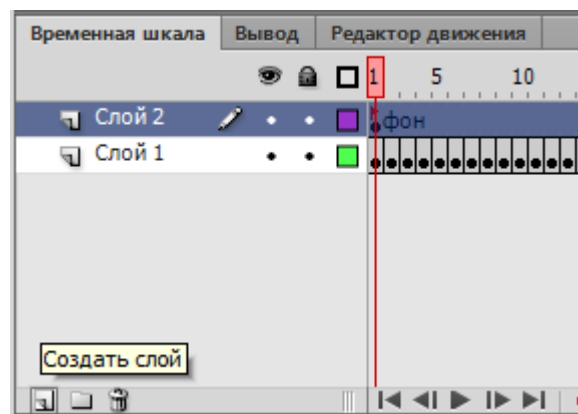


Рисунок 4.12 Добавление нового слоя

13. Установить указатель на временной шкале в начальное положение (рис. 4.12).

14. Поменять с помощью мыши слои местами, т.к. фоновый слой должен быть снизу (рис. 4.13).

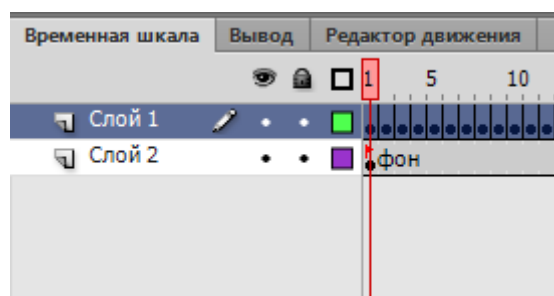


Рисунок 4.13 Расположение слоев

15. Выполнить экспорт и проверить работу ролика.

Самостоятельная работа 4.1. Создать покадровую анимацию «Парашют». В ролике должен быть фон, ключевую роль в котором должен играть обрыв, с которого парашютист будет разбегаться и прыгать. Во время прыжка должен распускаться парашют. Протестировать ролик в программе и в браузере.

Задание 4.2. Создать анимацию движения «Спутник».

Порядок работы:

1. Запустить Adobe Flash Professional CS6.
2. На первом слое нарисовать фон – звездное небо.
3. На втором слое нарисовать спутник.
4. Сохранить спутник в библиотеке символов.

5. Выделить спутник и в контекстном меню выбрать «Создать анимацию движения».
6. Переместить спутник в новое местоположение.
7. Выполнить тестирование ролика в программе.
8. Экспортировать ролик в файл Спутник.gif.
9. Добавить в ролик падающую звезду.
10. Протестировать и выполнить экспорт.

Самостоятельная работа 4.2. Создать анимацию движения «Корабль». В ролике должен быть фон – волнующееся море и солнце. Первоначально корабль должен двигаться прямолинейно. Ролик сохранить с именем Кораблик. Затем изменить траекторию движения, чтобы был эффект качания на волнах. Сохранить ролик с именем «Шторм».

Лабораторная работа 5. Работа в видеоредакторе для нелинейного монтажа Lightworks

Задание 5.1. Создать новый проект в видеоредакторе Lightworks.

Порядок работы:

1. Подготовить 6 видеофайлов продолжительностью не более 30 секунд каждый. Для этого можно воспользоваться любыми аппаратными и программными средствами.
2. Запустить программу Lightworks. Откроется окно запуска программы (рис 1).

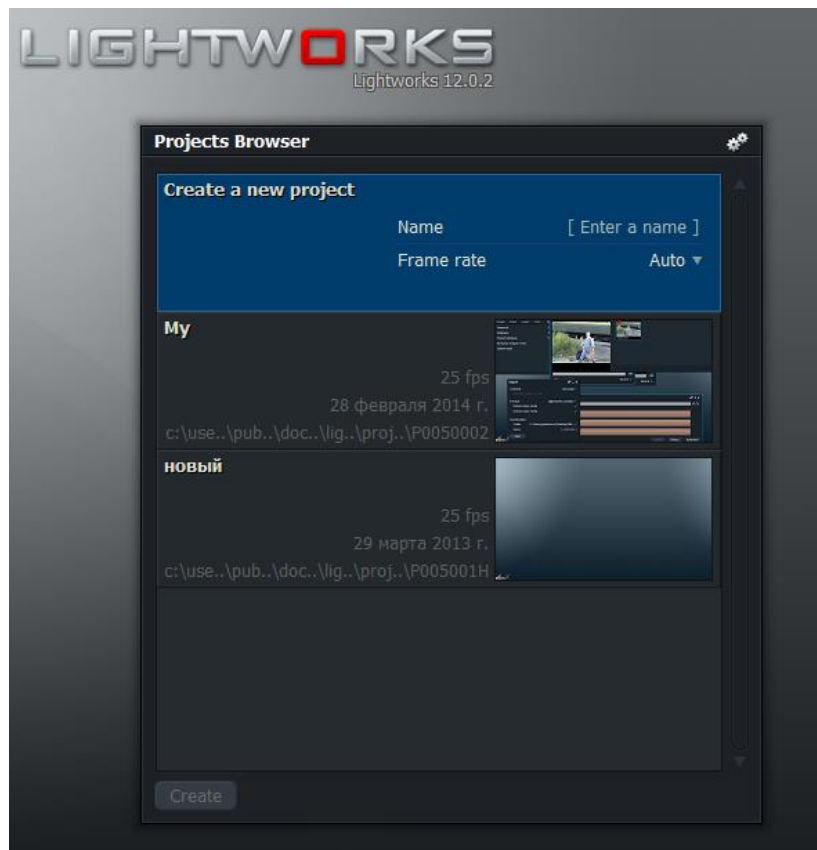


Рисунок 5.1 Окно запуска программы

3. В окне Projects Browser выбрать Create new project. Ввести имя проекта [Enter a name] – Фильм и для параметра Frame rate выбрать значение 25 fps (рис.5.2).

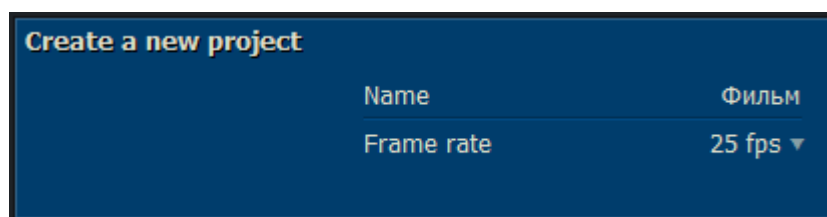


Рисунок 5.2 Обязательные параметры проекта

4. Нажать кнопку Create. Загрузится окно программы (рис.5.3).

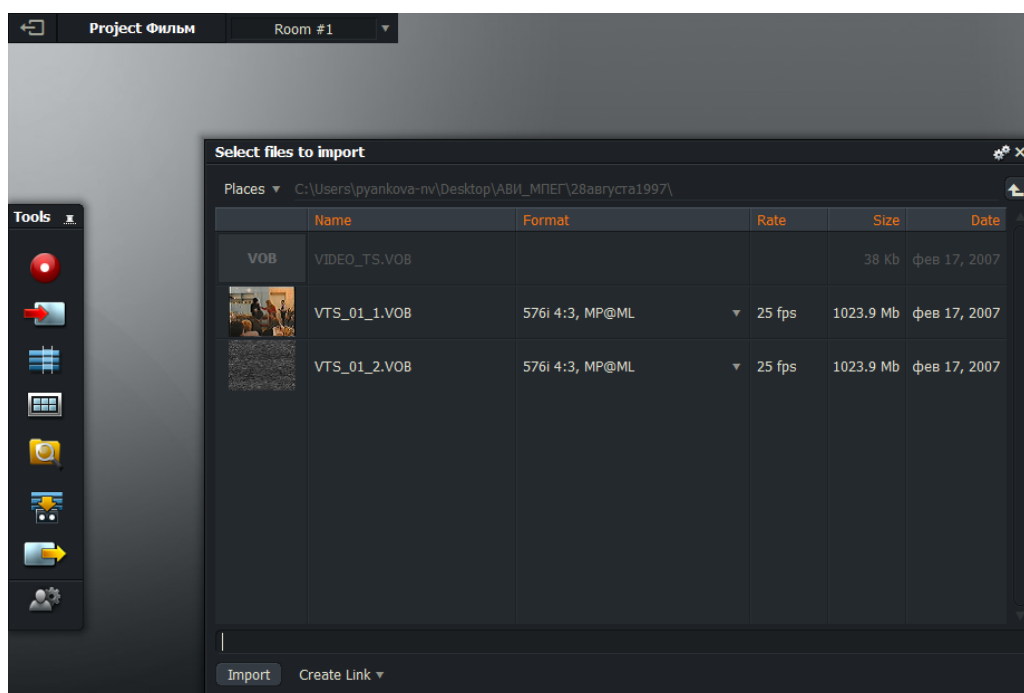


Рисунок 5.3 Рабочее окно программы

Задание 5.2. Выполнить настройку параметров нового проекта ФИЛЬМ.

Порядок работы:

1. Указать путь к папке, в которой будет храниться новый проект:
Project Фильм – Details – Location – D:\ФИО (рис. 5.4).

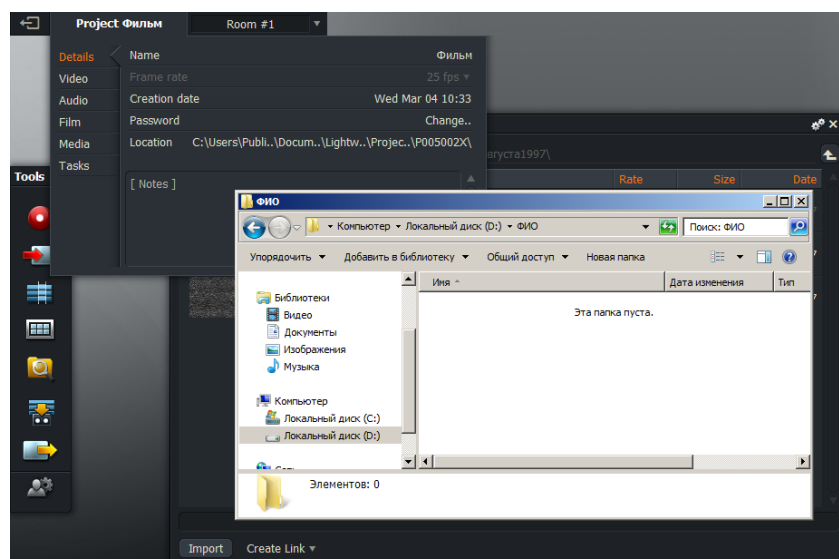


Рисунок 5.4 Место хранения нового проекта

2. Установить параметру Project Фильм – Video (Main) – Output format значение 1080p (sf) 25 fps (рис.5.5).

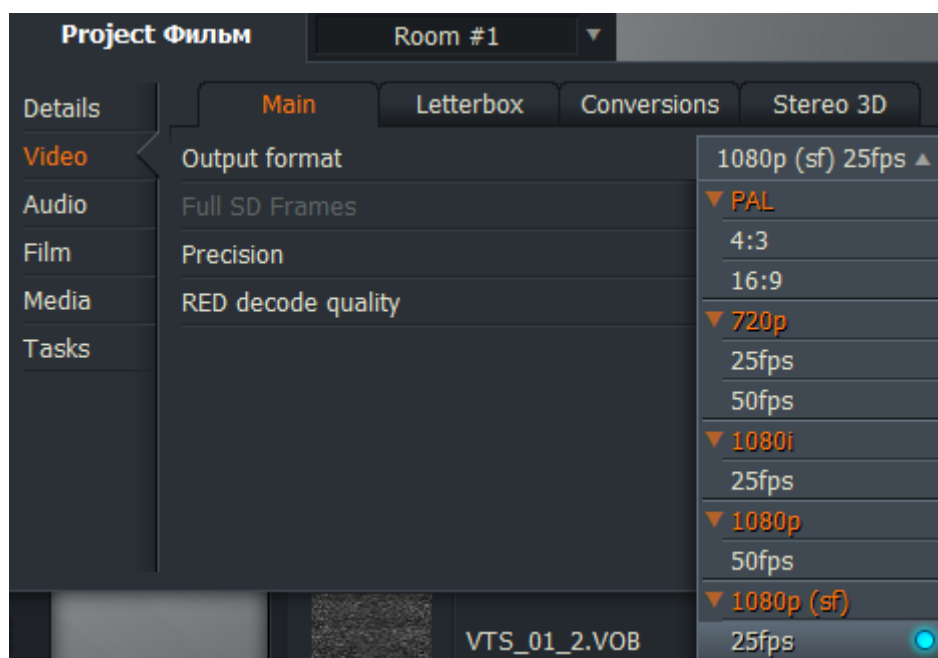


Рисунок 5.5 Параметр вывода видео

3. Значения остальных параметров оставить без изменения.
4. Чтобы закрыть окно параметров проекта достаточно щелкнуть по Project Фильм повторно.
5. Выйти из проекта с помощью кнопки выхода (рис. 5.6). Завершения работы программы при этом не происходит.

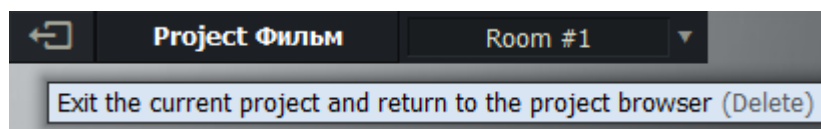


Рисунок 5.6 Кнопка выхода из текущего проекта

6. Обратить внимание на появление нового проекта Фильм в окне Projects Browser.

Задание 5.3. Выполнить экспорт видеофайлов в проект ФИЛЬМ.

Порядок работы:

1. Открыть проект Фильм.
2. В панели инструментов Tools воспользоваться инструментом



для вызова окна Select files to import (рис.5.7).

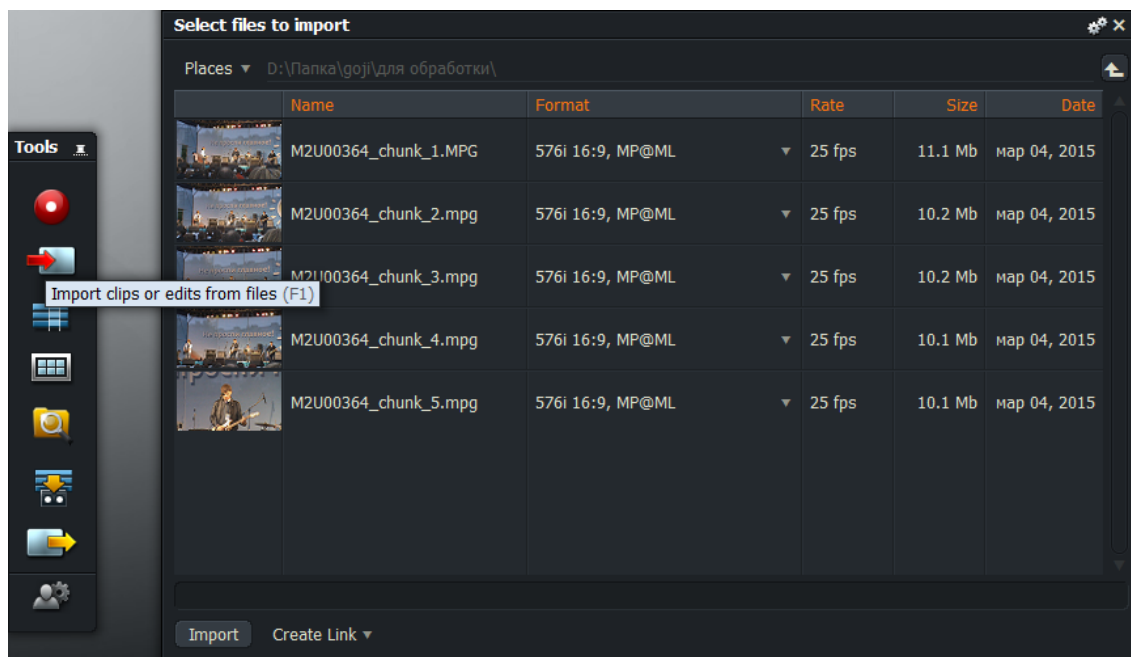


Рисунок 5.7 Окно выбора файлов для импорта

3. Щелкнуть по кнопке Places, чтобы указать путь к папке, в которой находятся файлы для импорта.

4. Выделить первые три импортируемых файла и нажать кнопку Import (рис. 5.8).

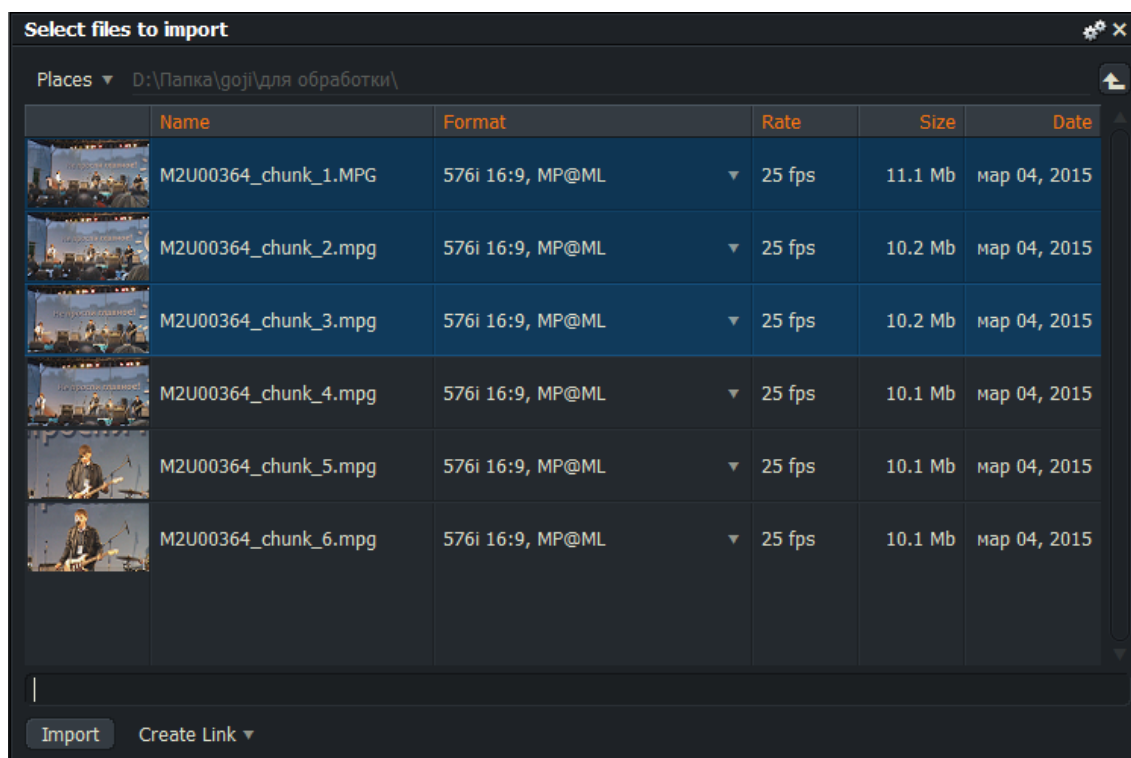


Рисунок 5.8 Выбор файлов для импорта

5. Результат импорта показан на рисунке 5.9.

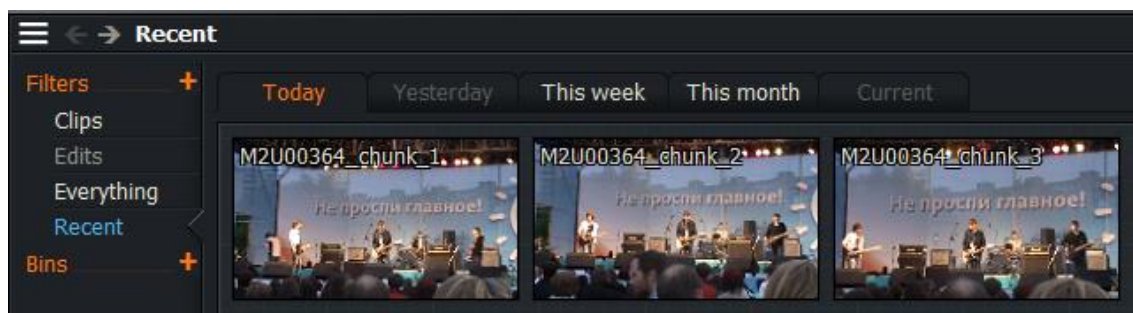
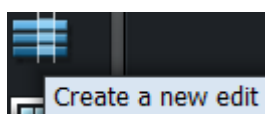


Рисунок 5.9 Окно просмотра содержимого проекта

6. Импортировать в проект оставшиеся три видеофайла. Выполнить самостоятельно.

7. В окне проекта найти инструмент создания нового окна



редактирования и вызвать контекстное меню, в котором выбрать пункт New edit settings.. Установить параметры согласно рисунку 5.10.

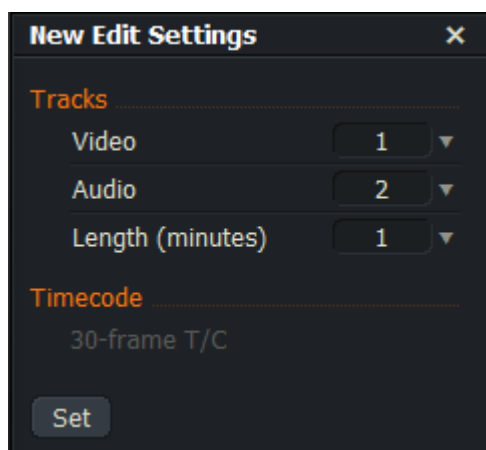


Рисунок 5.10 Параметры окна редактирования

8. Войти в режим редактирования (рис.5.11).

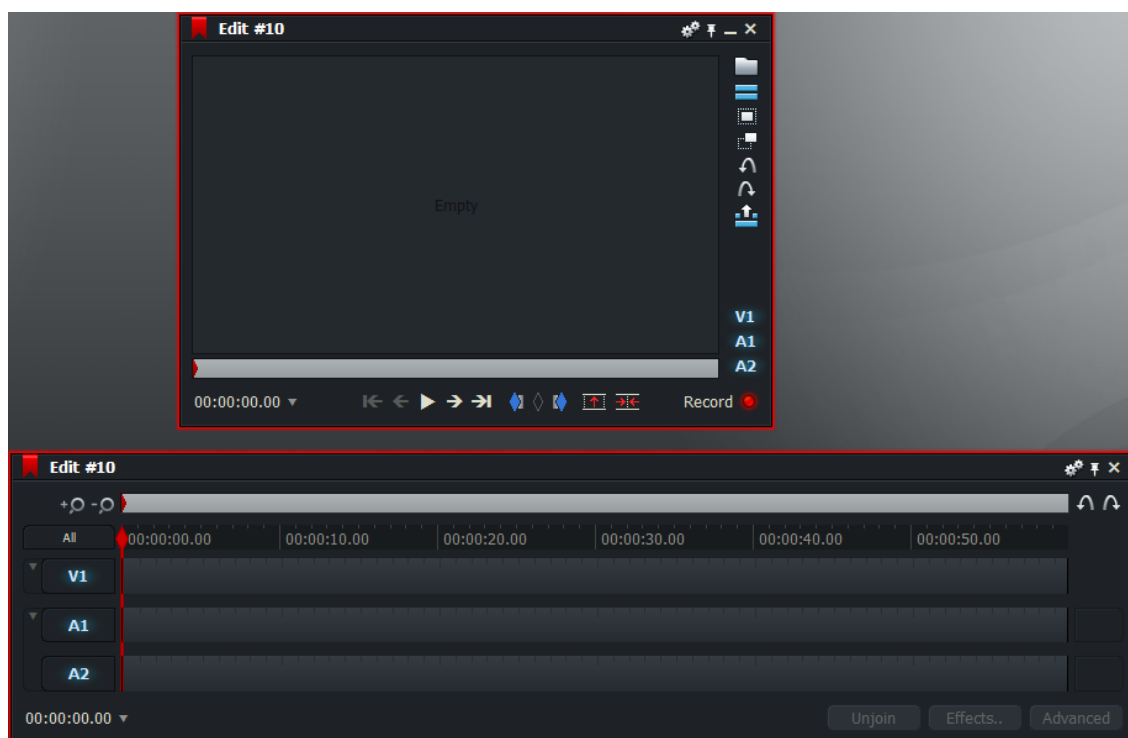


Рисунок 5.11 Две составляющие режима редактирования

9. Двойным щелчком открыть первый видеофайл. В окне видеофайла установить маркеры начала и конца фрагмента, который будет добавлен в проект из первого видеофайла. Кнопки установки маркеров показаны на рисунке 5.12.

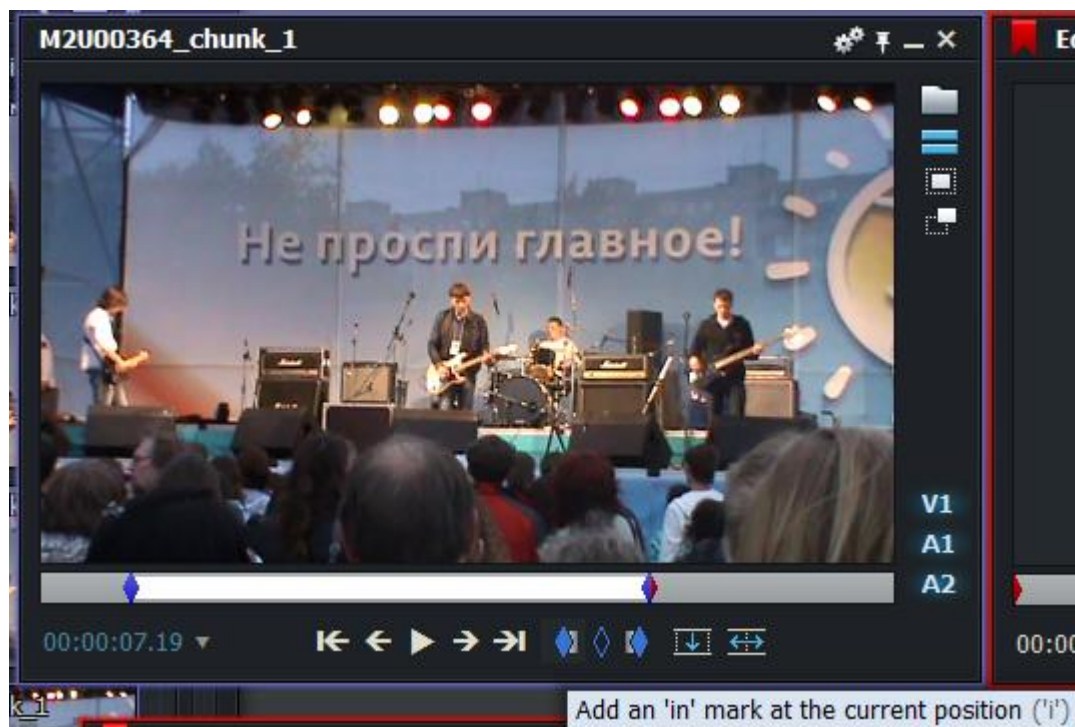


Рисунок 5.12 Установка маркеров начала и конца фрагмента

10. Добавить в окно редактирования выделенный фрагмент с помощью кнопки . Результат показан на рисунке 5.13.

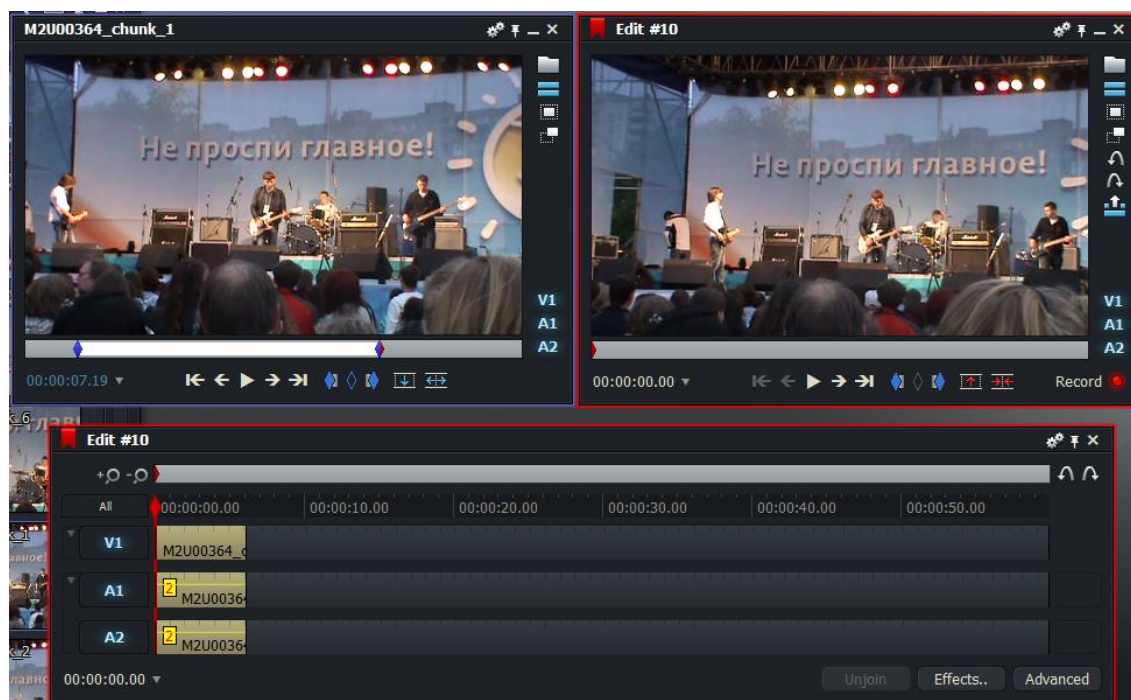


Рисунок 5.13 В окно редактирования добавлен первый фрагмент

11. Добавить в окно редактирования фрагменты всех остальных видеофайлов. Выполнить самостоятельно. Результат показан на рисунке 5.14.

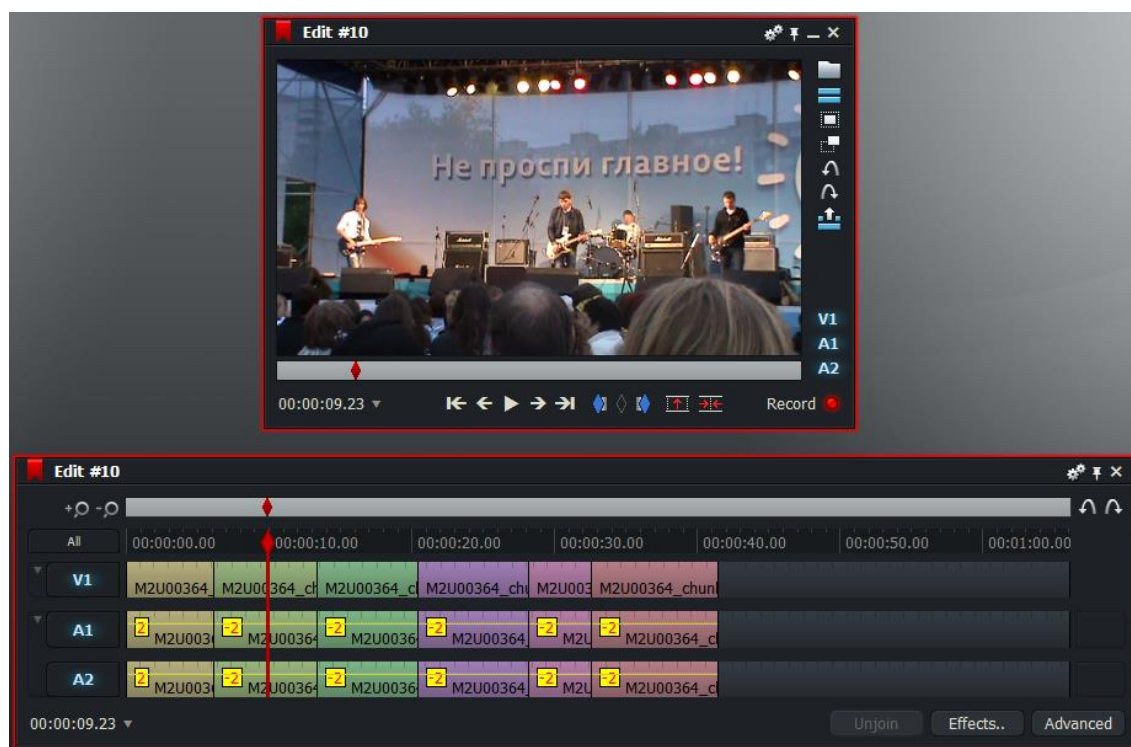


Рисунок 5.14 Окно редактирования после добавления

12. Добавить переходы между первым и вторым фрагментами. Для этого нужно установить курсор между первым и вторым фрагментом и нажать кнопку Effects..(рис.5.15).

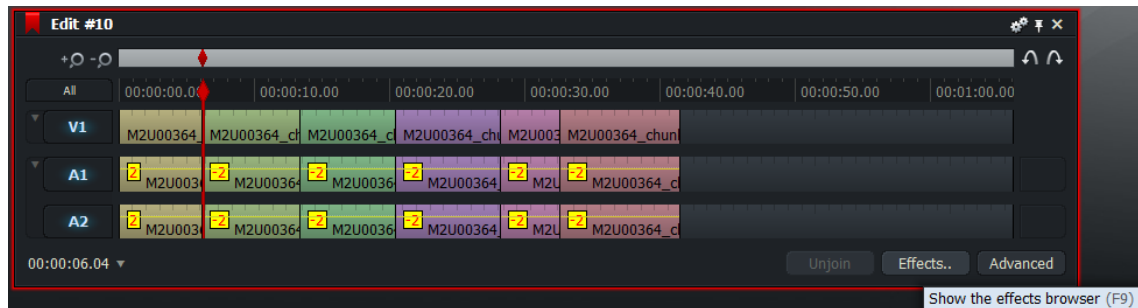


Рисунок 5.15 Кнопка добавления эффекта

13. Откроется окно добавления эффектов Add effects, в котором нужно выбрать параметры, показанные на рисунке 5.16, и нажать кнопку Add.

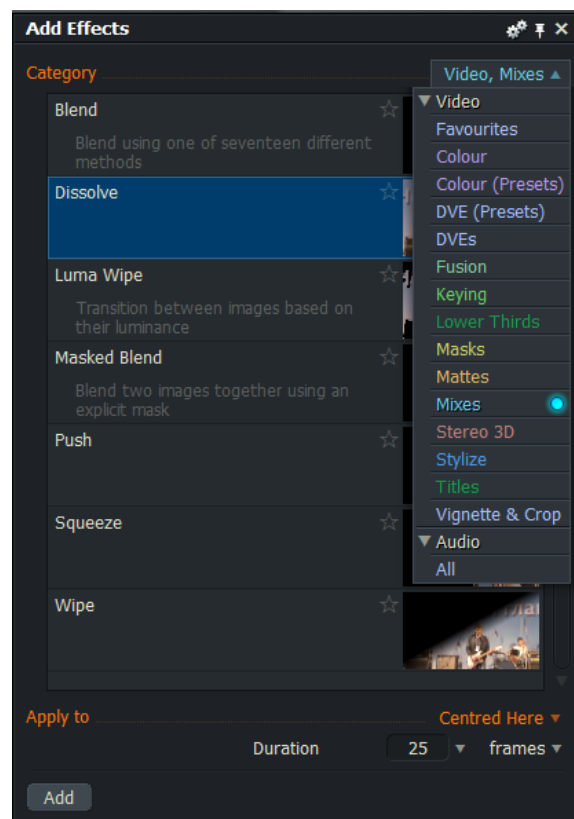


Рисунок 5.16 Окно добавления эффектов

14. Результат добавления показан на рисунке 5.17.

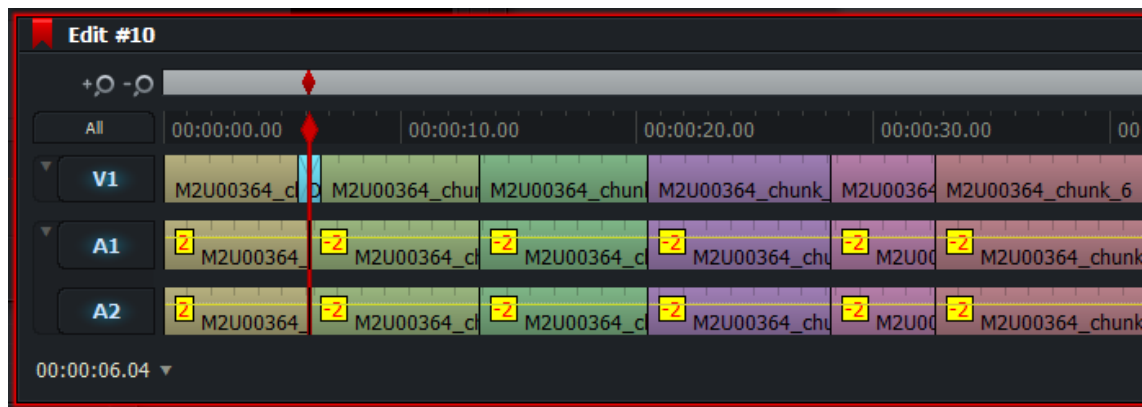
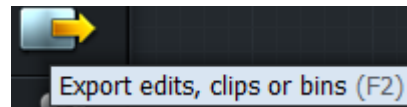


Рисунок 5.17 Добавлен эффект между первым и вторым фрагментами

15. Добавить переходы между всеми остальными фрагментами.
Выполнить самостоятельно.



16. Экспортировать фильм в формате MP4

Установить параметры, показанные на рисунке 5.18.

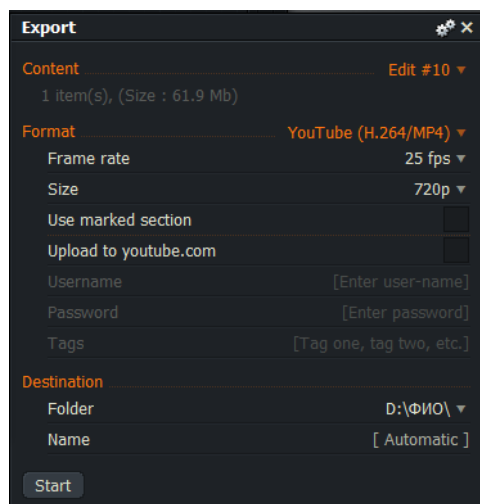


Рисунок 5.18 Параметры экспорта видео

17. На экране будет показано окно Tasks (рис.5.19).

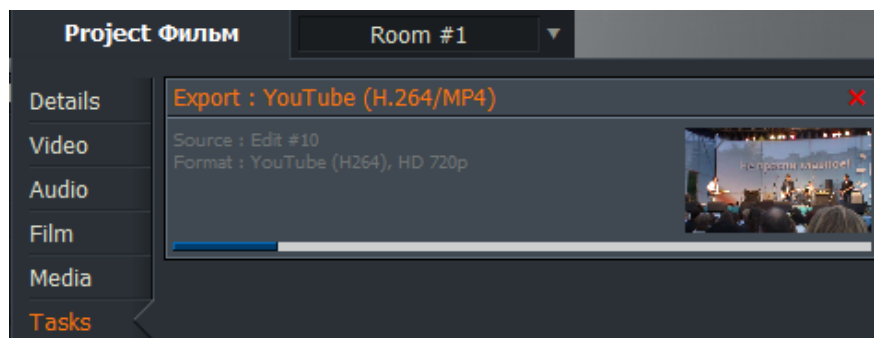


Рисунок 5.19 Окно экспорта видео

Самостоятельная работа 5.1. Записать и отредактировать видеофрагменты с соблюдением следующих обязательных настроек, параметров и требований:

1. Продолжительность сцены не более 120 с.
 2. Развитие сюжета: завязка- экспозиция-кульминация-развязка.
 3. Голос за кадром.
 4. Рекламные паузы.
 5. Титры/Субтитры.
 6. Кадр не короче 2х секунд.
 7. Параллельный и последовательный монтаж.
 8. Масштаб изображения через 1.
 9. Монтаж по ориентации в пространстве.
 10. Перебивка.
- Экспортировать ролик.

Лабораторная работа 6. Настройка служб Windows Media

Задание 6.1. Установить службы потокового мультимедиа (с помощью диспетчера сервера добавить службы потокового мультимедиа на сервер second.company.com).

Порядок работы:

1. Войти на second как пользователь с административными привилегиями.
2. Установить пакет обновления Службы Windows Media. Данный пакет представляет собой загружаемую надстройку Windows Server 2008. Пакет находится на сервере в папке: Файлы студентов\Службы потокового мультимедиа\Windows6.0-KB934518-x86-Server.msu. Для копирования файла надстройки с реального компьютера на виртуальную машину нужно установить дополнения гостевой ОС и подключить папку с надстройкой к внешним папкам. Для доступа к добавленной внешней папке открыть Мой

компьютер и в адресной строке ввести \\vboxsvr. Установку выполнить самостоятельно.

3. Установить роль Веб-сервер (IIS). Выполнить самостоятельно.

4. Включить компонент Возможности рабочего стола: в Диспетчере сервера добавить нужный компонент (рис.6.1.). Выполнить самостоятельно.

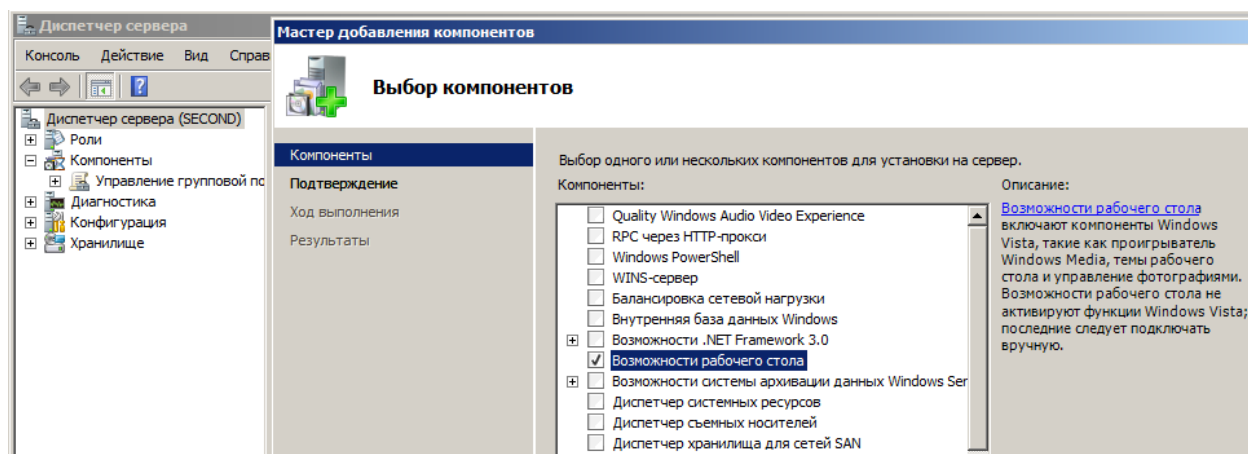


Рисунок 6.1. Окно мастера добавления компонентов

5. В Диспетчере сервера щелкнуть правой кнопкой мыши объект Роли и выбрать команду Добавить роли. Параметры новой Роли приведены в таблице 6.1.

Таблица 6.1 Параметры новой роли сервера

Параметр	Значение
Перед началом работы	Щелкнуть кнопку Далее.
Роли сервера	Выбрать роль Службы потокового мультимедиа.
Службы ролей	Выбрать три службы: Сервер Windows Media, Веб-администрирование и Агент ведения журнала (рис.6.2). Щелкнуть кнопку Далее.
Протоколы передачи данных	Оставить параметры по умолчанию. Щелкнуть кнопку Далее.
Веб-сервер/ Службы ролей	Если будет предложено, выполнить инструкции для добавления необходимых компонентов роли Веб-сервер (IIS).
Подтверждение	Просмотреть итоговые сведения о

	выбранных компонентах и щелкнуть кнопку Установить. После завершения процесса установки щелкнуть кнопку Заккрыть.
--	---

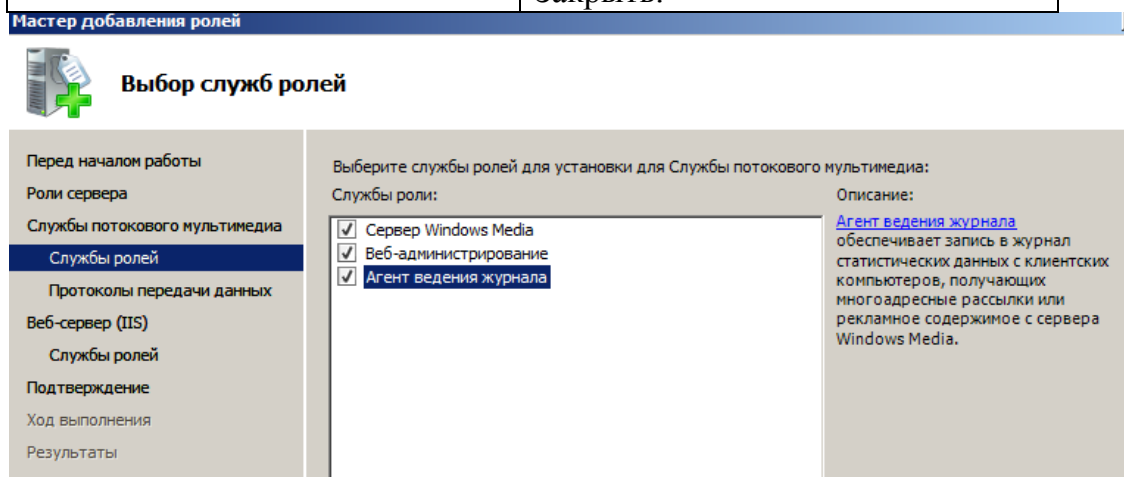


Рисунок 6.2 Выбор служб ролей

6. В Диспетчере сервера развернуть объект Роли и выбрать Службы потокового мультимедиа. Прочитать сведения в областях События, Системные службы, Службы ролей и Справка и поддержка (рис. 6.3). Заккрыть Диспетчер сервера.

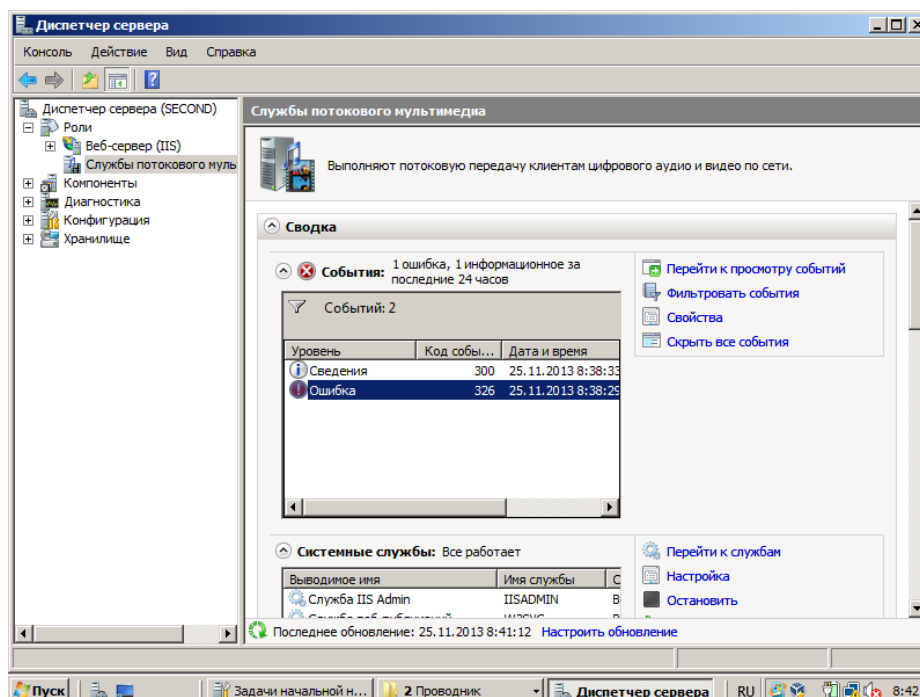


Рисунок 6.3 Службы потокового мультимедиа

7. В группе программ Администрирование запустить Службы

Windows Media. Консоль автоматически подключится к локальному серверу Windows Media. Развернуть объект сервера и просмотреть конфигурацию по умолчанию (рис. 6.4). После этого закрыть консоль Службы Windows Media.

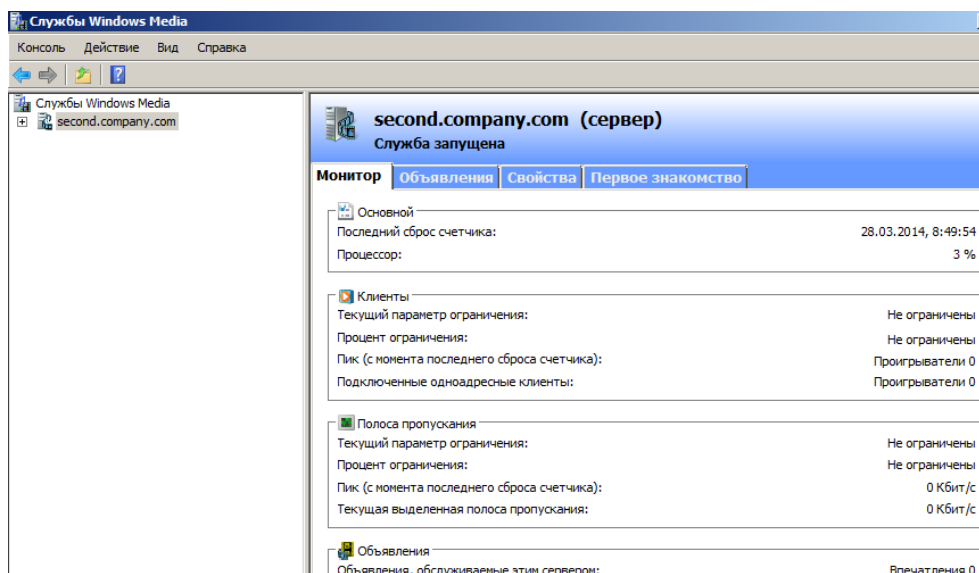


Рисунок 6.4 Конфигурация локального сервера Windows Media
(по умолчанию)

8. Открыть Проводник Windows и найти папку %System Drive%\Wmpub. Просмотреть содержимое по умолчанию, размещенное в этой папке. Папка wmpub содержит образцы файлов, которые можно использовать для тестирования.

9. Закрыть Проводник Windows и выйти с сервера.

Задание 6.2. Создать и проверить новый пункт публикации служб Windows Media. Пункт публикации будет обеспечивать доступ по запросу к нескольким образцам файлов мультимедиа, включенным в роль Службы потокового мультимедиа (рис.6.5). Затем протестировать доступ к содержимому, подключившись к видеофайлу с помощью Internet Explorer. Для выполнения тестов на локальном компьютере должен быть установлен компонент Возможности рабочего стола.

Порядок работы:

1. Войти на second как пользователь с административными привилегиями.

2. Открыть Проводник Windows и создать копию папки wmroot в каталоге %SystemDrive%\Wmpub. Задать для копии папки имя CompanyVideo. Эта папка будет играть роль корневого каталога нового пункта публикации. После этого закрыть Проводник Windows.

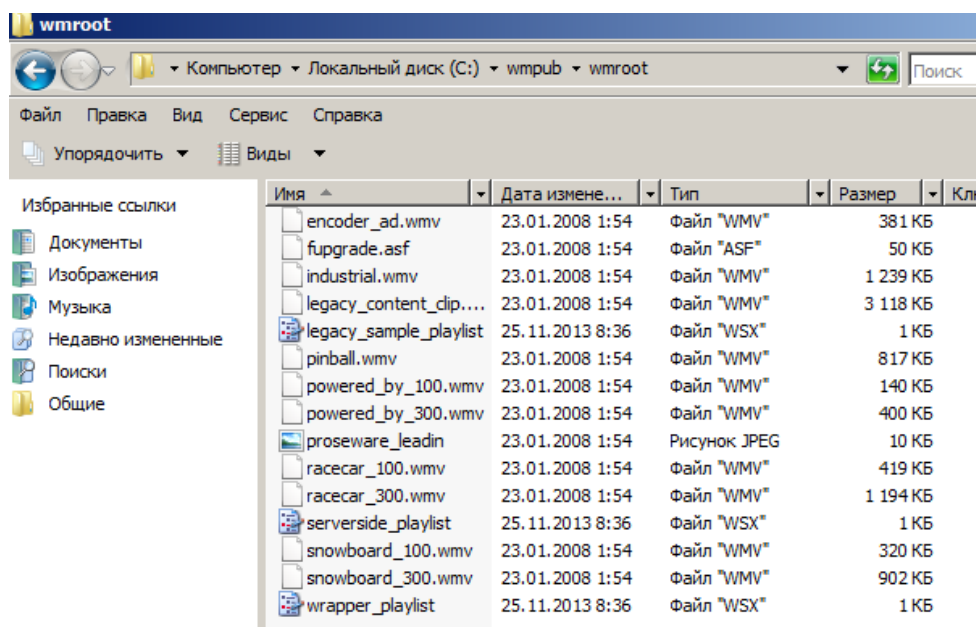


Рисунок 6.5 Образцы файлов мультимедиа для тестирования

3. В группе программ Администрирование открыть консоль Службы Windows Media.

4. Развернуть объект second, щелкнуть правой кнопкой мыши Пункты публикации и применить команду Добавить пункт публикации (мастер) (рис. 6.6).

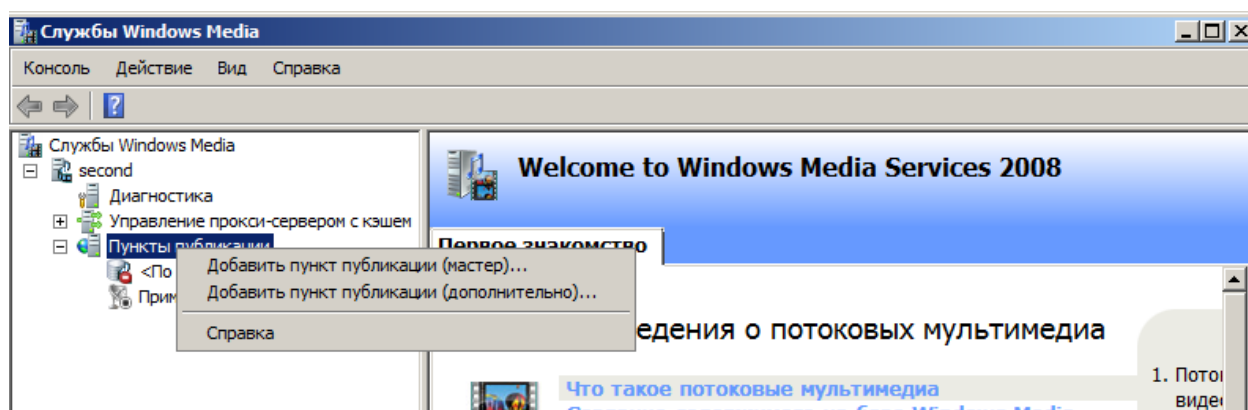


Рисунок 6.6 Добавление нового пункта публикации

5. Для запуска Мастера добавления пункта публикации щелкнуть кнопку Далее.

6. На странице Имя пункта публикации ввести имя CompanyVideo (рис. 6.7). Щелкнуть кнопку Далее.

The screenshot shows a Windows-style dialog box titled "Мастер добавления пункта публикации" (Master of adding a publication point). The main heading is "Имя пункта публикации" (Name of the publication point). Below it, a text box contains "CompanyVideo". A "Совет" (Tip) section at the bottom left lists characters to avoid: "< > \ ? % & ' # \" { } | ^ [] *". Navigation buttons at the bottom are "< Назад", "Далее >", "Отмена", and "Справка".

Мастер добавления пункта публикации

Имя пункта публикации
Имя, включенное в этот адрес URL, используется клиентом для подключения к содержимому.

Используйте понятные имена, например, тип содержимого, передаваемого с помощью потоковой передачи от пункта публикации. Введите имя.

Имя:
CompanyVideo

Совет
При названии пункта публикации не используйте следующие знаки:
< > \ ? % & ' # \" { } | ^ [] *

< Назад Далее > Отмена Справка

Рисунок 6.7 Ввод имени нового пункта публикации

7. На странице Тип содержимого выбрать параметр Файлы (мультимедиа или списки воспроизведения) и щелкнуть кнопку Далее (рис.6.8).

The screenshot shows the same dialog box, but at the "Тип содержимого" (Content type) step. It lists four options with radio buttons. The fourth option, "Файлы (мультимедиа или списки воспроизведения) в каталоге", is selected. A "Совет" (Tip) section at the bottom left explains that this type allows for streaming other content types. Navigation buttons at the bottom are the same as in the previous step.

Мастер добавления пункта публикации

Тип содержимого
Указание типа содержимого для потоковой передачи.

Выберите один из следующих параметров.

- ☐ Кодировщик (прямой поток)
- ☐ Список воспроизведения (смесь файлов и прямых потоков данных, объединяемая в непрерывный поток)
- ☐ Один файл (удобен для широкого вещания архивированного файла)
- ☒ Файлы (мультимедиа или списки воспроизведения) в каталоге (удобен для предоставления доступа для воспроизведения по запросу с помощью одного пункта публикации)

Совет
Можно также выполнять потоковую передачу другого типа содержимого с помощью диалогового окна "Расширенные возможности добавления пункта публикации".

< Назад Далее > Отмена Справка

Рисунок 6.8 Выбор типа содержимого для потоковой передачи

8. На странице Тип пункта публикации выбрать сценарий Пункт публикации по запросу. Этот сценарий позволит пользователям подключаться ко всем доступным видеофайлам (при наличии соответствующих разрешений), а также управлять воспроизведением во время приема потока. Щелкнуть кнопку Далее.

9. На странице Каталог указать путь к папке, созданной в пункте 2 (рис.6.9). Установить флажок Включить доступ к содержимому каталога с помощью подстановочных знаков. В результате пользователи смогут вручную вводить имя видеофайла для получения прямого доступа на сервере. Щелкнуть кнопку Далее.

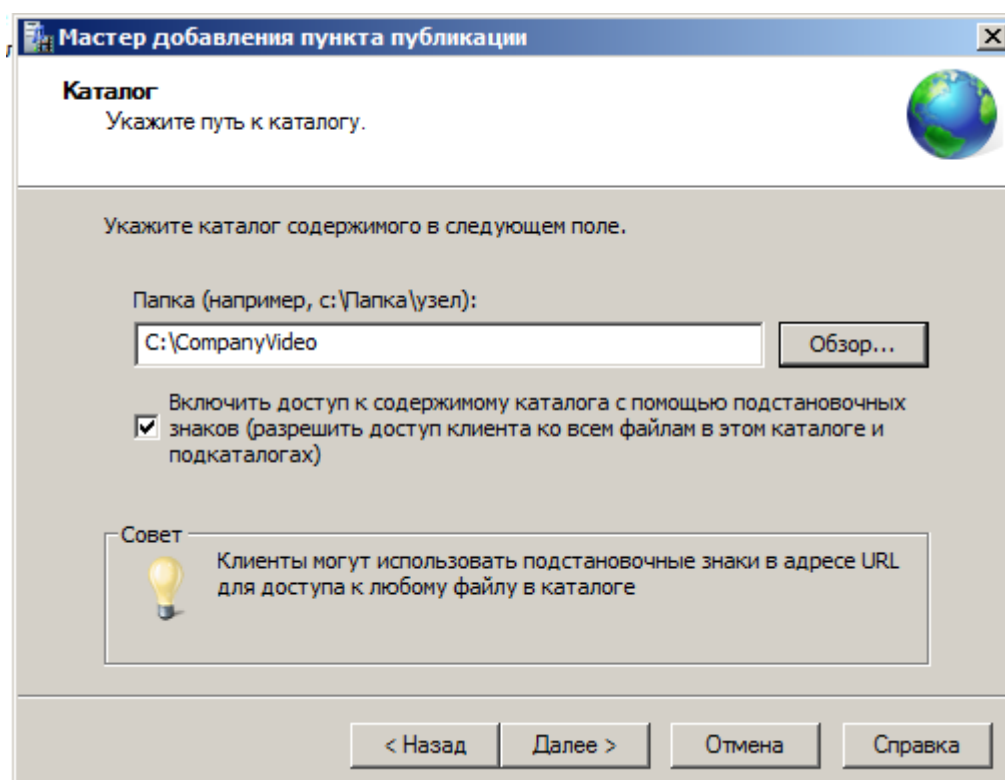


Рисунок 6.9 Указание пути к каталогу

10. На странице Воспроизведение содержимого оставить опции по умолчанию и щелкнуть кнопку Далее.

11. На странице Журнал одноадресного вещания оставить опцию по умолчанию и щелкнуть кнопку Далее.

12. На странице Сведения о пункте публикации проверить выбранные параметры и щелкнуть кнопку Далее.

13. На последней странице Мастера добавления пункта публикации оставить опции по умолчанию и щелкнуть кнопку Готово.

14. После создания пункта публикации автоматически откроется Мастер одноадресных объявлений (рис.6.10). Щелкнуть кнопку Далее.

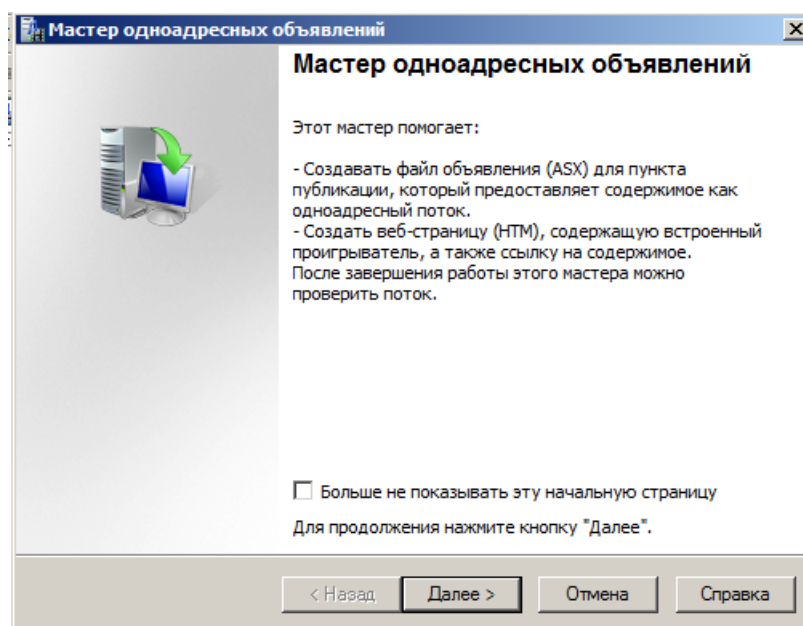


Рисунок 6.10 Окно Мастера одноадресных объявлений

15. На странице Каталог по запросу щелкнуть кнопку Обзор и выбрать файл serverside_playlist.wsx в папке, созданной в шаге 2 (рис. 6.11). Щелкнуть кнопку Далее.

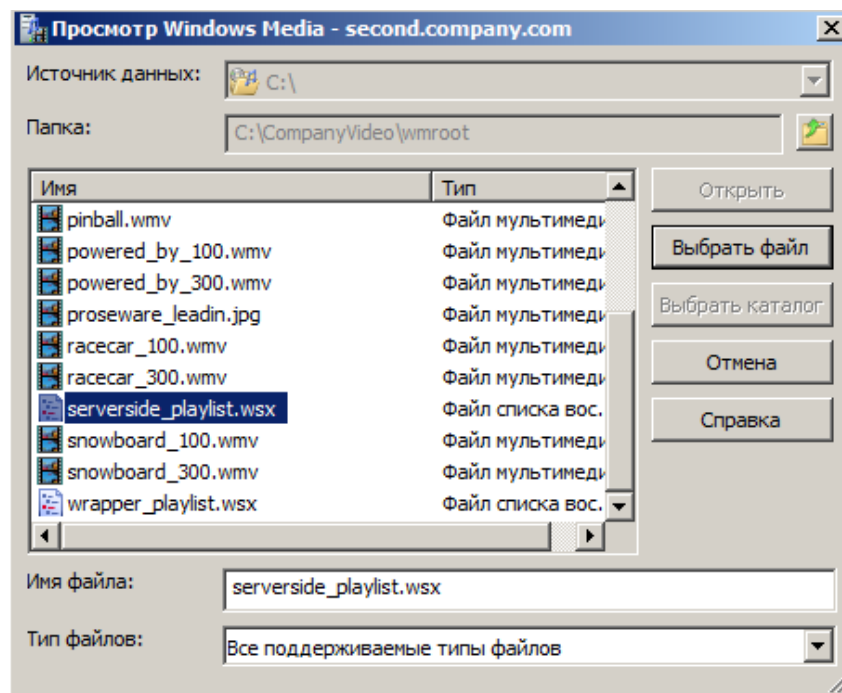


Рисунок 6.11. Выбор файла мультимедиа

16. На странице **Доступ к содержимому** запомнить URL, который можно использовать для доступа к содержимому (рис.6.12). Этот URL будет использоваться для проверки объявления. Щелкнуть кнопку **Далее**.

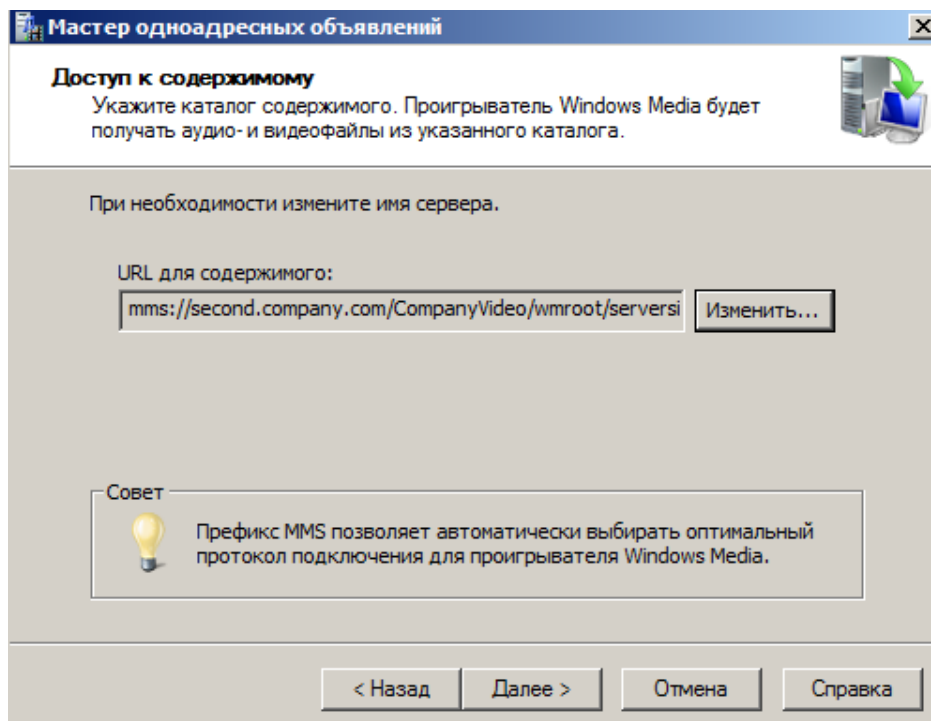


Рисунок 6.12 Доступ к содержимому

17. На странице **Сохранить параметры объявления** оставить путь по умолчанию. Он указывает размещение объекта **Default Web Site**,

установленного вместе с ролью Веб-сервер (IIS). Кроме того, установить флажок для создания веб-страницы с параметрами по умолчанию. Щелкнуть кнопку Далее.

18. На странице Изменение метаданных объявления ввести заголовок Обучение. Щелкнуть кнопку Далее.

19. На последней странице Мастера одноадресных объявлений установить флажок Проверить файлы после закрытия и щелкнуть кнопку Готово.

20. В диалоговом окне Проверочное одноадресное объявление щелкните первую кнопку Проверить, чтобы непосредственно проверить объявление (рис. 6.13). Должен запуститься Проигрыватель Windows Media и автоматически воспроизвести видеофайл из пункта публикации. После воспроизведения видео закрыть проигрыватель Windows Media.

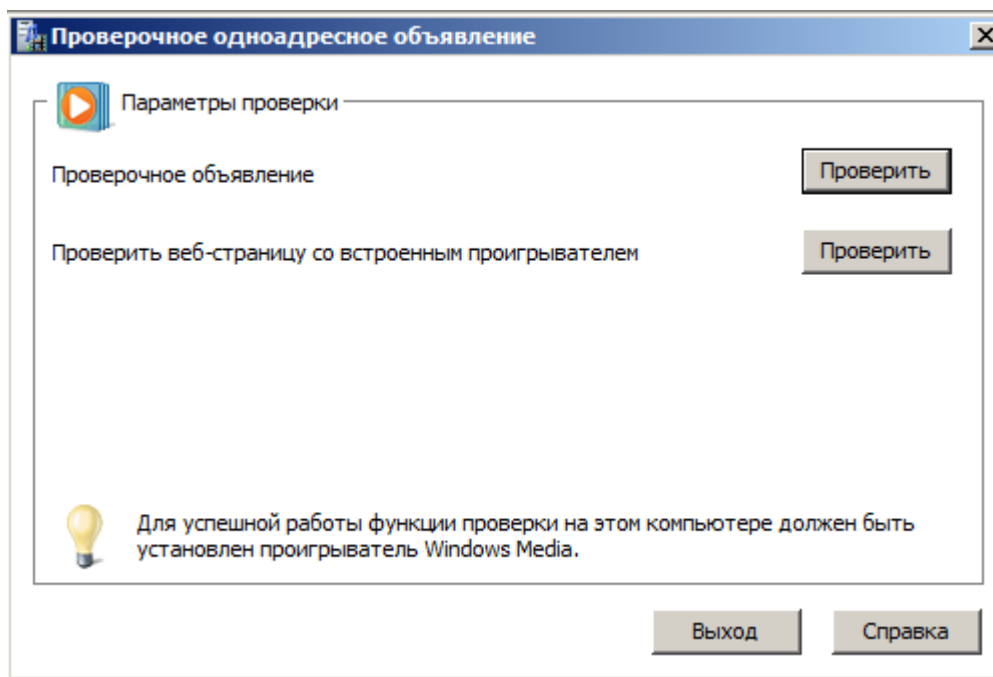


Рисунок 6.13 Проверочное одноадресное объявление

21. В диалоговом окне Проверочное одноадресное объявление щелкнуть вторую кнопку Проверить. Затем закрыть консоль Службы Windows Media.

Самостоятельная работа 6.1. Создать и проверить новый пункт публикации служб Windows Media. Пункт публикации будет обеспечивать широкополосный доступ к файлу мультимедиа. Протестировать доступ к содержимому, подключившись к видеофайлу с помощью Internet Explorer.

Заключение

Лабораторный практикум «Мультимедиа технологии» включает 6 лабораторных работ по обработке информации разного типа.

Первая лабораторная работа посвящена верстке веб-страницы с применением таблиц стилей. Для создания веб-страницы можно воспользоваться любым текстовым редактором, т.к. в основе – текстовый файл.

Вторая лабораторная работа связана созданием сценариев на языке Javascript, что делает веб-страницы более функциональными.

В третьей лабораторной работе создается музыкальный проект в LMMS - бесплатной межплатформенной альтернативе коммерческим программам типа FL Studio. Включает создание мелодий и битов, синтез и микширование звуков, и аранжировку сэмплов. Подключение MIDI клавиатуры, дружелюбный и современный интерфейс.

Четвертая лабораторная работа рассматривает приемы flash-анимации. Приведены примеры создания покадровой анимации и анимации движения.

В пятой лабораторной работе для видеомонтажа используется свободно распространяемый программный продукт Lightworks. Lightworks – это мощный видеоредактор для нелинейного монтажа. Он предоставляет широкий набор продвинутых, но простых в использовании инструментов и удобный интерфейс.

Шестая лабораторная работа связана настройкой роли Службы потокового мультимедиа в Windows Server 2008, которая обеспечивает большой выбор возможностей управления аудио- и видеосодержимым и его представления для пользователей. Кроме того, она включает административные средства и опции конфигурации, соответствующие многим техническим и бизнес-требованиям.

Список использованных источников

1. Развертывание и настройка Windows Server 2008. Учебный курс Microsoft/ Пер. с англ. – М.: Издательство «Русская Редакция». 2008. – 640 стр.
2. Флэш анимация и дизайн. <http://flash-animated.com> [Дата обращения: 24.04.2014]
3. Мир 3D.ru. <http://www.mir3d.ru/dictionary/75/884> [Дата обращения: 2.08.2014]
4. Lightworks: The professional editor for everyone. <http://www.lwks.com> [Дата обращения: 30.05.2014, 15.02.2015]
5. Центральный javascript ресурс. <http://javascript.ru> [Дата обращения: 2.03.2014]
6. Каскадные таблицы стилей CSS. <http://www.programmersclub.ru/alar-css> [Дата обращения: 08.02.2014]