

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ



ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«БАШКИРСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ»

Кафедра информатики и  
информационных технологий

**Б1.В.ДВ.06.02 РАЗРАБОТКА ИНТЕРНЕТ-РЕСУРСОВ**

**Методические указания к лабораторным работам  
и самостоятельной работе обучающихся  
Часть 1**

Направление подготовки (специальность)  
**38.03.05 БИЗНЕС-ИНФОРМАТИКА**

Профиль подготовки  
**ИНФОРМАЦИОННЫЕ СИСТЕМЫ В БИЗНЕСЕ**

Квалификация (степень) выпускника  
**БАКАЛАВР**

**Уфа 2019**

**УДК 004.7**  
**ББК 32.9**  
**М 54**

Рекомендовано к изданию методической комиссией экономического факультета (протокол № 8 от «28» марта 2019 г.)

Составитель: ст. преподаватель кафедры информатики и информационных технологий Прокофьева С.В.

Ответственный за выпуск: заведующий кафедрой информатики и информационных технологий д.т.н, доцент Беляева А.С.

г.Уфа, ФГБОУ ВО Башкирский ГАУ, кафедра информатики и информационных технологий

**Оглавление**

Анализ структуры и дизайна существующих интернет-проектов .....	4
Разработка интернет-ресурсов средствами языка HTML .....	8
Создание списков и таблиц средствами языка HTML .....	11
Создание форм средствами языка HTML .....	14
Размещение мультимедийных элементов средствами языка HTML.....	19
Использование технологии CSS при создании интернет-ресурсов .....	23
Блочная верстка интернет-ресурсов .....	28
Создание элементов дизайна интернет-ресурсов средствами графических редакторов.....	33
Специализированные программы для создания интернет-ресурсов .....	39

## Лабораторная работа 1

### Анализ структуры и дизайна существующих интернет-проектов

#### Цель лабораторной работы

Изучение структуры и дизайна существующих проектов в сети Internet.

#### Задачи работы

Определение критериев оценки структуры и дизайна web-сайтов.

#### Требования к организации рабочего места

Лабораторная работа выполняется студентами в компьютерных классах кафедры информатики и информационных технологий. Необходимое программное обеспечение: ОС Windows XP(Vista), обозреватель MS Internet Explorer или Opera, текстовый процессор MS Word, графический редактор Adobe PhotoShop.

### 1 Общие положения

#### 1.1 Основные принципы web-дизайна

Основными принципами web-дизайна являются:

- целевая аудитория;
- эргономичность;
- комфорт для глаз;
- технологичность.

### 2 Порядок выполнения работы

2.1 Проанализировать структуру и дизайн трех существующих Интернет-проектов (сайтов) одной тематической направленности из задания для самостоятельной работы (п. 5). Вариант задания указывается преподавателем. Рекомендуется осуществить просмотр в разных браузерах и с разным разрешением экрана.

2.2 Заполнить таблицу 5, проставив баллы от 1 до 5.

Таблица 5

Критерии оценки интерактивных сайтов

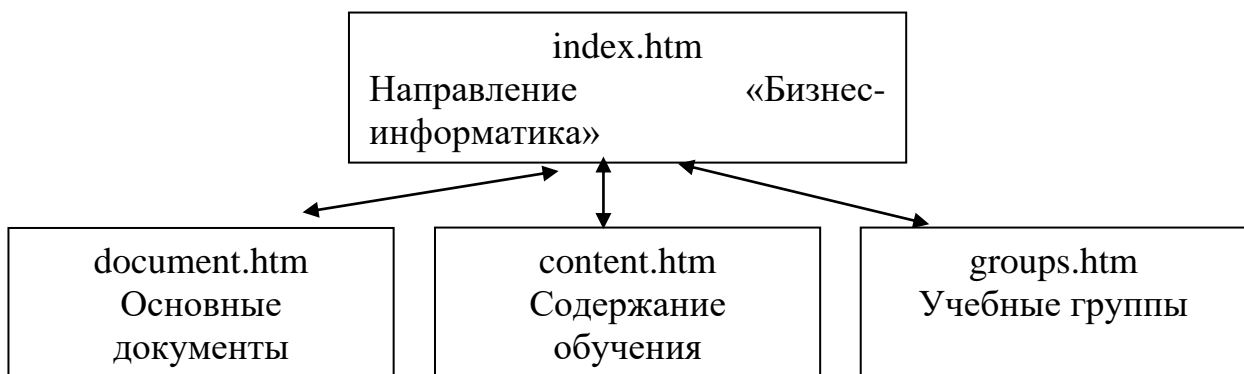
Критерий оценки	Сайт 1	Сайт 2	Сайт 3
1	2	3	4
соответствие замыслу			
<b>ДИЗАЙН</b>			

1	2	3	4
существует единый стиль для всех страниц			
графика существенно дополняет содержание			
соответствующая цветовая палитра			
графика - небольшого размера			
нет назойливой анимации, мигания			
наличие alt-текста для изображений			
<b>СОДЕРЖАНИЕ</b>			
достаточная наполненность всего сайта			
оригинальность замысла			
удобочитаемые блоки текста			
отсутствие ошибок			
соблюдение авторских прав			
наличие новостей сайта			
<b>НАВИГАЦИЯ</b>			
с главной страницы понятна основная структура сайта			
гиперссылки легко определяются			
смысловая организация страниц однотипна			
на каждой странице есть ссылка на меню			
Есть карта сайта			
<b>ТЕХНОЛОГИЧНОСТЬ</b>			
интерактивность: - тесты - голосования (с просмотром результатов) - модели			
наличие обратной связи: - адреса эл.почты - гостевой книги			

- форума (с модерированием)			
применение фреймов оправдано			
наличие флэш- , видео- , аудио- роликов			
корректность просмотра в разных браузерах			
основное разрешение: 800*600			
дополнительные сервисы			

2.3 Дополнить таблицу своими критериями (не менее пяти). Определить для этих критериев диапазон оценок (или способ вычисления оценок). Оценить сайты по этим дополнительным критериям. Сделать выводы.

2.4 Создайте проект сайта по тематике творческого задания, т. е. определить количество (от 3 до 6), взаимосвязь и тематику web-страниц. Проработать структуру, общий дизайн сайта и содержание главной страницы. Например, для темы «Направление подготовки бакалавров Бизнес-информатика» структура сайта может быть такой:



2.5 В графическом редакторе сделать эскиз главной страницы сайта:

- отметить расположение и размеры главного баннера, навигационных кнопок и других элементов страницы;
- подобрать цветовое оформление;
- определить шрифт заголовков, подзаголовков и основного текста.

2.6 Создать папку для хранения файлов сайта (например, Site), в которой собрать необходимый текстовый и графический материал (необходимо оптимизировать объем и количество графических файлов, предназначенных для размещения на страницах вашего сайта).

Примечание. Не следует использовать символы русского алфавита в именах файлов, которые включаются в состав веб-узла, так как многие операционные системы серверов сети Интернет работают только с англоязычными именами.

2.7 Показать созданный макет будущего сайта преподавателю. Обсудить элементы макета и дизайна. Внести необходимые изменения и согласовать их с преподавателем.

2.8 Оформить отчёт.

### **3 Требования к оформлению отчета**

Отчет о проделанной работе должен содержать:

- название и цель работы;
- описание последовательности выполнения работы, результаты анализа структуры и дизайна сайтов;
- письменные ответы на вопросы для самоконтроля знаний п.4 по указанию преподавателя.

### **4 Вопросы для самоконтроля знаний**

- 1) В чем заключается принцип ориентирования web-дизайна на конкретную аудиторию?
- 2) В чем заключается принцип эргономичности web-дизайна?
- 3) В чем заключается принцип того, что web-дизайн должен быть красивым?
- 4) В чем заключается требование технологичности web-дизайна?
- 5) Что такое анти-алиасинг?
- 6) Привести примеры критериев оценки Интернет-проектов.

### **5 Варианты тем заданий для самостоятельного выполнения**

- 1) Аграрные вузы.
- 2) Предприятия агропромышленного комплекса.
- 3) Города республики Башкортостан.
- 4) Сайты районных администраций РБ.
- 5) Персональные сайты.
- 6) Фермерские хозяйства.
- 7) Спорт.
- 8) Музыка.
- 9) Детские центры.
- 10) Школы.
- 11) Туристические бюро.
- 12) Базы отдыха.
- 13) Строительные организации.
- 14) Библиотеки.
- 15) Коммерческие организации.

## **Лабораторная работа 2**

### **Разработка интернет-ресурсов средствами языка HTML**

#### **Цель работы**

Научиться разрабатывать документы, предназначенные для публикации в сети Интернет.

#### **Задачи работы**

Изучение возможностей создания web-приложений. Знакомство с языком разметки гипертекстовых документов HTML.

#### **Требования к организации рабочего места**

Необходимое программное обеспечение: ОС Windows, текстовый редактор (Блокнот либо NotePad++, PHPDesigner и подобные им редакторы кода), браузеры (MS Internet Explorer, Opera или другие).

### **1 Теоретические положения**

#### **1.1 Создание Web-документов**

Размещение собственных материалов в Интернете включает два этапа: подготовку материалов и их публикацию. Подготовка материалов состоит в создании документов, имеющих формат, принятый в Интернете, то есть web-страниц, написанных на языке HTML. Публикация материалов, то есть открытие к ним доступа, осуществляется после решения организационных вопросов, связанных с получением дискового пространства на web-сервере для их размещения.

Для создания Web-документов используется язык HTML (HyperText Markup Language - язык разметки гипертекста). Гипертекст, то есть расширенный текст, включает дополнительные элементы: иллюстрации, ссылки, вставные объекты.

#### **1.2 Понятие тегов, их виды, правила записи**

Документ, который написан на языке HTML представляет собой простой текст, в который вставлены теги (флаги разметки). Теги указывают, как должен быть расположен текст на экране, в каком месте должны находиться рисунки, хранящиеся в других файлах и т.д. С помощью тегов формируются связи с другими ресурсами Интернет.

Теги - это определенные последовательности символов, заключенные между знаками < (меньше) и > (больше). Прописные и строчные буквы в написании тегов значения не имеют.

Теги бывают парные (например, <TITLE> и </TITLE>) - открывающие и закрывающие и непарные (например, <BR>). Отличительным признаком закрывающего тега является символ /.

### 1.3 Структура документа HTML

Первый тег `<!DOCTYPE>` указывает, в соответствии с каким стандартом HTML написана web-страница. В соответствие со стандартом HTML5 этот тег выглядит так:

```
<!DOCTYPE html>
```

Сам документ HTML всегда должен начинаться с тега `<HTML>` и заканчиваться соответствующим закрывающимся тегом `</HTML>`. Внутри документа выделяются два основных раздела: раздел заголовков и тело документа. Раздел заголовков содержит информацию, описывающую документ в целом, и ограничивается тегами `<HEAD>` и `</HEAD>`. В частности, раздел заголовков должен содержать название web-страницы, которое при просмотре в браузере появится в заголовке окна документа. Название web-страницы располагается между тегами `<TITLE>` и `</TITLE>`.

Например,

```
<TITLE>Моя страничка</TITLE>
```

Основное содержание размещается в теле документа, которое ограничивается парным тегом `<BODY>`.

#### Пример 1

Простейший документ HTML, содержащий все теги, определяющие структуру, может выглядеть следующим образом:

```
<HTML>
<HEAD><TITLE>Заголовок документа</TITLE></HEAD>
<BODY>
Содержание документа
</BODY>
</HTML>
```

## 2 Порядок выполнения работы

2.1 Создать папку для хранения файлов, созданных при выполнении задания. Также в этой папке рекомендуется размещать используемые картинки.

2.2 В текстовом редакторе (Блокноте или специализированном редакторе кода) создать документ HTML, который должен содержать все теги, определяющие структуру, заголовок и тело документа. Разместить на web-странице контент, разработанный в соответствии с одной из тем п.5 (вариант выдается преподавателем) или предложенной самим студентом по согласованию с преподавателем.

На странице должны присутствовать следующие элементы:

- заголовки разных уровней (теги `<h1>`, `<h2>` и т.д.);
- абзацы (`<p>`);
- переходы на новые строки (`<br>`);

- якоря и ссылки на разделы внутри страницы (<a>).  
Сохранить файл как index.htm.

*Примечание. В приложении Блокнот могут возникнуть проблемы с сохранением файла как web-страницы. Рекомендуется при сохранении в имени файла указать имя и расширение, например: index.htm, а в типе файла выбрать вариант «Все файлы». Если указана кодировка (например: <meta charset="utf-8">), то при сохранении файла в приложении Блокнот следует указать ту же кодировку.*

2.3 Протестировать созданный файл в разных браузерах (вызывать контекстное меню для файла index.htm, команда Открыть, выбрать браузер).

2.4 Создать еще несколько HTML-страниц в рамках своей темы и организовать по ним навигацию с помощью гиперссылок.

2.5 Оформить отчет.

### **3 Требования к оформлению отчета**

Отчет о проделанной работе должен содержать:

- название и цель работы;
- HTML-код web-страницы;
- скриншоты, полученные при просмотре страницы в браузере;
- письменные ответы на вопросы для самопроверки знаний п. 4 по указанию преподавателя.

### **4 Вопросы для самоконтроля знаний**

- 1) Для каких целей создаются документы HTML?
- 2) Что такое теги?
- 3) Какую структуру должен иметь документ HTML?
- 4) Как создать абзац?
- 5) Как перейти на новую строку в пределах абзаца?
- 6) Как добавить гиперссылку в документ HTML?

### **5 Варианты тем для выполнения задания**

- 1) Страница-визитка фирмы или организации.
- 2) Страница промо-акции.
- 3) Страница-учебник по основным возможностям языка HTML.
- 4) Страница-портфолио специалиста.

## Лабораторная работа 3

### Создание списков и таблиц средствами языка HTML

#### Цель работы

Научиться разрабатывать документы, предназначенные для публикации в сети Интернет.

#### Задачи работы

Изучение возможностей создания web-приложений со сложным форматированием.

#### Требования к организации рабочего места

Необходимое программное обеспечение: ОС Windows, текстовый редактор (Блокнот либо NotePad++, PHPDesigner и подобные им редакторы кода), браузеры (MS Internet Explorer, Opera или другие).

### 1 Теоретические положения

#### 1.1 Создание таблиц

Таблицы в документах HTML используются не только для показа данных в строках и столбцах, но и широко применяются для точного размещения элементов web-страниц (пунктов меню, иллюстраций и т.д.). Таблица вставляется в документ HTML парным тегом **<table>**.

Таблица может содержать заголовок, определяемый парным тегом **<caption>**. У тега **<caption>** всего один параметр, определяющий положение заголовка таблицы - **ALIGN**. Он может принимать два значения: **top** - над таблицей и **bottom** - под таблицей.

Например, теги `<CAPTION ALIGN="top">Список абонентов </CAPTION>` определяют заголовок с текстом “список абонентов” и расположат его над таблицей.

Строки таблицы задаются при помощи парных тегов **<TR>**. Закрывающие теги **</TR>** можно опускать.

#### 1.2.7 Списки

Для создания **нумерованных списков** используются теги **<ol>** и **<li>**. Тегом **<ol>** отмечается начало/окончание всего списка. Тегом **<li>** обозначают начало/конец отдельного элемента списка.

По умолчанию элементы списка нумеруются по порядку 1, 2, 3... При помощи атрибута **TYPE** можно изменить стиль нумерации (таблица 1).

Таблица 1 - Обозначения в нумерованных списках

Значение TYPE	Нумерация
<b>A</b>	<b>A, B, C..</b>
<b>a</b>	<b>a, b, c..</b>
<b>I</b>	<b>I, II, III..</b>
<b>i</b>	<b>i, ii, iii..</b>

Если дополнить уже существующий список новыми значениями, то браузер автоматически пересчитает его.

При помощи атрибутов `start` и `value` можно изменить порядок нумерации списка. **start** - служит для задания начального номера списка, отличного от 1. **value** - дает возможность назначить произвольный номер любому элементу списка.

Для создания **маркированных списков** применяют теги `<ul>` и `<li>`. Тегом `<ul>` отмечается начало/окончание всего списка. Тегом `<li>` обозначают начало/конец отдельного элемента списка. По умолчанию элементы списка маркируются черным кружочком. При помощи атрибута **type** можно изменить стиль маркирования.

В пределах одного списка можно использовать различную маркировку элементов списка.

## 2 Порядок выполнения работы

2.1 Создать папку для хранения файлов, созданных при выполнении задания. Также в этой папке рекомендуется размещать используемые картинки.

2.2 В текстовом редакторе (Блокноте или специализированном редакторе кода) создать документ HTML, который должен содержать все теги, определяющие структуру, заголовок и тело документа. Разместить на web-странице контент, разработанный в соответствии с одной из тем п.5 (вариант выдается преподавателем) или предложенной самим студентом по согласованию с преподавателем.

На странице должны присутствовать следующие элементы:

- таблица (`<table>`);
- список (`<ul>` или `<ol>`).

Сохранить файл как `table.htm`.

Протестировать созданный файл в разных браузерах (вызывать контекстное меню для файла `index.htm`, команда Открыть, выбрать браузер).

2.3 Оформить отчет.

## 4 Требования к оформлению отчета

Отчет о проделанной работе должен содержать:

- название и цель работы;
- HTML-код web-страницы;
- скриншоты, полученные при просмотре страницы в браузере;
- письменные ответы на вопросы для самопроверки знаний п. 4 по указанию преподавателя.

## 4 Вопросы для самоконтроля знаний

- 1) Для каких целей создаются документы HTML?
- 2) Что такое теги?

- 3) Какую структуру должен иметь документ HTML?
- 4) Как создать абзац?
- 5) Как перейти на новую строку в пределах абзаца?
- 6) Как добавить гиперссылку в документ HTML?
- 7) Как в документ HTML вставить рисунок?
- 8) Как создать таблицу?

### **5 Варианты тем для выполнения задания**

- 5) Страница-визитка фирмы или организации.
- 6) Страница промо-акции.
- 7) Страница-учебник по основным возможностям языка HTML.
- 8) Страница-портфолио специалиста.

## Лабораторная работа 4

### Создание форм средствами языка HTML

#### Цель работы

Научиться разрабатывать документы, предназначенные для публикации в сети Интернет.

#### Задачи работы

Изучение возможностей создания web-приложений. Знакомство с языком разметки гипертекстовых документов HTML.

#### Требования к организации рабочего места

Необходимое программное обеспечение: ОС Windows, текстовый редактор (Блокнот либо NotePad++, PHPDesigner и подобные им редакторы кода), браузеры (MS Internet Explorer, Opera или другие).

### 1 Теоретические положения

#### 1.1 Создание формы

HTML предоставляет возможность для организации диалога между пользователем и сервером, то есть позволяет организовать ввод информации и передачу ее на сервер для обработки. Эту функцию берут на себя формы. Форма представляет собой панель диалога, содержащую строки ввода, кнопки, списки и так далее.

Форма начинается с тега **<form>**, а заканчивается тегом **</form>**. Тег **<form>** имеет следующие параметры:

- **name** – имя формы, которое может быть использовано для обработки средствами языка JavaScript.
- **action** – указывает действие, которое будет произведено с введенными данными.
- **method** – способ отправки данных серверу.

С помощью параметра **action** можно перенаправить введенные данные на собственный почтовый ящик следующим образом:  
`action="mailto:электронный_адрес"`

Например, `action="mailto:xxx_mail@mail.ru"`

Для параметра **method** чаще всего используют одно из двух значений: **GET** и **POST**. Во втором случае данные будут посланы в скрытом виде.

Для описания ряда элементов формы используют тег **<input>**. Его параметр **type** определяет создаваемый элемент. Параметр **name** задает уникальное имя элемента. Это имя не должно повторяться для других элементов в пределах одной HTML-страницы.

## 1.2 МЕТА - теги: описание и правила создания

МЕТА-теги используются в пределах заголовка страницы и предназначены для того, чтобы включить любую полезную информацию, не определенную другими HTML тегами. Такая информация может быть извлечена серверами / клиентами для использования в идентификации, индексации и создании каталогов Ваших страниц.

МЕТА-тег не виден в браузере при просмотре страницы, но значительно увеличивает размер страницы. Если нужно, сайт находился в числе первых, выдаваемых поисковыми машинами при запросах пользователей, то нельзя игнорировать использование МЕТА-тега. Кроме того, необходимо правильно заполнять содержимое МЕТА-тегов.

## 2 Порядок выполнения работы

2.1 Выполнить следующий пример, создав web-страницу в редакторе Блокнот:

```
<HTML> <HEAD>
<TITLE> Пример формы </TITLE>
</HEAD>
<BODY>
<P ALIGN=LEFT>
<FORM      NAME="talk"          ACTION="mailto:xxx_main@mail.ru"
METHOD="POST">
<HR>
<CENTER> Пожалуйста, заполните мою анкету: </CENTER>
<HR>
Ваше имя:
<INPUT TYPE="TEXT" NAME="name" SIZE="35">
Ваш e-mail:
<INPUT TYPE="TEXT" NAME="e-mail" SIZE="35"><BR>
Ваш возраст (лет):
<INPUT TYPE="RADIO" NAME="years" VALUE="o-le15"> Младше 15
<INPUT TYPE="RADIO" NAME="years" VALUE="o-15-25" > 15-25
<BR>
Место вашего проживания:
<SELECT NAME="city">
<OPTION SELECTED> Уфа </OPTION>
<OPTION> Москва </OPTION>
<OPTION> С.-Петербург </OPTION>
<OPTION> Другое </OPTION>
</SELECT>
Вам понравился мой сайт?
<INPUT TYPE="CHECKBOX" NAME="mark" VALUE="yes"><BR><BR>
Ваши пожелания:<BR>
<TEXTAREA NAME="comment" ROWS=5 COLS=50>
```

Теперь осваивай CGI:))

```
</TEXTAREA>
```

```
<BR><INPUT TYPE="SUBMIT" VALUE="Послать запрос">
```

```
</FORM></P>
```

```
</BODY>
```

```
</HTML>
```

Просмотреть страницу в браузере. Проанализировать расположение и вид элементов формы.

2.2 Внести изменения в созданный в п.2.1 документ:

- добавить три радиокнопки для возрастов 26-35, 36-60, старше 60;
- радиокнопку «15-25» сделать выделенной по умолчанию (см. п. 1.2);
- в список «Место вашего проживания» добавить Екатеринбург;
- в конце формы разместить кнопку для восстановления значений по умолчанию с надписью «Очистить форму» (см. п.1.2);
- перед кнопкой SUBMIT добавить текст «Введите ваш пароль на сайте» и поле для ввода пароля (см. п 1.2).

2.3 В редакторе Блокнот разработать новый документ HTML с формой по варианту из п. 5.1 (задается преподавателем).

Предусмотреть кнопки для отправки данных (на свой электронный почтовый ящик) и для очистки формы. Для более удачного взаиморасположения элементов формы использовать таблицу. Выполнить цветовое оформление страницы. Результат просмотреть в браузере.

2.4 Добавить в документ META-теги, с помощью которых:

- определить кодировку;
- указать список ключевых слов (согласно теме из п. 5.1);
- описать содержимое страницы (согласно теме из п. 5.1).

2.5 Добавить META-тег, выполнив задание из п. 5.2. Вариант задания указывается преподавателем.

2.6 Оформить отчет.

### **3 Требование к оформлению отчета**

Отчет о проделанной работе должен содержать:

- название и цель работы;
- вид в браузере (скриншот) и HTML-код документа, полученный в результате выполнения п.п. 2.1 и 2.2;
- вид в браузере (скриншот) и HTML-код документа, полученный в результате выполнения п.п. 2.3 - 2.5;
- письменные ответы на вопросы для самопроверки знаний п. 4 по указанию преподавателя.

### **4 Вопросы для самоконтроля знаний**

4.1 Что такое форма?

4.2 Как создать форму на web-странице?

4.3 Какие параметры могут быть у тега <FORM>?

- 4.4 Как создать текстовое поле?
- 4.5 Как создать раскрывающийся список?
- 4.6 Как создать группу радиокнопок?
- 4.7 Как создать текстовую область?
- 4.8 Как создать флажок?
- 4.9 Как создать поле для ввода пароля?
- 4.10 Для чего предназначено скрытое поле и как его создать?
- 4.11 Как создать кнопку для отправки данных?
- 4.12 Как создать кнопку для очистки формы?
- 4.13 Для чего предназначены meta-теги?

## **5 Задания для самостоятельного выполнения**

### **5.1 Разработка форм**

5.1.1 Форма-анкета для абитуриента вуза (фамилия, имя, отчество абитуриента, год и дата рождения, год окончания школы, оценки по отдельным предметам, участие в олимпиадах, на какую специальность поступает, дополнительная информация).

5.1.2 Форма-анкета для посетителя сайта (имя посетителя, никнейм, e-mail, ICQ, Skype, возраст, пол, страна, адрес собственного ресурса, оценка сайта, пожелания владельцу сайта).

5.1.3 Форма регистрации на форуме (фамилия, имя, отчество, год и дата рождения, пол, ник на форуме, пароль, подтверждение пароля, часовой пояс, подпись под сообщениями, соглашение с правилами).

5.1.4 Форма для оформления покупки в Интернет-магазине (фамилия, имя, отчество покупателя, выбор формы доставки, выбор формы оплаты, адрес, сотовый телефон, подтверждение покупки, дополнительная информация, пароль).

5.1.5 Форма для тестирования по дисциплине «Мировые информационные ресурсы» (фамилия, имя студента, группа, тесты разного вида: один из многих, много из многих, соответствие, да/нет, поле ввода).

5.1.6 Форма-анкета для выпускника вуза (фамилия, имя, отчество выпускника, год поступления в вуз, год окончания вуза, работает ли по специальности, место работы, e-mail и телефон для связи, пожелания сегодняшним студентам, дополнительная информация).

5.1.7 Форма для поиска документов в базе данных (категория, ключевые слова для поиска, дополнительные условия поиска: формат файла, язык, «с включением слов», «с точной фразой», «с отсутствием слов»).

5.1.8 Форма для заполнения сведений о кредите в банке (фамилия, имя, отчество клиента, дата рождения, номер паспорта, адрес постоянного проживания, фамилия сотрудника, вид кредита, филиал банка, сумма кредита, примечание).

5.1.9 Форма выдачи книг библиотекарем (фамилия библиотекаря, номер читательского билета, тематический раздел, УДК, название книги, автор, срок возврата, продление срока, примечание).

5.1.10 Форма для заполнения сведений о клиенте гостиницы (фамилия, имя, отчество клиента, номер паспорта, адрес постоянного проживания, дата приезда, дата отъезда, номер комнаты, оплата, фамилия администратора, примечание).

5.1.11 Форма-анкета службы занятости (фамилия, имя, отчество соискателя, возраст, e-mail, контактный телефон, адрес, семейное положение, специальность, учебное заведение, год окончания, стаж работы, последнее место работы и должность, профессиональные навыки и умения, дополнительная информация).

5.1.12 Форма для заказа билета на поезд (откуда, куда, дата поездки, количество взрослых, количество детей, количество младенцев до 5 лет, категория вагона, ФИО, дата рождения, паспорт, e-mail, телефон, адрес доставки, дополнительная информация).

5.1.13 Форма для бронирования авиабилета (откуда, куда, туда и обратно (флажок), дата вылета туда, время, дата вылета обратно, время, количество взрослых, количество детей 2 – 11 лет, количество детей до 2 лет без места, количество детей до 2 лет с местом, класс, авиакомпания).

5.1.14 Форма регистрации на социальном ресурсе (имя, фамилия, пол, дата рождения, страна проживания, город, загрузить фото, e-mail, секретный вопрос, ответ на секретный вопрос, логин, пароль, подтверждение пароля).

## **5.2 В МЕТА-теге указать следующую информацию о документе.**

5.2.1 Сведения об авторе.

5.2.2 Кому принадлежат авторские права.

5.2.3 E-mail издателя страницы.

5.2.4 Адрес сайта (указать произвольный).

5.2.5 Программное средство, с помощью которого была создана данная страница.

5.2.6 Чтобы страница перечитывала сама себя через 60 секунд.

5.2.7 Чтение другой страницы через 5 секунд (указать URL страницы в Интернете, например <http://mir2010.ucoz.ru>).

5.2.8 Запрет кеширования страницы на компьютере пользователя.

5.2.9 Возможность индексирования данного документа.

5.2.10 Возможность индексирования всех документов, на которые есть ссылки в данном HTML-файле.

5.2.11 Запрет индексирования данного документа

5.2.12 Запрет индексирования всех документов, на которые есть ссылки в данном HTML-файле.

5.2.13 Возможность индексирования данного документа и запрет индексирования всех документов, на которые есть ссылки в данном HTML-файле.

5.2.14 Запрет индексирования данного документа и возможность индексирования всех документов, на которые есть ссылки в данном HTML-файле.

## **Лабораторная работа 5**

### **Размещение мультимедийных элементов средствами языка HTML.**

#### **Цель работы**

Научиться разрабатывать документы, предназначенные для публикации в сети Интернет.

#### **Задачи работы**

Изучение возможностей создания web-приложений. Знакомство с языком разметки гипертекстовых документов HTML.

#### **Требования к организации рабочего места**

Необходимое программное обеспечение: ОС Windows, текстовый редактор (Блокнот либо NotePad++, PHPDesigner и подобные им редакторы кода), браузеры (MS Internet Explorer, Opera или другие).

### **1 Общие положения**

**Внедренный элемент веб-страницы** – данные, хранящиеся в отдельных от веб-страницы файлах.

**Кодеки** – алгоритмы, используемые для кодирования и декодирования определенных типов звуковых и видеопотоков, для их воспроизведения.

**Потоковая мультимедиа** – мультимедиа, непрерывно получаемая от провайдера потокового вещания (например, телевидение).

#### **1.1 Работа с графикой**

Современными веб-браузерами поддерживаются не все существующие графические форматы, поэтому не всякое изображение может быть размещено в рамках веб-страницы.

Различают форматы растровых и векторных изображений. Растровые изображения хранятся в файлах с расширениями *jpg*, *gif*, *bmp*, *tiff*, *png*, *psd*. Векторные – *swf*, *cdr*, *max*, *ai*.

Из вышеперечисленных форматов хранения графики, в веб-разработке используются *JPEG*, *PNG* и *GIF* для растровой графики и *SWF* для векторной. Все три формата для растровой графики используют сжатие графической информации.

#### **1.2 Добавление графики на веб-страницу**

Структура мультимедийной информации принципиально отличается от структуры текстовой, поэтому напрямую мультимедиа не может быть описана в *html*-коде. Вся необходимая разработчику мультимедиа содержится в отдельных файлах, ссылки на которые, в виде соответствующих тегов прописываются в *html*-коде.

Для добавления изображения средствами *html* используется непарный тег `<img>`, содержащий обязательный атрибут `src`, указывающий расположение графического файла для отображения.

К числу необязательных атрибутов `<img>` относятся:

- align – задает тип выравнивания изображения;
- alt – задает текст, отображаемый в случае, если картинка не загрузилась;
- border – определяет толщину рамки вокруг изображения;
- height – задает высоту изображения;
- hspace – задает величину горизонтального отступа от изображения до ближайшего контента;
- ismap – определяет, является ли изображение картой (т.е. к различным частям изображения "привязаны" разные ссылки);
- vspace – задает величину вертикального отступа от изображения до ближайшего контента;
- width – задает ширину изображения;
- usemap – определяет ссылку на <map>, содержащий координаты клиентской карты - изображения.

<img> является встроенным тегом, т.е. он не может использоваться вне блока. Пример: <p></p>

### 1.3 Основы работы с видео и звуком

В спецификации HTML5 предусмотрено два тега для работы с аудио и видео соответственно: <audio> и <video>.

Данные теги являются компонентами собственной среды браузера. Это означает, что не используется никаких сторонних средств для воспроизведения мультимедиа информации, что, во-первых, повышает безопасность, во-вторых, за счет более тесной интеграции, позволяет обходиться меньшим количеством аппаратных ресурсов для воспроизведения мультимедиа, и, в-третьих, позволяет избежать ряда проблем отображения информации (визуальное пересечение с остальным контентом).

Помимо этого, использование <audio> и <video> позволяет организовать управление из веб-сценариев.

Также существуют и недостатки указанных тегов, спецификация HTML5 поддерживает далеко не все кодеки, строго говоря, из спецификации исключены все упоминания об обязательной поддержке каких-либо кодеков.

Частичным решением проблемы кодеков может служить использование элемента <source>, позволяющий объявить несколько источников мультимедиа, из которых браузер выберет наиболее подходящий.

Пример:

```
<audio>
  <source src="sound1.ogg">
  <source src="sound1.mp3">
</audio>
```

### 1.4 Вставка аудио и видеороликов

Для вставки аудиоролика в HTML5 используется парный тег <audio>. Текст, помещающийся внутри данного тега будет отображаться в браузерах, не поддерживающих <audio>.

Базовый код для вставки аудио:

```
<audio src="sound1.mp3"> </audio>
```

Или, для обеспечения универсального воспроизведения различными браузерами:

```
<audio>
  <source src="sound1.ogg">
  <source src="sound1.mp3">
</audio>
```

Тег <audio> содержит следующие атрибуты:

- `autoplay` – при его добавлении, воспроизведение файла начинается сразу же после загрузки страницы;
- `controls` – добавляет панель управления к аудио;
- `loop` – воспроизведение видео повторяется с начала, после его завершения;
- `preload` – используется для загрузки файла вместе с загрузкой самой страницы, игнорируется, если использован `autoplay`;
- `src` – задает путь к файлу для воспроизведения.

Пример:

```
<audio autoplay controls src="1.mp3">
  Тег <audio> не поддерживается
</audio>
```

Видео добавляется аналогичным образом:

```
<video src="video1.avi"> </video>
```

Атрибуты тега <video>:

- `autoplay` – при его добавлении, воспроизведение файла начинается сразу же после загрузки страницы;
- `controls` – добавляет панель управления к видео;
- `height` – задает высоту области, для воспроизведения видео.
- `loop` – воспроизведение аудио повторяется с начала, после его завершения;
- `poster` – указывает путь к изображению, пока видео не воспроизводится, или недоступно;
- `preload` – используется для загрузки видеофайла вместе с загрузкой самой страницы, игнорируется, если использован *autoplay*;
- `src` – задает путь к файлу для воспроизведения.
- `width` – задает ширину области, для воспроизведения видео.

## 2 Порядок выполнения работы

2.1 В папку с web-сайтом, созданным на предыдущих лабораторных поместить изображения, видео и аудио-файлы (скопировать из сети Интернет по адресу, указанному преподавателем; также можно использовать самостоятельно разработанный контент).

2.1 На страницах, созданных на предыдущих лабораторных работах разместить картинки (не менее трех), видео (не менее одного), звуковое сопровождение (не менее одного).

2.2 Составить отчет по проделанной работе.

### **3 Требования к оформлению отчета**

Отчет по лабораторной работе включает в себя:

- цель и задачи работы;
- HTML-код страниц с мультимедиа объектами;
- ответы на вопросы для самоконтроля знаний (п.4) по указанию преподавателя.

### **4 Вопросы для самоконтроля знаний**

- 1) Как в документ HTML вставить рисунок?
- 2) Какие растровые форматы изображений могут быть размещены на web-страницах?
- 3) Какие векторные форматы изображений могут быть размещены на web-страницах?
- 4) Для чего предназначен атрибут align тега img?
- 5) Для чего предназначен атрибут alt тега img?
- 6) Как в документ HTML добавить аудио-сопровождение?
- 6) Как в документ HTML вставить видео-файлы?

## Лабораторная работа 6

### Использование технологии CSS при создании интернет-ресурсов

#### Цель работы

Научиться разрабатывать документы, предназначенные для публикации в Internet.

#### Задачи работы

Изучение возможностей создания web-приложений с использованием языка CSS.

#### Требования к организации рабочего места

Необходимое программное обеспечение: ОС Windows, текстовый редактор Блокнот (либо NotePad++, PHPDesigner и подобные им редакторы кода), браузер (MS Internet Explorer, Opera или другой).

### 1.1 Каскадные таблицы стилей

Детальное форматирование документов HTML возможно при использовании технологии каскадных таблиц стилей (Cascade Style Sheet — CSS). *Каскадные таблицы стилей* являются следствием дальнейшего развития HTML и дают возможность перейти на следующий уровень представления информации. Таблицы стилей позволяют разделить смысловое содержимое странички и его оформление. Как правило, таблицу стилей нужно написать всего один раз при создании сайта. К тому же таблица стилей может быть единой для целого сайта. И, следовательно, не нужно будет повторять одни и те же описания стилей на каждой из страниц. Размещение всей стилевой информации в одном внешнем файле открывает и другие полезные возможности: изменив содержимое только одного стилевого файла, можно в считанные секунды сменить весь дизайн сайта.

Таблицу стилей можно рассматривать как шаблон, который управляет форматированием тегов HTML в Web-документе. Таблица стилей представляет собой набор правил форматирования элементов HTML. Любое правило каскадных таблиц стилей состоит из двух частей: селектора и определения. Селектором может быть любой тег HTML, для которого правило определяет, каким образом необходимо его форматировать. Определение состоит также из двух частей: свойства и значения, разделенных знаком двоеточия.

Рассмотрим правило

```
h3 {color: silver}
```

Селектором является элемент h3, а определение, заключенное в фигурные скобки, задает значение цвета шрифта (свойство color) серым цветом (значение silver).

Таблица стилей связывается с документами различными способами:

– **связывание** — используется одна таблица стилей для форматирования одной или нескольких страниц HTML; **описания стилей вынесены во внешний файл**. Диапазон воздействия простирается на все

файлы, в которые включено описание. В случае, если потребуется изменить внешний вид сайта, достаточно будет скорректировать лишь этот файл.

Создается стилевой файл с описанием всех нужных классов (например, с именем `mystyle.css`):

```
.header { text-align : center; font-size : 27pt; }
.red { color :red; }
p { text-align : center; font-size : 12pt; }
```

А потом ссылка на него внедряется в документ при помощи тега `<link>`:

```
<head> .... <link rel="stylesheet" type="text/css"
href="css/mystyle.css" title="MyStyleSheet"> .... </head>
```

Это самый удобный способ, и для основной таблицы стилей рекомендуется пользоваться именно им.

– **внедрение** — задаются правила таблицы стилей непосредственно в самом документе. Действие распространяется на всю страничку. Определение стилей происходит при помощи классов, которые представляют собой списки с определением всех необходимых параметров оформления.

При использовании этого метода описание стилей необходимо разместить в секции заголовка:

```
<head> ....
<style type="text/css">
.header {
text-align :center;
font-size : 27pt; }
.red { color : red; }
</style>
</head>
```

Теперь эти стили можно применять в любом месте html-кода. Для этого используется следующая конструкция:

```
<p class=header>Этот текст написан стилем header<p>
<p class=red>Этот текст написан красным цветом<p>
```

Кроме определения новых классов можно переопределять стандартные теги. Например, тег `<p>`:

```
<style type="text/css">
p { text-align : center; font-size :12pt; }
</style>
```

Теперь весь текст, заключенный в теги `<p></p>`, будет выглядеть так, как определено данным стилем. Это очень удобно и позволяет легко адаптировать уже существующие странички к использованию стилей. Кроме того, это несколько уменьшает объем файла за счет отсутствия лишних атрибутов `class`.

– **встраивание** — правила задаются в теге документа и позволяют изменять форматирование конкретных элементов страницы:

```
<p style="color:red; text-align:center;">
Этот текст переопределен стилем
</p>
```

При помощи атрибута style можно определить нужные стилевые параметры в любом теге. Стилль действует в пределах одного тега. Но насколько вырастет размер файла, и насколько неудобно будет его исправлять, если указывать стилль у каждого тега.

## 1.2 Каскадность стилей

Каскадность заключается в том, что стили могут переопределяться. Приведенный выше список способов внедрения стилей соответствует порядку переопределения. Нижерасположенный способ может переопределять вышерасположенный.

Например, мы определили во внешнем стилевом файле, что текст в теге <p> должен быть написан при помощи шрифта высотой 10 пунктов. Но если в заголовке странички мы дополнительно укажем, что тот же текст в теге <p> должен быть написан шрифтом в 12 пунктов, то текст будет выведен именно таким кеглем - т.е. стилль в заголовке странички переопределил стилль во внешнем файле.

## 2 Порядок выполнения работы

Рекомендуется все полученные в результате работы файлы, а также используемые картинки размещать в одной, специально созданной папке. Работа выполняется в приложении Блокнот (также вместо Блокнота могут использоваться специализированные редакторы кода: Блокнот++, PHPDesigner и подобные им). Для просмотра результата используется браузер.

### 2.1 Связывание таблицы стилей с документом

Выполнить пример 1 (для создания текста использовать Блокнот)

Пример 1.

**Листинг файла Tab11.css.**

```
BODY {background-color:yellow }
```

```
H3 {text-align:center }
```

```
P {color: #0000FF }
```

Пояснение. В файле с именем Tab11.css задается таблица каскадных стилей, определяющая формат того документа, с которым она будет связана: Правила предписывают определить желтый цвет фона для всего документа, заголовков третьего уровня расположить по центру, цвет шрифта в документе должен быть синим. В качестве значений цветов можно использовать зарезервированные слова, а также цветовую модель RGB.

### Листинг документа с форматированием.

```
<HTML><HEAD><title>Связывание документов с таблицей стилей</title>
```

```
<LINK REL="STYLESHEET" type="text/css" href="Tab11.css">
```

```
</HEAD><BODY>
```

```
<h3> JavaScript </h3>
```

Язык сценариев JavaScript предназначен для создания интерактивных HTML-документов. Основой языка является понятие объекта.

```
<h3> DHTML </h3>
```

Динамический HTML (DHTML) является дальнейшим развитием языка HTML, добавляет к HTML набор свойств, позволяющих определять внешний вид документа и управлять слоями. При создании web-страницы с помощью DHTML у разработчика появляются дополнительные возможности.

<h3>CSS</h3>

CSS (англ. Cascading Style Sheets — каскадные таблицы стилей) — технология описания внешнего вида документа, написанного языком разметки. CSS используется создателями веб-страниц для задания цветов, шрифтов, расположения и других аспектов представления документа.

2.2 Внести изменения в файл Tab11.css, в соответствии с вариантом из п. 5. Вариант выполнения задается преподавателем.

2.3 Оформить отчет.

#### **4 Вопросы для самоконтроля знаний**

- 1) Что собой представляет каскадная таблица стилей?
- 2) Как формируется правило таблицы стилей?
- 3) Как таблица стилей связывается с документами HTML?
- 4) Назначение слоев на web-страницах.
- 5) С помощью какого тега создаются слои?

#### **5 Варианты задания для самостоятельного выполнения**

Внести изменения в файл Tab11.css, установив цвет фона, шрифт и выравнивание заголовков 3 уровня в соответствии с вариантом.

5.1 Цвет фона: GREY

Цвет шрифта абзаца RED

Цвет шрифта заголовков 3 уровня: YELLOW

Выравнивание заголовков 3 уровня: по правому краю

Выравнивание текста абзаца: по ширине

5.2 Цвет фона: YELLOW

Цвет шрифта абзаца: NAVY

Цвет шрифта заголовков 3 уровня: BLUE

Выравнивание заголовков 3 уровня: по левому краю

5.3 Цвет фона: RED

Цвет шрифта абзаца: BLACK

Цвет шрифта заголовков 3 уровня: WHITE

Выравнивание заголовков 3 уровня: по правому краю

5.4 Цвет фона: LIME

Цвет шрифта абзаца: BLUE

Цвет шрифта заголовков 3 уровня: PURPLE

Выравнивание заголовков 3 уровня: по левому краю

5.5 Цвет фона: SILVER

Цвет шрифта абзаца: TEAL

Цвет шрифта заголовков 3 уровня: MAROON

Выравнивание заголовков 3 уровня: по правому краю

## 5.6 Цвет фона: LIME

Цвет шрифта абзаца: GREEN

Цвет шрифта заголовков 3 уровня: TEAL

Выравнивание заголовков 3 уровня: по левому краю

## 5.7 Цвет фона: FUCHSIA

Цвет шрифта абзаца: PURPLE

Цвет шрифта заголовков 3 уровня: RED

Выравнивание заголовков 3 уровня: по правому краю

## 5.8 Цвет фона: OLIVE

Цвет шрифта абзаца: YELLOW

Цвет шрифта заголовков 3 уровня: LIME

Выравнивание заголовков 3 уровня: по левому краю

## 5.9 Цвет фона: MAROON

Цвет шрифта абзаца: WHITE

Цвет шрифта заголовков 3 уровня: WHITE

Выравнивание заголовков 3 уровня: по правому краю

## 5.10 Цвет фона: GREEN

Цвет шрифта абзаца: LIME

Цвет шрифта заголовков 3 уровня: AQUA

Выравнивание заголовков 3 уровня: по левому краю

## 5.11 Цвет фона: BLACK

Цвет шрифта абзаца: FUCHSIA

Цвет шрифта заголовков 3 уровня: RED

Выравнивание заголовков 3 уровня: по правому краю

## 5.12 Цвет фона: NAVY

Цвет шрифта абзаца: AQUA

Цвет шрифта заголовков 3 уровня: SILVER

Выравнивание заголовков 3 уровня: по левому краю

## 5.13 Цвет фона: PURPLE

Цвет шрифта абзаца: SILVER

Цвет шрифта заголовков 3 уровня: LIME

Выравнивание заголовков 3 уровня: по правому краю

## 5.14 Цвет фона: YELLOW

Цвет шрифта абзаца: RED

Цвет шрифта заголовков 3 уровня: BLUE

Выравнивание заголовков 3 уровня: по левому краю  
отступ от текста сверху: 20 px

## 5.15 Цвет фона: WHITE

Цвет шрифта абзаца: GREY

Цвет шрифта заголовков 3 уровня: SILVER

Выравнивание заголовков 3 уровня: по левому краю  
отступ от текста снизу

## Лабораторная работа 7

### Блочная верстка интернет-ресурсов

#### Цель работы

Научиться разрабатывать документы, предназначенные для публикации в Internet.

#### Задачи работы

Изучение возможностей создания web-приложений с использованием блочной верстки.

#### Требования к организации рабочего места

Необходимое программное обеспечение: ОС Windows, текстовый редактор Блокнот (либо NotePad++, PHPDesigner и подобные им редакторы кода), браузер (MS Internet Explorer, Opera или другой).

## 1 Теоретические положения

### 1.1 Блоки

Один из способов представления информации на Web-странице — представление на слоях с помощью блоков. Блоки являются элементами окна, они могут перекрываться. С помощью JavaScript блоки можно перемещать, скрывать или отображать. Блоки, известные как перемещаемые элементы, составляют часть технологии CSS. Один из способов определения блока — использование тега **<div>**. Чтобы задать блок с помощью тега **<div>**, следует заключить содержимое блока в пару тегов **<div>** и **</div>** и определить свойства блока с помощью параметра STYLE, например так

```
<div id="l1" STYLE=" position: absolute; left=300; top=150; width=120; visibility: visible"> <img src=pl.gif width=100></div>
```

Данный фрагмент кода определяет перемещаемый блок, содержащий изображение, смещенный относительно левого верхнего угла окна браузера на 300 пикселей по горизонтали и на 150 пикселей по вертикали, ширина блока — 120 пикселей. При абсолютном позиционировании элемент позиционируется независимо и может перекрывать другие элементы. В рассмотренном примере элемент позиционирован относительно тела документа.

Блоки можно использовать при создании меню, оглавлений, навигационных панелей.

### Пример 1. Несложный пример блочной вёрстки

Пусть имеется макет (рисунок 1). Макет условный — просто разметка, т.к. цель примера — наглядно объяснить принцип создания блочной вёрстки. Зная базу, далее можно усложнить макет.

Согласно макету, страница сайта будет содержать пять блоков: «шапку», боковую панель с навигационным меню, основной блок с контентом и «подвал».



Рисунок 1 Условный макет страницы

Сначала создадим HTML-страницу: обозначим структуру, разметим её. HTML-код будет таким:

```
<html>
<head>
  <meta charset="utf-8">
  <link rel="stylesheet" type="text/css" href="style.css">
  <title>Пример</title>
</head>
<body>
  <div id="block">
    <div class="header"><h1>Шапка</h1></div>
    <div class="menu"><h2>Меню</h2></div>
    <div class="content"><h2>Контент</h2></div>
    <div class="footer"><h2>Подвал сайта</h2></div>
  </div>
</body>
</html>
```

`<div id="block">` — это блок-родитель, внутри которого расположились все остальные блоки. Если открыть HTML-файл в браузере, не подключая таблицу стилей, страница будет выглядеть как на рисунке 2.

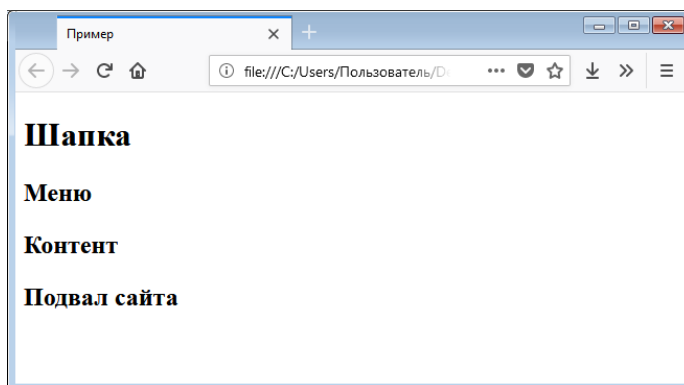


Рисунок 2 HTML-файл без подключенной таблицы стилей

Теперь добавим файл CSS (style.css), код которого приведён ниже.

```
body {
    background: #f3f2f3;
    color: #000000;
    font-family: Trebuchet MS, Arial, Times New Roman;
    font-size: 14px;
}

div#block { width:900px; margin:30px auto; background-color:#dddddd;}
div.header {width:900px; height:100px; background-color:#717dc9;}
div.menu{width:148px; height:350px; float:left; border-right:2px dashed
#717dc9;}
div.content {width:750px; float:right;}
div.footer {width:900px; height:70px; background-color:#717dc9;
clear:both;}
p {text-align: justify; text-indent: 20px;padding-left: 20px;padding-right:
20px;}
h1 {color: #dddddd; padding-left: 170px;padding-top: 10px; font-size: 36px;}
h2 {padding-left: 20px;}
h3 {padding-left: 14px;padding-top: 5px;}
```

С помощью стилей были последовательно оформлены содержимое тега **<body>** и всех находящихся внутри контейнеров. HTML-страница приняла другой вид (рисунок 3).

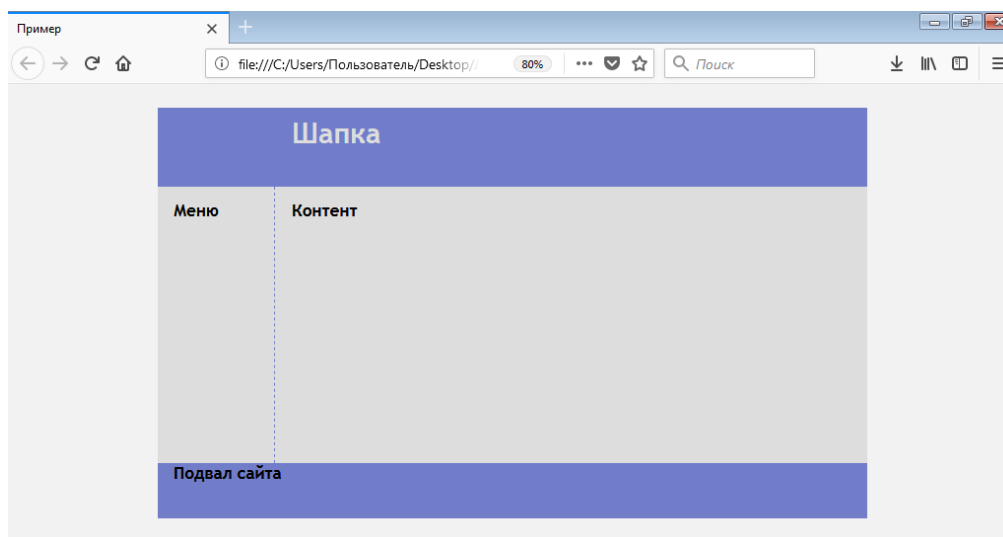


Рисунок 3 HTML-файл с подключенной таблицей стилей

Дальше можно наполнять сайт содержимым и усложнять оформление, но делаться это будет по изложенному выше принципу.

Пример с контентом:

```

<!doctype html>
<html lang="ru">
<head>
  <meta charset="utf-8">
  <link rel="stylesheet" type="text/css" href="style.css">
  <title>Пример</title>
</head>
<body>
  <div id="block">
    <div class="header">
      <h1>Бизнес-информатика</h1>
    </div>
    <div class="menu">
      <p><a href="index.htm">Главная</a></p>
      <p><a href="obuch.htm">Обучение</a></p>
      <p><a href="praktika.htm">Практика</a></p>
    </div>
    <div class="content">
      <h2>О направлении подготовки</h2>
      <div style="float:left; margin: 5px 10px 5px 20px">
        
      </div>
      <p>Бизнес-информатика – один из новейших научных подходов к
построению и применению систем информации и коммуникации в бизнес-
процессах. Это направление построено на стыке нескольких научных
дисциплин: информатики, экономики и менеджмента.</p>
      <p>Бизнес-информатика» - специальность XXI-го века, она предоставляет
широкие возможности в выборе сферы деятельности. Выпускник по
направлению «Бизнес-информатика» – это эрудированный специалист широкого
профиля, получивший фундаментальные знания в области информатики и
профессиональные знания в области экономики, менеджмента и маркетинга.</p>
      <p>Выпускники направления «Бизнес-информатика» могут работать
менеджерами по поддержке информационных систем, программистами, бизнес-
аналитиками, начальниками информационных управлений и отделов (на
предприятиях, в банках, налоговых инспекциях, страховых организациях и т.д.),
специалистами в области информационных технологий, администраторами баз
данных, преподавателями, системными администраторами.</p>
    </div>
    <div class="footer">
      <h2>&copy; 2018</h2>
    </div>
  </div>

```

```
</body>  
</html>
```

## 2 Порядок выполнения работы

Рекомендуется все полученные в результате работы файлы, а также используемые картинки размещать в одной, специально созданной папке. Работа выполняется в приложении Блокнот (также вместо Блокнота могут использоваться специализированные редакторы кода: Блокнот++, PHPDesigner и подобные им). Для просмотра результата используется браузер.

2.1 Создать главную страницу сайта на одну из тем, указанных в п.5 (выбирается по указанию преподавателя, также обучающийся может предложить свою тему и согласовать ее с преподавателем). При создании страницы использовать CSS и блочную верстку. За основу можно взять разметку из Примера 1.

Создать другие страницы сайта (не менее двух).

Примечание. Если указана кодировка (например, `<meta charset="utf-8">`), то при сохранении файла в приложении Блокнот следует указать ту же кодировку.

2.2 Проверить валидность кода (валидатор <https://validator.w3.org>).

2.3 Оформить отчет о проделанной работе (см. п.3).

## 3 Оформление отчета

В отчете следует привести:

- тему, цель, задачи работы;
- листинг кода html-файла;
- листинг css-файла;
- скриншот страницы, открытой в браузере;
- ответы на контрольные вопросы.

## 4 Вопросы для самоконтроля знаний

- 1) Что собой представляет каскадная таблица стилей?
- 2) Как формируется правило таблицы стилей?
- 3) Как таблица стилей связывается с документами HTML?
- 4) Назначение слоев на web-страницах.
- 5) С помощью какого тега создаются слои?

## 5 Варианты тем для создания web-страницы

- Направление подготовки «Бизнес-информатика».
- Что такое CSS.
- Страница сайта фирмы или организации.
- Главная страница интернет-магазина.

## **Лабораторная работа 8**

### **Создание элементов дизайна интернет-ресурсов средствами графических редакторов**

#### Цель работы

Научиться редактировать некачественные изображения и оформлять их.

#### Задачи работы

Освоить приемы подготовки графических изображений, предназначенных для публикации на web-страницах.

#### **Требования к организации рабочего места**

Лабораторная работа выполняется студентами в компьютерных классах кафедры информатики и информационных технологий. Необходимое программное обеспечение: ОС Windows XP(Vista), обозреватель MS Internet Explorer или Opera, текстовый процессор MS Word, графический редактор Adobe PhotoShop.

## **1 Общие положения**

### **1.1 Компьютерная графика и ее применение на web-страницах**

Компьютерная графика – специальная область информатики, изучающая методы и средства создания и обработки изображений с помощью программно-аппаратных вычислительных комплексов.

В зависимости от способа формирования изображений, компьютерную графику принято делить на:

- растровую;
- векторную;
- фрактальную.

Отдельным предметом считается трехмерная графика, изучающая приемы и методы построения объемных моделей объектов в виртуальном пространстве.

Эти виды отличаются принципами формирования изображения при отображении на экране монитора или при печати на бумаге. Каждый вид используется в определенной области.

Растровую графику применяют при разработке мультимедийных проектов. Иллюстрации, выполненные средствами растровой графики, чаще вводятся в компьютер с помощью сканера, а затем обрабатываются специальными программами - графическими редакторами.

Программные средства для работы с векторной графикой наоборот предназначены для создания иллюстраций на основе простейших геометрических элементов. В основном применение векторной графики - это оформительские работы.

Создание фрактальной художественной композиции состоит не в рисовании, а скорее в программировании. Программные средства для работы с фрактальной графикой предназначены для автоматической генерации изображений путем математических расчетов. Например, таким образом можно получить заставки на ТВ.

## 1.2 Графический редактор Adobe PhotoShop

В большом классе программ для обработки растровой графики особое место занимает пакет PhotoShop компании Adobe. На сегодняшний день эта программа является стандартом в компьютерной графике. Основные возможности, которые предоставляет PhotoShop для работы с изображениями:

- изменение режимов изображения;
- корректировка цвета;
- деформация, вращение, движение изображения;
- работа со слоями; редактирование слоев по отдельности, создание между ними различных эффектов;
- прекрасные возможности выделения и работы с фрагментами;
- огромное количество фильтров;
- много инструментов для рисования;
- наложение текста.

Обычно редактор PhotoShop используется для того, чтобы рисовать, стилизовать, заниматься фотомонтажом, исправлять плохие фотографии и создавать собственные изображения. Но несмотря на то, что PhotoShop предоставляет неплохие возможности для создания изображений, основное назначение его - редактирование готовых рисунков.

## 1.3 Пример создания макета сайта в Photoshop

Photoshop — популярная среди веб-дизайнеров программа, для многих даже стандарт. И если речь заходит о макете сайта, то в большинстве случаев под ним подразумевается файл формата .PSD.

Перед непосредственно созданием макета необходимо определиться со структурой сайта, продумать навигацию, представлять себе, где какие блоки и элементы будут располагаться.

Подавляющее большинство веб-страниц состоит из:

- верхней части (шапки), в которой располагается логотип, слоган, навигационное меню и т. д.;
- основного блока, в котором размещается контент;
- боковой панели (сайдбара);
- нижней части (футера), в которой обычно хранится вспомогательная информация, сведения об авторских правах, полезные ссылки и т. д.

Рассмотрим условный пример макета, схему шаблона, но по описываемому алгоритму можно работать с любым макетом.



Рисунок 2 Условный макет сайта

1. Создать документ будущего макета. При создании нужно определить размер.
2. Разграничить макет с помощью направляющих, вытягиваемых из линеек. Чтобы включить линейки, нужно либо нажать **Ctrl+R**, либо выполнить команду **Просмотр - Линейки** главного меню программы.

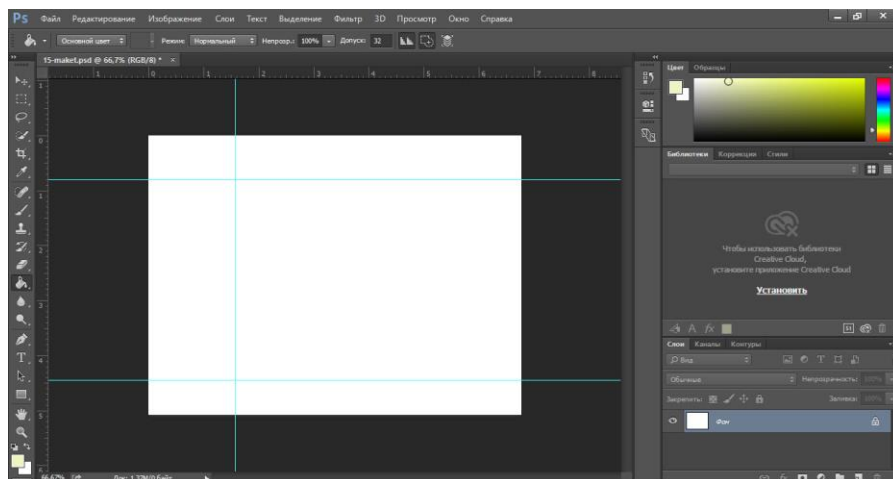


Рисунок 3 Документ с направляющими


Для вытягивания горизонтальной направляющей следует щёлкнуть на верхней линейке кнопкой мыши и, не отпуская её, перемещать указатель вниз до тех пор, пока направляющая не займёт нужное положение. Чтобы вытянуть вертикальную линию, нужно проделать то же самое, только перемещать указатель вправо от расположенной слева линейки.

В итоге шаблон должен быть разграничен (рисунок 3).

3. Нарисовать макет. Добавить графику, линии, кнопки и прочие элементы.
4. Добавить цвета.
5. Разрезать макет.

Уже готовый макет необходимо разрезать «на кусочки», чтобы использовать его элементы в дизайне сайта, часть которого будет реализована

через HTML, часть — через CSS, а часть — при помощи как раз тех самых картинок, на которые макет будет разрезан. Для разрезания макета нужно:

- Выключить слои с ненужными элементами макета — текстом, кнопками, формами и т. д. Чтобы спрятать слой, щёлкнуть на глазке в его строчке на панели «Слой».
- Подогнать вытянутые из линеек направляющие под структурные элементы сайта.
- Выбрать инструмент Раскройка . Он расположен в группе инструментов Рамка.
- Чтобы вырезать кусок макета, нужно подвести указатель к верхнему левому углу вырезаемой прямоугольной области, при нажатой левой кнопке мыши выделить эту область. Вырезанная часть будет пронумерована. Синий цвет номера означает, что фрагмент активен и его можно редактировать. Отрезанный фрагмент выделен и имеет маркеры, благодаря которым его размер и положение можно корректировать (рисунок 4).

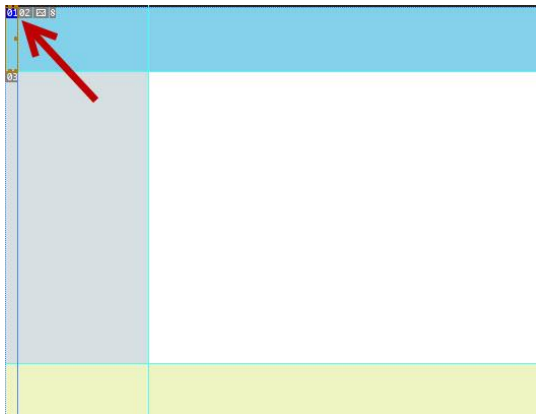


Рисунок 4 Выделенный кусок макета

5. Аналогичным образом сделать следующий разрез, и так до тех пор, пока работа не будет завершена.

6. Когда всё сделано, сохранить результат. Для этого выполнить команду Файл - Экспортировать - Сохранить для Web (старая версия), проверить настройки (выбрать формат, качество и т. д.), нажать Сохранить и указать папку, в которой будет создан каталог графических файлов. Внутри выбранной директории появится папка **images** с графическими файлами, каждый из которых содержит часть макета.

7. Далее можно сохранить кнопки, иконки и прочие, ранее скрытые, элементы.

## 2 Порядок выполнения работы

2.1 Создать папку для хранения файлов сайта (например, Site), в которой собрать необходимый текстовый и графический материал (необходимо оптимизировать объем и количество графических файлов, предназначенных для

размещения на страницах вашего сайта). Макет сайта и его графическое наполнение создается по одной из тем п.5.

Примечание. Не следует использовать символы русского алфавита в именах файлов, которые включаются в состав веб-узла, так как многие операционные системы серверов сети Интернет работают только с англоязычными именами.

2.2 Для подготовки графического материала (фотографий, схем и т.д.) можно использовать следующие возможности программы **Photoshop**:

- изменить тоновый диапазон изображения (см. п. 1.3.5);
- сделать коррекцию цветового баланса с помощью команд

Изображение - Автоматическая цветовая коррекция, Изображение - Коррекция - Сочность, Изображение - Коррекция - Цветовой баланс;

- повысить контрастность изображения Изображение - Автоконтраст;
- уменьшить размер изображения Изображение - Размер изображения (выполняется в последнюю очередь).

2.3 В графическом редакторе **Photoshop** сделать эскиз главной страницы сайта:

- отметить расположение и размеры главного баннера, навигационных кнопок и других элементов страницы;
- подобрать цветовой оформление;
- добавить графический материал, подготовленный по заданию п. 2.2;
- определить шрифт заголовков, подзаголовков и основного текста;
- выполнить нарезку макета;
- сохранить как web-публикацию.

2.4 В графическом редакторе CorelDraw сделать логотип (или эмблему) сайта. Для создания такого изображения могут быть использованы:

- рисунок фигуры и выполнение заливки, можно также применить интерактивные инструменты;
- верхнюю часть можно нарисовать с помощью замкнутой кривой и произвести ее заливку;
- создание изогнутой траектории (кривые Безье) и размещение текста по траектории.

Экспортировать картинку в один из растровых форматов (jpeg, gif, png). Добавить на страницу, созданную в п. 2.3.

2.5 Оформить отчет.

### 3 Требования к оформлению отчета

Отчет о проделанной работе должен содержать:

- название и цель работы;
- описание последовательности выполнения работы, результаты разработки дизайна сайта;

- письменные ответы на вопросы для самоконтроля знаний п.4 по указанию преподавателя.

#### **4 Вопросы для самоконтроля знаний**

- 1) В чем заключается принцип ориентирования web-дизайна на конкретную аудиторию?
- 2) В чем заключается принцип эргономичности web-дизайна?
- 3) В чем заключается принцип того, что web-дизайн должен быть красивым?
- 4) В чем заключается требование технологичности web-дизайна?
- 5) Понятие компьютерной графики.
- 6) Виды компьютерной графики.
- 7) Какие инструменты PhotoShop используются для выделения участка изображения?
- 8) Какие действия с выделенной областью позволяет произвести PhotoShop?
- 9) Что такое тоновая коррекция?
- 10) Где можно увидеть тоновый диапазон изображения в PhotoShop?
- 11) Понятие слоя в PhotoShop.
- 12) Как создать слой в PhotoShop ?

#### **5 Варианты тем заданий для самостоятельного выполнения**

- 1) Аграрные вузы.
- 2) Предприятия агропромышленного комплекса.
- 3) Города республики Башкортостан.
- 4) Сайты районных администраций РБ.
- 5) Персональные сайты.
- 6) Фермерские хозяйства.
- 7) Спорт.
- 8) Музыка.
- 9) Детские центры.
- 10) Школы.
- 11) Туристические бюро.
- 12) Базы отдыха.
- 13) Строительные организации.
- 14) Библиотеки.
- 15) Коммерческие организации.

## **Лабораторная работа 9**

### **Специализированные программы для создания интернет-ресурсов**

**Цель работы.** Научиться разрабатывать документы, предназначенные для публикации в Internet.

**Задачи работы.** Знакомство с возможностями пакет приложений MS Office по созданию документов HTML.

#### **Требования к организации рабочего места**

Лабораторная работа выполняется студентами в компьютерных классах кафедры информатики и информационных технологий. Необходимое программное обеспечение: ОС Windows XP (Vista), обозреватель MS Internet Explorer, пакет приложений MS Office, включающий в себя, в том числе, текстовый процессор MS Word, редактор гипертекстовых документов MS Front Page, текстовый процессор MS Excel, приложение MS Power Point.

### **1 Общие сведения**

#### **1.1 Проектирование web-сайтов средствами редактора MS FrontPage**

Одним из средств для создания и редактирования web-сайтов является приложение MS FrontPage. Программа FrontPage обладает интегрированным интерфейсом, удобным для создания и редактирования web-страниц. Панели инструментов и меню имеют дружественный интерфейс с другими приложениями Microsoft Office и являются полностью настраиваемыми. Используя технологию фреймов при создании сайтов, можно значительно улучшить их внешний вид, функциональность, а также без дополнительных сложностей организовать навигацию.

После запуска MS FrontPage на экране появляется окно приложения, в котором располагаются страницы будущего сайта. Под строкой заголовка окна находится строка меню, при помощи которой можно вызвать любую команду MS FrontPage.

По команде **Вид (Страница)** или выборе определенного файла в режиме папок вызывается встроенный редактор страниц (рисунок 2). При этом внизу экрана появляются четыре кнопки управления режимами: (**Конструктор**, **С разделением**, **Код**, **Просмотр**). **Конструктор** - режим редактирования как в обычном текстовом редакторе; **Код** - редактирование HTML-кода; **Просмотр** - просмотр готового вида страницы; **С разделением** - совмещает режимы **Кода** и **Конструктора** в соседних подокнах.

#### **1.2 Создание web-страниц средствами приложений пакета MS Office**

##### **1.2.1 Основные возможности пакета MS Office по созданию web-страниц**

Для создания web-страницы можно использовать возможности нескольких программ пакета Microsoft Office. Например, сохранить данные в виде web-страницы можно в Microsoft Excel, после чего использовать Microsoft

Access для добавления сгруппированных элементов управления страницей данных или сценариев. Кроме того, web-страницу можно открыть в Microsoft FrontPage и добавить в нее темы, чтобы придать ей вид, сходный с видом других web-страниц на web-узле.

Рассмотрим разные способы использования данных из программ Word, Excel и PowerPoint при подготовке Web-страниц.

### 1.2.2 Использование буфера обмена Windows

Программа FrontPage входит в состав Microsoft Office, который является в настоящий момент одним из наиболее популярных офисных пакетов. Значительный объем информации, который вы собираетесь размещать на Web-узле, может быть представлен в виде документов Word, Excel, PowerPoint или в базе данных Access.

Наиболее простой способ переноса данных из документов Microsoft Office — использование буфера обмена Windows, (команды Правка Копировать и Правка Вставить (Paste) на стандартной панели инструментов или На панели Буфер обмена (Clipboard) щелкнуть мышью на нужном фрагменте).

### 1.2.3 Использование команды **Файл меню Вставка**

Для размещения на Web-странице данных других форматов можно использовать команду Вставка ► Файл. При выполнении этой команды открывается диалоговое окно Выбор файла. Список Тип файлов данного окна содержит форматы, которые можно импортировать в программу FrontPage.

В том случае, если программа FrontPage не знает, с использованием какого формата открыть выбранный файл, на экране появляется диалоговое окно **Открыть файл как**, предлагающее применить один из следующих форматов: HTML, RTF или txt.

### 1.2.4 Добавление файла без преобразования формата

Программа FrontPage позволяет с помощью команды **Файл ► Импорт** добавлять в Web-узел файлы без преобразования их исходного формата. Список форматов, для которых применима данная команда, ограничен. С помощью этой команды можно импортировать в Web-узел файлы в HTML формате, графические файлы форматов GIF и JPEG, а также документы Microsoft Office: Word, Excel, PowerPoint.

### 1.2.5 Сохранение документов Microsoft Office в HTML-формате

Программы пакета Microsoft Office Word, Excel и PowerPoint позволяют сохранять документы, подготовленные ими, в формате Web-страницы и просматривать их в обозревателе. Кроме этого, все эти программные продукты держат команды, позволяющие размещать гиперссылки в документе.

Для сохранения документа в формате Web-страницы в меню **Файл** их программ размещена команда **Сохранить как web-страницу**. При выборе этой команды открывается диалоговое окно **Сохранение документа**. В списке **Тип файла** по умолчанию установлено значение **Web-страница**, при котором

сохраненный документ получает расширение HTML и может открыться в обозревателе.

Для того чтобы просмотреть подготовленную Web-страницу в обозревателе, нет необходимости выходить из программы, достаточно воспользоваться командой **Файл ► Web ► Предварительный просмотр web-страницы**. Запустится обозреватель Microsoft Internet Explorer с загруженным в него документом.

### 1.2.6 Создание гиперссылок

В документах, созданных в программах Word, Excel и PowerPoint, можно разместить гиперссылки. Во всех перечисленных выше программах для создания гиперссылки используется диалоговое окно **Добавление гиперссылки**, которое можно открыть, воспользовавшись командой **Вставка ► Гиперссылка**.

### 1.2.7 Создание HTML-документа в Word

Документ, подготовленный в Microsoft Word 2003, можно сохранить в HTML-формате. Для этого предназначена команда **Файл ► Сохранить как Web-страницу**. Кроме этого, создавая новый документ, можно сразу использовать как шаблон Web-страницы, выполнив следующие действия:

1) Запустить программу Microsoft Word 2003.

2) Для создания нового документа выбрать команду **Файл ► Создать**.

3) На панели **Создание документа** в разделе **Создание** выбрать значок **Веб-страница**. Создается пустой документ, имеющий по умолчанию расширение HTML. Можно приступить к вводу информации.

### 1.2.8 Использование анимации PowerPoint

Программа PowerPoint, входящая в состав Microsoft Office, позволяет на профессиональном уровне подготовить слайды для проведения презентаций. Благодаря тесной интеграции, существующей между программными продуктами данного пакета, можно в программе PowerPoint создать слайд, разместить на нем ссылки и сохранить его в виде Web-страницы.

### 1.2.9 Копирование с помощью «мыши»

Перемещение и копирование данных между документами из различных программ можно осуществлять перетаскиванием фрагментов документов с помощью мыши. Обе программы должны поддерживать механизм работы с составными документами (OLE). Окна программ нужно расположить так, чтобы исходный и конечный документы были открыты и видимы.

## 2 Порядок выполнения работы

### 2.1 Создание web-страниц с помощью текстового процессора MS Word.

#### 2.1.1 Выполнение примера на создание личной web-страницы.

В начале выполнения работы следует создать новую папку, в которой будут храниться все созданные файлы.

Запустить приложение MS Word.

Создание личной web-страницы можно начинать с создания пустой web-страницы или воспользоваться шаблоном. На личную страницу поместить информацию о себе, своих текущих делах и планах, учебе, увлечениях, друзьях и т.п. Личную страницу можно украсить графическими элементами (рисунками, линиями, маркерами). Присвоить название web-странице (выбрать команду **Файл ► Свойства**, на вкладке **Документ** в поле **Название** ввести нужный заголовок).

### 2.2.1 Создание web-узла

В web-узел входят:

- web-документы из Приложения А или другие по указанию преподавателя;
- web-документ с гиперссылками на другие документы;
- web-документ, размечающий рамки.

Создание каждого из документов web-узла следует начинать с создания пустой web-страницы. На каждой из этих страниц следует набрать и отформатировать текст обычными средствами Word.

В первом документе вставить картинку из библиотеки ClipArt и бегущую строку **Вид - Панели инструментов- Веб-компоненты- Бегущая строка**).

Во втором документе вставить картинку из библиотеки ClipArt и разделить страницу графической линией.

В третьем документе создать список с графическими маркерами.

Оформить страницы, применив к ним тему.

Во втором документе слово *факультетах* преобразовать в гиперссылку на документ 3.

Создать новую страницу фреймов, разделить ее на две рамки (**Формат - Рамки - Новая рамка сверху**). В одной из рамок создать гиперссылки на документы 1, 2 и 3, например, главная университет факультеты

При добавлении гиперссылок указывать, что документы 1, 2 и 3 должны выводиться в другой рамке.

Указать документ, который будет выводиться в другой рамке при первоначальной загрузке (**Формат - Рамки - Свойства рамки**, заполнить поле **Начальная страница**).

Сохранить страницу рамок. Просмотреть страницу рамок в Internet Explorer, проверить работу гиперссылок.

### 2.1.3 Добавление гиперссылки

На личную web-страницу добавить гиперссылку для перехода на web-узел. Гиперссылка на web-узел может быть помещена непосредственно в текст личной web-страницы, например, Я учусь в БГАУ.

## 2.2 Создание и оформление web-узла в редакторе MS Front Page

### 2.2.1 Создание и оформление web-узла

Создать и оформить web-узла, содержащий web-страницы. Для этого необходимо выполнить команду **Файл ( Создать ( Одностраничный Web-**

узел, в окне диалога выбрать вариант **Одностраничный Web-узел**. Затем переключиться в режим **Переходы** (вкладка внизу окна Front Page), далее дважды щелкнуть по названию файла index.htm в центре экрана. Затем следует расположить на листе данные, указанные на рисунке 1: ввести текст по образцу, расположить картинки.

Для оформления внешнего вида домашней страницы необходимо воспользоваться командой **Формат/Шрифт**, в которой нужно выбрать цвет шрифта и гарнитуру написания. Далее необходимо сохранить страницу.

Для создания страницы с фреймами, указанной на рисунке 2 выполнить команду **Файл/ Создать / Другие шаблоны страниц**, в появившемся диалоговом окне **Шаблоны страниц** выбрать вкладку **Страница рамок**, затем выбрать шаблон **Объявление и оглавление**, и щелкнуть на кнопке **ОК**. После этого перейти в режим редактирования страницы, затем нужно щелкнуть на кнопке **Создать страницу** в каждом фрейме.



Рисунок 1 Пример домашней страницы

2.2.2 Для оформления страницы верхнего фрейма (рисунок 2) вставить таблицу из одной строки и трех столбцов. Затем в столбцы таблицы поместить рисунки, далее переместить границы столбцов по размеру каждого из рисунков и выровнять их по центру. Затем щелкнуть правой кнопкой мыши на таблице и в появившемся контекстном меню выбрать команду **Свойства таблицы....** В группе элементов **Границы** задать нулевой размер границы таблицы для того, чтобы граница стала невидимой.

Для создания кнопок меню в левой части страницы с фреймами также следует использовать таблицу, состоящую из одного столбца и пяти строк. Каждая кнопка помещается в строку таблицы командой **Вставка/Меняющаяся кнопка**. В диалоговом окне **Меняющиеся кнопки** следует выбрать вид кнопки и задать текст, который будет располагаться на кнопке.

В правой части страницы с фреймами следует ввести текст, оформив его соответствующим образом.

После этого необходимо выполнить команду **Файл - Сохранить**, для сохранения новых страниц. Каждая страница будет сохранена под своим именем. Сохраняемая страница будет выделена рамкой синего цвета. Например, для страницы верхнего фрейма следует задать имя baner.htm, для левого - mnu.htm, для правого - text.htm, а для самой страницы с фреймами — имя start.htm. В результате должна получиться страница с фреймами, представленная на рисунке 2.

После создания страницы с фреймами start.htm следует приступить к созданию других страниц, которые будут содержать информацию по каждому из пунктов меню, находящегося в левом фрейме.

**Примечание.** Данные, содержащиеся на создаваемых web-страницах можно перенести, воспользовавшись командами по работе с буфером обмена из электронного варианта методических указаний по FrontPage, установленного на компьютерах.

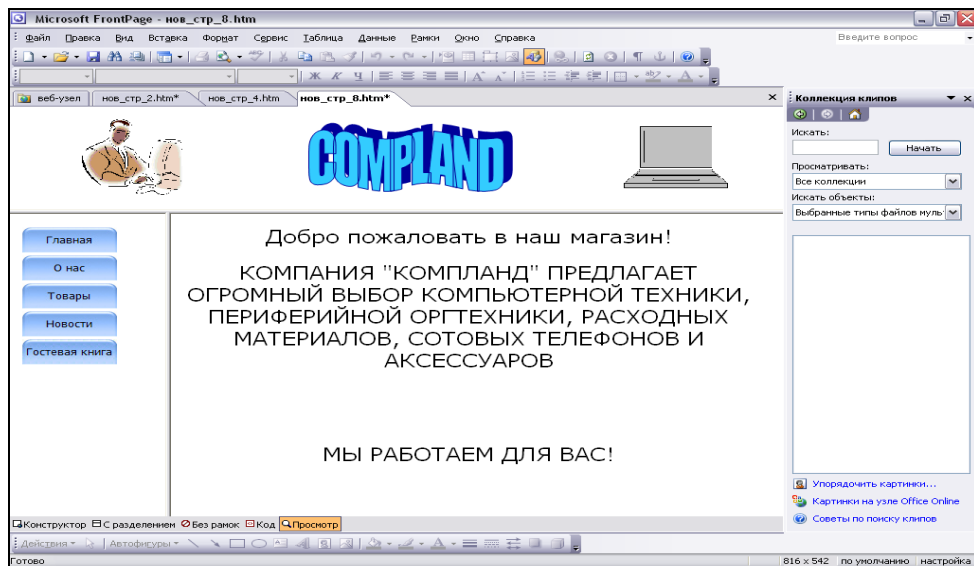


Рисунок 2 Пример создания страницы с фреймами

2.2.3 Подготовить новую страницу в текстовом процессоре MS Word, расположив на ней следующую информацию:

Наши адреса

450075 Уфа, ул. Айская 115	Офис. Режим работы: понедельник- пятница с 10.00 до 20.00
450001 Уфа, ул. Ленина 25	Магазин Compland. Режим работы: понедельник- воскресенье с 10.00 до 20.00

Телефон/факс: Compland: (3457) 28-31-45, 58-64-23, 52-08-32 (многоканальный), 8-9174098888 (звонки из городов Белебей, Бирск, Благовещенск, Дюртюли, Ишимбай, Нефтекамск, Октябрьский, Салават, Стерлитамак, Туймазы бесплатные).

Наш адрес в Интернет: <http://www.compland.ru/>

Наш e-mail: [compland@compland.ru](mailto:compland@compland.ru)

Сотрудники компании: Директор - Ахметов Айрат Гафурович, главный бухгалтер - Зарипова Эльза Фанилевна, начальник отдела продаж - Некрасова Софья Ефимовна, начальник гарантийного отдела - Калганов Денис Петрович.

Затем нужно сохранить страницу, задав ей имя o\_nas.htm в папке, где хранятся другие файлы данного web-узла.

2.2.4 В приложении MS Excel создать страницу, на которой будут располагаться сведения о товарах, предлагаемых к продаже в магазине, например:

Наименование товара	Цена, у.е.	Цена, руб.
22" MONITOR Samsung SyncMasner 1200NF0.24 TCO"99	731.46	22208
19" MONITOR Samsung SamTron 96B TCO"95	232.21	7050
17" MONITOR 0.20 Samsung SyncMasner 757MB TCO"99	2383.71	7248
17" MONITOR 0.20 Samsung SyncMasner 757NF TCO"99	254.04	7713
17" MONITOR 0.23 Samsung 753S	163.59	4967
17" MONITOR Samsung SamTron 76B TCO"99	154.4	4688
Cannon A4 Copy бумага (500листов, 80г/м <sup>2</sup> )	3,66	112
EPSON S04114 4"x6" бумага Photo (20 листов, 194 г/м <sup>2</sup> ) 720 dpi	6,87	209
Hp 92296U A4 пленка LaserJet Monochrome Transparency Film (50 листов) для лазерного принтера	25,76	783
Acorp 56EMSF-2 EXT 56K VI (RTL) V.92	47.25	1435
Acorp 56PIM-2 PCI 56K VI (RTL) V.92 Winmodem	22.61	687

Рассчитать последний столбец с учетом текущего курса доллара. Вставить таблицу из Excel во FrontPage, используя буфер обмена Windows или с помощью мыши.

Сохранить страницу, задав ей имя tovar.htm.

2.2.5 Далее создать страницу, содержащую новости. Предварительно в приложении Power Point создать слайд, поместив на ней следующий текст о проводимых акциях.

Сохранить слайд в формате HTML командой **Файл ► Сохранить как web-страницу**. Используя раскрывающийся список **Папка** открывшегося диалогового окна **Сохранение документа** найти на диске и открыть Web-узел, в который нужно добавить страницу со слайдом. В поле **Имя файла** введите имя Web-страницы new.htm. Нажмите кнопку **Сохранить**. Перейти в программу FrontPage, к открытому Web-узлу, на панели **Список папок** найти созданную в программе PowerPoint Web-страницу. Она имеет расширение htm и содержит на значке логотип программы PowerPoint. Для редактирования файла дважды щелкнуть на нем мышью. Запускается PowerPoint, и можно приступать к модификации информации, размещенной в файле.

2.2.6 Создать форму обратной связи. Для этого в режиме редактирования следует ввести текст по образцу (рисунок 3), далее командой **Вставка ► Форма** вставить соответствующие поля форм. В данной лабораторной работе не предусматривается обработка данных на сервере, поэтому кнопку Submit (Отправить) следует удалить. Для очистки формы будет использоваться кнопка Сброс (Reset). Для удобства ее следует переименовать.

The screenshot shows a web form titled "Нам нужен Ваш отклик!". It contains the following elements:

- A text input field for "Фамилия" (Surname) labeled "текстовое поле One Line TextBox".
- A text input field for "Имя" (Name) labeled "текстовое поле One Line TextBox".
- A dropdown menu for "Укажите город проживания" (Specify your city of residence) with "Уфа" (Ufa) selected, labeled "ниспадающее меню Drop-Down menu".
- A checkbox for "Использовать для связи с Вами только E-mail" (Use only E-mail for communication with you) labeled "флажок Check Box".
- A scrolling text area for "Ваши комментарии" (Your comments) labeled "текстовое поле с полосой прокрутки Scrolling TextBox".
- Three "Очистить" (Reset) buttons, one for each input field, and a larger "кнопки Reset" button.

Рисунок 3 Пример формы обратной связи

Для этого необходимо выполнить двойной щелчок мышью по кнопке, в поле **Значение или подпись** ввести новое название **Очистить**, закончить диалог.

Для создания элементов ниспадающего меню следует выделить поле **Раскрывающийся список**, вызвать его контекстное меню и выбрать там пункт **Свойства поля формы**, в диалоговом окне **Свойства раскрывающегося списка** нажать кнопку **Добавить** и в поле **Вариант** ввести название города, например Уфа, **ОК**. Затем снова нажать кнопку **Добавить** и ввести названия городов Стерлитамак, Салават, Ишимбай, Туймазы, Бирск. Сохранить страницу, задав ей имя otclik.htm.



Рисунок 4 Структура web-узла

2.2.7 Затем следует перейти в режим **Переходы (Вид ► Переходы)** и перетащить мышью из поля **Список папок** файлы start, otclik, o\_nas, new, tovar к блоку **index.htm**, расположенному в центре экрана, чтобы получилась следующая структура web-узла (рисунок 4). Находясь в режиме **Переходы** удобно перемещаться по созданной структуре и использовать ее при редактировании web-страниц.

2.2.8 Для организации навигации по учебному web-сайту необходимо загрузить главную страницу Index.htm, выделить на ней текст **Далее**, затем выполнить команду **Вставка(Гиперссылка** и в диалоговом окне в поле URL указать имя файла Start.htm.

После этого аналогичным образом необходимо связать гиперссылками элементы левого фрейма с соответствующими страницами: **О нас** со страницей o\_nas.htm. Затем связать пункт меню левого фрейма **Товары** с файлом tovar.htm, **Новости** со страницей new.htm, **Гостевая книга** со страницей otclik.htm. Проверить работоспособность созданных гиперссылок. Сохранить отредактированную страницу.

2.2.9 После того, как web-сайт спроектирован, для его оформления рекомендуется применить стандартную тему. Для этого нужно выполнить команду **Формат/Тема**.

**2.3** Выполнить индивидуальные задания для самостоятельной работы по указанию преподавателя из п.5.

### **3 Оформить отчет.**

Отчет о проделанной работе должен содержать:

- название и цель работы;
- описание последовательности выполнения работы;
- дискету с файлами web-узлов;
- письменные ответы на контрольные вопросы п.2.3 по указанию преподавателя.

### **4 Вопросы для самопроверки знаний**

- 1) В каких режимах можно работать с приложением MS Front Page?
- 2) В чем особенность буфера обмена приложения MS Front Page?
- 3) Какими способами в приложении Word можно создать web-страницу?
- 4) Какие файлы находятся в папке, автоматически создающейся при сохранении web-страницы?
- 5) Как осуществить импорт файла в приложение MS Front Page?
- 6) Как происходит сохранение документов приложений из пакете MS Office в HTML-формате?
- 7) Как добавить гиперссылку в документе приложения MS Office?
- 8) Как с помощью «мышь» скопировать данные из приложения MS Office во Front Page?

## 5 Задания для самостоятельного выполнения

5.1 Создать web-узел факультета, включающий следующие страницы: «Главная» (наименование учебного заведения и факультета, фамилии декана и его заместителей), «Кафедры» (названия кафедр, фамилии заведующих, преподаваемые дисциплины), «Специальности» (перечень специальностей, выпускающие кафедры).

5.2 Создать web-узел кафедры, включающий следующие страницы: «Главная» (наименование учебного заведения, факультета и кафедры, фамилия, имя, отчество заведующего кафедрой), «Сотрудники» (фамилия, имя, отчество сотрудника, ученое звание, должность), «Наука» (научные направления, в которых работают сотрудники кафедры, сведения о защите диссертаций).

5.3 Создать web-узел курса, включающий следующие страницы: «Главная» (наименование учебного заведения и факультета, специальность и номер курса), «Группы» (количество групп на курсе, старосты, краткая характеристика каждой группы), «Учеба и досуг» (о достижениях в учебе и увлечениях студентов).

5.4 Создать web-узел учебной группы, включающий следующие страницы: «Главная» (наименование учебного заведения и факультета, номер курса и группы, краткая характеристика группы), «Учеба» (изучаемые дисциплины, участие в олимпиадах и конференциях, о результатах последней сессии), «Увлечения» (спортивные достижения, творчество, учеба на ФОП).

5.5 Создать web-узел коммерческой фирмы, включающий следующие страницы: «Главная» (название фирмы, приглашение к сотрудничеству), «В продаже» (наименования товаров, их характеристики и цены), «Новинки» (новые товары, скидки).

5.6 Создать web-узел туристического агентства, включающий следующие страницы: «Главная» (название агентства, адрес, приглашение воспользоваться услугами), «Путешествия» (страна, количество дней, экскурсии, услуги, цены), «Горящие туры» (страна, дата отъезда).

5.7 Создать web-узел кадрового агентства, включающий следующие страницы: «Главная» (название агентства, адрес, приглашение воспользоваться услугами), «Вакансии» (предприятие, должность, оклад, требования к соискателям), «Резюме» (специальность и возраст работника, образование, профессиональные качества, желаемые должность и оклад).

5.8 Создать web-узел сельскохозяйственного предприятия, включающий следующие страницы: «Главная» (название предприятия, адрес), «Производство» (сведения о цехах, оборудовании, реконструкции предприятия), «Продукция» (производимая продукция, ее качество и цены).

5.9 Создать web-узел магазина, включающий следующие страницы: «Главная» (название магазина, адрес, приглашение прийти за покупками), «Информация о товарах» (наименование товара, производитель, цена, гарантийный срок), «Специальное предложение» (скидки, подарки покупателям).

5.10 Создать web-узел газеты, включающий следующие страницы: «Главная» (наименование печатного издания, тематика, где распространяется), «В номере» (названия статей, краткое содержание), «Новость дня» (содержание, источник информации).

5.11 Создать web-узел района, включающий следующие страницы: «Главная» (название района, место расположения, районный центр), «Территория и население» (площадь территории, плотность населения, число населенных пунктов, самые крупные из них), «Хозяйства» (название хозяйств, их местонахождение, производимая продукция).

5.12 Создать web-узел спортивно-оздоровительного комплекса, включающий следующие страницы: «Главная» (название комплекса, адрес, предложение воспользоваться услугами), «Спорт и здоровье» (перечень спортивных секций и оздоровительных групп, спортивные залы, предлагаемый инвентарь), «Медицинские услуги» (перечень услуг).

5.13 Создать web-узел научного работника, включающий следующие страницы: «Главная» (фамилия, имя, отчество, ученое звание, место работы), «Наука» (направление научных исследований), «Публикации» (название публикации, место и год издания).

5.14 Создать web-узел транспортного хозяйства, включающий следующие страницы: «Главная» (название хозяйства, адрес), «Транспорт» (виды транспорта, количество единиц транспорта, назначение), «Маршруты» (номера маршрутов, начальный и конечный пункты, время в пути).

5.15 Создать web-узел метеорологической службы, включающий следующие страницы: «Главная» (название службы, адрес), «Погода сегодня» (температура, скорость ветра, облачность, осадки), «Прогноз на неделю» (дата, прогнозируемые температура и осадки).

### **Библиографический список**

- 1) Евсеев, Д. А. Web-дизайн в примерах и задачах: учебное пособие для студентов высш. учеб. заведений, обуч. по специальности "Прикладная информатика" и др. эконом. специальностям / Д. А. Евсеев, В. В. Трофимов ; под ред. В. В. Трофимова ; Санкт-Петербургский гос. ун-т экономики и финансов. - Москва : Кнорус, 2014. - 263 с.
- 2) Немцова, Т. И. Компьютерная графика и web-дизайн. Практикум [Текст] : учеб. пособие для студ. учреждений сред. проф. образования : допущено М-вом образования РФ / Т. И. Немцова, Ю. В. Назарова ; под ред. Л. Г. Гагариной. - М. : ФОРУМ : ИНФРА-М, 2011. - 287 с. + 1 эл. опт. диск (CD-ROM).
- 3) Вершинин, М. И. Электронный каталог : проблемы и решения [Текст] : [учебное пособие] / М. И. Вершинин. - СПб. : Профессия, 2009. - 231 с.
- 4) Васильев В. В. Практикум по Web-технологиям [Электронный ресурс] : учебное пособие / В.В. Васильев, Н.В. Сороколетова, Л.В. Хливненко. - М.: Форум, 2009. - 416 с. - Режим доступа: <http://www.znaniium.com/bookread.php?book=166294>

## ПРИЛОЖЕНИЕ А

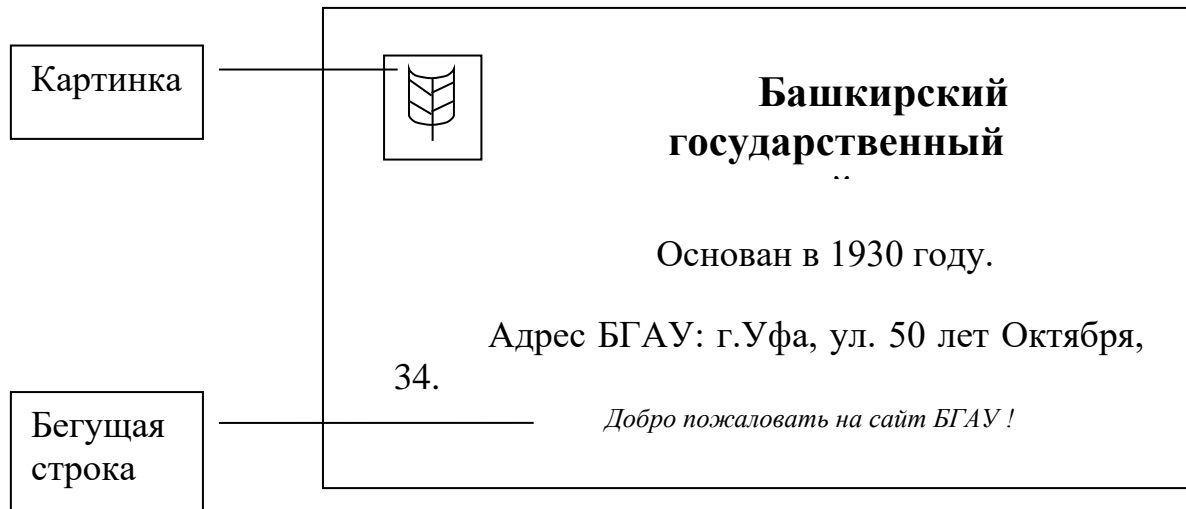


Рисунок А1 Схема документа 1 (главная страница)

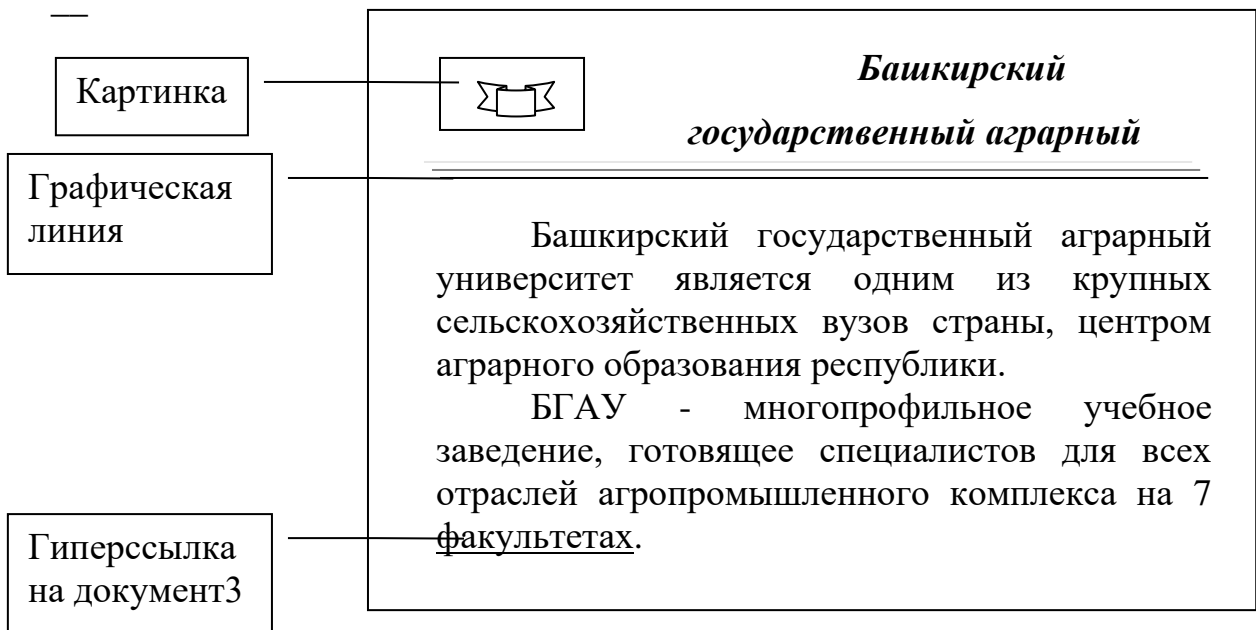
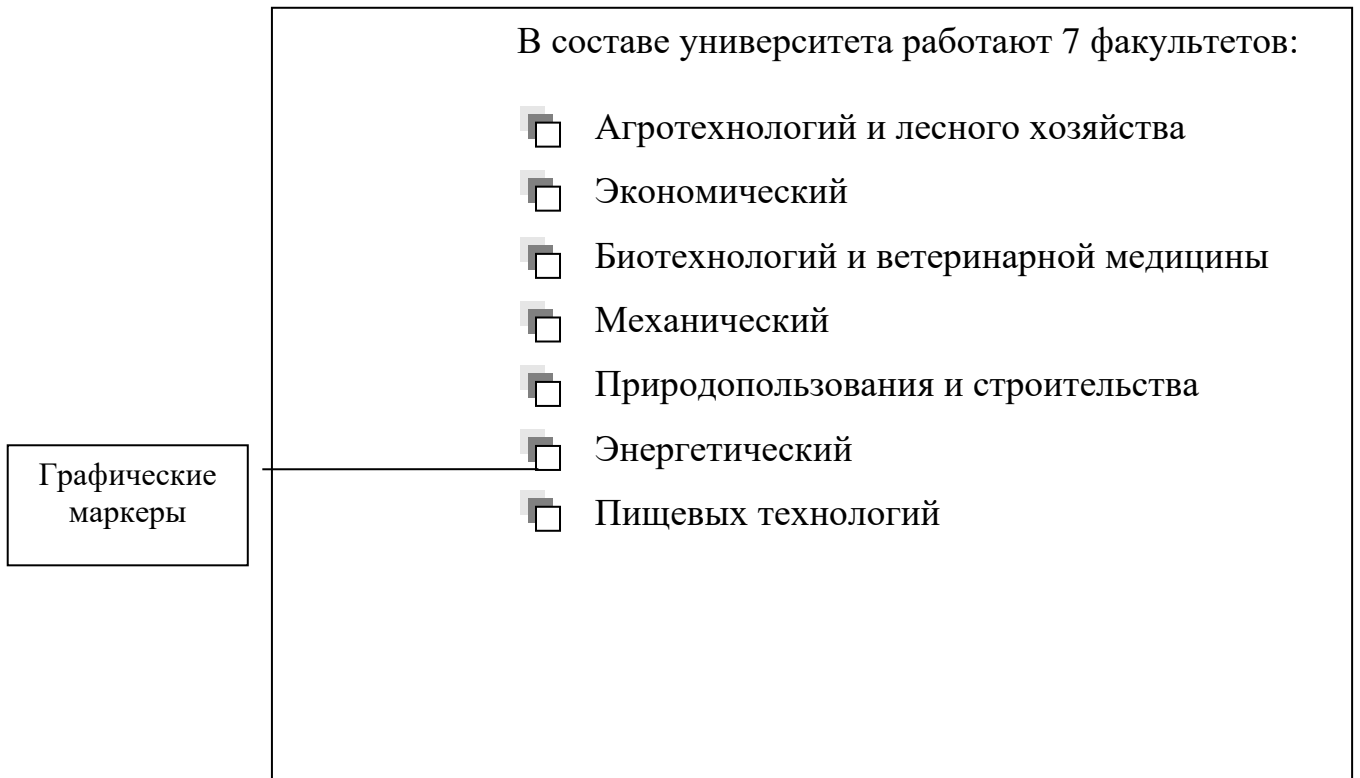


Рисунок А2 Схема документа 2 (университет)

**Продолжение приложения А****Рисунок А3 Схема документа 3 (факультеты)**

