

Факультет маркетинга, менеджмента, предпринимательства  
Кафедра «Основы бизнеса»

## **ТЕХНОЛОГИИ ПРОЕКТИРОВАНИЯ WEB-ДОКУМЕНТОВ**

Методическое пособие

по выполнению лабораторных работ

по дисциплинам «Компьютерные информационные технологии»,

«Системы и средства компьютерных телекоммуникаций в бизнесе»

для студентов специальностей 1-26 02 01 «Бизнес-администрирование»,

1-26 02 03 «Маркетинг», 1-25 01 03 «Мировая экономика»,

1-25 01 07 «Экономика и управление на предприятии»

*Учебное электронное издание*

**Минск 2010**

УДК 004.67(076.5)  
ББК 32.97

**Авторы:**

*И.Е. Ругалёва, Н.В. Дашкевич*

**Рецензенты**

*И.И. Краснова*, доцент кафедры «Экономика и управление на транспорте» БНТУ, кандидат экономических наук, доцент;

*С.С. Карпович*, и. о. заведующего кафедрой «Новые материалы и технологии» ИПК и ПК БНТУ, кандидат технических наук.

В методическом пособии приведены краткие теоретические сведения описания основ гипертекстовой разметки HTML, являющейся основой разработки электронных документов для Интернета. Материал, представленный в пособии, содержит пояснительные иллюстрации и практические примеры с пошаговыми инструкциями и раскрывает все стороны создания электронных HTML – документов – от форматирования текста до стилевых шаблонов CSS. После каждой темы приведены варианты индивидуальных заданий.

Белорусский национальный технический университет  
пр-т Независимости, 65, г. Минск, Республика Беларусь  
тел. (017) 293 91 97  
Регистрационный номер № БНТУ/ФММП51-17.2010

© Ругалёва И.Е., Дашкевич Н.В., 2010  
© БНТУ, 2010

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	6
<i>Лабораторная работа № 1</i>	
СТРУКТУРА ГИПЕРТЕКСТОВОГО ДОКУМЕНТА .....	7
1.1. Типовое задание .....	12
1.2. Контрольные вопросы.....	14
1.3. Контрольные задания.....	14
<i>Лабораторная работа № 2</i>	
КАСКАДНЫЕ ТАБЛИЦЫ СТИЛЕЙ.....	16
2.1. Типовое задание .....	19
2.2. Контрольные вопросы.....	22
2.3. Контрольные задания.....	22
<i>Лабораторная работа № 3</i>	
ФОРМАТИРОВАНИЕ ТЕКСТА .....	23
3.1. Типовое задание .....	28
3.2. Контрольные вопросы.....	32
3.3. Контрольные задания.....	32
<i>Лабораторная работа № 4</i>	
ФОРМАТИРОВАНИЕ ТЕКСТА С ПОМОЩЬЮ CSS .....	38
4.1. Типовое задание .....	52
4.2. Контрольные вопросы.....	56
4.3. Контрольные задания.....	56
<i>Лабораторная работа № 5</i>	
ГИПЕРССЫЛКИ И ДОБАВЛЕНИЕ ИЗОБРАЖЕНИЙ .....	57
5.1. Типовое задание .....	60
5.2. Контрольные вопросы.....	62
5.3. Контрольные задания.....	62
<i>Лабораторная работа № 6</i>	
ИСПОЛЬЗОВАНИЕ ГИПЕРССЫЛОК И ДОБАВЛЕНИЕ ИЗОБРАЖЕНИЙ В CSS ..	65
6.1. Типовое задание .....	81
6.2. Контрольные вопросы.....	83
6.3. Контрольные задания.....	84
<i>Лабораторная работа № 7</i>	
ПОСТРОЕНИЕ СПИСКОВ .....	85
7.1.Типовое задание .....	87
7.2. Контрольные вопросы.....	90
7.3. Контрольные задания.....	90

<i>Лабораторная работа № 8</i>	
ПОСТРОЕНИЕ СПИСКОВ В КАСКАДНЫХ ТАБЛИЦАХ .....	95
8.1. Типовое задание .....	99
8.2. Контрольные вопросы.....	101
8.3. Контрольные задания.....	101
<i>Лабораторная работа № 9</i>	
ОРГАНИЗАЦИЯ И ФОРМИРОВАНИЕ ТАБЛИЦ.....	102
9.1. Типовое задание .....	103
9.2. Контрольные вопросы.....	105
9.3. Контрольные задания.....	105
<i>Лабораторная работа № 10</i>	
СОЗДАНИЕ ТАБЛИЦ С ПОМОЩЬЮ CSS.....	108
10.1. Типовое задание .....	118
10.2. Контрольные вопросы.....	121
10.3. Контрольные задания.....	121
<i>Лабораторная работа № 11</i>	
ПОСТРОЕНИЕ ФОРМ .....	123
11.1. Типовое задание .....	128
11.2. Контрольные вопросы.....	130
11.3. Контрольные задания.....	130
<i>Лабораторная работа № 12</i>	
СОЗДАНИЕ ФОРМ С ИСПОЛЬЗОВАНИЕМ СТИЛЕЙ .....	133
12.1. Типовое задание .....	142
12.2. Контрольные вопросы.....	144
12.3. Контрольные задания.....	144
<i>Лабораторная работа № 13</i>	
ОРГАНИЗАЦИЯ ФРЕЙМОВ.....	145
13.1 Типовое задание .....	146
13.2. Контрольные вопросы.....	149
13.3. Контрольные задания.....	149
<i>Лабораторная работа № 14</i>	
СОЗДАНИЕ МАКЕТА САЙТА С ФРЕЙМОФНОЙ СТРУКТУРОЙ В РЕДАКТОРЕ MICROSOFT FRONTPAGE .....	152
14.1. Типовое задание .....	157
14.2. Контрольные вопросы.....	170
14.3. Контрольные задания.....	170
<i>Лабораторная работа № 15</i>	
СОЗДАНИЕ ПАНЕЛИ ГИПЕРССЫЛОК ДЛЯ НАВИГАЦИИ ПО СТРАНИЦАМ САЙТА .....	171
15.1. Типовое задание .....	176
15.2. Контрольные вопросы.....	184
15.3. Контрольные задания.....	184

*Лабораторная работа № 16*

СОЗДАНИЕ WEB-СТРАНИЦ В РЕДАКТОРЕ MICROSOFT PUBLISHER.....	186
16.1. Типовое задание .....	187
16.2. Контрольные вопросы.....	206
16.3. Контрольные задания.....	207

*Лабораторная работа № 17*

ТЕХНОЛОГИЯ ПРОЕКТИРОВАНИЯ WEB-СТРАНИЦ СРЕДСТВАМИ ОФИСНЫХ ПРИЛОЖЕНИЙ.....	208
17.1. Типовое задание .....	209
17.2. Контрольные вопросы.....	217
17.3. Контрольное задание .....	218
ЛИТЕРАТУРА .....	219
ПРИЛОЖЕНИЯ.....	220
Приложение А. Структура отчета .....	220
Приложение Б. Основные терминологические понятия .....	221
Приложение В. Краткая справка по использованию языка HTML.....	222
Приложение Г. Список основных параметров для задания стилей .....	227

## ВВЕДЕНИЕ

В пособии рассматриваются основные возможности языка разметки гипертекстовых документов HTML (HyperText Markup Language): форматирование текста, построение таблиц и списков, организация гиперссылок, фреймов и форм, таблиц, стилей, которые используются при построении Web-страниц.

Вследствие постоянного развития HTML-технологии, возникновения новых свойств и параметров, а также появления новых версий популярных браузеров, возможности HTML приобретают все большее значение в ходе создания Web-сайтов различной сложности и тематики. Знание основ языка HTML становится обязательным и неотъемлемым атрибутом многих специалистов в области интернет-технологий, превращается в систему знаний, необходимых практически каждому пользователю Всемирной сети.

Пособие содержит лабораторные работы, структурное содержание которых одинаковое. В начале каждой из них дается краткий теоретический материал, затем рассматриваются типовые задания по соответствующей теме и их решения на языке HTML с подробным объяснением, конце предлагаются варианты заданий практического характера.

Пособие может быть полезно при изучении дисциплин «Компьютерные информационные технологии», «Системы и средства компьютерных телекоммуникаций в бизнесе».

## Лабораторная работа № 1

### СТРУКТУРА ГИПЕРТЕКСТОВОГО ДОКУМЕНТА

**Цель работы:** приобрести начальные навыки создания простейших Internet-документов; научиться выполнять форматирование созданных Web-страниц.

#### Методические указания

##### 1. Создание Web-документов

Управляющие конструкции языка HTML (Hyper Text Markup Language) называются тегами (дескрипторами) и вставляются непосредственно в текст документа. Все теги заключаются в угловые скобки <...>. Сразу после открывающей скобки помещается ключевое слово, определяющее тег, например <div>.

Теги HTML бывают парными и непарными. Непарные теги оказывают воздействие на весь документ или определяют разовый эффект в том месте, где они вставлены. При использовании парных тегов в документ добавляются открывающий и закрывающий теги, которые воздействуют на часть документа, заключенную между ними. Закрывающий тег отличается от открывающего наличием символа "/" перед ключевым словом (</div>).

Закрытие парных тегов выполняется так, чтобы соблюдались правила вложения: <B><I>Текст</I></B>

К открывающему тегу может быть добавлен атрибут, представляющий собой дополнительные ключевые слова, отделяемые от ключевого слова, определяющего тег, и от других атрибутов пробелами и размещаемые до завершающего тег символа ">". Способ применения некоторых атрибутов требует указания значения атрибута.

Значение атрибута отделяется от ключевого слова атрибута символом "=" и заключается в кавычки: <H1 align="left">

При отображении документа HTML сами теги не отображаются, но влияют

на способ отображения документа *HTML*.

Существует два способа формирования документов *HTML*.

Первый способ состоит в разметке существующего или создаваемого документа вручную. Эта работа выполняется в текстовом редакторе или редакторе *HTML*.

Второй способ заключается в формировании документа непосредственно на экране и автоматической его разметке. В этом способе необязательно знание языка *HTML*. Разметка выполняется специальным редактором, работающим по принципу **WYSIWYG**, например, *FrontPage Express*.

Во втором способе используются средства форматирования вместо средств описания, что может приводить к нежелательным последствиям.

Воспроизведение документа *HTML* выполняется программами – браузерами, например *Internet Explorer*.

Документ *HTML* является блочным элементом и сам состоит из блоков. Два основных блока – это *head* (определяет свойства документа) и *body* (определяет тело документа). Каждый из них включает другие блоки.

*HTML*-документ состоит из стандартных элементов разметки: заголовки, списки, таблицы, параграфы и т.д., которые разделены на два типа: строчные и блочные. К блочным относятся параграф, список, таблица. К строчным относятся начертание, текст гипертекстовых ссылок.

## **2. Определение *HTML* – файла**

Первый тег, который должен находиться в любом документе, - это `<html>...</html>`. Этот тег указывает на то, что в данный документ содержится в себе *HTML*-текст. Все, что вы напишите в своем документе, должно находиться внутри данного тега:

```
<html>
...
  текст документа
...
</html>
```



Вы указали, что ваш документ содержит команды *HTML*, вы должны разбить документ на две части – заголовок и собственно текст.

### 3. Определение заголовка

Определение заголовка должно содержаться внутри тега `<head>...</head>`:

```
<html>
<head>..
описание заголовка...
</head>...
    текст документа...
</html>.
```

В разделе описания заголовка можно указать заглавие документа, для этого используется тег `<title>...</title>`. Когда браузер встречает этот тег, он отображает все, что находится внутри него, как заглавие.

Пример того, как используется данный тег:

```
<html>
<head>
<title>Это заглавие документа</title>
</head>
текст документа
...
</html>
```

### 4. Определение тела документа

Весь остальной *HTML*-документ, включая весь текст, содержится внутри тега. Теперь наш документ выглядит примерно так:

```
<html>
<head>
<title>Это заглавие документа</title>
</head>
<body>
...

```

текст документа

...

</body>

</html>

## 5. Правила синтаксиса

1. Взаимное расположение элементов **HTML, HEAD, TITLE, BODY** должно быть стандартным на любой странице.

<html>

<head>

<title>Это заглавие документа</title>

</head>

...

текст документа

...

</html>.

2. Всегда использовать конечные теги (не забывать </p>, </h1>, </table> и др.).

3. Не нарушать правила вложения тегов.

Правильно: <H1>Заголовок крупный <H2> Заголовок поменьше </h2>  
</h1>

Неправильно:<H1>Заголовок крупный <H2> Заголовок поменьше  
</h1> </h2>

4. Любая полезная информация должна находиться между начальным и конечным тегами, указывающими ее формат.

5. Все атрибуты располагаются в начальном теге.

6. Принято начальные теги писать ЗАГЛАВНЫМИ буквами (<H1>), а конечные строчными буквами (</h1>).

7. К структуре HTML документа часто относят тег <Meta>, который содержит специальную служебную информацию, не отображающуюся при просмотре Web-страницы.

Тег < Meta > обычно содержит атрибуты со следующим назначением:

*name = "Expires" content = "Дата"* – «срок годности» документа;  
*name = "Author" content = "Имя автора"* – имя автора Web-страницы;  
*http - equiv - "Content - Type"* – используется для указаний инструкций пользователю;

*content = "text/html; charset = windows - 1251"* – указание на то, что в документе содержится текст в форме **HTML** и определена кодовая страница

К базисным элементам гипертекстового документа относятся:

- признаки начала и конца документа;
- область заголовка документа;
- имя документа;
- тело документа;
- комментарии;
- элементы заголовка;
- горизонтальные линии.

1. `<html>` – тег указывает на начало HTML-документа.

2. `<head>` – тег указывает на начало области заголовка Web-страницы.

Служит для формирования областей структуры документа.

3. `<title> Структура Web-страницы</title>` – элемент для размещения заголовка Web-страницы. Строка отображается в заголовке окна браузера.

4. `<Meta http-equiv="Content-Type" content="text/html; charset=windows-1251">` – этот тег несет служебную информацию и не отображается в экране браузера. В данном случае идет речь о кодировке Web-странички. Достаточно каждый раз вставлять этот тег в таком виде на свою страничку, тогда ваша страничка будет использовать кодировку windows-1251, наиболее распространенную на сегодняшний день.

5. `<Meta name="Author" content="Ivanov Ivan">` – имя автора Web-страницы.

6. `<Meta name="Keywords" content="WWW, HTML, document, страничка, структура">` – набор ключевых слов для поиска. Раньше этими словами

пользовались поисковые машины, для отбора сайтов по тематике запроса.

Сегодня эти слова поисковыми машинами практически не используются, однако полезно вставить этот тег на свою страничку и указать в нем ключевые слова, отражающие содержание вашего сайта.

7. `</head>` – конец области заголовка Web-страницы.

8. `<body >` – начало собственно содержимого Web-страницы, может содержать дополнительные атрибуты для цветового оформления Web-страницы: *bgcolor, text, link, vlink*.

9. `</ body >` – конец содержимого Web-страницы.

10. `</текст комментария>` – тег, позволяющий формировать комментарии, которые браузером игнорируются и не отображаются.

11. `</html>` – конец HTML-документа.

### ***1.1. Типовое задание***

***Пример 1.*** Создайте простейшие HTML-файлы.

1. Запустите программу Блокнот (Notepad).

2. Наберите в окне редактора простейший текст файла HTML, т.е. следующий код:

```
<html>
  <head>
    <title>Учебный файл HTML</title>
  </head>
  <body>
    Расписание занятий на вторник
  </body>
</html>.
```

3. Сохраните код под любым именем с расширением .html или .htm. Сохраните файл в своей папке под именем *RASP.htm*.

4. Для просмотра созданной Web-страницы загрузите браузер Microsoft Internet Explorer.

5. Выполните команду ***Файл(File) / Открыть(Open) / Просмотр (Обзор, Browse)***, найдите в своей папке файл *RASP.htm* и загрузите его. Убедитесь, что

название Web-страницы ("*Учебный файл HTML*"), отразилось в верхней, статусной строке браузера. На экране вы увидите результат своей работы.

Результат отображения HTML-документа в браузере приведен на рис. 1.1.

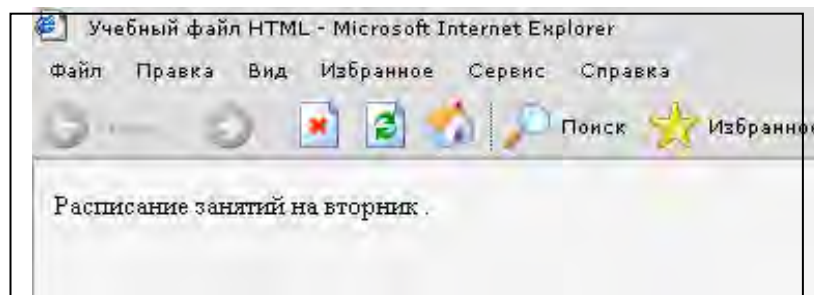


Рис. 1.1. Отображение HTML-документа в браузере

**Пример 2.** Вставить в созданный документ метатеги.

1. `<meta http-equiv="content-type" content="text/html; charset=Кодировка">`

При указании этого мета тега браузер будет правильно отображать текст страницы: `<meta http-equiv="content-type" content="text/html; charset=Windows-1251">`

2. `<meta name="author" content="имя автора">`

Этот мета тег указывает имя автора сайта.

Например: `<meta name="author" content="Петров ">`

**Пример 3.** Управление расположением текста на экране.

При отображении HTML-документов браузеры автоматически размещают текст на экране, *не принимая* во внимание встречающиеся в файле переводы строк и идущие подряд пробелов.

1. При необходимости выполните п. 5 примера 1.

2. Откройте первоисточник Web-страницы – в меню браузера *Вид (View)*, *Источник (в виде HTML)* откроется окно со стандартной программой *Блокнот (Notepad)*, в котором ваша Web-страница представлена в командах HTML.

3. Внесите изменения в текст файла **HTML**, расположив слова "Расписание", "занятий", "на вторник" на разных строках:

```
<html>
<head>
<title> Учебный файл HTML </title>
</head>
<body>
    Расписание
    занятий
    на вторник
</body>
</html>
```

4. Сохраните внесенные изменения в файле *RASP.htm*, с помощью команд *Файл (File)*, *Сохранить (Save)*. Закройте программу Блокнот (Notepad).

5. Просмотрите с помощью браузера Microsoft Internet Explorer новую полученную Web-страницу используя клавишу F5 или с помощью команд *Вид (View)*, *Обновить (Refresh)*. Изменилось ли изображение текста на экране?

*Примечание.* В дальнейшем после внесения изменений в Web-страницу всегда выполняйте п.п. 4-5.

### ***1.2. Контрольные вопросы***

1. Что такое гипертекст?
2. Какова структура простейшего HTML-документа.
3. Как задается начало текстового абзаца?
4. Какие возможности начертания текста предоставляет язык HTML?
5. В чем заключается разметка гипертекста средствами HTML?
6. Дайте характеристику программам, открывающим документы HTML.
7. Чем непарные теги отличаются от парных?
8. Что такое атрибуты тега? Как они записываются?
9. Для чего используются мета теги?

### ***1.3. Контрольные задания***

1. Пользуясь средствами Проводника, создайте в своей рабочей папке файловую структуру, приведенную на рис. 1.2.

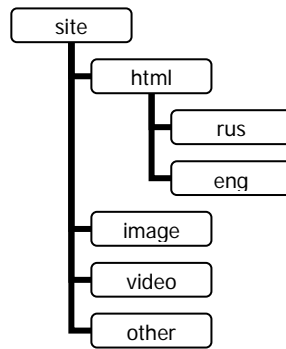


Рис. 1.2

2. Средствами приложения блокнот создайте HTML документ, при просмотре которого в браузерах в строке заголовка окна отображается текст «Содержимое строки заголовка», в основной области – текст «*так выглядит документ HTML*».

3. Сохраните документ под именем **grop.html** в папке **site\html\rus**.

4. Измените документ **grop.html** таким образом, чтобы при просмотре в браузере в строке заголовка отображался текст «*Общий вид документа html*», а в основной части документа – текст:

*«Вот некоторые приёмы работы,  
которые мы освоим:  
разбиение текста  
выравнивание текста  
создание заголовков  
работа с таблицами».*

5. Сохраните изменения. **Файл/ Сохранить.**

**Примечание.** Для просмотра изменений выполните команду **Вид/ Обновить.**

6. Просмотрите изменённый документ в браузере, при необходимости отредактируйте файл при помощи "Блокнота".

*Лабораторная работа № 2*  
**КАСКАДНЫЕ ТАБЛИЦЫ СТИЛЕЙ**

**Цель работы:** приобрести начальные навыки создания простейших Internet-документов, используя таблицы стилей.

***Методические указания***

Каскадные таблицы стилей применяются для оформления Web-страниц. Назначить определенный стиль для какой-либо части Web-документа можно тремя способами.

**1. Встроенный стиль**

*Синтаксис:*

`<тэг style="свойство: значение"> текст </тэг>`

Данный стиль будет применяться для определенного места в документе. Для изменения стиля нескольких слов или даже символов можно использовать атрибут style совместно с тэгом `<span>`, например:

Часть текста `<SPAN STYLE="color: blue">голубого</SPAN>` цвета

Результат отображения: Часть текста **голубого** цвета

**2. Вложенный стиль**

*Синтаксис:*

```
<head>
  <style type="text/css">
    селектор1 {свойство: значение}
    селектор2 {свойство: значение}
    ...
  </style>
</head>
```

Данный стиль будет применяться для фрагментов документа, выделенных с помощью отмеченных селекторов.

Селектором называется имя стилей, в котором указаны параметры форматирования. Селекторы делятся на несколько типов: селекторы тегов,



классы и идентификаторы.

В качестве селектора может использоваться любой из тэгов языка HTML, если к нему возможно употребить какое-нибудь свойство.

Селектор может состоять из нескольких тэгов (контекстный селектор). В этом случае значение указанного свойства будет применено только к тому фрагменту документа, который помечен сразу всеми участвующими в декларировании тэгами с соблюдением указанной последовательности вложения тэгов.

Классы используются для создания стилей, которые можно применять к любому тегу HTML, для выделений или изменения стиля блока текста; они обозначаются точкой перед именем. Обращение к селектору класса происходит через параметр `class`, значением которого выступает имя класса без точки.

Спецификация CSS позволяет назначать различные декларации одинаковым селекторам. Для этого селектору сопоставляются различные классы, каждому из которых назначается своя декларация.

*Синтаксис:*

**селектор.класс {свойство: значение}**

Можно также описать класс без явного указания селектора.

*Синтаксис:*

**.класс {свойство: значение}**

Связывание какого-либо тэга с объявленным классом осуществляется следующим образом.

*Синтаксис:*

**<тэг class="класс"> текст </тэг>**

*Идентификаторы* обычно используются совместно с клиентскими программами, чтобы через них можно было динамически управлять параметрами стиля; для определения того, что это идентификатор, к имени добавляется символ решетки. Стиль любого тега можно связать с идентификатором, если к тегу добавить параметр **id** с указанием имени идентификатора без символа решетки.

Идентификатор отличается от класса только тем, что его нельзя применить к какому-то конкретному элементу HTML. Стиль, описываемый с помощью идентификатора, можно применять к любой метке.

```
<html>
<head>
<style type="text/css">
#cursive { font-style: italic }           /* Курсивный текст */
#oblique { font-style: oblique } /* Наклонный текст */
}
</style>
</head>
<body>
<p id=cursive><span style="font-style: normal"> Курсивный текст </span>
хорошо использовать для цитат и прямой речи </p>
<p id=oblique>Наклонный шрифт пригодится для создания ненавязчивого
выделения в тексте </p>
</body>
</html>
```

Идентификаторы и классы обычно выполняют одну роль, поэтому они одинаково часто встречаются в текстах примеров.

### 3. Присоединенный стиль

*Синтаксис:*

```
<head>
<link rel="stylesheet" type="text/css"
href="file.css">
</head>
```

Данный стиль описан в отдельном файле (**file.css**). Файл может располагаться на другом компьютере в сети. В этом случае атрибут **href** содержит ссылку на URL нужного компьютера.

Представленные выше три метода назначения стилей описывают иерархию стилевых решений, которая вступает в силу, когда разные декларации предписываются одним и тем же тэгам. В этом случае приоритет отдается тому решению, которое занимает более высокий уровень иерархии. Если же назначаемые декларации стоят на одной ступени иерархии стилей, предпочтение

отдается последнему описанному стилевому решению. Однако значение любого свойства в декларации можно сделать обязательным, т.е. назначаемым безусловно.

*Синтаксис:*

{свойство: значение!important}

## **2.1. Типовое задание**

**Пример 1.** Вывод всех фрагментов документа, отмеченных тэгом **h1**, полужирным шрифтом.

```
<head>
  <style type="text/css">
    h1 {font-weight: bold}
  </style>
</head>
<body>
  <h1>здесь используются стили</h1>
  здесь стилей нет
</body>
```

**Пример 2.** Группировка селекторов, имеющих одинаковую декларацию.

Три одинаковых декларации

```
h1 {font-weight: bold}
h2 {font-weight: bold}
h3 {font-weight: bold}
```

могут быть заменены одной конструкцией вида

```
h1, h2, h3 {font-weight: bold}
```

Стиль в фигурных скобках можно применять к форматированию заголовков сразу трех уровней.

Можно также сделать так, чтобы стиль применялся только к некоторой последовательности тегов. Например, чтобы весь курсив внутри абзаца был окрашен в зеленый цвет. В этом случае выполняется следующая

последовательность тегов: сначала открывается абзац, а потом начинается курсив.

```
<STYLE TYPE="text/css">
P I { color: green }
</STYLE>
```

Если курсивный текст расположен вне абзаца, то стиль к нему применен не будет.

**Пример 3.** Использование контекстного селектора.

```
<head>
  <style type="text/css">
    b i {color: red}
  </style>
</head>
<body>
  <b>здесь стилей нет</b>
  <i>здесь стилей нет</i>
  здесь стилей нет
  <b><i>здесь используются стили</i></b>
  <i><b>здесь НЕ ИСПОЛЬЗУЮТСЯ стили</b></i>
</body>
```

**Пример 4.** Назначение ряда свойств одному тэгу.

```
<head>
  <style type="text/css">
    h1 {font-weight: bold;
        font-size: 20pt}
  </style>
</head>
<body>
  <h1>здесь используются стили</h1>
  здесь стилей нет
</body>
```

**Пример 5.** Использование классов селекторов.

```
<head>
  <style type="text/css">
    .red {color: red}
    .blue {color: blue}
    .green {color: green}
  </style>
</head>
<body>
  <h1 class="red">красный заголовок</h1>
  <h1 class="green">зеленый заголовок</h1>
  <div class="blue">синий текст</div>
</body>
```

**Пример 6.** Использование иерархии стилей.

```
<head>
  <style type="text/css">
    .red {color: red}
    .blue {color: blue!important}
  </style>
</head>
<body>
  <h1 class="red" style="color: green">
    зеленый заголовок </h1>
  <h1 class="blue" style="color: green">
    синий заголовок </h1>
</body>
```

## 2.2. Контрольные вопросы

1. Основные методы применения таблиц стилей для оформления Web-страниц.
2. Типы селекторов и их назначение.
3. Иерархия стилевых решений.

## 2.3. Контрольные задания

1. Выделите в окне браузера произвольный текст без форматирования. Задайте цвет шрифта, используя встроенный стиль в соответствии с вариантом.

	Вариант					
	1	2	3	4	5	6
Цвет	lime	maroon	silver	teal	navy	fuchsia

2. Используя идентификатор, задайте курсивное начертание для следующего абзаца текста.
3. Выведите в окне браузера заголовок H1 без форматирования. Используя селекторы, задайте для заголовка следующие параметры форматирования.

	Вариант					
	1	2	3	4	5	6
Цвет	green	red	teal	yellow	red	blue
Размер шрифта	20pt	12pt	14pt	16pt	22pt	18pt

4. Используя классы селекторов, задайте цвет текста для следующего абзаца в соответствии с вариантом.

	Вариант					
	1	2	3	4	5	6
Цвет	maroon	navy	fuchsia	pink	silver	olive

5. В виде отдельного файла mystyle.css создайте таблицу стилей, в которой задайте контекстный селектор. Примените его к произвольному фрагменту документа, который помечен сразу всеми участвующими в декларировании тэгами с соблюдением указанной последовательности вложения тэгов.

*Лабораторная работа № 3*  
**ФОРМАТИРОВАНИЕ ТЕКСТА**

**Цель работы:** приобрести начальные навыки создания простейших Internet-документов; научиться выполнять форматирование созданных Web-страниц.

**Методические указания**

Текст Web-страницы – произвольные символы, определяемые по умолчанию как текстовые данные.

Основные группы тегов, выполняющих форматирование текста Web-страницы:

`<P>...</p>` – тег начала и конца абзаца. Данный тег может использовать атрибут *align*.

`<Pre>...</pre>` – тег вывода текста, определенного пользователем заранее;

`<Center>...</center>` – тег для центрирования текста любого содержания;

`<Div>...</div>` – тег центрирования текста в зависимости от значения атрибута *align*;

`<Big>...</big>` – тег увеличенного размера шрифта;

`<Small> ... </small>` – тег уменьшенного размера шрифта;

`<Tt>...</tt>` – тег для обозначения шрифта телетайпа (моноширинного шрифта);

`<Strike>...</strike>` – тег для создания перечеркнутого текста;

`<U>...</u>` – тег для формирования подчеркнутого текста;

`<I>...</i>` – тег вывода текста курсивом;

`<Sub>...</sub>` – тег для написания нижних индексов;

`<Sup>...</sup>` – тег для написания верхних индексов;

`<Font>...</font>` – тег для определения типа, размера и цвета шрифта.

Все характеристики тега *<Font>* определяются с помощью специальных атрибутов:

- *size* = абсолютный размер шрифта. Абсолютный размер принимает значения от 1 до 7. Размер шрифта может задаваться не в абсолютном, а в относительном виде по отношению к базовому размеру, например, *size* = + число или *size* = - число. Для базового размера, равного двум, относительный размер может находиться в пределах от -1 до +5.

- *color* = цвет – атрибут цвета шрифта.

*<Bdo>...</bdo>* – изменение направления текста на обратный с атрибутом *dir*.

Например:

*<Bdo dir = "RTL">* текст, направление которого меняется *</bdo>*

*<Marquee>...</marquee>* – текст на странице будет представлен в виде бегущей строки. Атрибут *direction* используется для определения движения текста слева направо или справа налево.

Например:

*< Marquee direction = left, right> текст </marquee>* – движение текста слева направо.

Для оформления цвет указывается по имени (английский эквивалент), например, red (красный), green (зеленый), blue (синий), или тремя двухразрядными шестнадцатеричными числами, которые определяют цвет и двумя двухразрядными шестнадцатеричными числами, которые определяют цвет их определяется парами шестнадцатеричных чисел соответственно. Названия цветов и их коды представлены в табл. 3.1.

Таблица 3.1

Русское название цвета	Английское название цвета	Код цвета
1	2	3
Аквамарин	Aqua	#00FFFF
Белый	White	#FFFFFF



1	2	3
Желтый	Yellow	#FFFF00
Зеленый	Green	#008000
Каштановый (бордовый)	Maroon	#800000
Красный	Red	#FF0000
Оливковый	Olive	#808000
Пурпурный	Purple	#800080
Светло-зеленый	Lime	#00FF00
Серебристый	Silver	#C0C0C0
Серый	Gray	#808080
Сизый	Teal	#008080
Синий	Blue	#0000FF
Ультрамарин	Navy	#000080
Фуксиновый	Fuchsia	#FF00FF
Черный	Black	#000000

### 1. Атрибуты тега < Body >

*bbgcolor* = *цвет* – цвет фона Web-страницы;

*text* = *цвет* – цвет текста на Web-странице;

*link* = *цвет* – цвет текста гиперссылок;

*vlink* = *цвет* – цвет использованных гиперссылок.

### 2. Атрибуты текста

Из параметров текста документа реально применяется только один – *text*, который задает цвет основного текста на странице (значение параметра может быть введено аналогично цвету фона документа): <BODY *text* = "black">.

### 3. Атрибуты гиперссылок

Параметры гиперссылок (связей с внутренними или внешними документами) определяют цвета активных(*alink*), не посещенных (*link*) и посещенных (*vlink*) ссылок:

```
<body link ="#0000FF" alink ="#0000FF" vlink ="blue">
```

Теги  $\langle H1 \rangle$ ,  $\langle H2 \rangle$ ...  $\langle H6 \rangle$  — тег  $\langle H \rangle$  с указанием одной из цифр (от 1 до 6) задает определенный размер заголовка всего текста целиком или его конкретного фрагмента. Заголовок  $\langle H6 \rangle$  будет минимальным, а  $\langle H1 \rangle$  — самым большим. Особенностью тегов  $\langle Hn \rangle$  является то, что они уже подразумевают отступ от текстовой части и разработчикам *HTML*-документов не приходится делать отступ самостоятельно

```
<html>
  <head>
    <title>Заголовки H1-H6</title>
  </head>
  <body bgcolor="#FFFFFF" text="black" link="#00FF00"
    alink="#00FF00" vlink="blue">
    <H1>Заголовок1</H1>
    <H2>Заголовок 2</H2>
    <H3>Заголовок 3</H3>
    <H4>Заголовок 4</H4>
    <H5>Заголовок 5</H5>
    <H6>Заголовок 6</H6>
  </body>
</html>
```

Для тегов вида  $\langle Hn \rangle$  обычно указывается одно из трех значений атрибута этого тега `align`:

*align = center* — выравнивание заголовка по центру;

*align = left* — выравнивание заголовка по левому краю;

*align = right* — выравнивание заголовка по правому краю.

Разбиение текста Web-страницы на части выполняют горизонтальные линии, которые описывает тег  $\langle Hr \rangle$ .

$\langle Hr \rangle$  — тег для задания горизонтальной линии и может использовать дополнительно атрибуты:

*align* = выравнивание линии;

*size* — толщина линии в пикселах;

*width* = длина линии в пикселах или процентах;

*color* = цвет линии.

#### 4. Атрибуты фона

Параметрами фона документа являются *bgcolor*, *background* и *backgroundproperties.bgcolor* устанавливает цвет фона, значение которого может быть введено в символьном эквиваленте, в шестнадцатеричном коде или в формате цветовой модели **RGB**.

Система указания цвета в **HTML** основана на трех основных цветах: красном, зеленом и синем (модель **RGB** — **Red, Green, Blue**). Любое значение RGB может быть преобразовано в шестнадцатеричный формат (от 00 до FF с приставкой # (читается "дизз")). Некоторым значениям упомянутых моделей соответствует символьное название цвета. Таким образом, один и тот же цвет можно указать тремя возможными способами.

Например, три варианта установки цвета фона (белого):

- `<body bgcolor="white">`
- `<body bgcolor="#FFFFFF">`
- `<body bgcolor="255,255,255">`

Параметр *backgroundproperties* позволяет накладывать на фон документа графическое изображение: `<body background="images/bg.gif">`.

Параметр *backgroundproperties* поддерживается только браузером Microsoft Internet Explorer и позволяет менять свойства фона документа.

Например, конструкция

```
< body background = "imagesbg.gif" bgpropeties="fixed">
```

позволит прокручивать содержание документа, оставляя фоновое графическое изображение в зафиксированном виде.

Описанные параметры не являются обязательными, однако рекомендуется использование *bgcolor*, так как пользователь в настройках своего браузера может поставить любой цвет фона, а разработчик, считая, что белый цвет является

основным по умолчанию, может не указать этот параметр. В результате вместо подразумеваемого белого цвета, фон может оказаться черным, зеленым и т. д., что способно привести к нарушению оформления документа. Также наряду с графическим изображением фона рекомендуется использовать и параметры цвета на тот случай, если рисунок не загрузится, тогда браузер отобразит цвет.

## 5. Атрибуты границ документа

Параметры границ HTML-документа создают отступы заданного размера от верхнего, нижнего, левого и правого краев документа, которым соответствуют параметры `topmargin`, `bottommargin`, `left-margin` и `rightmargin`.

Значения для них задаются в пикселах:

```
<body topmargin="5" bottommargin="5" leftmargin="10" rightmargin="10">
```

Однако данные параметры не воспринимаются некоторыми браузерами. В частности, Netscape объединяет упомянутые параметры в две группы – горизонтальные и вертикальные отступы:

```
<body marginwidth="10" marginheight="5">
```

Чтобы учесть особенности всех браузеров, то в HTML-конструкцию надо подставить и те, и другие параметры:

```
< body topmargin ="5" bottommargin ="5" leftmargin ="10" rightmargin ="10">
```

### 3.1. Типовое задание

**Пример 1.** Отформатируйте текст, как представлено ниже:

**Форматирование текста**

---

#### ВНЕШНИЙ ВИД

**полужирный шрифт**

*курсивный шрифт*      подчеркнутое      начертание

перечеркнутое начертание

Печатная машинка

Обычный текст и элемент Sub

Обычный текст и элемент Sup

---

#### ЗАДАНИЕ ЦВЕТА, РАЗМЕРА И ТИПА ШРИФТА

Задан зеленый цвет шрифта размера 4 типа Courier

---

#### ЗАДАНИЕ АБСОЛЮТНЫХ РАЗМЕРОВ ШРИФТОВ

---

Шрифт размера 7

Шрифт размера 6

Шрифт размера 5

Шрифт размера 4

Шрифт размера 3

Шрифт размера 2

Шрифт размера 1

---

## ЗАДАНИЕ ОТНОСИТЕЛЬНЫХ РАЗМЕРОВ ШРИФТОВ

---

Шрифт размера +4

Шрифт размера +3

Шрифт размера +2

Шрифт размера +1

Шрифт размера +0

Шрифт размера -1

Шрифт размера -2

---

Большой шрифт

(используется элемент Big)

Маленький шрифт (используется элемент Small)

---

## ОПРЕДЕЛЕНИЕ СТРУКТУРЫ ТЕКСТА

---

Заголовок 1

**Заголовок 2**

**Заголовок 3**

**Заголовок 4**

**Заголовок 5**

**Заголовок 6**

---

Использование элемента Center

---

Выравнивание текста (Div)

---

Элемент ***Br*** обеспечивает разрыв строки

---

Эта строка, хоть и длинная, не должна разрываться браузером, потому что текст этой строки находится внутри элемента ***Nobr***

---

Использование элемента ***Pre***.

Он запрещает форматирование  
текста браузером и выводит его так, как  
определил пользователь.

---

Направление текста можно изменить  
ьтинемзи онжом атскет еинелварпаН

---

**Пример 2.** Рассмотрите и проанализируйте решение практического задания:

```
<Html>
<Head>
<Title> Форматирование текста </title></head>
<Meta http-equiv="Content-Type" content="text/html; charset=windows-1251 ">
<! цвет фона серебристый, цвет текста синий >
<Body bgcolor=silver text=blue>
<Center>
<Font size=6 color="navy"> <B>Ф</b></font>
<Font size=5><I>ормат<Font color="lime">
<Sub>ирование </sub></i> </font>
<Font size=2 color="purple">m</font><Font size=3
color="blue">e</font><Font size=4color="green">к</font>
<Font size=5 color="maroon">c</font><Font size=6 color="fuchsia">m</font>
<Font size=7 color="orange">a</font>
</center>
<! горизонтальная линия>
<Hr align=center size=6 color=maroon>
<Center>
<Font color=red><H3> ВНЕШНИЙ ВИД</H3> </font>
</center>
<Hr>
<P><B>полужирный шрифт </b>
<P><I>курсивный шрифт</i>
<P><U>подчеркнутое начертание</u>
<P><Strike>перечеркнутое начертание</strike>
<P><Tt> Печатная машинка</tt>
<P> Обычный текст и <Sub> элемент Sub </sub>
<P> Обычный текст и <Sup> элемент Sup </sup>
<Hr size=4 color=fuchsia>
<Center>
<Font color =red>
<H3>ЗАДАНИЕ ЦВЕТА, РАЗМЕРА И ТИПА ШРИФТА</H3>
</font>
</center>
<Hr>
<P><Font size=4 color="green" face="Courier"> Задан зеленый цвет шрифта
размера 4 типа Courier</font>
<Hr size=4 color=fuchsia>
<Center>
<Font color=red>
<H3>ЗАДАНИЕ АБСОЛЮТНЫХ РАЗМЕРОВ ШРИФТОВ </H3>
</font>
```

```

</center>
<Hr>
<P><Font size=7> Шрифт размера 7</font>
<P><Font size=6> Шрифт размера 6</font>
<P><Font size=5> Шрифт размера 5</font>
<P><Font size=4> Шрифт размера 4</font>
<P><Font size=3> Шрифт размера 3</font>
<P><Font size=2> Шрифт размера 2</font>
<p> <Font size=1 > Шрифт размера 1 </font>
<Hr size=4 color =fuchsia>
<Center>
<Font color=red>
<H3>ЗАДАНИЕ ОТНОСИТЕЛЬНЫХ РАЗМЕРОВ ШРИФТОВ</H3>
</font>
</center>
<Hr>
<P><Font size=+4> Шрифт размера +4</font>
<P><Font size=+3> Шрифт размера +3</font>
<P><Font size=+2> Шрифт размера +2</font>
<P> <Font size~+1 > Шрифт размера +1 </font>
<P><Font size=+0> Шрифт размера +0</font>
<P><Font size=-1> Шрифт размера -1</font>
<P><Font size=-2> Шрифт размера -2</font>
<Hr size=4 color=fuchsia>
<P> <Big> Большой шрифт (используется элемент Big) </big>
<P> <Small> Маленький шрифт (используется элемент Small)
</small>
<Hr size=4 color=fuchsia>
<Center>
<Font color=red><H3>ОПРЕДЕЛЕНИЕ СТРУКТУРЫ ТЕКСТА</H3>
</font>
</center>
<Hr>
<H1>Заголовок 1</h1>
<H2>Заголовок 2</h2>
<H3>Заголовок 3</h3>
<H4>Заголовок 4</h4>
<H5>Заголовок 5</h5>
<H6>Заголовок 6</h6>
<Hr>
<P> <Center> Использование элемента Center </center>
<Hr>
<P> <Div align="right">Выравнивание текста (Div) </div>
<Hr>

```

`<P>` Элемент `Br` обеспечивает разрыв `<Br>` строки `<Hr>`  
`<P><Nobr>` Эта строка, хоть и длинная, не должна разрываться браузером, потому что текст этой строки находится внутри элемента `Nobr` `</nobr>`  
`<Hr>`  
`<Pre>` Использование элемента `Pre`.  
Он запрещает форматирование текста браузером и выводит его так, как определил пользователь. `</pre>`  
`<Hr>`  
`<P>`Направление текста можно изменить  
`<Bdo dir="RTL">`Направление текста можно изменить `</Bdo>`  
`<Hr size=4 color=fuchsia>`  
`</Body></HTML>`

### 3.2. Контрольные вопросы

1. Структура HTML-документа.
2. Использование цвета в HTML-документах.
3. Элементы языка HTML, задающие шрифт.
4. Специальные тэги языка HTML.
5. Можно ли на языке HTML задать бегущую строку?

### 3.3. Контрольные задания

Отформатируйте текст по указанному ниже образцу, после его заглавия организуйте бегущую строку: «Работа выполнена студентом: Фамилия имя», каждое слово заглавия наберите своим цветом, в конце текста проведите горизонтальную линию толщиной, равной номеру варианта. Оформите фон вашего сайта, используя готовый background, который будет найден на Web-странице Internet.

#### **Вариант 1**

ASKING t<sup>h</sup>e W<sub>ay</sub>

- <sup>1</sup> Excuse me, <sup>2</sup> can you tell me where Prater Street is, please?
- <sup>3</sup> Take the second on the left and then ask again.
- <sup>4</sup> Is it far?
- <sup>5</sup> No, it's only about five minutes' walk.



— <sup>6</sup> Thank you.

— <sup>7</sup> That's OK

---

<sup>1</sup> набрать большим шрифтом, красным цветом, курсивом;

<sup>2</sup> вывести, используя тег *<Pre>*;

<sup>3</sup> выровнять по правому краю, набрать шрифтом размера +3, чередовать цвета букв;

<sup>4</sup> изменить направление текста;

<sup>5</sup> набрать шрифтом с перечеркнутым начертанием, разорвать строку после слова «only»;

<sup>6</sup> набрать малым шрифтом типа Arial и выровнять по центру;

<sup>7</sup> набрать, как заголовок 2.

### ***Вариант 2***

**DISTANCE** C<sub>A</sub><sup>LL</sup>

— <sup>1</sup> Number, please.

— <sup>2</sup> I'd like to make a long-distance call to Minsk 00375172777222. Make it personal, please.

— <sup>3</sup> Who do you want to speak to?

— <sup>4</sup> Sergei.

— <sup>5</sup> What is your number, please?

— <sup>6</sup> This is Ivanov. My room number is 201, <sup>7</sup>the Forum Hotel.

---

<sup>1</sup> набрать малым шрифтом, типа Courier, желтого цвета; слово «please» вывести как нижний индекс;

<sup>2</sup> набрать большим шрифтом, предотвратить разрыв этой строки браузером;

<sup>3</sup> набрать шрифтом с перечеркнутым начертанием, изменить направление текста;

<sup>4</sup> выровнять по правому краю, набрать, как заголовок 1;

<sup>5</sup> набрать полужирным шрифтом, разорвать строку после слова «number»;

<sup>6</sup> набрать шрифтом печатной машинки, чередовать цвета букв;

<sup>7</sup> вывести, используя тег *<Pre>*.

### Вариант 3

## FEELING UNWELL

— <sup>1</sup> Doctor, I feel shivery and I've got a pain in my stomach.

— <sup>2</sup> Where does it pain you?

— <sup>3</sup> Here, doctor.

— <sup>4</sup> How long have you had it?

— <sup>5</sup> It came on yesterday.

— <sup>6</sup> What have you been eating of late?

— <sup>7</sup> Well, I think, doctor, I've overeaten fruits and vegetables these days.

— <sup>8</sup> I'll give you a prescription and you'll buy these capsules at a chemist's shop.

---

<sup>1</sup> набрать шрифтом печатной машинки, разорвать строку после слова «and»;

<sup>2</sup> вывести, используя тег <Pre>;

<sup>3</sup> набрать шрифтом размера +4 с перечеркнутым начертанием, выровнять по центру, чередовать цвета букв;

<sup>4</sup> набрать шрифтом типа Courier, синего цвета, фразу «have you had it» вывести как нижний индекс;

<sup>5</sup> набрать курсивным шрифтом, изменить направление текста;

<sup>6</sup> выровнять по правому краю; фразу «you been eating» вывести как верхний индекс;

<sup>7</sup> набрать, как заголовок 2, шрифтом с подчеркиванием, зеленого цвета;

<sup>8</sup> набрать большим шрифтом, предотвратить разрыв этой строки браузером.

### Вариант 4

## IN A CAFE

— <sup>1</sup> What are you going to have, sir?

— <sup>2</sup> A bottle of pineapple juice <sup>3</sup> for me, <sup>4</sup> please.

— <sup>5</sup> Cakes? Biscuits? A slice of fruit cake? A slice of chocolate tart?

— <sup>6</sup> I think I'll have a slice of chocolate tart.

— <sup>7</sup> Right. In a minute, sir.

---

- <sup>1</sup> выровнять по правому краю, набрать, как заголовок 4, шрифтом размера +3 с перечеркнутым начертанием, чередовать цвета букв;
- <sup>2</sup> вывести, используя тег <Pre>;
- <sup>3</sup> набрать курсивным шрифтом, увеличивая каждый раз размер букв;
- <sup>4</sup> слово «please» вывести как верхний индекс;
- <sup>5</sup> набрать большим шрифтом, предотвратить разрыв этой строки браузером;
- <sup>6</sup> набрать шрифтом типа Arial, синего цвета; фразу «I'll have» вывести как нижний индекс, разорвать строку после слова «have»;
- <sup>7</sup> набрать шрифтом печатной машинки.

### **Вариант 5**

#### **BUYIGN S<sup>O</sup>U<sub>v</sub>E<sup>N</sup>IR**

- <sup>1</sup> Good morning, sir. May I help you?
  - <sup>2</sup> Hello. I want a lips harmonica. Have you got any?
  - <sup>3</sup> Yes, of three colours: red, green and black.
  - <sup>4</sup> Will you, please, show me a green one?
  - <sup>5</sup> Here it is.
  - <sup>6</sup> Where is it made?
  - <sup>7</sup> In China.
  - <sup>8</sup> Well, I like it and I'll take it. I want 20 pieces. How much are they?
  - <sup>9</sup> 30 German marks.
  - <sup>10</sup> Where must I pay?
  - <sup>11</sup> The cash-desk is on your right.
- 

- <sup>1</sup> набрать маленьким шрифтом, типа Courier, красного цвета; фразу «May I help you?» вывести как нижний индекс;
- <sup>2</sup> набрать шрифтом печатной машинки, выровнять по правому краю;
- <sup>3</sup> набрать курсивным шрифтом, чередовать цвета букв;
- <sup>4</sup> вывести, используя тег <Pre>;
- <sup>5</sup> набрать шрифтом с перечеркнутым начертанием, уменьшая каждый раз

размер букв;

<sup>6</sup> центрировать, набрать, как заголовок 1;

<sup>7</sup> набрать шрифтом размера +2 с подчеркиванием, чередовать цвета букв;

<sup>8</sup> набрать большим шрифтом, предотвратить разрыв этой строки

браузером;

<sup>9</sup> набрать полужирным шрифтом размера –1, изменив направление текста;

<sup>10</sup> набрать шрифтом печатной машинки, разорвать строку после слова «*must*»;

<sup>11</sup> выровнять по правому краю, фразу «*on your right*» вывести как верхний индекс

### **Вариант 6**

#### **ACCOMMODATION AT A HOTEL**

— <sup>1</sup> Can I have a double room from now until Monday?

— <sup>2</sup> You can have Room 5, overlooking the sea.

— <sup>3</sup> What does it cost?

— <sup>4</sup> 40000 lire a day, not counting the service.

— <sup>5</sup> Can I have a look at this room, please?

— <sup>6</sup> Yes, of course. Come this way.

— <sup>7</sup> Can you show me something a little cheaper, but also with a view of the sea?

— <sup>8</sup> Yes, of course. I won't keep you a moment

---

<sup>1</sup> набрать шрифтом с перечеркнутым начертанием, увеличивая каждый раз размер и чередуя буквы;

<sup>2</sup> выровнять по правому краю, фразу «overlooking the sea» вывести как верхний индекс;

<sup>3</sup> набрать курсивным шрифтом размера +3, изменив направление текста;

<sup>4</sup> набрать шрифтом печатной машинки, разорвать строку после слова «not»;

<sup>5</sup> центрировать, набрать, как заголовок 2;

<sup>6</sup> вывести, используя тег `<Pre>`;

<sup>7</sup> набрать большим шрифтом, предотвратить разрыв этой строки браузером;

<sup>8</sup> набрать маленьким шрифтом, типа Courier, зеленого цвета; фразу «*of course*» вывести как нижний индекс.

## ФОРМАТИРОВАНИЕ ТЕКСТА С ПОМОЩЬЮ CSS

**Цель работы:** приобрести начальные навыки форматирования созданных Web-страниц с помощью таблиц стилей.

### Методические указания

#### 1. Форматирование абзаца

Авторы различных документов традиционно разделяют свои тексты на последовательности абзацев, отражающих некоторую законченную мысль.

Начало нового абзаца задается тэгом **<p>**. Абзацы отделяются друг от друга пустой строкой. Предоставляется возможным задавать следующие атрибуты формата абзаца.

Красная строка (отступ первой строки)

*Синтаксис:*

```
<p style="text-indent: значение"> текст </p>
```

Значение атрибута может быть задано:

- абсолютной величиной: **in** (дюйм), **cm** (сантиметр), **mm** (миллиметр), **pt** (1 пункт  $\approx 0,353$  мм), **pc** (1 пика = 12 pt);
- относительной величиной: **em** (вычисленное значение свойства **font-size** шрифта абзаца), **ex** (вычисленное значение свойства **x-height** шрифта абзаца), **px** (пиксел), **%** (процент ширины открытого окна браузера).

**Пример 1.** Отступ первой строки в 20 мм.

```
<p style="text-indent: 20 mm"> текст </p>
```

**Пример 2.** Отступ первой строки на расстояние, равное 20% открытого окна браузера.

```
<p style="text-indent: 20 %"> текст </p>
```

**Пример 3.** Отступ первой строки на расстояние на 20% превышающее размер шрифта абзаца.

```
<p style="text-indent: 1.2 em"> текст </p>
```

## 2. Поля (**margin**) и отступы (**padding**)

Горизонтальные и вертикальные отступы от края браузера до содержимого веб-страницы встроены в браузер по умолчанию. Отступы задаются параметрами **margin** и **padding**. Наличие двух атрибутов вместо одного требуют интересы разных браузеров, **margin** — Internet Explorer, а **padding** — Opera и Netscape. Совмещение разных параметров гарантирует, что показываться веб-страница в разных браузерах будет одинаково.

*Синтаксис:*

```
<p style="text-indent: значение"> текст </p>
```

Значение атрибута может быть задано:

- абсолютной величиной: **in** (дюйм), **cm** (сантиметр), **mm** (миллиметр), **pt** (1 пункт  $\approx 0,353$  мм), **pc** (1 пика = 12 pt);
- относительной величиной: **em** (вычисленное значение свойства **font-size** шрифта абзаца), **ex** (вычисленное значение свойства **x-height** шрифта абзаца), **px** (пиксел), **%** (процент).

*Пример 1.* Отступ первой строки в 20 мм.

```
<p style="text-indent: 20 mm"> текст </p>
```

Управлять отдельными отступами от разных краев экрана можно с помощью параметров **margin-top**, **margin-bottom**, **margin-right** и **margin-left**, которые соответственно изменяют расстояние от верхнего, нижнего, правого и левого края окна браузера (рис. 4.1). Прежде чем их использовать, следует задать нулевое значение параметрам **margin** и **padding** (рис. 4.1).

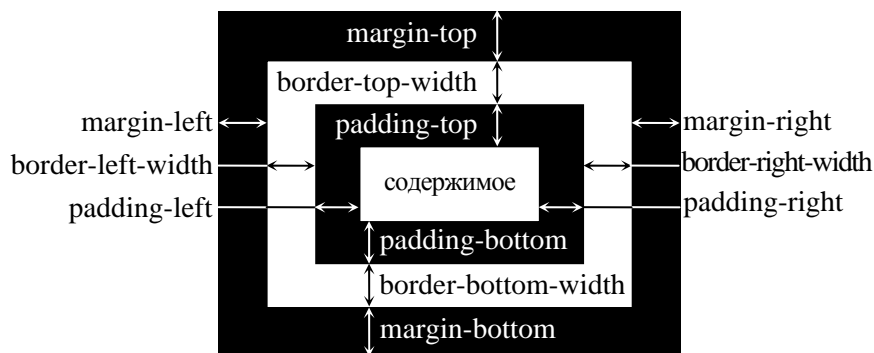


Рис. 4.1. Обозначения отступов от разных краев экрана

**Пример 2.** Изменение верхнего отступа от края браузера

```
<html>
<head>
<style type="text/css">
BODY {
    margin: 0px;      /* Убираем отступы */
    padding: 0px;    /* Убираем поля */
    margin-top: 10px /* Добавляем отступ сверху */
}
</style>
</head>
<body>
...
</body>
</html>
```

Селектор **BODY**, к которому применяется стиль, задает отступы на всей веб-странице целиком. То же можно сделать и применительно к отдельным элементам, например заголовкам (пример 3).

**Пример 3.** Изменение отступов у заголовка

```
<html>
<head>
<style type="text/css">
H1 { margin-top: 0px;      /* Отступ сверху */
      margin-bottom: 1em    /* Отступ снизу */}
</style>
</head>
<body>
<h1>Заголовок</h1>
```



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

Для сокращения кода можно использовать универсальный параметр `margin`, как показано в примере 4. Три значения разделенных пробелом определяют отступ сверху, одновременно слева и справа, и снизу.

**Пример 4.** Использование атрибута `margin`

```
<html>
<head>
<style type="text/css">
P {
margin: 0.5em 0px 1em      /* Отступ сверху, справа-слева и снизу */
}
</style>
</head>
<body>
<h1>Заголовок</h1>
<p>Основной текст</p>
</body>
</html>
```

### 3. Обрамление

Обрамление (**border**) – это формирование границы вокруг текстового блока.

Граница формируется с помощью трех параметров: толщина (`width`), стиль (`style`) и цвет (`color`). А объединяет эти три параметра свойство **border**:

**border-color:<Цвет>;border-width:<Значение>;border-style:<Стиль>.**

**Пример 1.** Создание границы вокруг текстового блока.

```
<STYLE TYPE="text/css">
P {color: red; border-style: solid; border-width: 1pt;}
</style>
```

Допустимые значения для **border-width**:

- **thin** – тонкая граница;
- **medium** – граница средней толщины;
- **thick** – толстая граница;

Допустимые значения для **border-style**:

- **none** – ну нет никакой границы (принято по умолчанию);
- **solid** – обычная граница **double** делает границу двойной;
- **inset** – создает эффект вдавленного текста;
- **outset** – эффект выпуклости;
- **groove** – это как будто сначала сделали **outset**, а потом – **inset**;
- **ridge** – то же, что и **groove**, только все цвета в границе – наоборот.

Можно для каждой стороны границы задать разную толщину. Необходимо вставить вместо свойства **border-width** соответствующие конструкции из схемы выше. Каждое из свойств задает толщину границы для соответствующей стороны. Если для какой-то из сторон толщина не задана, то она будет по умолчанию равна нулю.

#### 4. Создание линии возле текста

Для создания линий используется свойство CSS – **border**, которое устанавливает рамку вокруг блока. В частных случаях, для создания линии лишь с одной стороны элемента, используются параметры **border-bottom**, **border-top**, **border-left** и **border-right**, они соответственно задают линию внизу, сверху, слева и справа.

Значение этих атрибутов перечисляются через пробел и сразу устанавливают тип линии, ее толщину и цвет. Стиль линии может принимать одно из восьми значений, как показано на рис. 4.2, а толщина задается, как правило, в пикселах. В примере 1 показано создание вертикальной линии рядом с текстом.

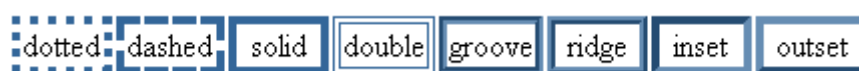


Рис. 4.2. Типы линий

### *Пример 1.* Вертикальная линия слева от текста

```
<html>
<head>
<style type="text/css">
P.line {
border-left: solid 2px black; /* Линия слева от текста */
margin-left: 20px;           /* Отступ слева от края до текста */
padding-left: 7px           /* Расстояние от линии до текста */
}
</style>
</head>
<body>
<p class="line"> Значение этих атрибутов перечисляются через пробел и сразу
устанавливают тип линии, ее толщину и цвет. </p>
</body>
</html>
```

Чтобы стиль параграфа оставить неизменным, в данном примере введен новый класс **line**, который устанавливает возле параграфа вертикальную линию. Данный класс, кроме того, еще задает смещение 20 пикселей от левого края окна до текста с помощью параметра **margin-left** и отступ от линии до текста параметром **padding-left**, без него текст будет соприкасаться с линией слишком плотно. Линия в примере сплошная, черного цвета и толщиной два пиксела.

### **Результат отображения в браузере:**

```
| Значение этих атрибутов перечисляются через
| пробел и сразу устанавливают тип линии, ее
| толщину и цвет.
```

Точно также линии можно создавать для любых блочных элементов (теги `<TABLE>`, `<P>`, `<DIV>`), а также для встроенных элементов (тег `<SPAN>`), как показано в примере 2.

### *Пример 2.* Горизонтальная линия внизу текста

```
<html>
<head>
<style type="text/css">
```

```

.line {
border-bottom: 1px dotted blue
}
</style>
</head>
<body>
<p><span class="line"> Значение этих атрибутов </span> перечисляются через
пробел и сразу устанавливают тип линии, ее толщину и цвет. </p>
</body>
</html>

```

## 5. Форматирование символов

Форматирование символов дает возможность авторам Web-страниц придавать текстам вид, соответствующий типографским традициям, а также авторским замыслам оформления документа.

Изменение начертания шрифта и его размера происходит через свойства CSS, которые описаны в табл. 4.1.

Таблица 4.1

Свойство	Значение	Описание	Пример
font-family	имя шрифта	Задаёт список шрифтов	P {font-family: Arial, serif}
font-style	normal italic oblique	Нормальный шрифт Курсив Наклонный шрифт	P {font-style: italic}
font-variant	normal small-caps	Капитель (особые прописные буквы)	P {font-variant: small-caps}
font-weight	normal lighter bold bolder 100-900	Нормальная жирность Светлое начертание Полужирный Жирный 100-светлый шрифт, 900-самый жирный	P {font-weight: bold}
font-size	normal pt px %	нормальный размер пункты пиксели проценты	font-size: normal font-size: 12pt font-size: 12px font-size: 120%

Кроме изменения параметров шрифтов, можно управлять и свойствами всего текста. Значения свойств приведены в табл. 4.2.

Таблица 4.2

Свойство	Значение	Описание	Пример
line-height	normal множитель точно %	Интерлиньяж (межстрочный интервал)	line-height: normal line-height: 1.5 line-height: 12px line-height: 120%
text-decoration	none underline overline line-through blink	Убрать все оформление Подчеркивание Линия над текстом Перечеркивание Мигание текста	text-decoration: none
text-transform	none capitalize uppercase lowercase	Убрать все эффекты Начинать С Прописных ВСЕ ПРОПИСНЫЕ все строчные	text-transform: capitalize
text-align	left right center justify	Выравнивание текста	text-align: justify выравнивание по ширине
text-indent	точно %	Отступ первой строки	text-indent: 15px; text-indent: 10%

Атрибуты форматирования можно задавать как для всего текста целиком, так и для отдельных фрагментов текста.

### 5.1. Гарнитура (шрифт)

Гарнитура определяет художественное изображение шрифта, позволяющее различать его среди других шрифтов.

*Синтаксис:*

```
<span style="font-family: имя_шрифта1, имя_шрифта2, семейство_шрифтов"> текст </span>
```

Задавая определенный шрифт, автор Web-страницы предполагает наличие такового на компьютере пользователя. В большинстве случаев такой подход оправдан тем, что наиболее распространенные шрифты входят в стандартную

поставку операционных систем или популярных программных пакетов. Если же заданных шрифтов все же не окажется, браузер выполнит подстановку любого другого шрифта из указанного семейства шрифтов.

*Пример 1.* Вывод фразы бессерифным шрифтом.

```
<span style="font-family: Helvetica"> текст </span>
```

*Пример 2.* Задание общего семейства шрифтов.

```
<span style="font-family: Helvetica, Arial, sans-serif"> текст </span>
```

*Пример 3.* Название шрифта, состоящего из нескольких слов, следует заключать в кавычки или апострофы (если название указывается внутри конструкции, уже содержащей кавычки).

```
<span style="font-family: 'Comic Sans MS'"> текст </span>
```

## 5.2. Начертание

На практике отдельно изменяют ширину символов и их наклон.

*Синтаксис:*

```
<span style="font-weight: ширина_символов"> текст </span>
```

или

```
<span style="font-style: наклон"> текст </span>
```

Ширина символов может быть задана параметрами **normal** (обычный) и **bold** (полужирный).

Наклон символов определяется параметрами **normal** (обычный) и **italic** (наклонный).

*Пример 1.* Использование стилей.

Выделение слова `<span style="font-weight: bold">` полужирным `</span>` шрифтом.

Выделение слова `<span style="font-style: italic">` наклонным `</span>` шрифтом.

*Пример 2.* Совместное использование нескольких стилей.

Выделение слов `<span style="font-weight: bold; font-style: italic">` наклонным полужирным `</span>` шрифтом.

### 5.3. Размер (кегель)

*Синтаксис:*

```
<span style="font-size: размер"> текст </span>
```

Предусмотрено три метода указания размера шрифта:

- абсолютный размер: задается один из семи базовых размеров (**xx-large**, **x-large**, **large**, **medium**, **small**, **x-small**, **xx-small**);
- относительный размер: задается либо один из двух параметров – **larger** (на шаг больше текущего базового размера) и **smaller** (на шаг меньше), либо относительная величина – **em** (значение свойства **font-size** текущего текста), **ex** (значение свойства **x-height** текущего текста), **px** (пиксел), **%** (процент от размера шрифта, установленного в браузере как размер шрифта по умолчанию);
- явное указание размера: задается в дюймах (**in**), сантиметрах (**cm**), миллиметрах (**mm**), пунктах (**pt**), пиках (**pc**).

*Пример 1.* Указание самого большого базового размера шрифта.

```
<span style="font-size: xx-large"> текст </span>
```

*Пример 2.* Если текущий размер шрифта имеет значение **small**, то следующая конструкция выведет слово, имеющее базовый размер **medium**.

```
<span style="font-size: larger"> текст </span>
```

*Пример 3.* Вывод текста размером 12 пунктов (пт).

```
<span style="font-size: 12 pt"> текст </span>
```

### 5.4. Видоизменение

На практике отдельно рассматривают капитализацию символов (написание текста прописными буквами, но меньшего, чем предусмотрено для данного шрифта в нормальном значении размера), оформление текста, смещение текста (надстрочный и подстрочный).

*Синтаксис:*

```
<span style="font-variant: значение"> текст </span>
```

ИЛИ

```
<span style="text-decoration: значение"> текст </span>
```

ИЛИ

```
смещение текста <sup> вверх </sup> и <sub> вниз </sub>
```

Параметр капитализации шрифта может принимать значения **normal** (обычный) и **small-caps** (малые прописные).

При оформлении текста используются следующие значения: **none** (элементы оформления отсутствуют), **underline** (подчеркнутый), **line-through** (зачеркнутый), **overline** (надчеркнутый).

*Пример 1.* «Текстовая» реализация математического выражения  $x_i, i = \overline{1,5}$ .

`<i> x<sub>i</sub>, i = </i> <span style="text-decoration: overline"> 1,5 </span>`

*Пример 2.* Смещение фрагмента текста на 10 пт вниз и вверх (данная конструкция поддерживается *только* MS Internet Explorer версии не ниже 6.0).

`<i> x<span style="position: relative; top: 10.0 pt">i</span><span style="position: relative; top: -10.0 pt">2</span>`

## 5.5. Преобразование регистра

*Синтаксис:*

`<span style="text-transform: значение"> текст </span>`

Результат действия определяется значением параметра: **none** (Естественный порядок чередования прописных и строчных букв в предложении), **capitalize** (Каждое Слово Начинается С Прописной Буквы), **uppercase** (ВСЕ БУКВЫ ПРОПИСНЫЕ), **lowercase** (все буквы строчные).

## 5.6. Расстояние между буквами и словами

*Синтаксис:*

`<span style="letter-spacing: значение"> текст </span>`

или

`<span style="word-spacing: значение"> текст </span>`

Значение атрибута может быть задано параметром **normal** (в этом случае расстояние задается браузером, исходя из размера шрифта и выравнивания текста) или выражено конкретной величиной (в сантиметрах, дюймах, пунктах и др.).

*Пример 1.* Установка расстояния в 10 миллиметров между символами текста.

`<span style="letter-spacing: 10 mm"> текст </span>`



## 5.7. Цвет текста

*Синтаксис:*

```
<span style="color: цвет"> текст </span>
```

Существует следующие формы задания цвета:

- по его названию;
- по шестнадцатеричному значению;
- с помощью RGB.

## 6. Форматирование страниц

CSS имеет несколько опций для определения цвета текста и фоновых областей на веб-странице.

В табл. 4.3 перечислены свойства элементов, предназначенных для задания цвета.

Таблица 4.3

Свойство	Значение	Описание	Пример
Color	Цвет	Устанавливает цвет текста	P { color: #330000 }
background-color	Цвет transparent	Цвет фона	BODY { background-color: #6699FF }
background-image	URL none	Фоновый рисунок	BODY { background-image: url (bg.gif) }
background-repeat	repeat repeat-x repeat-y no-repeat	Повторяемость фонового рисунка	BODY { background-image: url (bg.gif) background-repeat: repeat-y }
background-attachment	scroll fixed	Прокручиваемость фона вместе с документом	BODY { background-image: url (bg.gif) background-attachment: fixed }
background-position	Проценты Пиксели top center bottom left right	Начальное положение фонового рисунка	BODY { background-position: left top }

## 6.1. Фон документа

Фон документа, можно задавать двумя способами:

### 1. Цвет фона

*Синтаксис:*

```
<body style="background: цвет">
```

### 2. Фоновое изображение

*Синтаксис:*

```
<body style="background: url('файл') повтор фиксация х-позиция  
у-позиция">
```

Атрибут **url** представляет собой сетевой адрес графического файла.

Значением по умолчанию для цвета фона является **transparent**, который устанавливает прозрачный фон. Для установки фонового рисунка используется абсолютный или относительный адрес к файлу. Рекомендуется задавать одновременно фоновый рисунок и цвет фона, который будет использоваться в случае недоступности файла изображения.

*Пример 1.* Цвет фона и фоновое изображение

```
<html>  
<head>  
<style type="text/css">  
BODY {  
background-color: #98560F; /* Цвет фона */  
background-image: url('/images/bg.gif') /* Путь к фоновому рисунку */  
}  
</style>  
</head>  
<body>  
<p> Это новый абзац...</p>  
</body>  
</html>
```

Как правило, фоновое изображение повторяется на странице столько раз, сколько необходимо для покрытия всего свободного пространства. Существует возможность управлять направлением повтора, используя свойство **background-repeat** или вообще запрещать его посредством команды **повтор**. Команда представляет собой одно из четырех ключевых слов: **repeat** (заполняется все доступное пространство), **no-repeat** (изображение выводится только один раз), **repeat-x** (повтор только по горизонтали), **repeat-y** (повтор только по вертикали).

Режим фиксации (команда **fixed**) запрещает перемещение фонового изображения при прокрутке содержимого экрана браузера.

## 6.2. Повторяемость фонового рисунка

```
<html>
<head>
<style type="text/css">
BODY {
background-image: url('/images/bg.gif');    /* Путь к фоновому рисунку */
background-repeat: repeat-y              /* Повторение фона по вертикали */
}
</style>
</head>
<body>
<p> Это новый абзац...</p>
</body>
</html>
```

Если изображение не повторяется, то представляется возможным указать его положение на странице с указанием координат или методов позиционирования. Положение фона определяется параметром **background-position**. В качестве методов позиционирования могут использоваться: по горизонтали – **left** (по левому краю), **center** (в центре), **right** (по правому краю), по вертикали – **top** (по верхнему краю), **center** (по центру), **bottom** (по нижнему краю). Положение можно, также, задавать в процентах, пикселах или других абсолютных единицах.

### 6.3. Положение фона

```
<html>
<head>
<style type="text/css">
BODY {
background-attachment: fixed;           /* Фиксируем фон */
background-image: url('mybg.gif');     /* Путь к фоновому рисунку */
background-repeat: no-repeat;         /* Отменяем повторение фона */
background-position: right bottom    /* Положение фона */
}
</style>
</head>
<body>
<p>Это новый абзац...</p>
</body>
</html>
```

В данном примере фон будет помещен в правый нижний угол страницы. Если нужно определить рисунок в левом верхнем углу, то надо задать следующий вид: **background-position: left top**.

Параметр **background-attachment: fixed** фиксирует фон, чтобы он оставался неподвижным при прокрутке содержимого окна браузера.

#### 4.1. Типовое задание

*Пример 1.* Отформатируйте текст с использованием CSS как представлено ниже:

### ЭТО ЗАГОЛОВОК ПЕРВОГО УРОВНЯ

---

Это заголовок третьего уровня, далее идут элементы форматирования

---

Горизонтальная линия внизу текста.

Вертикальная линия слева от текста.

Отступ первой строки в 20 мм.

У с т а н о в к а  
р а с с т о я н и я в  
1 0  
м и л л и м е т р о в  
м е ж д у  
с и м в о л а м и  
т е к с т а .

Установка расстояния в 20 миллиметров между словами. Написание текста малыми прописными буквами

### Подчеркнутый текст

Задаем размер текста в 12 pt

Выделение слов *наклонным полужирным* шрифтом

Гарнитура шрифта

Часть текста **красного** цвета

Динамический HTML-способ управления свойствами объектов на веб-странице, такие как, графика, текст, элементы форм и другое, без необходимости обновления всей страницы.

---

Курсивный текст *хорошо использовать для цитат и прямой речи*

*Наклонный шрифт пригодится для создания ненавязчивого выделения в тексте*

---

**Пример 2.** Рассмотрите и проанализируйте решение практического задания.

```
<html>
<head>
<style type="text/css">
BODY {
    background-color: purple; /* Цвет фона */
    background-attachment: fixed; /* Фиксируем фон */
    background-image: url(C:\Documents and Settings\Мои документы\Мои
рисунки\сердечко.bmp'); /* Путь к фоновому рисунку */
    background-repeat: no-repeat; /* Отменяем повторение фона */
    background-position: right bottom /* Положение фона */
}
```

```

H1 {
  text-align: center;
  margin-top: 0px;                /* Отступ сверху */
  margin-bottom: 1em             /* Отступ снизу */
  color: #fc0;
  font-family: Arial, Helvetica, Verdana, sans-serif;
  font-size: 150%;
  font-weight: lighter
}
H3 { font-family: 'Courier New'; font-size: 100%; font-weight: lighter }
P {
  color: yellow }
  margin: 0.5em 0px 1em         /* Отступ сверху, справа-слева и снизу */
}
P.line {
  border-left: solid 2px black; /* Линия слева от текста */
  margin-left: 20px;          /* Отступ слева от края до текста */
  padding-left: 7px          /* Расстояние от линии до текста */
}
HR{
  color: red;                 /* Цвет линии для браузера Internet Explorer*/
  height: 3px;                /* Высота линии */
  width: 800px;               /* Ширина линии */
  text-align: center}        /* Выравнивание по центру */
}
.line {
  border-bottom: 1px dotted blue
}
.underline {
border-bottom: 1px dashed blue /* Добавляем пунктирную линию под текстом */
}
#cursive { font-style: italic} /* Курсивный текст */
#oblique { font-style: oblique} /* Наклонный текст */
}
</style>

```

```
</head>
<body>
<h1> Это заголовок первого уровня </h1>
<HR>
<h3> Это заголовок третьего уровня, далее идут элементы форматирования
</h3>
<HR>
<p><span class="line"> Горизонтальная линия </span> внизу текста. </p>
<p class="line"> Вертикальная линия слева от текста. </p>
<p style="text-indent: 20 mm"> Отступ первой строки в 20 мм. </p>
<p><span style="letter-spacing: 10 mm"> Установка расстояния в 10 миллиметров
между символами текста. </span></p>
<p><span style="word-spacing: 20mm"> Установка расстояния в 20 миллиметров
между словами. </span></p>
<p><span style="text-transform: capitalize"> Каждое Слово Начинается С
Прописной Буквы </span></p>
<p><span style="font-variant: small-caps"> Написание текста малыми прописными
буквами </span></p>
<p><span style="text-decoration: underline "> Подчеркнутый текст </span></p>
<p><span style="font-size: 12 pt"> Задаем размер текста в 12 pt</span></p>
<p>Выделение слов <span style="font-weight: bold; font-style: italic"> наклонным
полужирным </span> шрифтом</p>
<p><span style="font-family: 'Comic Sans MS'"> Гарнитура шрифта </span></p>
<p>Часть текста <SPAN STYLE="color: red"> красного </SPAN> цвета</p>
<span class="underline">Динамический HTML</span> - способ управления
свойствами объектов на веб-странице, такие как, графика, текст, элементы форм
и другое, без необходимости обновления всей страницы.
<HR>
<p id=cursive><span style="font-style: normal"> Курсивный текст </span> хорошо
использовать для цитат и прямой речи </p>
<p id=oblique>Наклонный шрифт пригодится для создания ненавязчивого
выделения в тексте </p>
<HR>
</body>
</html>
```

## 4.2. Контрольные вопросы

1. Основные атрибуты CSS для форматирования.
2. Атрибуты CSS для управления шрифтами.
3. Свойства CSS для управления видом текста.
4. Опций CSS для определения цвета фона и текста.

## 4.3. Контрольные задания

1. Отформатируйте произвольный фрагмент текста, состоящий из 3 абзацев в соответствии с вариантом.

	Варианты					
	1	2	3	4	5	6
<b>1 абзац</b>	Отступ первой строки в 15 мм, выравнивание текста по ширине, цвет желтый	Отступ первой строки на расстояние, равное 10 % открытого окна браузера, шрифтом типа Courier	Отступ первой строки на расстояние на 10 % превышающее размер шрифта абзаца	Отступ сверху и снизу абзаца, выравнивание текста по центру, цвет розовый	Рамка вокруг текстового блока, оформленная стилем, создающим эффект выпуклости	Вертикальную линию справа от текста, задав ее тип, толщину и цвет
<b>2 абзац</b>	перечеркнутый полужирный шрифт, размер 12 pt, цвет красный	Мигание текста, шрифт размером 14 pt, цвет синий, подчеркнутый	Все слова текста начинать с прописных букв, шрифт текста типа Arial, курсив, размер 18 pt	Все буквы прописные, наклонный шрифт зеленого цвета размером 20 pt	Шрифт размером 22 pt красный цвет, полужирный	Шрифт текста типа Times New Roman, размер шрифта большого базового размера, нормальной жирности
<b>3 абзац</b>	Выровнять по центру, расстояние между буквами 10 мм	Выровнять по правому краю, расстояние между словами 10 мм	Выровнять по левому, расстояние между буквами 15 мм	Выровнять по ширине, расстояние между словами 20 мм	Выровнять по правому краю, расстояние между буквами 5 мм	Выровнять по центру, расстояние между словами 30 мм

2. Оформите фон сайта, используя фоновый рисунок, который прокручивается вместе с документом.



## ГИПЕРССЫЛКИ И ДОБАВЛЕНИЕ ИЗОБРАЖЕНИЙ

**Цель работы:** научиться использовать статические и динамические графические изображения в WWW-документах; научиться реализовывать связи между гипертекстовыми файлами и ссылки на точки внутри HTML-документа.

### *Методические указания*

При рассмотрении понятия «ссылка» следует выделить две ее составляющие части: место размещения ссылки в документе (элемент привязки, или якорь) и направление перехода по данной ссылке (адрес ссылки).

В качестве элементов привязки могут выступать: слова, группы слов (текстовые элементы) или изображения (графические элементы).

Текстовые элементы привязки обычно выделяются цветом и подчеркиваются, графические элементы (изображения) не подчеркиваются и не выделяются цветом, но часто обводятся рамкой. К графическим элементам привязки относятся маркеры, пиктограммы, реклама и др.

Ссылки на URL (адрес Web-страницы) подразделяются на относительные и абсолютные.

*Относительной ссылкой* называют путь к файлу относительно корневого каталога узла.

*Абсолютная ссылка* – это полный путь к файлу с указанием доменного адреса узла и пути к файлу на данном узле.

За организацию ссылок в языке HTML отвечает тег `< A >< /a >`, который чаще всего использует следующий шаблон:

- Произвольный текст `<A href= "Адрес ссылки">` текст для щелчка по нему мышью `</a >`.
- Атрибут `href` тега `<A>` может указывать на метку внутри страницы (*внутренние ссылки*), файл на локальном диске или ресурс *Internet*.

Организация внутренних ссылок, ссылок на закладку, позволяющая

осуществлять переход внутри текущей страницы, выполняется в два этапа:

а) сначала в нужных местах Web-страницы устанавливается метка

`<A name = "Метка" > </a>;`

б) на втором этапе определяется ссылка на метку по следующему шаблону: Текст подсказки `<A href = "#Метка">` текст для щелчка `</a>`.

Если в качестве адреса ссылки должно быть размещено указание на файл на локальном диске, то атрибут *href* принимает значение вида

`href= "file: //Диск: \ Путь к файлу"`.

Общий шаблон атрибута *href*: `href= "URL" или href = "Протокол: // Адрес ссылки"`.

Для обеспечения доступа к различным протоколам используются схемы доступа, представленные в табл. 5.1.

Таблица 5.1

Обозначение ресурса	Название ресурса
<i>ftp</i>	Доступ к архивам файлов при помощи протокола
<i>Gather</i>	Доступ к системе Gopher
<i>http</i>	Доступ к WWW
<i>mailto</i>	Отправка сообщения по электронной почте
<i>news</i>	Доступ к новостям USENET
<i>telnet</i>	Подключение к протоколу telnet
<i>wais</i>	Подключение к системе поиска WAIS

Приведем несколько примеров записи тега `< A > </a >`.

**Пример 1.** Организация гиперссылки на Web-страницу:

`<A href= "http: //www.bspu.unibel.by" > БНТУ </a>`

**Пример 2.** Формирование ссылки для отправки сообщения по электронной почте:

`<A href= "mailto: me@mycomp.com" > Пишите < a/ >`

**Пример 3.** Ссылка на раздел новостей USENET:

`<A href = "news: news.newusers.questions" > Новости </a >`

## Добавление изображений

Для хранения изображений используются десятки различных форматов: *gif*, *jpeg*, *psx*, *png*, *wmf*, *bmp* и др. Однако для работы с изображениями в документах HTML обычно выбирают форматы, которые распознаются браузерами *gif*, *jpg*, *png*. Для вставки изображения используется тег `<Img >` с атрибутом *src*, который указывает URL графического файла

`<Img src = "URL">`

Тег `<IMG >` использует атрибуты, значения которых имеют большую важность при размещении изображения на Web-странице. Основные атрибуты тега `<Img >` представлены в табл. 5.2.

Таблица 5.2

Название атрибута	Назначение /значение атрибута
<i>Align</i>	Выравнивание текста по: <i>top</i> – верхнему краю изображения <i>middle</i> – средней части изображения <i>bottom</i> – нижнему краю изображения <i>left</i> – показывает изображение слева <i>right</i> – показывает изображение в правой части окна
<i>Height</i>	Указывает высоту изображения в пикселях или процентах
<i>Width</i>	Указывает ширину изображения в пикселях или процентах
<i>Border</i>	Присваивает значение ширины рамки в пикселях вокруг изображения
<i>Vspace</i>	Расширяет свободное пространство в пикселях вокруг изображения по вертикали
<i>Hspace</i>	Расширяет свободное пространство в пикселях вокруг изображения по горизонтали

**Пример 4.** Выведите на Web-странице стандартную вставку для Windows из файла *Облака.bmp* по середине страницы высотой 250 пикселей, шириной 200

пикселей, шириной рамки 2 пиксела и отступами вокруг изображения по горизонтали – 100, по вертикали 50 пикселей

```
< Img src = "file: // c:\Windows\Облака.bmp" align= "middle" height = 250  
width = 200 border = 2 hspace = 100 vspace= 50.
```

При организации ссылки, в качестве которой используется изображение, применяется шаблон из комбинации двух тегов < A > < /a > и

```
< A href =« Адрес ссылки»> < Img src= «ссылка на рисунок»></a>
```

### 5.1. Типовое задание

Постройте на Web-странице гиперссылки следующего вида:

#### ГИПЕРССЫЛКИ

---

#### **ВНУТРЕННИЕ ССЫЛКИ**

---

Переход в конец документа

---

Переход к ссылке 1

---

Здесь расположена **ссылка 1**

---

#### **ССЫЛКИ НА АДРЕС ЭЛЕКТРОННОЙ ПОЧТЫ**

---

**Текстовый элемент привязки в ссылке**

Свои предложения высылайте автору этого пособия.

---

**Графический элемент привязки в ссылке**

Нажмите на картинку, чтобы отправить сообщение автору.



#### **ССЫЛКИ НА РЕСУРСЫ INTERNET**

---

БНТУ страница

---

Переход в начало документа.

Проанализируйте код HTML для организации ссылок и определите виды этих ссылок.

```
<Html>
```

```
<Head>
```

```
<title>Гиперссылки</title>
```

```
<Meta http-equiv= "Content-Type" content=«text/html; charset=windows-1251»>
```

```

<Body bgcolor=yellow>
<A name="top"></a>
<Center>
<Font size=6 color=lime><I> ГИПЕРССЫЛКИ </I></font>
</center>
<! горизонтальная линия>
<Hr align=center size=6 color=blue>
<Center>
<Font color=red><H3> ВНУТРЕННИЕ ССЫЛКИ </h3>
</font>
</center>
<Hr>
Переход в <A href="#bottom">концу</a> документа<P>
<Hr>
Переход к <A href="#S001"><B>ссылке 1</b></a><P>
<Hr>
Здесь расположена <B>ссылка 1</b><A name="S001"></a>
<Hr align=center size=4 color=blue>
<Center>
<Font color=red>
<H3>ССЫЛКИ НА АДРЕС ЭЛЕКТРОННОЙ ПОЧТЫ</H3>
</font>
</center>
<Hr>
<Center>
<Font color=green><H4>Текстовый элемент привязки в ссылке</H4>
</font>
</center>
Свои предложения высылайте
<A href="mailto:Ru@bnty.by">автору</a>
этого пособия.
<Hr>
<Center>
<Font color=green><H4>Графический элемент привязки в ссылке</H4>
</font>
</center>
<P>Нажмите на картинку, чтобы отправить сообщение автору.
<A href="mailto: Ru@bnty.by ">
<Img src="postbox.gif width="65" height=" 135"></a>
<Hr align=center size=4 color=blue>
<Center>

```

```

<Font color=red><H3>ССЫЛКИ НА РЕСУРСЫ INTERNET</H3>
</font>
</center>
<A href="http://www.bspu.unibel.by">БНТУ</a> страница
<Hr align=center size=4 color=blue>
<A name="bottom"></a><P>
Переход в <A href="#top">начало</a> документа.
<Hr align=center size=4 color=blue>
</body>
</html>

```

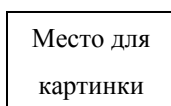
## 5.2. Контрольные вопросы

1. Вставка графических объектов в HTML-файлы.
2. Использование динамических изображений и видеозаписей.
3. Гипертекстовые связи между HTML-документами.
4. Ссылки на точки внутри документа.
5. Какой тег обеспечивает создание гиперссылки?
6. Как организовать гиперссылку внутри страницы?
7. Каким образом в качестве гиперссылки использовать изображение?

## 5.3. Контрольные задания

### Вариант 1

- Ссылка, по которой осуществляется переход в конец страницы.
- Ссылка на файл с расширением .htm, подготовленный в первой лабораторной работе.
- Нажатием на картинку осуществляется переход к ссылке \*.



- Ссылка на адрес Белорусского государственного университета  
**<http://www.bsu.unibel.by>**
- Ссылка на адрес электронной почты.
- Ссылка на \*.
- Ссылка, по которой осуществляется переход в начало страницы.

### **Вариант 2**

- Нажатием на картинку осуществляется переход в конец страницы.

Место для  
картинки

- Ссылка, по которой осуществляется переход к ссылке \*.
- Ссылка на адрес электронной почты.
- Ссылка на адрес Белорусского государственного педагогического университета  
**<http://www.bspu.unibel.by>**
- Ссылка на файл с расширением .htm, подготовленный в первой лабораторной работе.
- Ссылка на \*.
- Ссылка, по которой осуществляется переход в начало страницы.

### **Вариант 3**

- Ссылка, по которой осуществляется переход в конец страницы.
- Ссылка на адрес Белорусского государственного национального университета  
**<http://www.bspsa.unibel.by>**
- Ссылка, по которой осуществляется переход к ссылке \*.
- Нажатием на картинку осуществляется переход к файлу с расширением .htm, подготовленный в первой лабораторной работе.

Место для  
картинки

- Ссылка на адрес электронной почты.
- Ссылка на \*.
- Ссылка, по которой осуществляется переход в начало страницы.

### **Вариант 4**

- Ссылка, по которой осуществляется переход в конец страницы.
- Ссылка на \*.
- Ссылка на адрес Минского государственного лингвистического университета  
**<http://www.mslu.unibel.by>**
- Ссылка, по которой осуществляется переход к ссылке \*.
- Ссылка на файл с расширением .htm, подготовленный в первой лабораторной работе.
- Ссылка на адрес электронной почты.
- Нажатием на картинку осуществляется переход в конец страницы.

Место для  
картинки

### **Вариант 5**

- Ссылка, по которой осуществляется переход в конец страницы.
- Ссылка на адрес Полоцкого государственного педагогического университета **<http://www.psu.brest.by>**
- Ссылка на \*.
- Ссылка на адрес электронной почты (нажатием на картинку осуществляется отправка сообщения).

Место для  
картинки

- Ссылка на файл с расширением .htm, подготовленный в первой лабораторной работе.
- Ссылка, по которой осуществляется переход к ссылке \*.
- Ссылка, по которой осуществляется переход в начало страницы.

### **Вариант 6**

- Ссылка, по которой осуществляется переход в конец страницы.
- Ссылка на \*.
- Ссылка на адрес электронной почты.
- Ссылка на адрес Белорусского государственного экономического университета **<http://www.bseu.minsk.by>** (нажатием на картинку осуществляется загрузка браузером сайта этого университета).

Место для  
картинки

- Ссылка на файл с расширением .htm, подготовленный в первой лабораторной работе.
- Ссылка, по которой осуществляется переход в начало страницы.
- Ссылка, по которой осуществляется переход к ссылке \*.



*Лабораторная работа № 6*  
**ИСПОЛЬЗОВАНИЕ ГИПЕРССЫЛОК  
И ДОБАВЛЕНИЕ ИЗОБРАЖЕНИЙ В CSS**

**Цель работы:** научиться использовать ссылки, задавать альтернативные способы выделения ссылок в CSS; научиться использовать графические изображения; научиться реализовывать связи между гипертекстовыми файлами и ссылками на точки внутри HTML-документа.

**Методические указания**

**1. Карты-изображения**

Карты-изображения позволяют привязывать ссылки к разным областям одного изображения. Реализуется в двух различных вариантах – серверном и клиентском. В случае применения серверного варианта браузер посылает запрос на сервер для получения адреса выбранной ссылки и ждет ответа с требуемой информацией. Такой подход требует дополнительного времени на ожидание результата и отдельные файлы для каждой карты-изображения. В клиентском варианте карта располагается в том же HTML-документе, что и ссылка на изображение.

**2. Клиентский вариант карты-изображения**

Для указания того, что изображение является картой, используется параметр **usemap** тега **IMG**. В качестве значения используется ссылка на описание конфигурации карты (рис. 6.1).



Рис. 6.1. Закладки являющиеся ссылками, созданными с помощью карты

Чтобы сделать ссылки на закладки, показанных на рис. 4, следует использовать следующий код.

**Пример 1.** Использование карты-изображения.

```

<map name="map">
<area shape="poly" alt="Закладка 2" coords="210,27, 203,9, 202,6, 197,2,
192,1,120,1, 115,2, 110,6, 112,9, 119,27, 119,32, 211,32, 210,27"
href="2.html">
<area shape="poly" alt="Закладка 3" coords="302,27, 295,9, 293,6, 289,2,
283,1, 212,1, 206,2, 202,6, 203,9, 210,27, 211,32, 284,32, 303,32, 302,27"
href="3.html">
<area shape="poly" alt="Закладка 4" coords="302,27, 303,32, 394,32,
393,27, 386,9, 382,3, 375,1, 303,1, 298,2, 293,6, 295,9, 302,27"
href="4.html">
</map>
```

Для указания браузеру, что изображение является картой, используется параметр **usemap**. Он является ссылкой на описание конфигурации карты, которая задается тегом **MAP**. Значение параметра name данного тега должно соответствовать имени в **usemap**. Для задания активной области, являющейся ссылкой на HTML-документ, используется тег **AREA**. Тег **AREA** может иметь следующие параметры:

**SHAPE** – определяет форму активной области. Форма может быть в виде окружности (circle), прямоугольника (rect), многоугольника (poly).

**ALT** – добавляет альтернативный текст для каждой области. Служит лишь комментарием для ссылки, поскольку на экран никогда не выводится.

**COORDS** – задает координаты активной области. Координаты отсчитываются в пикселах от левого верхнего угла изображения, которому соответствует значение 0,0. Первое число является координатой по горизонтали, второе – по вертикали. Список координат зависит от формы области. Многоугольник предусматривает указание до 100 координат, и он должен быть замкнутым.

Для окружности задаются три числа - координаты центра круга и радиус.

```
<area shape=circle coords="230,340, 100" href=circle.html>
```

Для прямоугольника - координаты левого верхнего и правого нижнего угла.

```
<area shape=rect coords="24,18, 210,56" href=rect.html>
```

Для многоугольника задаются координаты его вершин (рис. 6.2).



Рис. 6.2. Координатные точки для полигона

HREF – определяет адрес ссылки для области. Правила записи такие же, как и для тега А.

### 3. Синтаксис определения карты ссылок обслуживаемой клиентом

```
<MAP NAME="mapname">
```

```
<AREA [SHAPE="shape"] COORD="x,y, ..." [HPEF="URL"] [NOHREF]>
```

```
</MAP>
```

#### **Пример 1**

```
<IMG SRC="Pict.jpg" USEMAP="#mymap"><BR>
```

```
<MAP NAME=mymap>
```

```
<AREA SHAPE=RECT COORDS="0,0,100,100" HREF=item1.html>
```

```
<AREA SHAPE= POLY COORDS="101,0,200, 0, 150,100" HREF=item2.html>
```

```
<AREA SHAPE= CIRCLE COORDS="250,50,50" HREF=item3.html>
```

```
</MAP>
```

#### 4. Использование гиперссылок

Гипертекстовые связи (гиперссылки или Web-ссылки) являются наиболее важным элементом Web-страниц. С их помощью документ получается связанным и структурированным, что позволяет пользователю получать необходимую ему информацию максимально быстро и удобно.

По умолчанию со ссылкой связана загрузка другого Web-ресурса. Это достигается путем выбора ссылки (щелчком мыши).

*Синтаксис:*

```
<a href="URL"> текст-ссылка</a>
```

Ссылка указывает на целевой объект (якорь), который может быть любым Web-ресурсом (изображением, видеоклипком, звуковым файлом, программой, HTML-документом, элементом в HTML-документе и т.д.).

Целевые якоря в HTML-документах могут указываться с помощью тэга **<a>** (с использованием атрибута **name**) или с помощью любого другого тэга (с использованием атрибута **id**).

*Синтаксис:*

```
<a name="якорь"> Текст, к которому идет переход </a>
```

или

```
<p id="якорь"> Текст, к которому идет переход </p>
```

Некоторые URL могут указывать на местоположение внутри ресурса. В этом случае URL заканчивается символом "#", за которым следует указатель (идентификатор фрагмента).

Используя специальный скрытый указатель, можно быстро переходить от одного раздела к другому внутри документа, не используя прокрутку экрана. После щелчка мышкой по ссылке браузер переместит вас на указанный раздел документа, а строка, в которой стоит указатель данного раздела (обычно, первая строка раздела или его заголовок) будет размещена на первой строке окна браузера (если данная строка не присутствует уже на экране).

Относительный URL не содержит информации о схеме доступа. Путь в нем указывает на ресурс на той же машине, на которой находится текущий документ. Относительные URL могут содержать компоненты относительного пути (например, ".." означает один уровень выше в иерархии) и идентификаторы фрагментов.

Атрибут **title** для тэга `<a>` позволяет отображать указанный текст в виде подсказки при наведении курсора мыши на гиперссылку.

**Пример 1.** Текстовая ссылка на другой HTML-документ, находящийся там же, где и текущий документ.

```
<a href="hpc.html">Карманные компьютеры</a>
```

**Пример 2.** Текстовая ссылка на графический файл, находящийся там же, где и текущий документ.

```
<a href="be.jpg">Медведь</a>
```

**Пример 3.** Графическая ссылка на HTML-документ, находящийся там же, где и текущий документ.

```
<a href="be.html"></a>
```

**Пример 4.** Текстовые ссылки на различные фрагменты текущего HTML-документа.

```
<h1>ОГЛАВЛЕНИЕ</h1>  
<p><a href="#chapter1">ГЛАВА 1</a><br>  
<a href="#chapter2">ГЛАВА 2</a><br>  
...продолжение оглавления...  
...тело документа...  
<h2><a name="chapter1"> ГЛАВА 1</a></h2>  
...содержание (текст) главы 1...  
<h2><a name="chapter2"> ГЛАВА 2</a></h2>  
...содержание (текст) главы 2...  
или  
<h1>ОГЛАВЛЕНИЕ/h1>  
<p><a href="#chapter1">ГЛАВА 1</a><br>  
<a href="#chapter2">ГЛАВА 2</a><br>  
...продолжение оглавления...  
...тело документа...  
<h2 id="chapter1"> ГЛАВА 1</a></h2>
```

...содержание (текст) главы 1...  
<h2 id="chapter2"> ГЛАВА 2</a></h2>  
...содержание (текст) главы 2...

Для управления видом различных ссылок в CSS используются псевдоклассы, которые указываются после селектора А через двоеточие. Псевдоклассы применяются в CSS, чтобы определять стиль элемента при изменении его состояния.

*Синтаксис:*

**А: псевдокласс** { свойство: значение }

В табл. 6.1 приведены допустимые псевдоклассы и их описания.

Таблица 6.1

Псевдоклассы	Описание
Link	Определяет стиль для обычной непосещенной ссылки.
Visited	Определяет стиль для посещенной ссылки.
Active	Определяет стиль для активной ссылки. Активной ссылка становится при нажатии на нее.
Hover	Определяет стиль для ссылки при наведении на нее мышью.

**Пример 5.** Использование псевдоклассов для управления цветами ссылок.

```
<html>
<head>
<style type="text/css">
A: link { color: # 003366 }
A: visited { color: # 660066 }
A: hover { color: # 800000 }
A: active { color: # FF0000 }
</style>
</head>
<body>
<a href=link1.html>Ссылка 1</a>|
```

```
<a href=link2.html> Ссылка 2</a>|
<a href=link3.html> Ссылка 3</a>
</body>
</html>
```

Псевдокласс **link** обычно не используется, поскольку он определяет то же самое, что и обычный селектор **A**, а именно стиль для непосещенных ссылок. Чтобы не было ненужного повторения, **link**, как правило, опускают.

Одно из наиболее популярных применений CSS — это скрытие подчеркивания у ссылок. При правильном применении отсутствие подчеркивания у ссылок может придать определенный эффект сайту. Часто делается, что при наведении курсора, ссылка становится подчеркнутой, меняет свой цвет или используется и тот и другой эффект одновременно (пример 6).

**Пример 6.** Подчеркивание у ссылки и изменение ее цвета

```
<html>
<head>
<style type="text/css">
A:link {
text-decoration: none /* Убирает подчеркивание для ссылок */
}
A:visited { text-decoration: none }
A:active { text-decoration: none }
A:hover {
text-decoration: underline; /* Делает ссылку подчеркнутой при наведении на нее
курсора */
color: red /* Цвет ссылки */
}
</style>
</head>
<body>
...
</body>
</html>
```

Еще один пример демонстрирует использование в ссылках подчеркивания и надчеркивания одновременно. При этом эффекте тонкие линии будут появляться над и под ссылкой при наведении на нее курсора. Это достигается применением параметра **text-decoration: underline overline** в селекторе **A:hover**.

*Пример 7.* Использование подчеркивания в ссылках

```
<html>
<head>
<style type="text/css">
A:link { text-decoration: none }
A:visited { text-decoration: none }
A:active { text-decoration: none }
A:hover {
text-decoration: underline overline; /* Ссылка подчеркнутая и надчеркнутая */
color: red /* Цвет ссылки */
}
</style>
</head>
<body>
...
</body>
</html>
```

Существует возможность изменять размер ссылки при наведении на нее курсора.

*Пример 8.* Изменение размера ссылки

```
<html>
<head>
<style type="text/css">
A:link { text-decoration: none }
A:visited { text-decoration: none }
A:active { text-decoration: none }
A:hover {
font-size: 24px; /* Размер шрифта */
font-weight: bold; /* Жирное начертание */
color: red /* Цвет ссылки */
}
</style>
</head>
```



```

<body>
...
</body>
</html>

```

Со ссылками, при помощи CSS, можно сделать так, что цвет ссылки, при наведении на нее курсора мыши, остается неизменным, но зато у нее появляется подчеркивание другого цвета, нежели сама ссылка.

**Пример 9.** Создание подчеркивание другого цвета




```












<html>
<head>
<style type="text/css">
A:link { color: blue; text-decoration: none }
A:hover { color: red; text-decoration: underline }
.link { color: blue }
</style>
</head>
<body>
<a href="link1.html"><span class="link">Ссылка</span></a>
</body>
</html>

```

При помощи CSS можно выбрать вид курсора при наведении его на разные ссылки. Для этого следует определить класс и в нем использовать один из типов курсора, описанных в табл. 6.2.

Таблица 6.2

Вид	Значение	Пример
1	2	3
	default	cursor:default
	crosshair	cursor:crosshair
	pointer	cursor:pointer

1	2	3
	move	cursor:move
I	text	cursor:text
	wait	cursor:wait
	help	cursor:help
	n-resize	cursor:n-resize
	ne-resize	cursor:ne-resize
	e-resize	cursor:e-resize
	se-resize	cursor:se-resize
	s-resize	cursor:s-resize
	sw-resize	cursor:sw-resize
	w-resize	cursor:w-resize
	nw-resize	cursor:nw-resize

**Пример 10.** Изменение курсора при наведении его на ссылку

```

<html>
<head>
<style type="text/css">
.movelink { cursor: move }
.helplink { cursor: help }
</style>
</head>
<body>
<a href="mypage.htm" class="movelink">ПЕРЕМЕСТИТЕ ЭТОТ ТЕКСТ</a>
<br>
<a href="mypage.htm" class="helplink">СПРАВКА</a>
</body>
</html>

```

Подчеркивание текста у ссылок можно использовать разное, например, в виде пунктирной линии. Для этого создадим новый стиль, в котором будем использовать параметр **border-bottom**, он создает линию под текстом (пример 11).

*Пример 11.* Создание пунктирного подчеркивания для ссылок

```
<html>
<head>
<style type="text/css">
A {
color: #f00                /* Цвет ссылок */
}
A:visited {
color: #666                /* Цвет посещенных ссылок */
}
A:hover {
text-decoration: none;    /* Убираем обычное подчеркивание */
border-bottom: 1px dashed blue /* Добавляем пунктирную линию под текстом */
}
</style>
</head>
<body>
<a href="1.html">Подчеркнутая ссылка</a>
</body>
</html>
```

В данном примере создается стиль для ссылки, который при наведении на нее курсором мыши создает у текста пунктирное подчеркивание синего цвета. Пунктирное подчеркивание появляется чуть ниже обычного подчеркивания ссылки.

Можно создать свой собственный вид подчеркивания ссылок, используя для этого графическое изображение. Рисунок подойдет не всякий, он должен обладать двумя качествами:

- картинка должна быть бесшовной, иными словами, состыковываться сама с собой по горизонтали, это необходимо для получения плавной неразрывной линии;
- высота линии не должна превышать 3 пиксела.

Высоту линии можно сделать и больше трех пикселей, если добавить к селектору **A** параметр **padding**.

На рис. 6.4 показано исходное изображение, которое годится для создания подчеркивания, а также его увеличенный вариант.

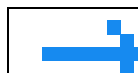


Рис. 6.4. Исходное изображение для создания подчеркивания ссылки

После создания рисунка в графическом редакторе, следует определить стиль, как показано в примере 12. Основным элементом для установки линии выступает стилевой параметр **background**. Это универсальный атрибут, который определяет характеристики фонового изображения, в частности, путь к графическому файлу, а также смещение фона относительно левого верхнего угла элемента. В данном случае требуется только сдвинуть фоновый рисунок вниз на высоту текста, для этого используется относительная единица **em**, равная высоте шрифта элемента. Кроме того, чтобы рисунок повторялся только по горизонтали, добавляем к параметру **background** аргумент **repeat-x**.

*Пример 12.* Добавление графической линии к ссылкам

```
<html>
<head>
<style type="text/css">
A {
text-decoration: none;           /* Убираем подчеркивание у ссылок */
padding: 3px;                   /* Поля вокруг текста ссылки */
white-space: nowrap             /* Нет переноса текста */
}
A:hover {
background: url('/images/arrow.gif') /* */
0px 1.1em                       /* Смещение рисунка по горизонтали и вертикали */
repeat-x                         /* Повторение фона по горизонтали */
}
</style>
</head>
```

```
<body>
<p><a href="link1.html">Ссылка 1</a></p>
<p><a href="link2.html">Ссылка 2</a></p>
<p><a href="link3.html">Ссылка 3</a></p>
</body>
</html>
```

Если текст ссылки достаточно длинный и занимает две и более строки, то рисунок отображается только под первой строкой. Чтобы ликвидировать эту особенность, для селектора А введен атрибут **white-space** со значением `nowrap`, он отменяет переносы в тексте и отображает его одной строкой.

Путь к изображению указывается с помощью ключевого слова `url`, после которого в скобках пишется адрес рисунка. Число **0px** в данном примере означает смещение изображения по горизонтали, а **1.1em** — вниз по вертикали. Вы можете незначительно менять это число в ту или иную сторону, подбирая наилучший вариант в зависимости от используемого рисунка. Результат использования графической линии в ссылках показан на рис. 6.5.

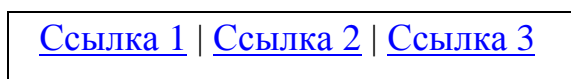


Рис 6.5. Использование графической линии в ссылках

Интересный эффект можно получить, если использовать анимированное изображение. Например, в качестве подчеркивания добавить статичный рисунок, а при наведении на ссылку он будет меняться на другой. Для этого уже потребуется две картинки, первая показана на рисунке 6, а вторая, статичная приведена на рис. 6.6.



Рис. 6.6. Изображение для анимации линии подчеркивания

После чего добавление линии происходит, как уже было описано ранее, за исключением того, что фоновый рисунок устанавливается для селектора **A** и псевдокласса **hover** (пример 13).

*Пример13.* Добавление анимированной линии к ссылкам

```
<html>
<head>
<style type="text/css">
A {
text-decoration: none;           /* Убираем подчеркивание у ссылок */
padding: 3px;                   /* Поля вокруг текста ссылки */
background: url('/images/arrow1.gif') /* Путь к исходному изображению */
0px 1.1em repeat-x              /* Положение фона */
}
A:hover {
background: url('/images/arrow2.gif') /* Путь к анимированному изображению */
0px 1.1em repeat-x              /* Положение фона */
}
</style>
</head>
<body>
<p><a href="link1.html">Ссылка 1</a></p>
<p><a href="link2.html">Ссылка 2</a></p>
<p><a href="link3.html">Ссылка 3</a></p>
</body>
</html>
```

У стилового параметра **background** для разных селекторов меняется только значение **url**. Остальные аргументы совпадают, чтобы не произошло смещения рисунков.

## 5. Графика

Можно использовать стили, чтобы убрать рамку для всех изображений, которые являются ссылками. Для этого применяется параметр **border** со значением **none**.

Конструкция **A IMG** определяет контекст применения стиля только для тега **IMG**, который находится внутри контейнера **A** и, следовательно, является ссылкой.

**Пример 1.** Использование стилей для отмены рамки изображений

```
<html>
<head>
<style type="text/css">
A IMG { border: none }
</style>
</head>
<body>
<a href=/index.html> <img src==/index/home.gif></a>
</body>
</html>
```

**Пример 2.** Цвет рамки, заданный с помощью стилей

```
<html>
<head>
<style type="text/css">
A IMG { border: yes; border-color: red }
</style>
</head>
<body bgcolor=blue>
<a href=/index.html> <img src=/index/home.gif></a>
</body>
</html>
```

**Пример 3.** Использование стилей для создания подрисовочной подписи

```
<html>
<head>
<style type="text/css">
# sign{
padding: 10px;           /* Поля вокруг изображения */
margin: 0px 10px 10px 0px; /* Отступы от рамки до текста */
background: #e0e0e0;    /* Цвет фона вокруг изображения */
border: 1px solid black; /* Параметры рамки */
width: 100px;          /* Ширина */
float: left;           /* Выравнивание по левому краю */
font-size: 90%         /* Размер текста подписи */
}
```

```

</style>
</head>
<body>
<div id=sign>
<img src=1.jpg width=100 height=110>Способ, когда текст под картинкой</div>
</body>
</html>

```

Текст можно расположить справа или слева от изображения, добавив новый селектор **caption** с именем к изображению.

*Пример 4.* Подрисовочная подпись справа от рисунка

```

<html>
<head>
<style type="text/css">
# sign{
padding: 10px; background: #e0e0e0;
border: 1px solid black; width: 190px;
float: left; font-size: 90%
}
# caption {
float: left; /* Подпись справа от картинки */
border: 1px solid black; /* Рамка вокруг изображения */
margin: 3px /* Отступы вокруг изображения */
}
</style>
</head>
<body>
<div id=sign>
<img src=1.jpg width=100 height=110 id=caption>Способ, когда текст справа от
изображения. <br>Способ, когда текст справа от изображения. <br>Способ,
когда текст справа от изображения. <br>Способ, когда текст справа от
изображения. <br> Способ, когда текст справа от изображения.</div>
</body>
</html>

```

Для обтекания картинки текстом можно применить стилевой атрибут **float**. Значение **right** будет выравнивать изображение по правому краю окна браузера, а текст размещать слева от рисунка. Значение **left**, выравнивает изображение по





левому краю, а текст – справа от рисунка. Элемент, для которого установлено значение **float**, обычно называется плавающим (т.е. создающим обтекание).

### *Пример 5.* Применение стилей

```
<html>
<head>
<style type="text/css">
# warp {
float: left;           /* Выравнивание изображения по левому краю */
margin-right: 10px    /* Отступы справа от изображения */
}
</style>
</head>
<body>
<img src=1.jpg width=100 height=110 id=warp>Применение стилей
</body>
</html>
```

### **6.1. Типовое задание**

*Пример 1.* Постройте на Web-странице гиперссылки так, что к ссылкам будут добавляться пунктирные линии при наведении на них курсором мыши, а вид самого курсора будет изменяться: для 2-ой ссылки – на двунаправленную стрелку , а 3-й ссылки – на вопрос  (с использованием CSS):

[Подчеркнутая ссылка](#)

[ПЕРЕМЕСТИТЕ ЭТОТ ТЕКСТ](#)

[СПРАВКА](#)

*Пример 2.* Рассмотрите и проанализируйте решение практического задания

```
<html>
<head>
<style type="text/css">
A:link {
```

```

color: #f00          /* Цвет ссылок */
}
A:visited {
color: #666          /* Цвет посещенных ссылок */
}
A:hover {
border-bottom: 1px dashed blue /* Добавляем пунктирную линию под текстом */
font-size: 24px;      /* Размер шрифта */
font-weight: bold;    /* Жирное начертание */
color: red            /* Цвет ссылки */
}
.movelink { cursor: move } /* Изменение курсора при наведении на него
мышью. Вид двунаправленной стрелки */
.helplink { cursor: help } /* Изменение курсора при наведении на него мышью. Вид
вопроса */
</style>
</head>
<body>
<a href="1.html">Подчеркнутая ссылка</a>
<p><a href="mypage.htm" class="movelink">ПЕРЕМЕСТИТЕ ЭТОТ
ТЕКСТ</a></p>
<p><a href="mypage.htm" class="helplink">СПРАВКА</a></p>
</body>
</html>

```

**Пример 3.** Вставьте графические изображения с использованием CSS, выровняв их по левому и правому краю, задав отступы вокруг изображения и рамки вокруг изображения.

**Пример 4.**Рассмотрите и проанализируйте решение практического задания


```
<html>
<head>
<style type="text/css">
  A IMG { border: yes; border-color: red; width: 100; height: 100}
  # warp {
float: left;           /* Выравнивание изображения по левому краю */
margin-right: 10px     /* Отступы справа от изображения */
}
  # caption {
border: 1px solid black; /* Рамка вокруг изображения */
margin: 3px           /* Отступы вокруг изображения */
}
</style>
</head>
<body>
<a href=текст1.html> </a>
<p>Применение стилей</p>
Способ, когда текст справа от
изображения.</div>
<div style="float: right">Изображение справа.</div>
</body>
</html>
```

## **6.2. Контрольные вопросы**


1. Использование гиперссылок.
2. Целевые якоря в HTML-документах
3. Использование псевдоклассов для задания стиля ссылок.
4. Изменение курсора при наведении его на ссылку.
5. Добавление графики к ссылкам.
6. Применение стилей для графических изображений.

### 6.3. Контрольные задания

Постройте на Web-странице гиперссылки в соответствии с вариантом. Вставьте графические изображения с использованием CSS, выровняв его по правому краю, задав рамки вокруг изображения.

**Вариант 1 и 4.** Задайте цвет использованной ссылки, к ссылкам будет добавляться пунктирная линия при наведении на них курсором мыши, а вид самого курсора будет изменяться: для 1-ой ссылки – на  .

**Вариант 2 и 5.** Задайте цвета использованной и неиспользованной ссылок, к ссылкам добавьте сплошную линию.

**Вариант 3 и 6.** Задайте подчеркивание ссылки цветом, отличным от самой ссылки. Установите вид курсора для 2-ой ссылки – на  .

## Лабораторная работа № 7

### ПОСТРОЕНИЕ СПИСКОВ

**Цель работы:** приобрести опыт создания списочных структур различных типов.

#### Методические указания

Все теги для создания списков могут быть условно разделены на две группы: одни определяют общий вид списка, а другие задают внутреннюю структуру списка.

Существуют, в основном, четыре вида списков, которые могут быть размещены на Web-странице: нумерованный (маркированный), нумерованный, вложенный (иерархический) и список с определениями.

*Ненумерованный список* (unordered list) строится по следующему шаблону:

Шаблон:

`<UL>`

`<Li>` Первая строка списка

`<Li>` Вторая строка списка

...

`<Li>` n-я строка списка

`</UL>`

Вид списка:

• Первая строка списка

• Вторая строка списка

• ...

• n-я строка списка

Для построения списка необходимо использовать два вида тегов:

1. `<UL>` `</ul>` — для обрамления списка

2. `<Li>` — для обозначения каждого пункта списка.

При построении *нумерованного списка* вместо тега `<UL>` `</ul>` используется тег `<OL>` `</ol>`, к которому добавляется атрибут *type* для определения способа нумерации. Некоторые возможные варианты нумерации представлены в табл. 7.1.

Таблица 7.1

Значение атрибута type	Способ нумерации
<i>type = 'a'</i>	a, b, c, d ...
<i>type = '4'</i>	4,5,6,7,8 ...
<i>type = '7'</i>	1,11, III, IV...
<i>type = 'B'</i>	B, C, D...

Атрибут *type* является частью тега `<OL>` и записывается, например, так: `<OL type= '1' >`.

*Список с определениями* (definition list) строится с помощью тегов трех наименований:

`<DL >` `< /dl >` – тег для определения начала и конца списка с определениями;

`<Dt >` — тег для описания пункта списка с определениями, в котором могут быть вложенности;

`<Dd >` – тег для описания вложенных пунктов.

Шаблон списка с определениями в кодах HTML и его внешний вид на странице представлен ниже.

Шаблон: `<DL>`

Вид списка:

`<Dt>` *Первая строка*

*Первая строка*

`<Dd >` *Определение 1*

*Определение 1*

`<Dd >` *Определение 2*

*Определение 2*

`<Dt>` *Вторая строка*

*Вторая строка*

`<Dd >` *Определение 3*

*Определение 3*

При формировании *вложенных списков* используются различные комбинации трех видов тегов: `<UL>` `</ul>`, `<OL>` `</ol>` и `<L>`.

Ненумерованный (*маркированный*) список может иметь своеобразный стиль маркеров, которые могут задаваться двумя способами с помощью атрибута *class* в теге `<OL>` с указанием кода маркера, например, `<OL class = "CO1">`, или атрибута *type* в теге `<UL>`, например, `<UL type= circle>`

## 7.1. Типовое задание

*Пример 1.* Сформируйте Web-страницу со списками следующего вида:

# СПИСКИ

---

## МАРКИРОВАННЫЙ СПИСОК

- Пункт 1 списка
  - Пункт 2 списка
  - Пункт 3 списка
- 

## НУМЕРОВАННЫЙ СПИСОК

(использован атрибут *value* = 4, нумерация осуществляется римскими числами)

IV. Пункт 1

V. Пункт 2

VI. Пункт 3

VII. Пункт 4

---

## ВЛОЖЕННЫЙ СПИСОК

- Пункт 1
    - 1. Пункт 1.1
    - 2. Пункт 1.2
  - Пункт 2
    - 1. Пункт 2.1
    - 2. Пункт 2.2
  - Пункт 3
    - 1. Пункт 3.1
    - 2. Пункт 3.2
- 

## СПИСОК С ОПРЕДЕЛЕНИЯМИ

Пункт 1

Определение пункта 1

Другое определение пункта 1

Пункт 2

Определение пункта 2

Пункт 3

Определение пункта 3

---

## ДВА СПОСОБА ПЕРЕОПРЕДЕЛЕНИЯ СТИЛЯ В МАРКИРОВАННОМ СПИСКЕ

- Пункт 1
  - Пункт 2
    - o Пункт 1 списка
    - o Пункт 2 списка
    - o Пункт 3 списка
- 

**Пример 2.** Разберите и проанализируйте алгоритм в кодах HTML.

```
<Html>
<Head>
<Title>Построение списков </title>
<Meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
<Style type="text/css">
OL.c01 {list-style-type: square}
</style>
</head>
<! цвет фона светло-зеленый>
<Body bgcolor=lime>
<Center>
<Font size=7 colors "purple ">c</font> <Font size=6
Color="blue">n</font><Font size=5 color ="green">u</font>
<Font size=4 color-"maroon">c</font><Font size=3
color="fuchsia">к</font><Font size=2 color="orange">u</font>
</center>
<! горизонтальная линия>
<Hr align-center size=6 color=orange>
<Center>
<Font color=red><H3>МАРКИРОВАННЫЙ СПИСОК<M>
</font>
</center>
<UL>
<Li>Пункт 1 списка
<Li>Пункт 2 списка
```



```

<Li>Пункт 3 списка
</ul>
<Hr>
<Center>
<Font color=red><H3>НУМЕРОВАННЫЙ СПИСОК <Br>
(использован атрибут value =4, нумерация осуществляется римскими
числами)</H3>
</font>
</center>
<OL type="I">
<Li value = 4>Пункт 1
<Li>Пункт 2
<Li>Пункт 3
</Li>Пункт 4
</ol>
<Hr>
<Center>
<Font color=red><H3> ВЛОЖЕННЫЙ СПИСОК</h3>
</font>
</center>
<UL>
<Li>Пункт 1
<OL>
<Li>Пункт 1.1
<Li>Пункт 1.2
</ol>
<Li>Пункт 2 <OL>
<Li>Пункт 2.1 <Li>Пункт 2.2
</ol>
<Li>Пункт 3 <OL>
<Li>Пункт 3.1 <Li>Пункт 3.2
</ol>
</ul>
<Hr>
<Center>
<Font color=red><H3>СПИСОК С ОПРЕДЕЛЕНИЯМИ</H3>
</font>
</center>
<Dt>Пункт 1
<Dd> Определение пункта 1
<Dd> Другое определение пункта 1
<Dt> Пункт 2

```

```

<Dd> Определение пункта 2
<Dt>Пункт 3
<Dd>Определение пункта 3
</dl>
<Hr>
<Center>
<Font color =red><H3>ДВА СПОСОБА ПЕРЕОПРЕДЕЛЕНИЯ
СТИЛЯ В МАРКИРОВАННОМ СПИСКЕ</H3>
</font>
</center>
<OL class="c01">
<Li> Пункт 1
<Li> Пункт 2
</ol>
<Hr>
<UL type=circle>
<Li>Пункт 1 списка
<Li>Пункт 2 списка
<Li>Пункт 3 списка
</ul>
<Hr>
</Body>
</Html>

```

## 7.2. Контрольные вопросы

1. Виды списков, размещаемых на Web-страницах.
2. Построение нумерованного, маркированного списков.
3. Построение иерархического списка.
4. Каков шаблон списка с определениями в HTML кодах.
5. Каково назначение атрибута *class*

## 7.3. Контрольные задания

Сформируйте списки следующих видов:

### **Вариант 1**

#### **Пункт 1**

- A. Пункт 1.1
  - ◆ Пункт 1.1.1

- ◆ Пункт 1.1.2
- ◆ Пункт 1.1.3
- В. Пункт 1.2
  - Пункт 1.2.1
  - Пункт 1.2.2
- Пункт 2
  - 2.Пункт 1
    - 1. Пункт 2.1.1
    - 2. Пункт 2.1.2
    - 3. Пункт 2.1.3
  - 3.Пункт 2
    - Пункт 2.2.1
    - Пункт 2.2.2
    - Пункт 2.2.3
- Пункт 3
  - Определение пункта 3
  - Другое определение пункта 3

## ***Вариант 2***

- Пункт 1
  - Пункт 1.1
    - 3. Пункт 1.1.1
    - 4. Пункт 1.1.2
    - 5. Пункт 1.1.3
  - Пункт 1.2
    - Пункт 1.2.1
      - Определение пункта 1.2.1
      - Другое определение пункта 1.2.1
    - Пункт 1.2.2
      - Определение пункта 1.2.2
      - Другое определение пункта 1.2.2
    - Пункт 1.2.3
      - Определение пункта 1.2.3
- Пункт 2
  - С. Пункт 2.1
  - Д. Пункт 2.2
- Пункт 3
  - Пункт 3.1
  - Пункт 3.2

### **Вариант 3**

#### III. Пункт 1

- Пункт 1.1
- Пункт 1.2
- Пункт 1.3

#### IV. Пункт 2

##### 1. Пункт 2.1

###### Пункт 2.1.1

Определение пункта 2.1.1

###### Пункт 2.1.2

Определение пункта 2.1.2

Другое определение пункта 2.1.2

###### Пункт 2.1.3

Определение пункта 2.1.3

##### 2. Пункт 2.2

###### ■ Пункт 2.2.1

В. Пункт 2.2.1.1

С. Пункт 2.2.1.2

###### ■ Пункт 2.2.2

#### V. Пункт 3

- Пункт 3.1
- Пункт 3.2

### **Вариант 4**

#### 5. Пункт 1

##### Пункт 1.1

Определение пункта 1.1

##### Пункт 1.2

Определение пункта 1.2

##### Пункт 1.3

Определение пункта 1.3

Другое определение пункта 1.3

#### 6. Пункт 2

##### ■ Пункт 2.1

###### 1. Пункт 2.1.1

- Пункт 2.1.1.1
- Пункт 2.1.1.2
- Пункт 2.1.1.3

###### 2. Пункт 2.

Ш. Пункт 2.1.2.1

- Пункт 2.2
  - Пункт 2.2.1
  - Пункт 2.2.2

7. Пункт 3

**Вариант 5**

■ Пункт 1

3. Пункт 1.1

Пункт 1.1.1

Определение пункта 1.1.1

Другое определение пункта 1.1.1

Пункт 1.1.2

Определение пункта 1.1.2

Пункт 1.1.3

А Пункт 1.1.3.1

В Пункт 1.1.3.2 4.

4. Пункт 1.2

- Пункт 1.2.1
- Пункт 1.2.2
  - Пункт 1.2.2.1
  - Пункт 1.2.2.2

■ Пункт 2

А. Пункт 2.1

В. Пункт 2.2

- Пункт 2.2.1
- Пункт 2.2.2
- Пункт 2.2.3

■ Пункт 3

**Вариант 6**

Пункт 1

○ Пункт 1.1

А. Пункт 1.1.1

■ Пункт 1.1.1.1

■ Пункт 1.1.1.2

■ Пункт 1.1.1.3

В. Пункт 1.1.2

○ Пункт 1.2

Пункт 1.2.1

Определение пункта 1.2.1

Другое определение пункта 1.2.1

Пункт 1.2.2

Определение пункта 1.2.2

Пункт 1.2.3

Пункт 2

Ш. Пункт 2.1

- Пункт 2.1.1
- Пункт 2.1.2

IV. Пункт 2.2

- A. Пункт 2.2.1
- B. Пункт 2.2.2
- C. Пункт 2.2.3

## ПОСТРОЕНИЕ СПИСКОВ В КАСКАДНЫХ ТАБЛИЦАХ

**Цель работы:** приобрести опыт создания списочных структур различных типов в каскадных таблицах.

### *Методические указания*

С помощью CSS можно создать маркированные и нумерованные списки, а также использовать в качестве маркера подходящее изображение. Для создания вложенных списков необходимо разместить одну пару тэгов (стартовый и завершающий) внутри другой.

Существует возможность задания следующих атрибутов списков и их элементов.

#### **1. Тип маркера**

*Синтаксис нумерованного списка:* `<ul style="list-style-type: тип">`

Допустимо задавать следующие значения типов маркера для нумерованных списков:

- **disc** (маркер имеет вид закрашенного круга – ●);
- **circle** (маркер имеет вид окружности – ○);
- **square** (маркер имеет вид закрашенного квадрата – ■).

`<ul style="list-style-image: URL('адрес')">`

В этом случае в качестве маркера нумерованного списка используется указанный графический объект.

*Синтаксис нумерованного списка:* `<ol style="list-style-type: тип">`

Допустимо задавать следующие значения типов маркера для нумерованных списков:

- **decimal** (маркер имеет вид обычных десятичных арабских цифр – 1, 2, 3, ...);
- **lower-alpha** (маркер имеет вид строчных латинских букв – a, b, c, ...);

- **upper-alpha** (маркер имеет вид прописных латинских букв – A, B, C, ...);
- **lower-roman** (маркер имеет вид римских цифр, представленных строчными буквами – i, ii, iii, ...);
- **upper-roman** (маркер имеет вид римских цифр, представленных прописными буквами – I, II, III, ...).

## 2. Положение блока маркера

*Синтаксис:*

```
<ul style="list-style-position: положение">
```

Блок маркера может находиться за пределами главного структурного блока (**outside**) или в первом строковом блоке (**inside**).

В табл. 8.1 перечислены свойства элементов, предназначенных для создания и изменения списков.

Таблица 8.1

Свойство	Значение	Описание	Пример
list-style	disc circle square decimal lower-roman upper-roman lower-alpha upper-alpha none	Вид маркера. Первые три используются для создания маркированного списка, а остальные – для нумерованного.	LI {list-style: circle}  LI {list-style: upper-alpha}
list-style-image	none URL	Устанавливает символом маркера любую картинку.	LI {list-style-image: url(check.gif)}
list-style-position	outside inside	Выбор положения маркера относительно блока строк текста.	LI {list-style-position: inside}

Поскольку тег <LI> наследует стилевые свойства тега <OL> или <UL>, который выступает в качестве его родителя, то можно устанавливать стиль как



для селектора UL, так и для селектора LI. Так, в примере 1 используется селектор UL, для него и задаются стилевые параметры.

**Пример 1.** Создание маркированного списка

```
<html>
<head>
<style type="text/css">
UL {
list-style: square;          /* Маркеры в виде квадрата */
list-style-position: outside; /* Маркеры размещаются за пределами текстового
блока */
color: navy                 /* Цвет текста списка */
}
</style>
</head>

<body>
<ul>
<li>Заголовок должен быть короче трех строк.</li>
<li>При названии разделов используйте уже устоявшиеся термины, такие как
гостевая книга, чат, ссылка, главная страница и другие.</li>
<li>Перед использованием специального термина или слова, решите, будет ли
оно понятно читателю.</li>
</ul>
</body>
</html>
```

В данном примере используются квадратные маркеры и их внешнее размещение относительно текста. Цвет – темно-синий.

Чтобы установить свое собственное изображение в качестве маркера применяется параметр **list-style-image**, как показано в примере 2.

**Пример 2.** Использование изображений в качестве маркера

```
<html>
<head>
<style type="text/css">
LI {
list-style-image: url('images/check.gif') /* Путь к файлу с маркером */
}

```

```

</style>
</head>
<body>
<ul>
<li>Заголовок должен быть короче трех строк.</li>
<li>При названии разделов используйте уже устоявшиеся термины, такие как
гостевая книга, чат, ссылка, главная страница и другие.</li>
<li>Перед использованием специального термина или слова, решите, будет ли
оно понятно читателю.</li>
</ul>
</body>
</html>

```

Результат данного примера показан ниже. В качестве маркеров используется маленькая картинка.

Некоторые примеры создания различных списков приведены в табл. 8.2.

Таблица 8.2

Код HTML	Пример
1	2
<code>&lt;li style="list-style: disc"&gt;</code>	Что следует учитывать при тестировании сайта: <ul style="list-style-type: none"> <li>• работоспособность всех ссылок</li> <li>• поддержку разных браузеров</li> <li>• читабельность текста</li> </ul>
<code>&lt;li style="list-style: circle"&gt;</code>	Что следует учитывать при тестировании сайта: <ul style="list-style-type: none"> <li>○ работоспособность всех ссылок</li> <li>○ поддержку разных браузеров</li> <li>○ читабельность текста</li> </ul>
<code>&lt;li style="list-style: square"&gt;</code>	Что следует учитывать при тестировании сайта: <ul style="list-style-type: none"> <li>▪ работоспособность всех ссылок</li> <li>▪ поддержку разных браузеров</li> <li>▪ читабельность текста</li> </ul>
<code>&lt;li style="list-style: decimal"&gt;</code>	Нумерованный список с арабскими цифрами: <ol style="list-style-type: none"> <li>1. текст</li> <li>2. текст</li> <li>3. текст</li> </ol>

1	2
<code>&lt;li style="list-style: lower-roman"&gt;</code>	Нумерованный список со строчными римскими цифрами: i. текст ii. текст iii. текст
<code>&lt;li style="list-style: upper-roman"&gt;</code>	Нумерованный список с заглавными римскими цифрами: I. текст II. текст III. текст
<code>&lt;li style="list-style: lower-alpha"&gt;</code>	Нумерованный список со строчными буквами латинского алфавита: a. текст b. текст c. текст
<code>&lt;li style="list-style: upper-alpha"&gt;</code>	Нумерованный список с заглавными буквами латинского алфавита: A. текст B. текст C. текст

### 8.1. Типовое задание

**Пример 1.** Создайте на Web-странице нумерованный список (с использованием CSS) следующего вида:

Виды списков

- [маркированный](#)
- [нумерованный](#)
- [многоуровневый](#)

**Пример 2.** Рассмотрите и проанализируйте решение практического задания:

```
<html>
<head>
<title>Шрифт в списках</title>
<style type="text/css">
UL {font-family: Georgia, "Times New Roman", Times, serif
/* Путь к файлу с маркером */
```

```

margin-top:0.5em          /* Отступ от верхнего края */
margin-bottom:0.5em      /* Отступ от нижнего края */
margin-left: 40px        /* Отступ от левого края */
}
LI{
padding-left: 20px      /* Отступ от маркера до текста */
}
LI SPAN{
color: blue             /* Цвет текста в списке */
}
</style>
</head>
<body>
<ul>Виды списков
<li style="list-style: circle"><span>Маркированный</span></li>
<li style="list-style: circle"><span>Нумерованный</span></li>
<li style="list-style: circle"><span>Многоуровневый</span></li>
</ul>
</body>
</html>

```

**Пример 3.** Создайте на Web-странице многоуровневый список (с использованием CSS) следующего вида (рис. 8.1):

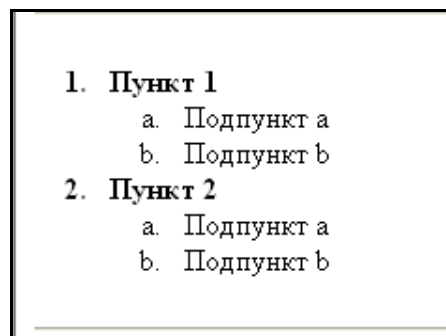


Рис. 8.1. Многоуровневый список

**Пример 3.** Рассмотрите и проанализируйте решение практического задания.

```

<html>
<head>
<style type="text/css">
OL LI { font-weight: bold}          /* Выделение пунктов */
OL OL { list-style: lower-alpha}    /* Тип нумерации подпунктов */

```

```

        OL OL LI { font-weight: normal} /* Оформление подпунктов */
    </style>
</head>
<body>
<hr>
    <ol>
    <li> Пункт 1 </li>
        <ol>
    <li> Подпункт а </li>
        <li> Подпункт b </li>
        </ol>
    <li> Пункт 2 </li>
        <ol>
    <li> Подпункт а </li>
        <li> Подпункт b </li>
        </ol>
    </ol>
<hr>
</body>
</html>

```

## 8.2. Контрольные вопросы

1. Виды списков в CSS.
2. Свойства CSS для управления видом списка.
3. Использование изображений в качестве маркера.

## 8.3. Контрольные задания

**Пример 1.** Создайте на Web-странице список (с использованием CSS) следующего вида:

Варианты					
1	2	3	4	5	6
Маркированный, тип маркера в виде диска, цвет текста – fuchsia	Нумерованный с заглавными римскими цифрами, цвет текста – maroon	Многоуровневый, тип маркеров 1-го уровня – арабские цифры, 2-го уровня – строчные римские цифры, цвет текста – olive	Маркированный, тип маркера в виде окружности, цвет текста – yellow	Нумерованный со строчными буквами латинского алфавита, цвет текста – navy	Многоуровневый, тип маркеров 1-го уровня – заглавные римские цифры, 2-го уровня – маркеры в виде квадрата, цвет текста – teal

## ОРГАНИЗАЦИЯ И ФОРМИРОВАНИЕ ТАБЛИЦ

**Цель работы:** приобрести навыки создания таблиц различных структур;

### Методические указания

Формирование таблицы на Web-странице сопровождается прорисовкой браузером ее рамок. Размер рамок автоматически согласовывается с размером окна просмотра браузера, размером текстов и изображений в ячейках таблицы.

При построении таблицы используются теги трех групп:

- 1) основной элемент описания таблицы — тег `< Table >... < /table >`;
- 2) определение строк в таблице — тег `< Tr >`;
- 3) описание ячеек в таблице — теги `< Td >` и `< Th >`.

Тег `<Table> </table>` использует целый ряд дополнительных атрибутов: *align*, *width*, *border*, *cellspacing*, *cellpadding*. В дальнейшем рассмотрим назначения каждого атрибута в отдельности.

Для выравнивания таблицы на странице применяется атрибут *align*:

- *align = "left"* — выравнивание таблицы по левому краю;
- *align = "center"* — выравнивание таблицы по центру;
- *align = "right"* — выравнивание таблицы по правому краю.

Ширина таблицы может быть задана в пикселах или в процентном отношении к ширине страницы, например, *width = 450* или *width = 40%*.

атрибут *border = ширина боковой грани* в пикселах,

атрибут *cellspacing = ширина фронтальных граней*,

атрибут *cellpadding = число пикселей* определяет расстояние размера пустого пространства вокруг текста в ячейках.

Сразу за тегом `< Table >` может быть размещен специальный тег для определения заголовка страницы `< Caption > < /caption >`.

С помощью атрибута *align* этом теге заголовок может располагаться над таблицей *align = "top"* или под таблицей *align = "bottom"*.

Каждую строку в таблице описывает тег `< Tr >`. Для выравнивания всех

ячеек в строке может использоваться атрибут *align* со своими стандартными значениями *left*, *center*, *right*.

Для описания в ячейках заголовков используется внутри тега `<Tr >` тег `< Th >`. При этом текст заголовка в ячейке обычно выделяется полужирным шрифтом.

Цвет шрифта текста в таблице, как правило, задается с помощью тега `< Font >` с атрибутом *color*, например, `< Font color = red >`, а цвет фона внутри ячейки задается атрибутом *bgcolor* внутри тега `< Th >`, например, `<Th bgcolor = "lime">`.

Тег `<Td>` определяет обычную ячейку в таблице.

В дальнейшем рассмотрим несколько примеров построения таблиц.

### 9.1. Типовое задание

**Пример 1.** Постройте таблицу следующего вида (рис. 9.1).

#### СТАНДАРТНАЯ ТАБЛИЦА

Заголовок таблицы	
Заголовок 1	Заголовок 2
Ячейка 1	Ячейка 2

Рис. 9.1. Стандартная таблица

Запись на языке HTML (фрагмент) представлена ниже:

```
<Center>  
<Font color=red><H3>СТАНДАРТНАЯ ТАБЛИЦА</H3>  
</font>  
</center>  
<Table border=4 cellspacing=3>  
<Caption> Заголовок таблицы </caption>  
<Tr> < Th bgcolor= "lime ">Заголовок 1  
      < Th bgcolor= "yellow "> Заголовок 2  
<Tr><Td> Ячейка 1  
      <Td> Ячейка 2  
</table >
```

**Пример 2.** Сформулируйте таблицу из шести ячеек с тремя заголовками и одной пустой ячейкой в правом верхнем углу (рис. 9.2).

### СПЕЦИАЛЬНАЯ ТАБЛИЦА

Заголовок таблицы		
	Заголовок 1	Заголовок 2
Заголовок 3	Ячейка 1	Ячейка 2

Рис. 9.2. Специальная таблица

Фрагмент построения таблицы на языке HTML:

```
<Center>  
<Font color=red><H3>СПЕЦИАЛЬНАЯ ТАБЛИЦА</H3>  
</font>  
</center>  
<Table border=4 cellspacing=3>  
<Caption> Заголовок таблицы </caption>  
<Tr><Th bgcolor="lime">  
  <Th bgcolor="lime">Заголовок 1  
  <Th bgcolor="lime">Заголовок 2  
  <Tr><Th bgcolor="lime">Заголовок 3  
  <Td>Ячейка 1  
  <Td> Ячейка 2  
</table >
```

Ячейки в таблице могут быть объединены с помощью атрибутов:

*rowspan* = *n*, где *n* (*n* > 1) — количество строк, выделяемых под ячейку;

*colspan* = *k*, где *k* (*k* > 1) — количество столбцов, выделенных под ячейку.

**Пример 3.** Постройте таблицу с объединенными ячейками (рис. 9.3).

### ОБЪЕДИНЕНИЕ ЯЧЕЕК

Заголовок	Заголовок 2
Ячейки 1	Ячейка 2
	Ячейка 3
	Ячейка 4

Рис. 9.3. Таблица с объединенными ячейками



Фрагмент алгоритма на языке HTML представлен ниже:

```
<Center>
<Font color=red><H3> ОБЪЕДИНЕНИЕ ЯЧЕЕК</H3>
</font>
</center>
<Table border=4 cellspacing=0 width=60%>
<Tr><Td bgcolor="yellow"><B>Заголовок 1</b>
<Td bgcolor="yellow">><B>Заголовок 2</b>
<Tr><Td rowspan=3 bgcolor="lime">Ячейка 1
<Td> Ячейка 2
<Tr><Td>Ячейка 3
<Tr><Td> Ячейка 4
</table>
```

При построении таблицы на языке HTML допускается вложение в ячейки других таблиц.

### 9.2. Контрольные вопросы

1. Структура таблицы в HTML-документе.
2. Основные атрибуты таблицы.
3. Вычерчивание рамок и управление размером и цветом ячеек таблицы.

### 9.3. Контрольные задания

Постройте таблицы следующих видов:

#### Вариант 1

Заголовок таблицы

	Заголовок 1	Заголовок 2		
Заголовок 3	Ячейка 1	Ячейка 1 Таблицы 2	Ячейка 2 Таблицы 2	Ячейка 3
Заголовок 4	Ячейка 2	Ячейка 3 Таблицы 2	Ячейка 4 Таблицы 2	Ячейка 4
Заголовок 5	Ячейка 5			

**Вариант 2**

Заголовок таблицы

Заголовок 1		Заголовок 2		
Ячейка				Заголовок 3
Ячейка 1 Таблицы 2	Ячейка 2 Таблицы 2	Ячейка 3 Таблицы 2		Заголовок 4
Ячейка 4 Таблицы 2	Ячейка 6			
Ячейка 5 Таблицы 2	Таблицы 2			

**Вариант 3**

Заголовок таблицы

Заголовок 1		Заголовок 2		
Ячейка				Заголовок 3
Ячейка 1 Таблицы 2	Ячейка 2 Таблицы 2	Ячейка 3 Таблицы 2		Заголовок 4
Ячейка 4 Таблицы 2	Ячейка 6			
Ячейка 5 Таблицы 2	Таблицы 2			

**Вариант 4**

Заголовок таблицы

Заголовок 1		Заголовок 2		
Ячейка				Заголовок 3
Ячейка 1 Таблицы 2	Ячейка 2 Таблицы 2	Ячейка 3 Таблицы 2		Заголовок 4
Ячейка 4 Таблицы 2	Ячейка 6			
Ячейка 5 Таблицы 2	Таблицы 2			

**Вариант 5**

## Заголовок таблицы

	Заголовок 1			Заголовок 2
Заголовок 3				Ячейка 4
Заголовок 4	Ячейка 1	Ячейка 2	Ячейка 3	Ячейка 5
Заголовок 5	Заголовок 1		Заголовок 2	Ячейка 6
	Таблицы 2		Таблицы 2	
	Ячейка 1	Ячейка 2	Ячейка 3	Ячейка 7
Таблицы 2	Таблицы 2	Таблицы 2		
	Ячейка 4	Ячейка 5		Ячейка 8
	Таблицы 2	Таблицы 2		

**Вариант 6**

## Заголовок таблицы

	Заголовок 1			Заголовок 2
Заголовок 3	Ячейка 1	Ячейка 2	Ячейка 3	Ячейка 4
Заголовок 4				Ячейка 5
Заголовок 5	Заголовок 1		Заголовок 2	Ячейка 6
	Таблицы 2		Таблицы 2	
	Ячейка 1	Ячейка 2	Ячейка 3	Ячейка 7
Таблицы 2	Таблицы 2	Таблицы 2		
	Ячейка 4	Ячейка 5		Ячейка 8
	Таблицы 2	Таблицы 2		

*Лабораторная работа № 10*  
**СОЗДАНИЕ ТАБЛИЦ С ПОМОЩЬЮ CSS**

**Цель работы:** приобрести навыки создания таблиц различных структур с помощью таблиц стилей.

**Методические указания**

**1. Задание заголовков**

При большом количестве таблиц на странице каждой из них необходимо задать заголовок, содержащий название таблицы и ее описание. Для этой цели можно получить универсальный код с помощью CSS, который задаст все необходимые атрибуты форматирования заголовка.

**Пример 1.** Установка заголовка сверху таблицы и выравнивание его по правому краю таблицы (рис. 10.1).

*Изменение добычи ресурсов по годам*

	<b>2007</b>	<b>2008</b>	<b>2009</b>
Нефть	43	51	79
Газ	33	57	68

Рис. 10.1. Выравнивание заголовка по правому краю

```
<html>
<head>
<style type="text/css">
  CAPTION {
    text-align: right;      /* Выравнивание по правому краю */
    font-style: italic;    /* Курсивный текст */
  }
</style>
</head>
<body>
  <table width=70% border=1 cellpadding=4 cellspacing=0 align=center>
  <caption>Изменение добычи ресурсов по годам</caption>
  <tr>
```

```

<th>&nbsp;   </th>
<th>2007</th>
<th>2008</th>
<th>2009</th>
</tr>
<tr>
<td>Нефть</td>
<td>43</td>
<td>51</td>
<td>79</td>
</tr>
<tr>
<td>Газ</td>
<td>33</td>
<td>57</td>
<td>68</td>
</tr>
</table>
</body>
</html>

```

Заголовок созданный при помощи тега **caption**, оказывается привязанным к таблице и не выходит за условные рамки, ограниченные ее шириной. Его можно задать, создав новый класс с именем **caption**, который применяется к параграфу (тегу **p**). Для этой цели добавляем отступы слева и справа через параметр **margin**. Ширина текстового блока совпадет с шириной таблицы. Расстояние от таблицы до текста регулируются значением **padding-bottom**.

*Пример 2.* Задание заголовка с использованием нового класса с именем **caption** (рис. 10.2).

	<b>2007</b>	<b>2008</b>	<b>2009</b>
Нефть	43	51	79
Газ	33	57	68

Рис. 10.2. Задание заголовка с именем **caption**

```

<html>
<head>
<style type="text/css">
  .caption {
    margin: 0px 15%;          /* Отступы сверху и сбоку от текста */
    padding-bottom: 4px;     /* Поле под текстом */
    text-align: right;       /* Выравнивание по правому краю */
    font-style: italic       /* Курсивный текст */
  }
</style>
</head>
<body>
  <p align-center class=caption> Изменение добычи ресурсов по годам</p>
  <table width=70% border=1 cellpadding=4 cellspacing=0 align=center>
    <tr>
      <th>&nbsp;</th>
      <th>2007</th>
      <th>2008</th>
      <th>2009</th>
    </tr>
    <tr>
      <td>Нефть</td>
      <td>43</td>
      <td>51</td>
      <td>79</td>
    </tr>
    <tr>
      <td>Газ</td>
      <td>33</td>
      <td>57</td>
      <td>68</td>
    </tr>
  </table>
</body>
</html>

```

## 2. Границы таблиц

В таблицах необходимо отделять содержимое одной ячейки от другой, используя границы таблицы. Для этого рекомендуется воспользоваться стилевым атрибутом **border**, применяя его к ячейкам таблицы. Рамки создаются

для каждой ячейки, но в местах их соприкосновения получается граница удвоенной толщины. Отслеживать соприкосновение линий и вместо двойной границы изображать одинарную можно с помощью свойства **border-collapse** со значением **collapse**. Добавление параметра убирает эту особенность, и толщина всех линий становится одинаковой (пример 3).

*Пример 3.* Установка рамок для ячеек (рис. 10.3).

	2007	2008	2009
Нефть	43	51	79
Газ	33	57	68

Рис. 10.3. Установка рамок для ячеек

```

<html>
<head>
<style type="text/css">
  Table {
    border-collapse: collapse /* Линия между ячейками отображается как одна */
  }
  TD {
    border: 2px solid green; /* Параметры рамки вокруг ячеек таблицы */
    text-align: center /* Выравнивание текста по центру */
  }
</style>
</head>
<body>
  <table width=500 cellpadding=2 cellspacing=0 align=center>
    <tr>
      <td>&nbsp;</td>
      <td>2007</td>
      <td>2008</td>
      <td>2009</td>
    </tr>
    <tr>
      <td>Нефть</td>
      <td>43</td>
      <td>51</td>

```

```

<td>79</td>
</tr>
<tr>
<td>Газ</td>
<td>33</td>
<td>57</td>
<td>68</td>
</tr>
</table>
</body>
</html>

```

Для создания односторонних линий внутри таблицы можно добавить для селектора **TD** рамку, но отменить линию справа и снизу, задавая соответствующим атрибутам значение **none**. Тогда при состыковке ячеек их границы не будут накладываться друг на друга, т.к. линия будет только одна. Но при этом методе формирования границ нет линий снизу и справа у самой таблицы. Добавляя параметры **border-right** и **border-bottom** к селектору **table**, в итоге получим нужную рамку (пример 4). Для односторонности следует, чтобы стиль, толщина и цвет границы во всех случаях совпадали.

*Пример 4.* Создание односторонних линий внутри таблицы

```

<html>
<head>
<style type="text/css">
  Table {
    border-right: 2px solid green; /* Граница у таблицы справа */
    border-bottom: 2px solid green; /* Граница у таблицы снизу */
  }
  TD {
    border: 2px solid green; /* Параметры рамки вокруг ячеек таблицы */
    border-right: none; /* Убираем линию справа */
    border-bottom: none; /* Убираем линию снизу */
    text-align: center /* Выравнивание текста по центру */
  }
</style>
</head>
<body>
  <table width=500 cellpadding=2 cellspacing=0 align=center>

```



```

<tr>
<td>&nbsp;</td>
<td>2007</td>
<td>2008</td>
<td>2009</td>
</tr>
<tr>
<td>Нефть</td>
<td>43</td>
<td>51</td>
<td>79</td>
</tr>
<tr>
<td>Газ</td>
<td>33</td>
<td>57</td>
<td>68</td>
</tr>
</table>
</body>
</html>

```

### 3. Управление размером таблиц

При ограничении места на веб-странице можно зафиксировать высоту таблицы и добавить полосы прокрутки, с помощью которых прокручивается ее содержимое. Кроме того, если верхнюю строку таблицы закрепить на одном месте, то становится удобно просматривать содержимое длинной таблицы. Чтобы создать такую конструкцию необходимо две таблицы и два контейнера *div*. Первый *div* назовем его *container*, необязателен, но требуется для того, чтобы добавить рамку вокруг области и установить общую ширину. Второй *div*, с именем *scroll* с помещенной внутрь него таблицей, задает полосы прокрутки и высоту отображения данных. Разделение одной таблицы на две необходимо, поскольку внутри тега *table* вставить *div* нельзя. Так как две таблицы – заголовок и содержимое – создаются независимо, то для совмещения колонок требуется задать ширину каждой колонки с помощью параметра *width*.

**Пример 5.** Фиксация высоты таблицы и добавление полосы прокрутки (рис. 10.4).



	2007	2008	2009
Нефть	45	51	79
Газ	33	57	68

Рис. 10.4. Фиксация высоты таблицы и добавление полосы прокрутки

```
<html>
<head>
<style type="text/css">
  TH {
    background: red;          /* Фон заголовка */
    color: white;            /* Цвет текста */
    text-align: left         /* Выравнивание текста заголовка*/
  }
  DIV.container {
    border: 1px solid black; /* Рамка вокруг таблицы */
    width: 400px            /* Ширина таблицы в пикселах */
  }
  DIV.scroll {
    height: 50px;           /* Высота области прокрутки */
    overflow: scroll         /* Добавляем полосы прокрутки */
  }
</style>
</head>
<body>
<div class=container >
<table width=100% border=0 cellpadding=4 cellspacing=0>
  <tr>
    <th width=100>&nbsp;</th>
    <th width=100>2007</th>
    <th width=100>2008</th>
    <th width=100>2009</th>
  </tr>
</table>
<div class=scroll>
<table width=100% border=0 cellpadding=4 cellspacing=0>
```

```

<tr>
<td width=100>Нефть</td>
<td width=100>43</td>
<td width=100>51</td>
<td width=100>79</td>
</tr>
<tr>
<td>Газ</td>
<td>33</td>
<td>57</td>
<td>68</td>
</tr>
</table>
</div>
</div>
</body>
</html>

```

К таблицам можно применять различные атрибуты форматирования: фон, вид линий, цвет линий, выравнивание текста в ячейках и т.д. В зависимости от этого таблицы будет выглядеть по-разному.

#### 4. Создание меню

Таблицы обладают некоторыми преимуществами по сравнению с другими элементами. Таблицы удобно использовать, менять параметры их отображения и выравнивать по любому краю содержимое ячеек. Используя тег *table* совместно со стилями, можно получить самый разнообразный вид меню. Например, можно добавить рамку вокруг таблицы через селектор *table*, а с помощью контекстного селектора *.menu TD* установить белые линии между ячейками и поля вокруг их содержимого (пример 6).

**Пример 6.** Создание меню с помощью таблиц (рис.10.5).

Первый элемент меню	Второй элемент меню	Третий элемент меню	Четвертый элемент меню
---------------------	---------------------	---------------------	------------------------

Рис. 10.5. Вид меню, созданного с помощью таблиц

```

<html>
<head>
<style type="text/css">
    TABLE.menu {
        background: pink;           /* Цвет фона меню*/
        width: 80%;                /* Ширина меню*/
        border: 1px solid black;    /* Рамка вокруг таблицы*/
        text-align: center         /* Выравнивание текста по центру*/
    }
    .menu TD {
        border: 1px solid white;   /* Линия между ячейками*/
        padding: 4px              /* Поля вокруг текста*/
    }
</style>
</head>
<body>
<table align=center cellspacing=0 class=menu>
    <tr>
        <td><a href=link1.html>Первый элемент меню</a></td>
        <td><a href=link2.html>Второй элемент меню</a></td>
        <td><a href=link3.html>Третий элемент меню</a></td>
        <td><a href=link4.html>Четвертый элемент меню</a></td>
    </tr>
</table>
</body>
</html>

```

При создании горизонтального меню необходимо подсвечивание фона ячейки таблицы при наведении на нее курсора мыши. Для создания такой подсветки текста воспользуемся псевдоклассом `hover`, который управляет стилем ссылки при наведении на нее курсора мыши. Чтобы в качестве ссылки

выступала вся ячейка целиком, а не только текст в ней, следует добавить для селектора A свойство *display:block*. В данном случае (пример 7) это свойство превращает ссылку в блочный элемент и подсветка будет происходить при наведении курсора мыши в любое место ячейки.

**Пример 7.** Подсвечивание фона ячейки таблицы (рис. 10.6).



Рис. 10.6. Подсвечивание фона ячейки горизонтального меню

```
<html>
<head>
<style type="text/css">
  TABLE.menu {
    background: pink;          /* Цвет фона меню*/
    width: 100%;             /* Ширина меню*/
    border: 1px solid black;  /* Рамка вокруг таблицы*/
    text-align: center       /* Выравнивание текста по центру*/
  }
  .menu TD {
    border: 1px solid white;  /* Линия между ячейками*/
  }
  .menu A {
    display: block;          /* Ссылка на всю ячейку*/
    padding: 4 px;          /* Поля вокруг ссылок*/
    width: 100%;            /* Работает в браузере Internet Explorer*/
  }
  .menu A:hover {
    background: blue;        /* Цвет фона наведения мыши*/
  }
</style>
</head>
<body>
<table align=center cellspacing=0 class=menu>
  <tr>
    <td><a href=link1.html>Первый элемент меню</a></td>
    <td><a href=link2.html>Второй элемент меню</a></td>
    <td><a href=link3.html>Третий элемент меню</a></td>
```

```

<td><a href=link4.html>Четвертый элемент меню</a></td>
</tr>
</table>
</body>
</html>

```

### 10.1. Типовое задание

**Пример 1.** Постройте простую таблицу с заголовком следующего вида (рис. 10.7).

	2007	2008	2009
Нефть	43	51	79
Газ	33	57	68

Рис. 10.7. Простая таблица с заголовком

Фрагмент алгоритма представлен ниже.

```

<html>
<head>
<style type="text/css">
  Table {
    border: 1px solid black; /* Рамка вокруг таблицы */
    text-align: center      /* Выравнивание текста ячеек по центру */
  }
  TH {
    background: black;      /* Цвет фона заголовка */
    color: white           /* Цвет текста заголовка */
  }
  TD. title {
    text-align: left       /* Выравнивание по левому краю */
  }
</style>
</head>
<body>
  <table width=300 border=0 cellpadding=4 cellspacing=0>
  <tr>
  <th>&nbsp;</th>
  <th>2007</th>
  <th>2008</th>

```

```

<th>2009</th>
</tr>
<tr>
<td class=title>Нефть</td>
<td>43</td>
<td>51</td>
<td>79</td>
</tr>
<tr>
<td class=title >Газ</td>
<td>33</td>
<td>57</td>
<td>68</td>
</tr>
</table>
</body>
</html>

```

**Пример 2.** Постройте таблицу с цветным заголовком следующего вида (рис. 10.8).

	2007	2008	2009
Нефть	43	51	79
Газ	33	57	68

Рис. 10.8. Таблица с цветным заголовком

Фрагмент алгоритма представлен ниже.

```

<html>
<head>
<style type="text/css">
Table {
border: 1px solid black;          /* Рамка вокруг таблицы */
text-align: center;             /* Выравнивание текста ячеек по центру */

```

```

border-collapse: collapse      /* Линия между ячейками отображается как
одна */
}
TH {
background: darkblue;         /* Цвет фона заголовка */
border-bottom: 1px solid white; /* Линия под заголовком */
color: white                  /* Цвет текста заголовка */
}
TD {
border-bottom: 1px dashed black /* Пунктирная линия под ячейками */
}
TD. title {
text-align: left;            /* Выравнивание по левому краю*/
font-weight: bold;          /* Жирное начертание */
background: green            /* Цвет фона левой колонки */
}
</style>
</head>
<body>
<table width=300 border=0 cellpadding=4 cellspacing=0>
<tr>
<th>&nbsp;</th>
<th>2007</th>
<th>2008</th>
<th>2009</th>
</tr>
<tr>
<td class=title>Нефть</td>
<td>43</td>
<td>51</td>

```



```
<td>79</td>
</tr>
<tr>
<td class=title >Газ</td>
<td>33</td>
<td>57</td>
<td>68</td>
</tr>
</table>
</body>
</html>
```

### ***10.2. Контрольные вопросы***

1. Атрибуты форматирования заголовков в CSS.
2. Стилиевые атрибуты для формирования границ таблицы.
3. Фиксация высоты таблицы и добавление полос прокрутки.
4. Создание меню с помощью стилей.
5. Подсвечивание фона ячеек таблицы.

### ***10.3. Контрольные задания***

Постройте таблицы следующих видов.

#### ***Вариант 1 и 4***

Задайте заголовок таблицы, создав новый класс с именем **caption**. Выровняйте его по правому краю.

Создайте таблицу с чередующимися колонками (цвет фона четных и нечетных колонок различный), используя два класса: **odd** – для нечетных колонок, идущий сразу после колонки с заголовками строк и **even** – для четных колонок. Слева от каждой колонки установите пунктирную линию, чтобы разделение содержимого ячеек происходило не только по цвету, но и с помощью границы между ними.

### ***Вариант 2 и 5***

Задайте заголовок таблицы, создав новый класс с именем **caption**. Выровняйте его по левому краю.

Создайте таблицу с цветным заголовком. Левую колонку выделите с помощью фона, а строки таблицы отделите друг от друга пунктирной линией. Примените свойство **border-collapse**, чтобы пунктирная линия в последних ячейках не совмещалась с нижней границей таблицы, т.е. линия между ячейками отображалась как одна.

### ***Вариант 3 и 6***

Задайте заголовок таблицы, создав новый класс с именем **caption**. Выровняйте его по центру.

Создайте таблицу с чередующимися строками (цвет фона четных и нечетных строк различный), используя класс **even**, который добавляется к тегу **TR** для всех строк через одну. Цвет фона остальных строк установите в селекторе **Body**.

## Лабораторная работа № 11

### ПОСТРОЕНИЕ ФОРМ

**Цель работы:** приобрести навыки создания форм различных типов.

#### Методические указания

Формы (формуляры) превращают WWW в интерактивную среду. Они реализуют простейший способ связи пользователя и сервера.

**Форма** – это набор полей для ввода текста, командные кнопки и флажки, содержимое которых возвращается серверу по указанию пользователя. Сервер в дальнейшем обрабатывает информацию и, если необходимо, возвращает пользователю ответы.

В дальнейшем рассмотрим использование некоторых тегов для создания форм и на фрагментах одного алгоритма виды таких форм.

Тег `< Input >` позволяет создавать различные элементы управления в форме в зависимости от своих атрибутов *type*, *value*, *checked*, *size* и *name*. Значения атрибута *type* представлены в табл. 11.1.

Таблица 11.1

Значение атрибута	Функциональное назначение
<i>type</i> = "text"	Создание поля ввода для текстовых данных
<i>type</i> = "password"	Создание строки для ввода пароля
<i>type</i> = "checkbox"	Создание флажков
<i>type</i> = "radio"	Формирование переключателей
<i>type</i> = "submit"	Описание кнопки, нажатие на которую подтверждает ввод информации в форму
<i>type</i> = "reset"	Описание кнопки для отмены ввода данных в форму
<i>type</i> = "image"	Построение кнопки с рисунком
<i>type</i> = "file"	Средство выбора файла для присоединения к форме

Атрибут *value* используется для определения произвольного текста в поле ввода (*type* = "text") или для надписи на кнопке (*type* = "submit").

Атрибут *checked* определяет переключатель, требуемый по умолчанию.

Атрибут *size* описывает длину поля ввода.

Атрибут *name* — имя поля (элемента управления).

**Пример 1.** Построение элементов управления различного вида с использованием тега `<Input >` (рис. 11.1).

**Поля ввода  
ЭЛЕМЕНТ INPUT**

**Ввод текстовой строки**

**Ввод пароля**

**Флажки**

**Переключатели**

**Кнопка подтверждения ввода**

**Кнопка с изображением**

**Кнопка очистки формы**

Рис. 11.1. Вид построенной формы

Код в HTML для построения элементов форм представлен ниже.

```
<Html>  
<Head>  
<Title>Создание форм </title>  
<Meta http-equiv="Content-Type" content="text/html; charset=windows-1251">  
</head>  
<body bgcolor=navy text=lime>  
<Hr color="red">  
<Center>  
<Font color=red> <H2>ПОЛЯ ВВОДА <Br> ЭЛЕМЕНТ INPUT</h2>  
</font>  
</center>  
<H3> Ввод текстовой строки </h3>  
<Input type="text" size=50>
```

```

<H3> Ввод пароля </h3>
<Input type="password">
<H3> Флажки </h3>
<Input type="checkbox " name = "F001" checked>
<Input type="checkbox" name="F001 ">
<H3> Переключатели </h3>
<Input type="radio " name="S001" value="Первы й">
<Input type="radio " name="S001" value="Второй ">
<Input type="radio " name="S001 " value="Третий " checked>
<H3> Кнопка подтверждения ввода </h3>
<Input type="submit" value="submit">
<H3> Кнопка с изображением </h3>
<Input type="image" src= "fon01.gif">
<H3> Кнопка очистки формы </h3>
<Input type= "reset" value="reset">
<H3> Файл </h3>
<Input type="file" name="photo" accept=" image/*">
<Hrcolor="red">
</body>
</html>

```

Для создания списка или меню на гипертекстовой странице применяется тег `<Select>` `</select>` с дополнительным внутренним тегом `<Option>`. Тег `<Option>` предназначен для создания пунктов списка и может иметь два атрибута: `selected` определяет пункт меню, выбираемый по умолчанию, а атрибут `value` требуется для обработки данных на стороне сервера.

**Пример 2.** Построение элементов управления в формах с помощью тега `<Select>` (рис. 11.2).

## ЭЛЕМЕНТ SELECT

Первый
Второй
Третий

Рис. 11.2 Элемент управления в форме

```
<Center>
<Font color=red><H2>ЭЛЕМЕНТ SELECT </h2>
</font>
</center>
<Select multiple>
<Option value=a> Первый
<Option value=b> Второй
<Option value=c> Третий
</select>
```

**Пример 3.** Построение элементов управления в формах с помощью тега `<Select >` (рис. 11.3).

### Выпадающее меню

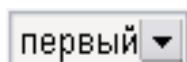


Рис. 11.3 Выпадающее меню

```
<H3> Выпадающее меню </h3>
<Select>
<option>первый </option>
<option>второй </option>
<option>третий </option>
</select>
```

Тег `<Textarea > </textarea >` описывает специальные области для ввода и просмотра текста. Для указания количества строк и столбцов используются атрибуты `rows` и `cols` соответственно.

**Пример 4.** Формирование области для ввода текста из 6 строк и 40 столбцов (рис. 11.4).

## ЭЛЕМЕНТ TEXTAREA

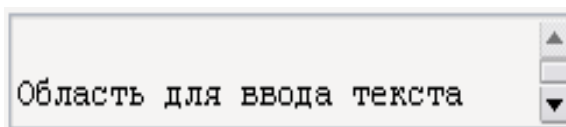


Рис. 11.4. Сформированная область для ввода текста

Фрагмент построения области в коде HTML:

```
<Center>  
<Font color=red><H2>ЭЛЕМЕНТ TEXTAREA </H2>  
</font>  
</center>  
<Textarea rows=6 cols=40>  
Область для ввода текста  
</textarea>
```

**Пример 5.** Построение кнопки с надписью и рисунком с помощью тега `< Button >` `< /button >` и `< Img >`.

Фрагмент в коде HTML:

```
<Center>  
<Font color=red><H2>КНОПКА С НАДПИСЬЮ И РИСУНКОМ</H2>  
</font>  
</center>  
<Button name="submit" value="submit" type="submit">  
Pencil</button>
```

**Пример 6.** Построение группы полей (рис. 11.5).

ГРУППА ПОЛЕЙ	
Заголовок группы	Имя: <input type="text"/>
	Фамилия: <input type="text"/>
<input type="text"/>	Телефон:1
	Фрагмент в коде HTML:

Рис. 11.5. Группа полей

```

<Center>
<Font color=red><H2>ГРУППА ПОЛЕЙ </H>
</font>
</center>
<Fieldset>
<Legend>Заголовок группы </Legend>
Имя: <Input name = "имя2 " type="text ">
Фамилия: <Input name="familiya2" type="text"><BR>
Телефон: <Input name="telefon2"type="text"><BR>
</fieldset>

```

### 11.1. Типовое задание

**Пример 1.** Постройте форму следующего вида (рис. 11.6).

**ПРИМЕР ФОРМЫ**

Имя:	<input type="text"/>
Фамилия:	<input type="text"/>
e-mail:	<input type="text"/>
Страна проживания:	Англия <input type="button" value="v"/>
Научный работник:	<input type="checkbox"/> Да <input type="checkbox"/> Нет
пол:	<input type="radio"/> М <input type="radio"/> Ж
Дополнительная информация:	<input type="text"/>
<input type="button" value="send"/> <input type="button" value="reset"/>	

Рис. 11.6. Пример формы



Проанализируйте представленный ниже код HTML и определите технологию построения управляющих элементов сложной формы:

```
<Html>
  <Head>
<Title>Пример формы </Title >
<Meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
</head>
<Body bgcolor=blue>
<Center>
<Font size=6 color=silver><i>ПРИМЕР ФОРМЫ</font> </i>
</center>
<p>
<Hr color= "red">
<Table border=3 bgcolor="lime">
<Form action= "mailto:Имя@сервер. домен " method= "post ">
<Tr>
<Td> <Label for= "имя" >Имя: </label>
<Td> <Input type= "text" id= "имя" >
<Tr>
<Td> <Label for= "фамилия" > Фамилия: </label>
<Td> <Input type="text" id= "фамилия" >
<Tr>
<Td> <Label for= "e-mail 1" >e-mail: </label>
<Td><Input type="text" id="e-mail">
<Tr>
<Td>Страна проживания: <Td> <Select> <Option>Англия </option>
<Option>Беларусь </option>
< Option >Германия </ option >
<Option> Франция </option> </select>
<Tr>
<Td>Научный работник:
<Td>
<Input type="checkbox" name= "F001" value="Да">Да
<Input type="checkbox" name="F001" value="Нет">Нет
<Tr>
<Td>пол:
<Td>&nbsp; &nbsp;
<Input type="radio " name= "poll" value = "Male "> М&nbsp;
<Input type="radio " name="poll" value="Female "> Ж
<Tr>
<Td>Дополнительная информация:
```

```

<Td><Textarea rows=6 cols=40>
</textarea>
<Tr>
<Td align = "bottom ">
<Input align= "bottom" type="submit" value= "send">
<Input type="reset" value= "reset" >
</form>
</table>
</body>
</html>

```

### 11.2. Контрольные вопросы

1. Какие теги используются для создания форм?
2. Каково назначение атрибутов *value*, *checked*, *size* ?
3. Каков порядок создания меню на гипертекстовой странице
4. Опишите технологию построения управляющих элементов сложной формы.

### 11.3. Контрольные задания

Постройте формы в виде таблиц.

#### Вариант 1

Вопрос 1	<input type="text"/> <input type="text"/> <input type="text"/>
Вопрос 2	Ответ 2.1 ▾
Вопрос 3	<input type="checkbox"/> Ответ 2.1 <input type="checkbox"/> Ответ 2.2 <input type="checkbox"/> Ответ 2.3
Вопрос 4	<input type="text"/>
Впрос 5	<input type="radio"/> Ответ 5.1 <input type="radio"/> Ответ 5.2 <input type="radio"/> Ответ 5.3
Вопрос 6	<input type="text"/> Обзор <input type="text"/>
Вопрос 7	<input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/>
Дополнительная информация	<input type="text"/>
<input type="button" value="send"/> <input type="button" value="reset"/> <input type="button" value="Кнопка с рисунком"/>	

### Вариант 2

	Вопрос 1	Вопрос 2	Вопрос 3
	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="button" value="Кнопка рисунка"/>	Вопрос 4	Вопрос 5	Вопрос 6
<input type="button" value="send"/>	Ответ 4.1 <input type="button" value="↓"/>	<input type="checkbox"/> Ответ 5.1 <input type="checkbox"/> Ответ 5.2	<input type="radio"/> Ответ 6.1 <input type="radio"/> Ответ 6.2
<input type="button" value="reset"/>	Вопрос 7		Вопрос 8
	<input type="text"/>	<input type="text"/>	
	<input type="button" value="Обзор"/>	<input type="text"/>	
Дополнительная информация	<input type="text"/>		

### Вариант 3

Вопрос 1	<input type="text"/>	<input type="text"/>	
Вопрос 2	<input type="radio"/> Ответ 2.1 <input type="radio"/> Ответ 2.2		
Вопрос 3	<input type="text"/>	<input type="text"/>	<input type="button" value="send"/>
Вопрос 4	<input type="checkbox"/> Ответ 4.1 <input type="checkbox"/> Ответ 4.2		<input type="button" value="Кнопка рисунка"/>
Вопрос 5	<input type="text"/>	<input type="button" value="Обзор"/>	
Вопрос 6	Ответ 6.1 <input type="button" value="↓"/>		<input type="button" value="reset"/>
Дополнительная информация	<input type="text"/>		

### Вариант 4

	<input type="button" value="send"/>	<input type="button" value="Кнопка рисунка"/>	<input type="button" value="reset"/>
Вопрос 1	<input type="text"/>	<input type="text"/>	
Вопрос 2	<input type="checkbox"/> Ответ 2.1	<input type="checkbox"/> Ответ 2.2	
Вопрос 3	<input type="text"/>	<input type="button" value="Обзор"/>	
Вопрос 4	Ответ 4.1 <input type="button" value="↓"/>		
Вопрос 5	<input type="radio"/> Ответ 5.1	<input type="radio"/> Ответ 5.2	
Вопрос 6	<input type="text"/>	<input type="text"/>	
Дополнительная информация	<input type="text"/>		

### Вариант 5

Вопрос 1		Вопрос 4
<input type="checkbox"/> Ответ 1.1 <input type="checkbox"/> Ответ 1.2 <input type="checkbox"/> Ответ 1.3 <input type="checkbox"/> Ответ 1.4	<input type="button" value="reset"/>	<input type="text"/> <input type="text"/>
Вопрос 2		Вопрос 5
<input type="text" value="Ответ 2.1"/> <input type="text" value="Ответ 2.2"/>	<input type="button" value="Кнопка рисунок"/>	<input type="text" value="Ответ 5.1"/> ▾
Вопрос 3	<input type="button" value="send"/>	Вопрос 6
<input type="text"/> <input type="text"/>		<input type="radio"/> Ответ 6.1 <input type="radio"/> Ответ 6.2
Дополнительная информация		
<input type="text"/>		

### Вариант 6

<input type="button" value="Кнопка рисунок"/>	Вопрос 1	<input type="radio"/> Ответ 1.1	<input type="radio"/> Ответ 1.2	<input type="radio"/> Ответ 1.3
	Вопрос 2	<input type="text" value="Ответ 2.1"/> ▾		
Дополнительная информация <input type="text"/>	Вопрос 3	<input type="text"/>		
	Вопрос 4	<input type="text" value="Ответ 4.1"/> <input type="text" value="Ответ 4.2"/> <input type="text" value="Ответ 4.3"/>		
	Вопрос 5	<input type="text"/>	<input type="text" value="Обзор"/>	
	Вопрос 6	<input type="checkbox"/> Ответ 6.1	<input type="checkbox"/> Ответ 6.2	<input type="checkbox"/> Ответ 6.3
<input type="button" value="send"/>		<input type="button" value="reset"/>		

## СОЗДАНИЕ ФОРМ С ИСПОЛЬЗОВАНИЕМ СТИЛЕЙ

**Цель работы:** приобрести навыки создания форм различных типов с использованием таблиц стилей.

### Методические указания

#### 1. Управляющие элементы

Форма HTML – это раздел документа, в котором содержатся обычная информация, разметка и специальные элементы, называемые *управляющими элементами* (флажки, кнопки, текстовые поля, поле со списком, переключатели и т.д.), а также метки этих управляющих элементов. Обычно пользователи заполняют форму, модифицируя управляющие элементы (вводя текст, выбирая пункты меню и т.д.) перед тем, как предоставить форму пользователя для обработки (например, на Web-сервер, на почтовый сервер и т.д.). Пользователи взаимодействуют с формами при помощи именованных *управляющих элементов*.

При создании однострочного текстового поля для задания параметров поля можно использовать стили, которые позволят изменить ширину текстового поля, цвет фона, тип шрифта и рамки вокруг поля (пример 1).

#### Пример 1. Создание однострочного текстового поля

```
<html>
<head>
<style type="text/css">
    .textfield {
        background: red;           /* Цвет фона под текстом*/
        color: blue;              /* Цвет символов */
        border: 2px solid black;  /* Рамка вокруг поля*/
        width: 50%;              /* Ширина поля в процентах */
        font-family: Arial, sans-serif; /* Тип шрифта*/
        font-size: 90%           /* Размер шрифта*/
    }
</style>
```

```

</head>
<body>
<FORM action="..." method="post">
<b>Введите ваше имя:
</b><br>
<INPUT type="text" name="user" maxlength=25 class=textfield>
<P>
<INPUT type="submit" value="Отправить">
</P>
</FORM>
</body>
</html>

```

В данном примере создается новый класс, который связывается с тегом **INPUT**. Также при помощи атрибутов **font-family** и **font-size** задается тип шрифта, а при помощи свойства **color** цвет символов (рис. 12.1).

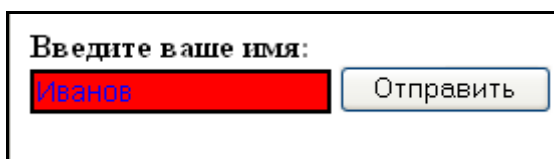


Рис. 12.1. Вид текстового поля в браузере

При оформлении многострочного поля применяются те же стилевые параметры, что и для однострочного текста, а именно можно изменять ширину, высоту, границу, цвет текста и фона (пример 2).

**Пример 2.** Создание многострочного текстового поля

```

<html>
<head>
<style type="text/css">
  TEXTAREA {
    background: red;           /* Цвет фона под текстом*/
    color: blue;              /* Цвет символов */
    padding: 5px;            /* Рамка вокруг поля*/
    border: 1px solid black;  /* Рамка вокруг поля*/
    width: 50%;              /* Ширина поля в процентах */
  }

```

```

</style>
</head>
<body>
<FORM action="..." method="post">
<b>Введите ваш отзыв:
</b><br>
<textarea rows=10>
</textarea>
<p>
<INPUT type="submit" value="Отправить">
</p>
</FORM>
</body>
</html>

```

Добавление рамки к полю через свойство **border** отменяет трехмерные эффекты. Вокруг текста при использовании рамок убираются поля, поэтому требуется их установить с помощью атрибута **padding** (рис. 12.2).

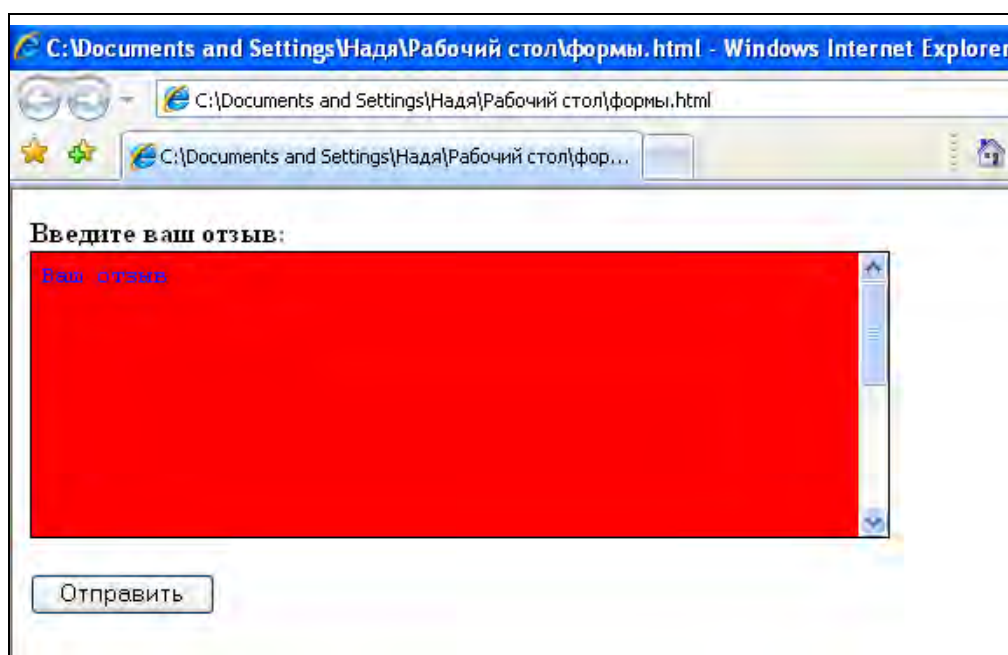


Рис. 12.2. Вид многострочного текстового поля в браузере

Стили позволяют устанавливать прямо в поле для ввода графические изображения с помощью тега **INPUT** в виде фона, используя для этого атрибут **background**.

**Пример 3.** Установка графического изображения в поле для ввода

```
<html>
<head>
<style type="text/css">
  INPUT.enter{
    height: 36 px;          /* Высота поля*/
    width: 200 px;         /* Ширина поля */
    padding-left: 35 px;   /* Отступ слева от края до текста*/
  }
</style>
</head>
<body>
<FORM action="..." method="post">
Логин: <input type="text" class=enter style="background: url (login.gif) no-repeat">
<br>
Пароль: <input type="password" class=enter style="background: url (pass.gif) no-
repeat">
<p>
<input type="submit" value="Отправить">
</p>
</FORM>
</body>
</html>
```

В качестве аргумента параметра **background** необходимо использовать **no-repeat**, тогда рисунок будет отображаться только один раз и не станет повторяться, как это задано для фона по умолчанию. Чтобы не писать текст поверх рисунка, к стилю тега **INPUT** следует добавить атрибут **padding-left**, который обеспечивает набор текста правее рисунка, а его значение зависит от ширины изображения.



При создании кнопок, их вид и цвет можно задавать, воспользовавшись стилями. Для этого к кнопке добавляют атрибут **background**, а также параметры шрифта, в частности цвет текста (пример 4).

*Пример 4.* Создание и вид кнопки (рис. 12.3).

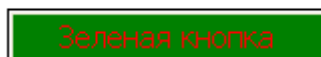


Рис. 12.3. Вид кнопки в браузере

```
<html>
<head>
<style type="text/css">
  .colorButton{
    background: green;           /* Цвет самой кнопки*/
    color: red;                 /* Цвет текста на кнопке*/
    font-family: Tahoma, sans-serif; /* Тип шрифта */
    font-size: 15 px;          /* Размер текста*/
  }
</style>
</head>
<body>
<FORM action="..." method="post">
<input type="button" class="colorButton" value="Зеленая кнопка">
</FORM>
</body>
</html>
```

При добавлении стилей вид кнопки отображается одинаково в разных браузерах. Кроме однотонного цвета на кнопке можно использовать и фоновый рисунок, это позволит сделать заливку кнопки градиентом (пример 5).

**Пример 5.** Фонový рисунок для кнопки (рис. 12.4).



Рис. 12.4. Вид кнопки с фоновым рисунком в браузере

```
<html>
<head>
<body>
<FORM action="..." method="post">
<input type="button" style="background: url(/images/bgbutton.gif)" value="Кнопка с
фоном">
</FORM>
</body>
</html>
```

Используя CSS можно оформлять переключатели. Допустимо устанавливать рамку вокруг каждого переключателя, менять цвет фона под ним, а также изменять расстояние между самими элементами и текстом рядом с ними.

**Пример 6.** Создание переключателей (рис. 12.5).

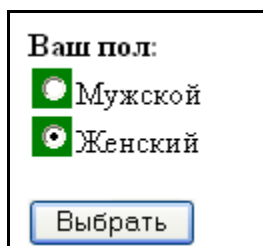


Рис. 12.5. Отображение переключателей в браузере

```
<html>
<head>
<style type="text/css">
.radiobutton{
background: green;           /* Цвет маркера*/
color: red;                 /* Цвет фона под переключателем*/
border: 1 px dashed black; /* Параметры рамки */
```

```

margin: 2px;                /* Отступы вокруг элемента*/
}
</style>
</head>
<body>
<FORM action="..." method="post">
<b>Ваш пол:</b><br>
<input name="Пол" type="radio" class="radiobutton">Мужской<br>
<input name="Пол" type="radio" class="radiobutton">Женский<br>
<p>
<input type="submit" value="Выбрать">
</p>
</FORM>
</body>
</html>

```

С помощью стилей можно изменять вид и положение флажков, задавать цвет фона, параметры рамки и расстояние между флажком и текстом (пример 7).

**Пример 7.** Создание флажков (рис. 12.6).

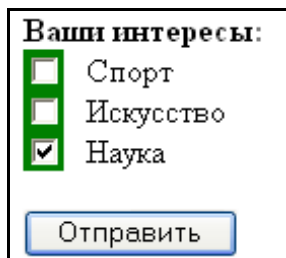


Рис. 12.6. Отображение флажков в браузере

```

<html>
<head>
<style type="text/css">
.col{
background: green;          /* Цвет маркера*/
color: red;                /* Цвет фона под переключателем*/
border: 1 px solid black; /* Параметры рамки */
margin-right: 7px;        /* Отступы вокруг элемента*/
}
</style>
</head>

```

```

<body>
<FORM action="..." method="post">
<b>Ваши интересы:</b><br>
<input class="col" type="checkbox" name="Interests1" value="Sports"> Спорт <br>
<input class="col" type="checkbox" name="Interests2" value="Art"> Искусство
<br>
<input class="col" type="checkbox" name="Interests3" value="Science"> Наука
<br>
<p>
<input type="submit" value="Отправить">
</p>
</FORM>
</body>
</html>

```

Браузеры совершенно по-разному отображают флажки, вид которых управляется через CSS.

## 2. Сложные формы

При создании сложной формы возникает необходимость в разделении логических блоков друг от друга. Этого можно добиться, применяя внутри тега FORM сочетания тегов и стилей. Элементы формы можно выделить, если использовать для них фоновый цвет и рамку, задавая их через CSS или другой подход, который состоит в применении тега FIELDSET. Этот контейнер группирует элементы формы, отображая вокруг них рамку. Чтобы добавить к рамке специальный заголовок, применяется контейнер LEGEND, который должен располагаться в теге FIELDSET. Внутри тега LEGEND допустимо использовать текст и теги форматирования, а также стили (пример 1).

*Пример 1.* Группировка элементов формы (рис. 12.7).

```

<html>
<body>
<FORM action="..." method="post">
<fieldset>
<legend style="font-weight: bold">Интересы</legend>
<input type="checkbox" name="Interests1" value="Sports"> Спорт <br>

```

```

<input type="checkbox" name="Interests2" value="Art"> Искусство <br>
<input type="checkbox" name="Interests3" value="Science"> Наука <br>
</fieldset>
</FORM>
</body>
</html>

```

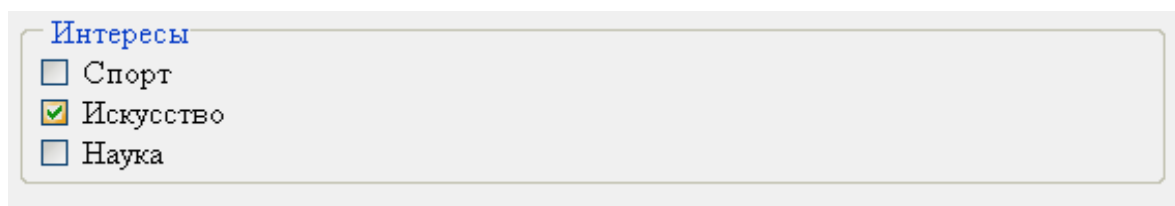


Рис. 12.7. Группировка элементов формы в браузере

С помощью стилей можно также задать отступы вокруг текста и цвет заголовка (пример 2).

**Пример 2.** Задание отступов и цвета заголовка (рис. 12.8).

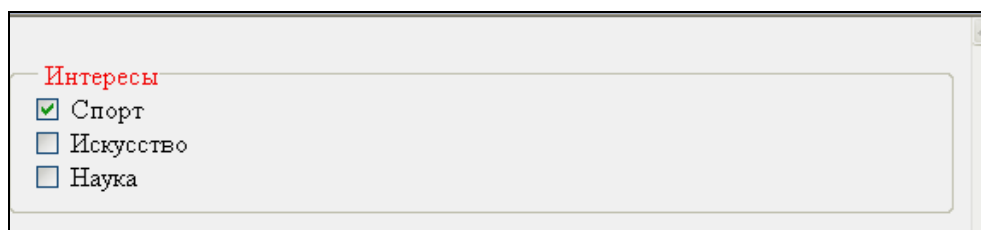


Рис. 12.8. Задание отступов и цвета заголовка

```

<html>
<body>
<FORM action="..." method="post">
<fieldset style="padding: 10 px">
<legend style="color: red">Интересы</legend>
<input type="checkbox" name="Interests1" value="Sports"> Спорт <br>
<input type="checkbox" name="Interests2" value="Art"> Искусство <br>
<input type="checkbox" name="Interests3" value="Science"> Наука <br>
</fieldset>

```

</FORM>

</body>

</html>

### 12.1. Типовое задание

**Пример 1.** Проанализируйте представленный ниже код HTML и определите технологию построения управляющих элементов сложной формы:

```
<html>
<head>
<style type="text/css">
.caption {
margin: 0px 15%;          /* Отступы сверху и сбоку от текста */
padding-bottom: 4px;     /* Поле под текстом */
color: yellow;           /* Цвет символов */
TEXTAREA {
background: white;       /* Цвет фона под текстом*/
color: blue;             /* Цвет символов */
padding: 5px;           /* Рамка вокруг поля*/
border: 1px solid black; /* Рамка вокруг поля*/
width: 50%;             /* Ширина поля в процентах */
.radiobutton{
background: green;      /* Цвет маркера*/
color: red;             /* Цвет фона под переключателем*/
border: 1 px dashed black; /* Параметры рамки */
margin: 2px;           /* Отступы вокруг элемента*/
.col{
background: green;     /* Цвет маркера*/
color: red;            /* Цвет фона под переключателем*/
border: 1 px solid black; /* Параметры рамки */
margin-right: 7px;    /* Отступы вокруг элемента*/
}
</style>
</head>
<body>
<table width=70% border=3 cellpadding=4 cellspacing=0 bgcolor="lime"
align=center>
<p class=caption> ПРИМЕР ФОРМЫ </p>
<FORM action="..." method="post">
<tr>
```

```

<Td><Label for="imyal">Имя: </label></Td>
<Td><Input type="text" id="imyal" maxlength=25 class=textfield></Td>
</tr>

<tr>
<Td><Label for="familiyal"> Фамилия: </label></Td>
<Td><Input type="text" id="familiyal" maxlength=25 class=textfield></Td>
</tr>
<Tr>
<Td><Label for="e-mail 1">e-mail: </label></Td>
<Td><Input type="text" id="e-mail" maxlength=25 class=textfield></Td>
</tr>
<Tr>
<Td>Страна проживания:</Td>
<Td> <Select>
<Option>Англия </option>
<Option>Беларусь </option>
<Option>Германия </option >
<Option> Франция </option> </select></Td>
</tr>
<Tr>
<Td>Научный работник:</Td>
<Td>
<p><Input class="col" type="checkbox" name="F001" гa1ие="Да">Да
<Input class="col" type="checkbox" name="F001" value="Нем">Нем</Td>
</tr>
<Tr>
<Td>пол:</Td>
<Td>&nbsp; &nbsp; &nbsp;
<Input class="radiobutton" type="radio" name="poll" value="Male"> М&nbsp; &nbsp; &nbsp;
<Input class="radiobutton" type="radio" name="poll" value="Female"> Ж</Td>
</tr>
<Tr>
<Td>Дополнительная информация:</Td>
<Td><Textarea rows=6 cols=40>
</textarea></Td>
</tr>
<Tr>
<Td align="bottom">
<Input align="bottom" type="submit" value="send">
<Input type="reset" value="reset"></Td></tr>

```

```
</form>  
</table>  
</body>  
</html>
```

### ***12.2. Контрольные вопросы***

1. Понятие и виды управляющих элементов.
2. Использование графических изображений в управляющих элементах.
3. Создание и форматирование кнопок с помощью стилей.
4. Опишите технологию построения управляющих элементов сложной формы.

### ***12.3. Контрольные задания***

Создайте формы в соответствии с вариантами из лабораторной работы № 11.



## ОРГАНИЗАЦИЯ ФРЕЙМОВ

**Цель работы:** приобрести опыт создания фреймовой структуры документа.

### Методические указания

Технология фреймов позволяет разделить Web-страницы на отдельные области.

*Фрейм* – область гипертекстового документа со своими полосами прокрутки.

Для создания страниц с фреймами необходимо обязательное построение двух групп HTML-файлов. К первой группе относятся документы содержания (*content*), которые обеспечивают заполнение информацией каждой области страницы. Ко второй группе относят документы-раскладки (*layout*), которые применяются для разделения Web-страницы на области.

Разделение Web-страницы на области (фреймы) определенных типов осуществляется в документах-раскладках с помощью тегов `<Frameset >` `</frameset >` и внутреннего тега `<Frame>`.

В теге `<Frameset >` за горизонтальное деление экрана отвечает атрибут *rows*, а за вертикальное деление - атрибут *cols*. Значение этих атрибутов выражается в пикселах, процентах или с помощью \* для обозначения оставшейся части экрана.

**Пример 1.** Разделить область просмотра по вертикали в отношении два к трем.

`cols = 40%, 60%`

**Пример 2.** Разделить область просмотра, в которой верхняя горизонтальная область имеет размер 100 пикселей, 40% доступного пространства под среднюю область, а все остальное – под нижнюю.

`rows = 100, 40%, *`

Внутри тега `<Frameset>` `</frameset>` устанавливается тег `<Frame>`,

который предназначен для организации ссылок на документы содержания.

Тег `<Frame>` имеет собственные атрибуты:

`src = "имя файла.htm"` – для вызова документа содержания;

`scrolling = "yes"` – обеспечивает прокрутку внутри области;

`scrolling = "no"` – запрещает прокрутку внутри области;

`scrolling = "Auto"` – разрешает построение полос прокрутки при необходимости.

Тег `<Noframes > </noframes >` применяется для предусмотрения ситуации, когда браузер не поддерживает фреймы.

### 13.1 Типовое задание

**Пример 1.** Разместите на Web-странице пять областей (фреймов), сопроводив центральный фрейм горизонтальной и вертикальной полосами прокрутки, как показано на рис. 13.1.



Рис. 13.1. Области (фреймы), размещенные на Web-странице

Для выполнения данного задания сначала необходимо описать пять отдельных документов содержания для каждой из областей (фреймов).

Документ содержания верхнего фрейма:

`<Html>`

`<Head>`

`< Title >Верхний фрейм </ Title >`

```

<Meta http-equiv="Content-Type" content="text/html; charset—windows-
1251">
</head>
<Body bgcolor="pink">
<Center><Font size=5 color="blue">Верхняя часть страницы</font>
</center>
</body>
</html>

```

Документ содержания нижнего фрейма (bottomfr.htm):

```

<Html>
<Head>
< Title >Нижний фрейм </ Title >
<Meta http-equiv="Content-Type" content="text/html; charset—windows-
1251">
</head>
<Body bgcolor=" yellow ">
<Center><Font size=5 color=" indigo">Верхняя часть страницы</font>
</center>
</body>
</html>

```

Документ содержания левого фрейма (left.htm):

```

<Html>
<Head>
< Title >Левый фрейм </ Title >
<Meta http-equiv="Content-Type" content="text/html; charset—windows-
1251">
</head>
<Body bgcolor=" white ">
<Center><Font size=5 color=" blue ">Верхняя часть страницы</font>
</center>
</body>
</html>

```

Документ содержания центрального фрейма (centerfr.htm):

```

<Html>
<Head>
< Title > Фрейм </ Title >
<Meta http-equiv="Content-Type" content="text/html; charset—windows-
1251">
</head>
<Body bgcolor=" lime ">
<Center><Font size=4 color=" blue ">Центральный фрейм </font>

```

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

Документ содержания правого фрейма (rightfr.htm):

```
<Html>
<Head>
< Title > Правый фрейм </ Title >
<Meta http-equiv="Content-Type" content="text/html; charset—windows-
1251">
</head>
<Body bgcolor=" white ">
<Center><Font size=4 color=" blue ">Правый фрейм </font>
</center>
</body>
</html>
```

После построения документов содержания строится документ – раскладка:

```
<Html>
<Head>
< Title > Организация фреймов </ Title >
<Meta http-equiv="Content-Type" content="text/html; charset—windows-
1251">
</head>
<! горизонтальное деление экрана>
<Frameset rows= "40%,20%, *">
<! вставка документа "topfr.htm" в верхний фрейм>
<Frame src="topfr. htm "noresize scrolling= "no ">
<! вертикальное деление центральной области экрана>
<Frameset cols = "40%, 20%, 40% ">
<! вставка документа "leftfr.htm" в левый фрейм>
<Frame src="leftfr. htm " scrolling= "no ">
<! вставка документа "centerfr.htm" в центральный фрейм>
<Frame src= "centerfr.htm" scrolling="yes" marginwidth="50" margin-
height="15">
<! вставка документа "rightfr.htm" в правый фрейм>
< Frame src="rightfr. htm " scrolling="no " >
</frameset>
<!вставка документа "bottomfr.hlm" в нижний фрейм>
<Frame src= "bottomfr. htm ">
</frameset>
<Noframes>
<Center<Hr color="blue">r><Font size=6">Фреймы </font></center>
```

<Hr color="blue">

Этот браузер не может воспроизводить фреймы

</noframes>

</frameset>

</html>

### 13.2. Контрольные вопросы

1. Охарактеризуйте технологию создания фреймов.
2. С помощью каких тегов осуществляется разделение Web-страницы на области.
3. Назначение атрибутов тега *Frameset*.
4. Как организовать ссылки на документы содержания.

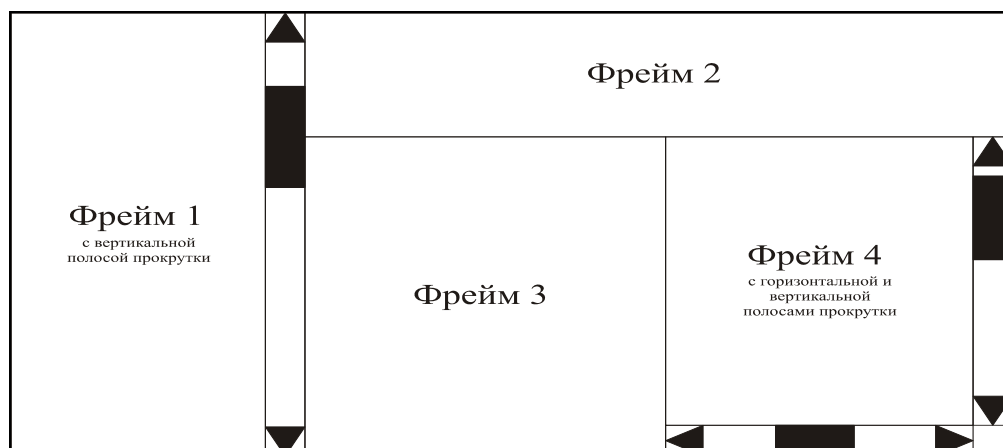
### 13.3. Контрольные задания

Разместите на Web-странице фреймы следующих видов.

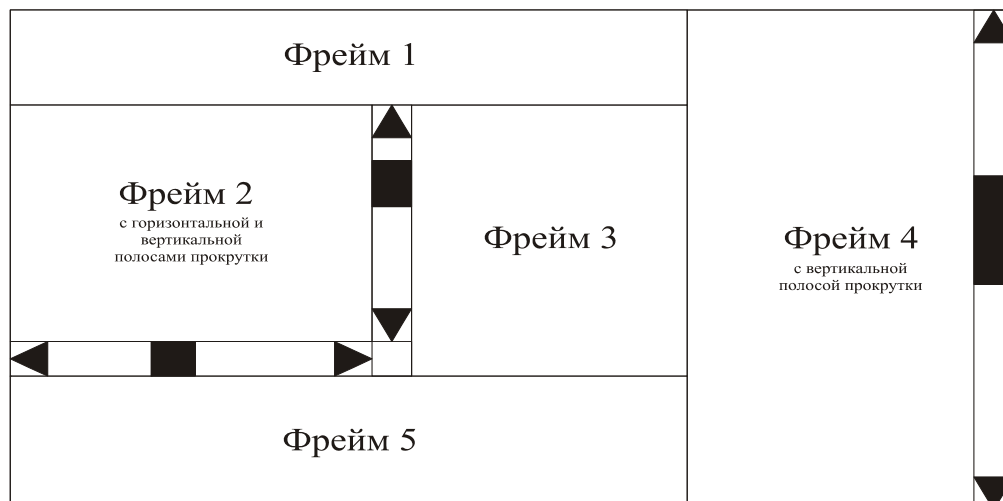
#### Вариант 1



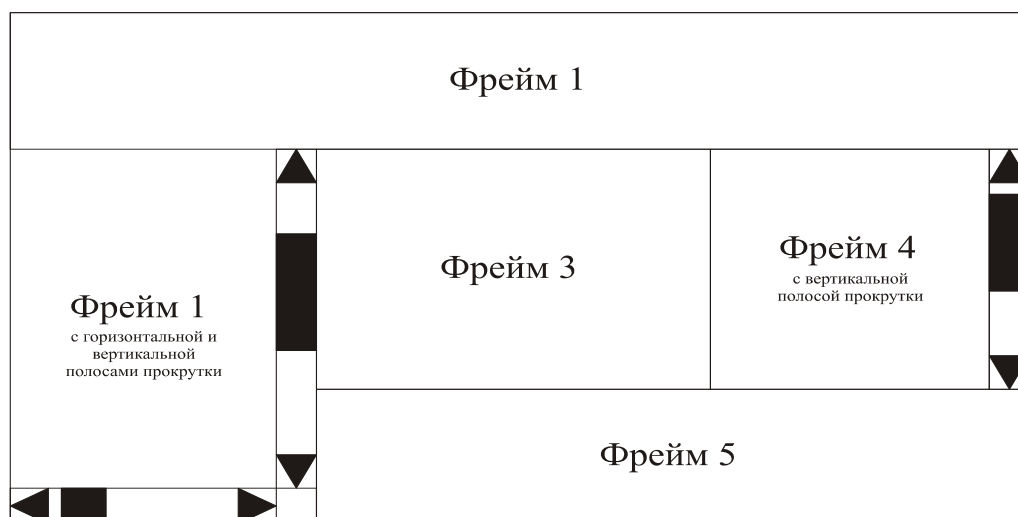
#### Вариант 2



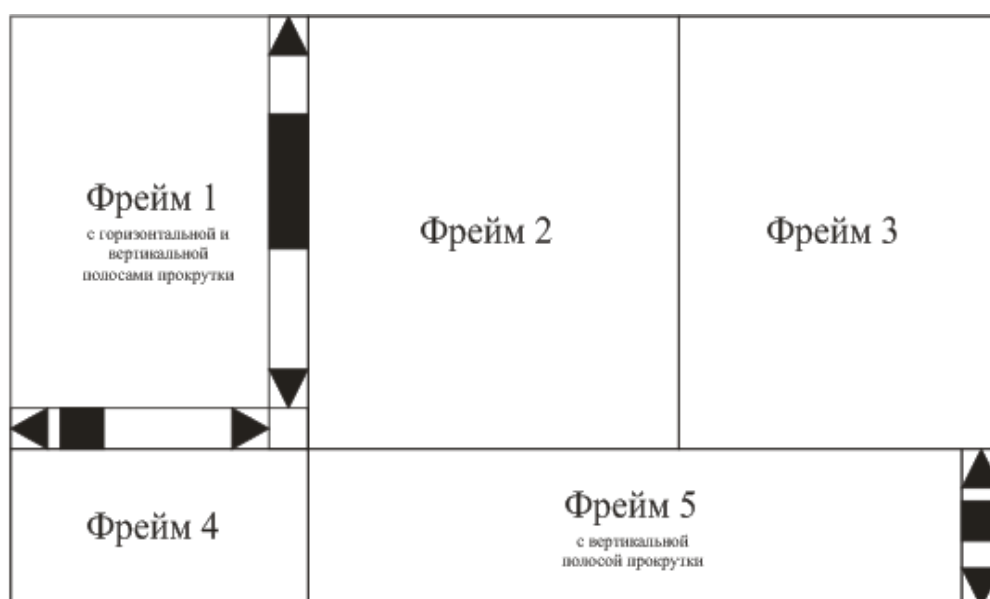
### Вариант 3



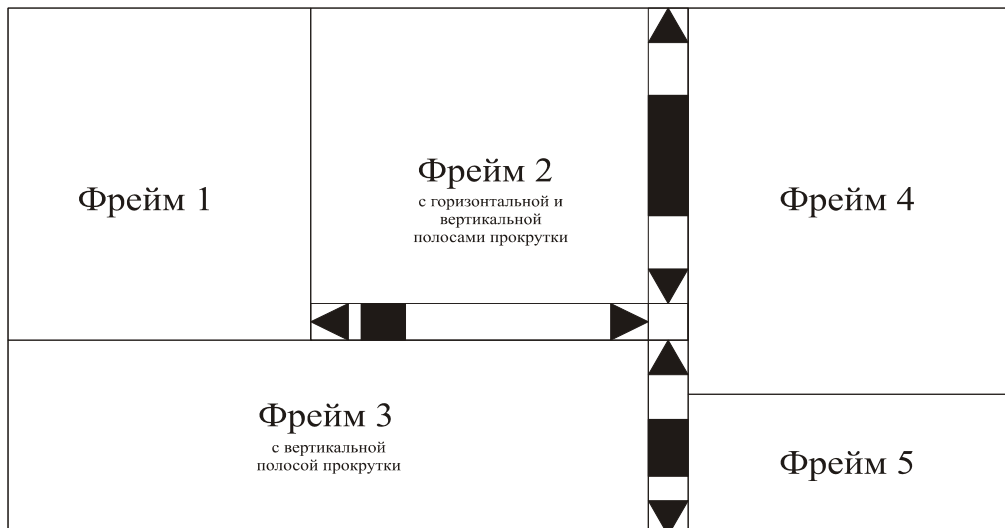
### Вариант 4



### Вариант 5



**Вариант 6**



## **СОЗДАНИЕ МАКЕТА САЙТА С ФРЕЙМОФНОЙ СТРУКТУРОЙ В РЕДАКТОРЕ MICROSOFT FRONTPAGE**

**Цель работы:** освоить основные приемы создания и информационного наполнения веб-страницы средствами HTML-редактора, создание макета сайта с фреймовой структурой главной страницы, вставки, настройки динамической кнопки для удобной навигации по сайту, изучить принципы макетирования сайта в среде Microsoft FrontPage.

### *Методические указания*

#### **1. Основные элементы интерфейса FrontPage**

Редактор Microsoft FrontPage относится к программным средствам визуального веб-конструирования. При работе с веб-редактором FrontPage можно обойтись без знания языка разметки гипертекстовых документов HTML. Веб-страница просто конструируется на экране и сохраняется в формате .html. Выполняемые при этом действия напоминают работу в текстовом редакторе Word.

Программа FrontPage включена в комплекс приложений Microsoft Office XP. Запуск редактора FrontPage осуществляется с помощью команды Пуск, Программы, Microsoft FrontPage.

Интерфейс программы представлен меню, основным окном документа (окно редактора), строкой состояния, основными панелями (*Стандартная*, *Форматирование*) и вспомогательными панелями инструментов (*Представления*, *Создание веб-страницы или узла*, *Список папок* и т. д.). Панели инструментов настраиваются так же, как и в других программах пакета MS Office .

После запуска FrontPage открывается окно, основные элементы которого представлены на рис. 14.1.



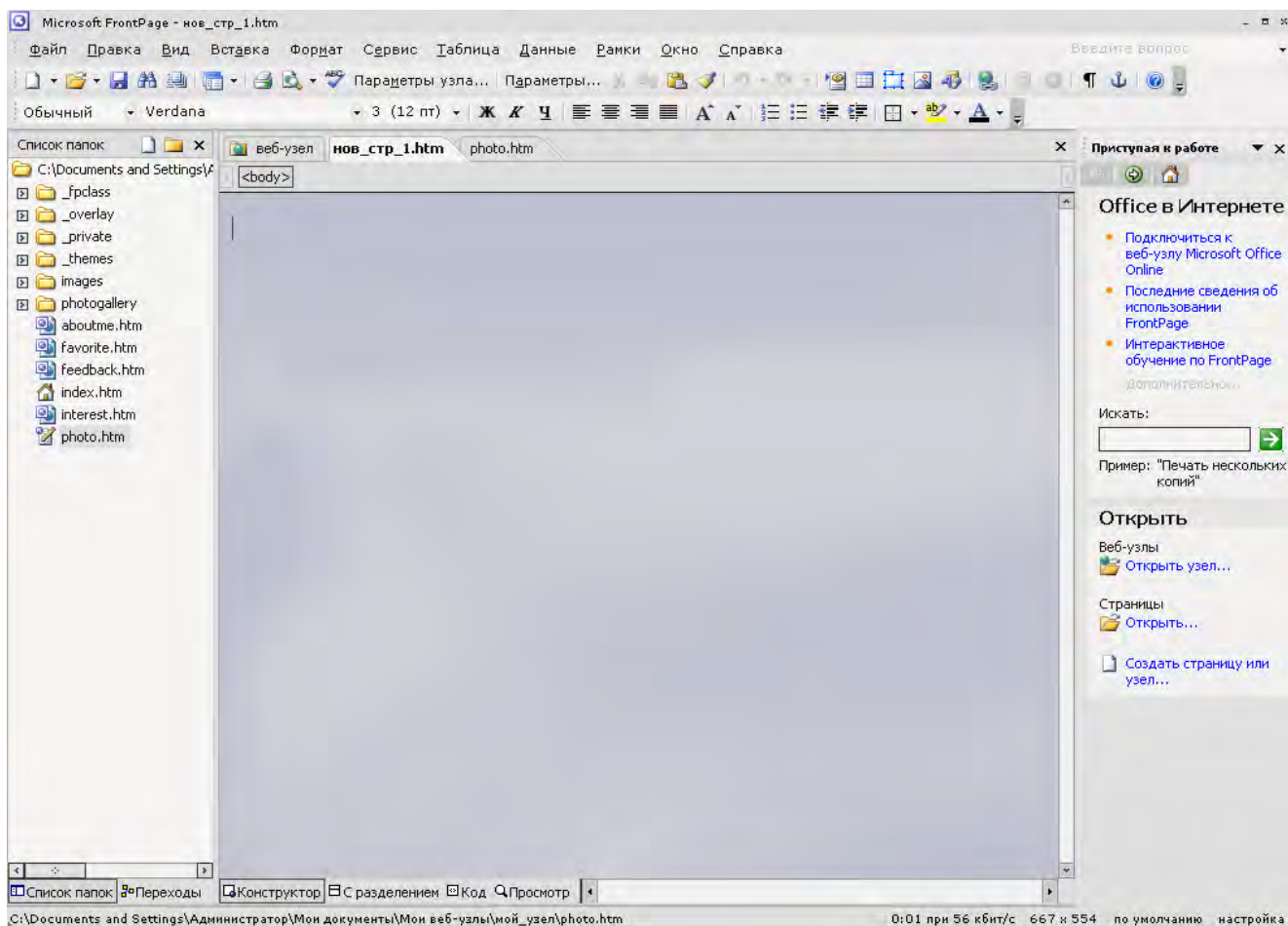


Рис. 14.1. Окно программы FrontPage

*Панели Стандартная, Форматирование и Рисование* по своим возможностям напоминают аналогичные панели текстового редактора Word.

Вид *Рабочей области* зависит от выбранного режима работы. В режиме *Конструктор* веб-страница конструируется из текстовых блоков и графических объектов. При этом автоматически генерируется ее HTML-код, который можно просматривать и редактировать в режиме *Код*. Режим *С разделением* является комбинацией этих двух режимов. Режим *Просмотр* позволяет просматривать созданные страницы. С помощью меню *Вид* можно выбрать и другие полезные при конструировании сайта режимы.

Режим *Страница* предназначен для создания и редактирования веб-страниц (рис. 14.2).

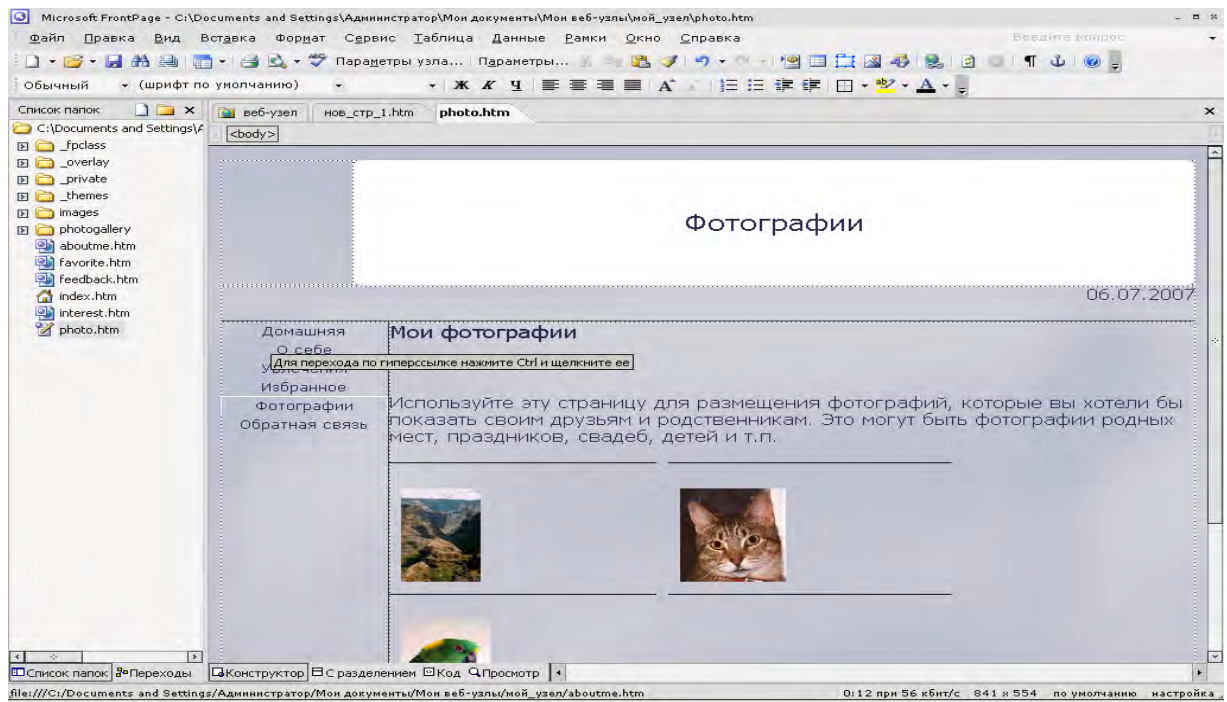


Рис. 14.2. Вид окна FrontPage в режиме редактирования страниц сайта

В режиме *Папки* просматривается структура папок сайта, выполняются файловые операции. Любые изменения файлов и папок сайта необходимо производить из среды разработки, так как это позволяет отследить изменения в структуре взаимосвязанных веб-страниц сайта (рис. 14.3).

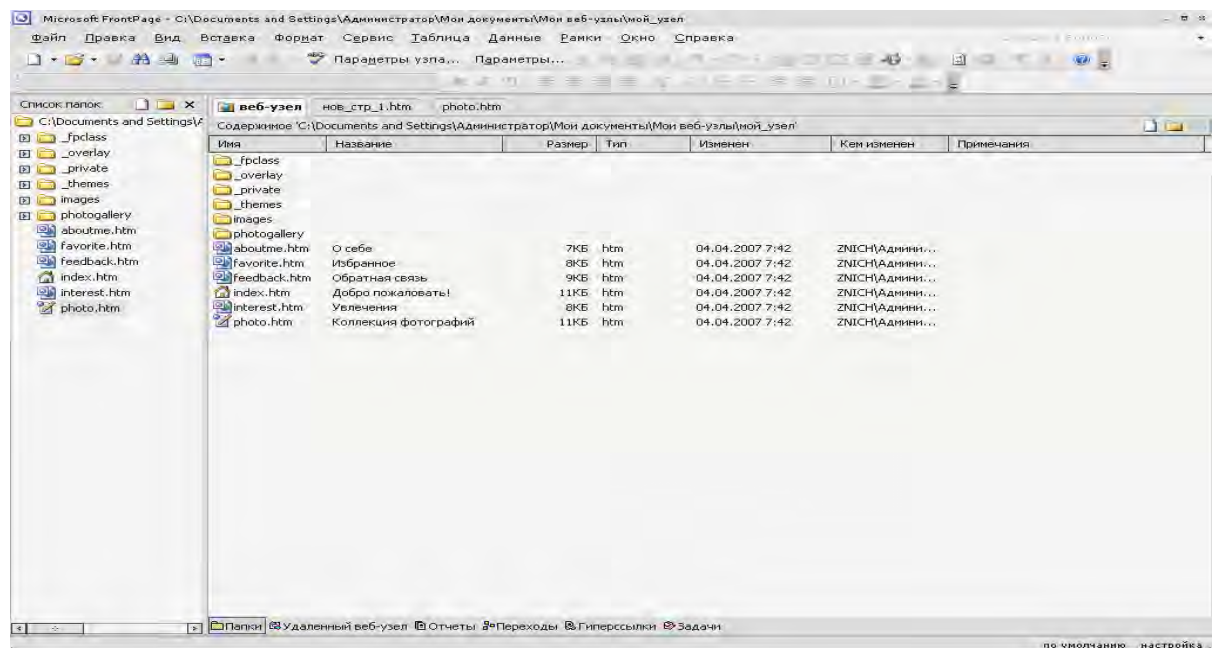


Рис. 14.3. Режим просмотра файлов и папок сайта

Режим *Переходы* отображает структуру связей между страницами и позволяет ее изменять (рис. 14.4).

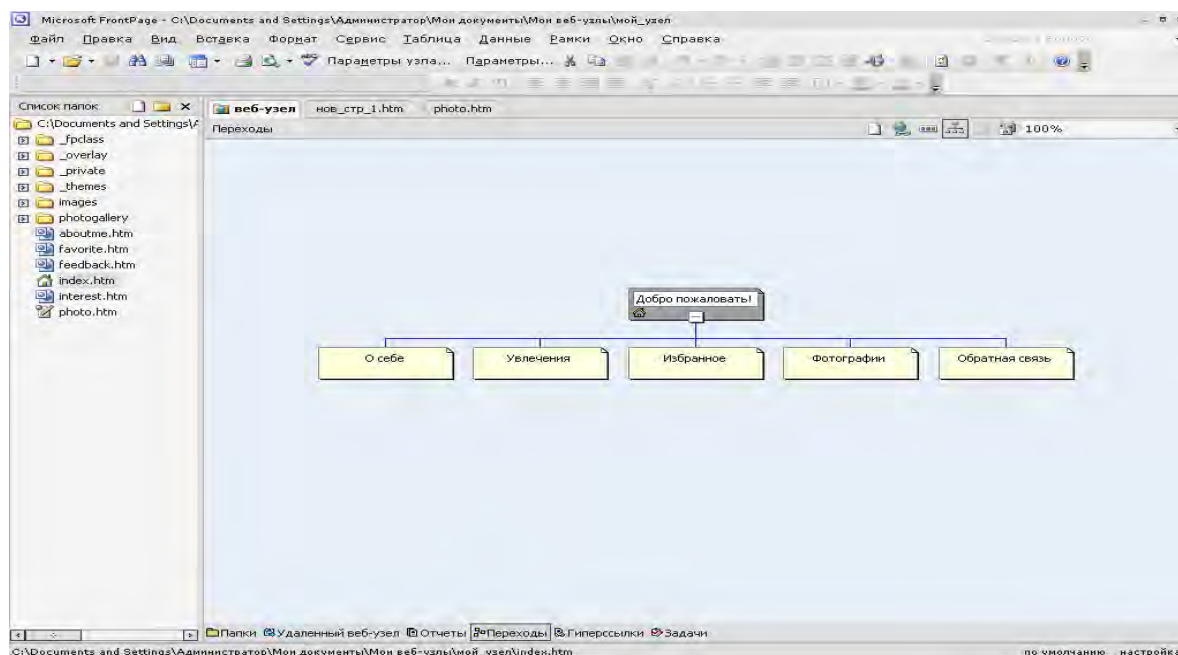


Рис.14.4. Режим отображения навигационной структуры сайта

Режим *Гиперссылки* наглядно отображает систему ссылок, обеспечивает их проверку и редактирование (рис. 14.5).

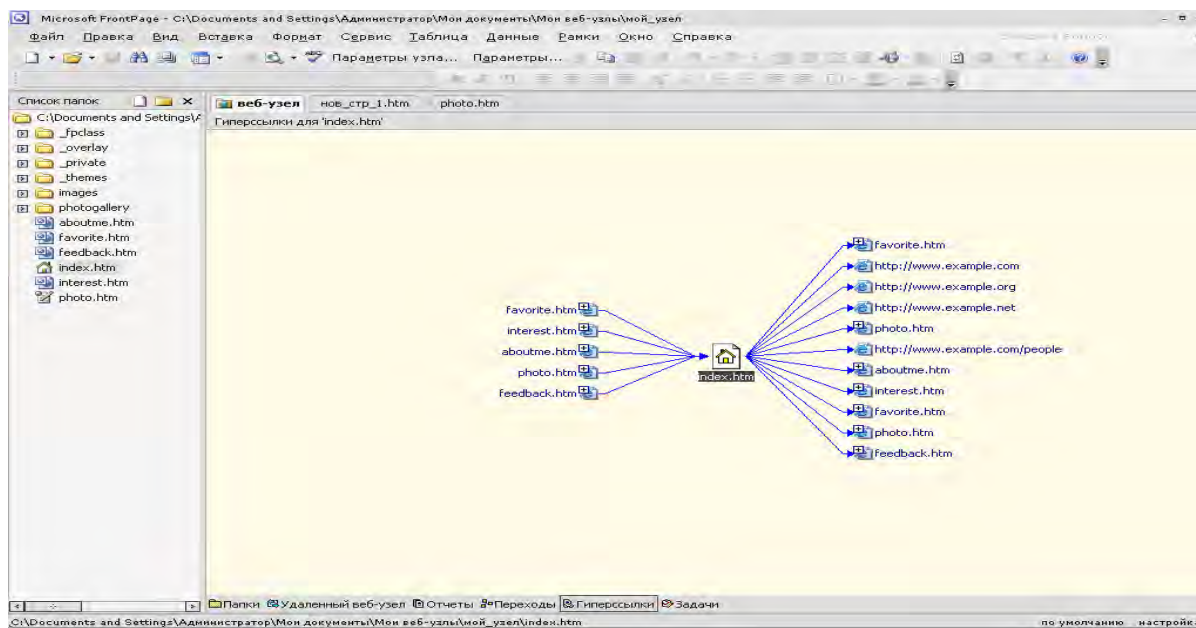


Рис. 14.5. Отображение взаимосвязей страниц сайта в режиме *Гиперссылки*

*Отчеты* — режим просмотра отчетов (рис. 4) с подробной информацией по каждому из них (рис. 14.6).

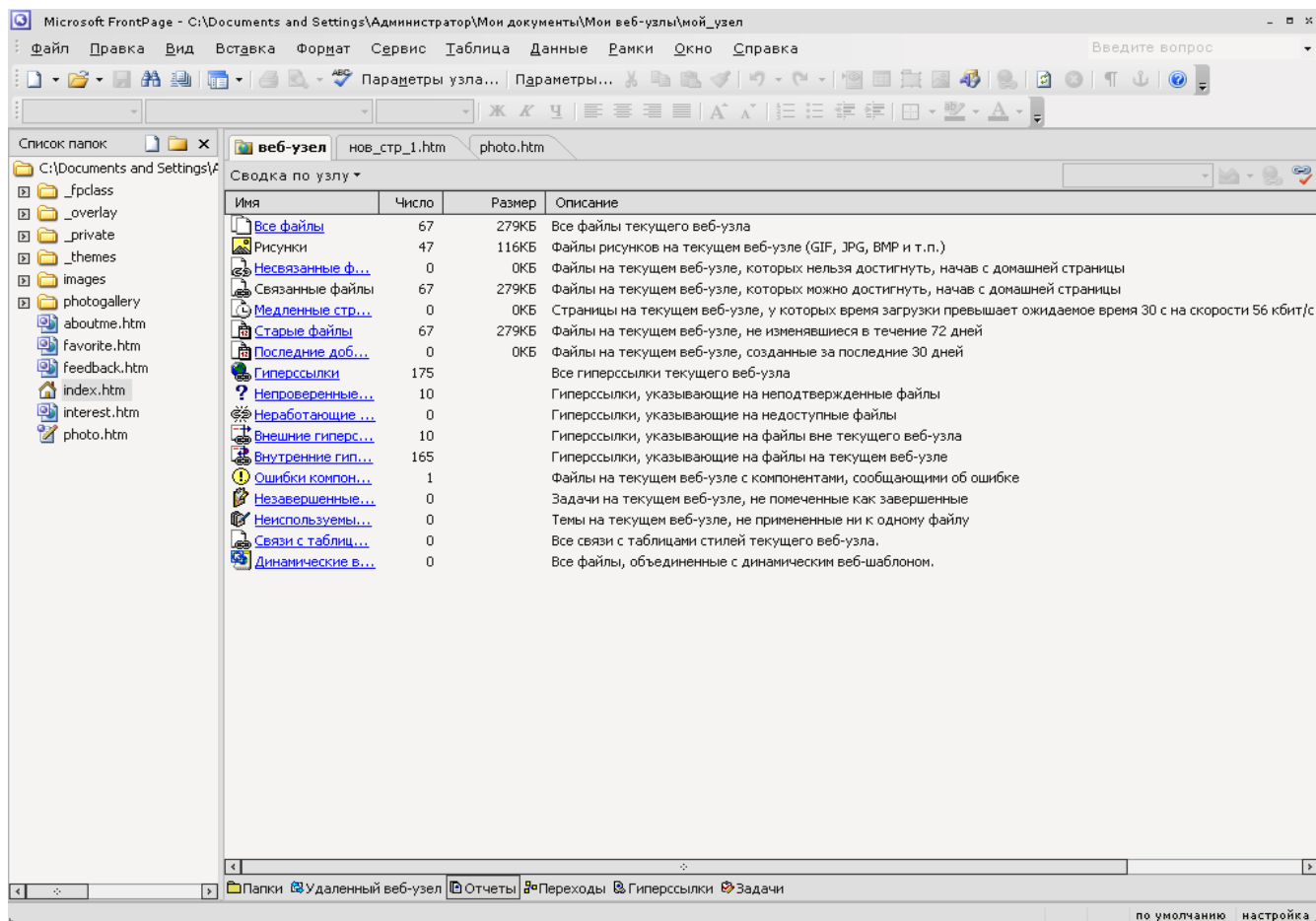


Рис. 14.6. Режим Отчеты

*Задачи* — режим отображения задач сайта. Фактически это записная книжка, особенно полезная при коллективной работе над сайтом. В этом режиме можно ввести описание задачи, которую в последующем необходимо

*Просмотр* — режим предварительного просмотра веб-страницы. Окончательный просмотр веб-страницы следует производить браузером (рис. 14.7).

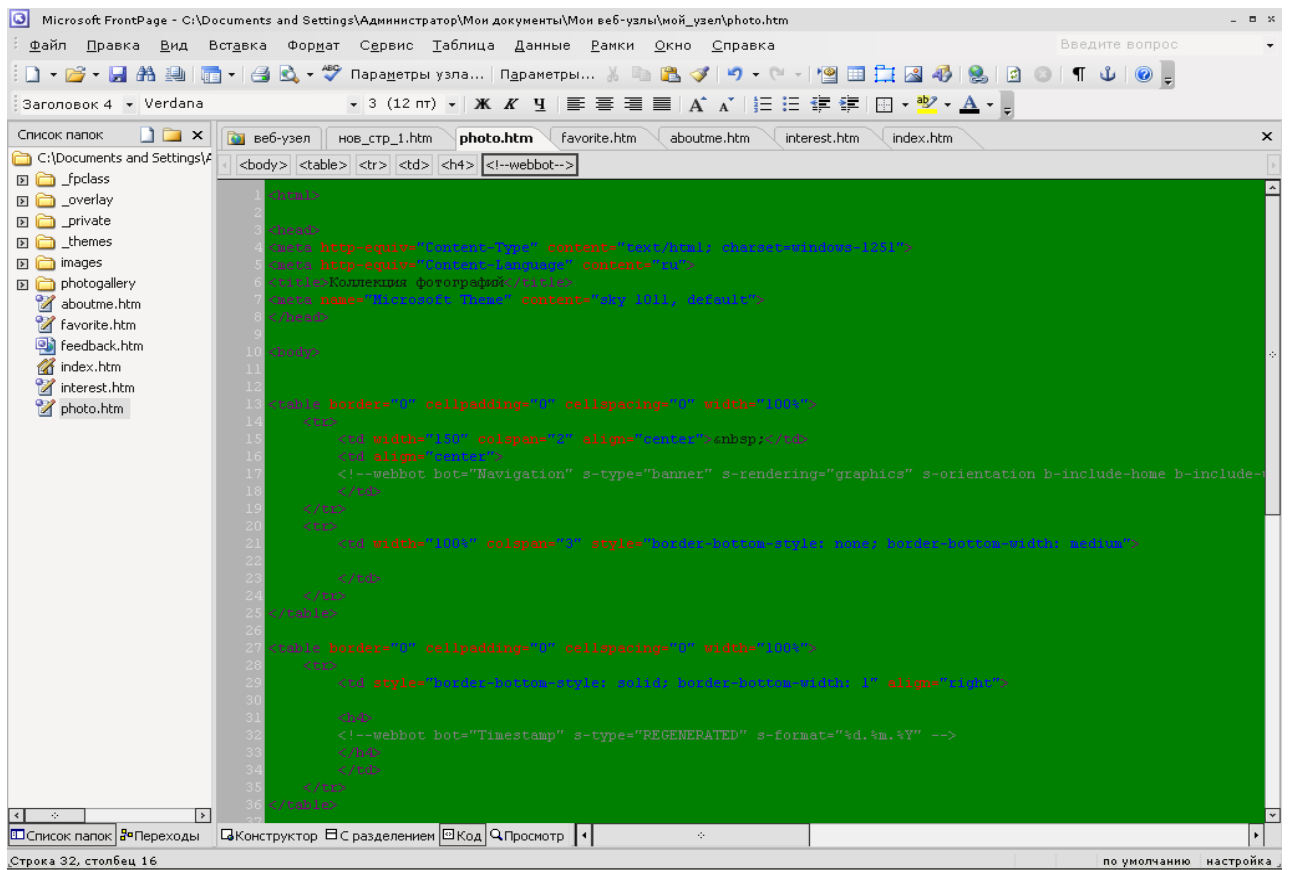


Рис. 14.7. Режим просмотра кода HTML-страницы

Важным параметром создаваемого веб-узла является правильная настройка его кодировки, которая осуществляется посредством команды *Настройка веб-узла* меню *Сервис*. Для русскоязычных сайтов следует установить язык сообщений сервера — русский; применяемый по умолчанию набор знаков для страниц — кириллица (Windows-1251). Эти установки будут применены к каждому создаваемому файлу веб-узла.

### 14.1. Типовое задание

**Пример 1.** Создайте макет сайта с фреймовой структурой главной страницы, представленной на рис. 14.8. Разработайте удобную навигацию по страницам сайта.

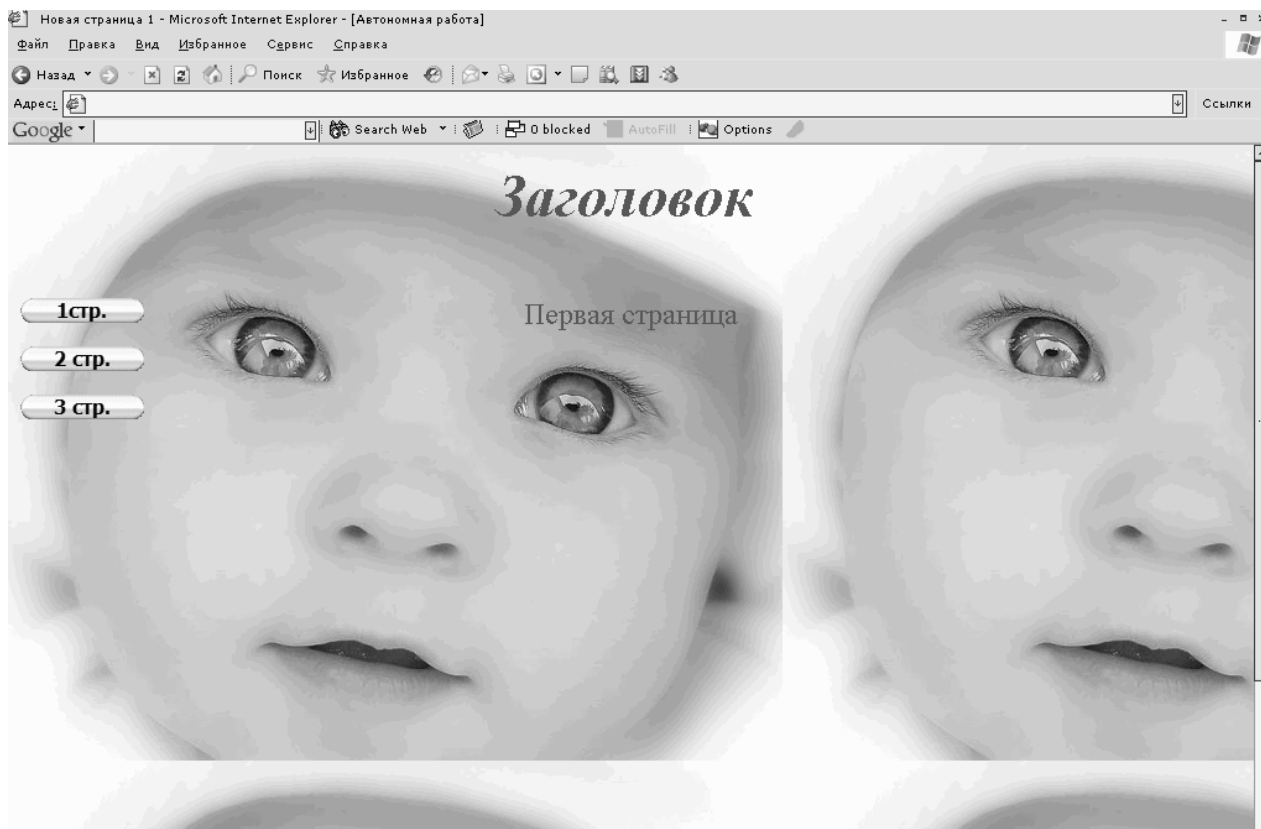


Рис. 14.8. Образец главной страницы сайта

### ***Порядок выполнения***

1. Создайте папку сайта с именем *site\_1*.
2. Откройте редактор Microsoft FrontPage.
3. Создайте сайт на основе папки *site\_1*:
  - a) выполните команду меню *Файл, Создать, Страница* или *Веб-узел*;
  - b) в панели *Создание веб-страницы* или узла в разделе *Создание* выберите *Пустой веб-узел* (рис. 14.9);

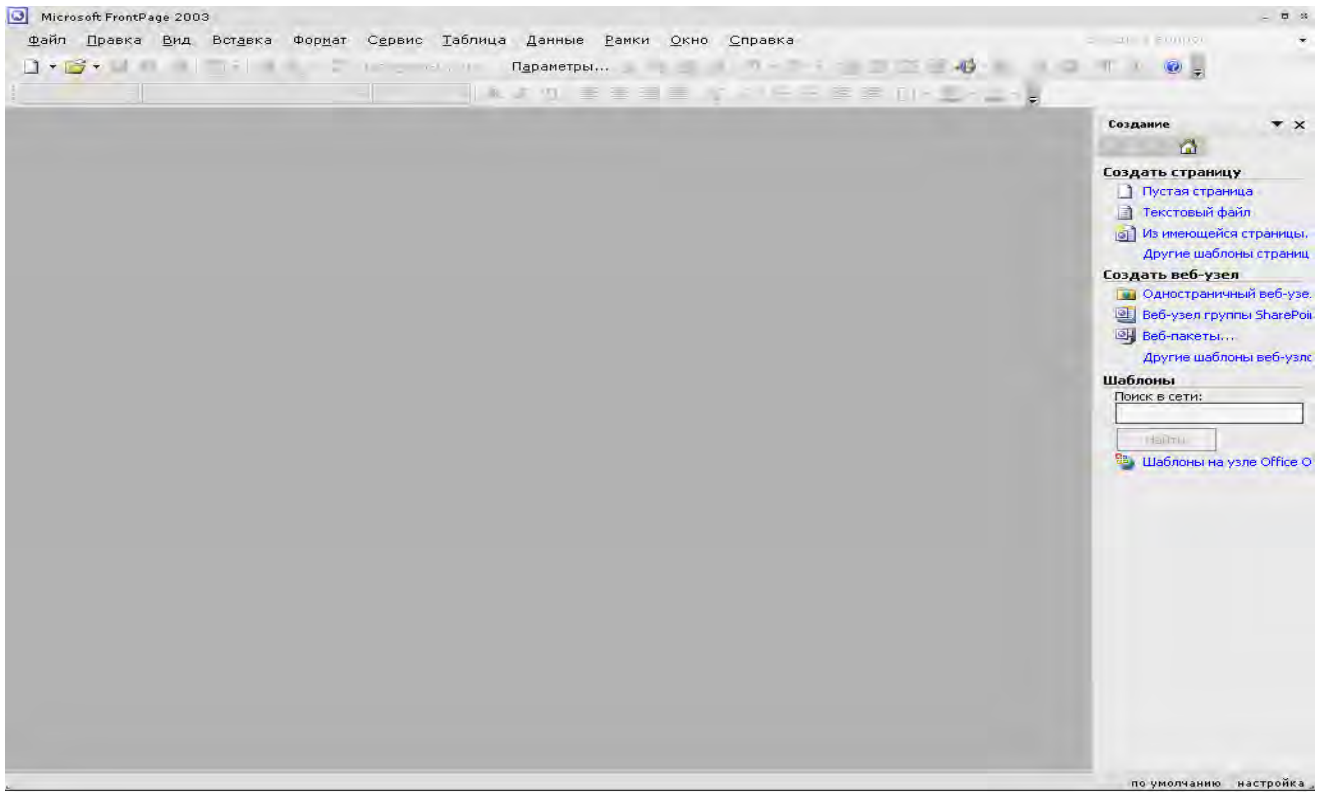


Рис. 14.9. Окно Редактора FrontPage

с) в окне *Шаблоны веб-узлов* в разделе *Параметры* укажите расположение нового веб-узла (используя экранную кнопку *Обзор*, выделите папку *site\_l*) (рис. 14.10. и 14.11.);



Рис. 14.10. Вид окна *Шаблоны Web-узлов*

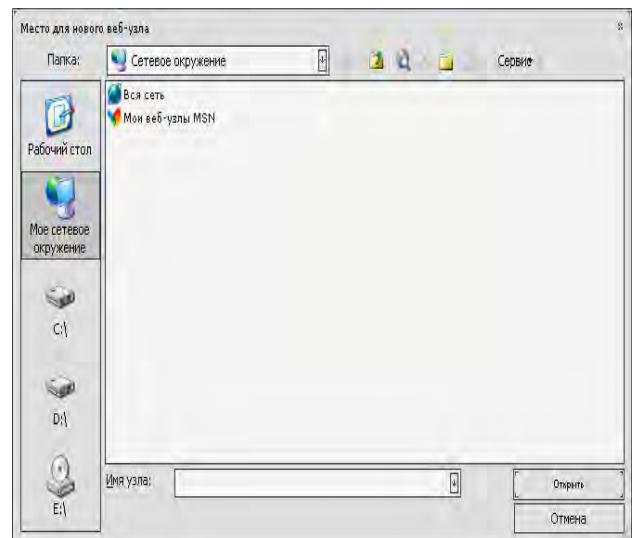


Рис. 14.11. Выбор папки для создания *Web-узла*

d) щелкните на кнопках *Открыть*, ОК.

4. Выполните настройку кодировки языка сайта: меню *Сервис/Настройка веб-узла*. В окне *Настройка веб-узла* (рис. 14.12) перейдите на вкладку *Язык* и выберите следующие установки:

a) язык сообщений сервера: *русский*;

b) применяемый по умолчанию набор знаков для страниц: *кириллица*;

c) Игнорировать раскладку яри выборе кодировки новых страниц.

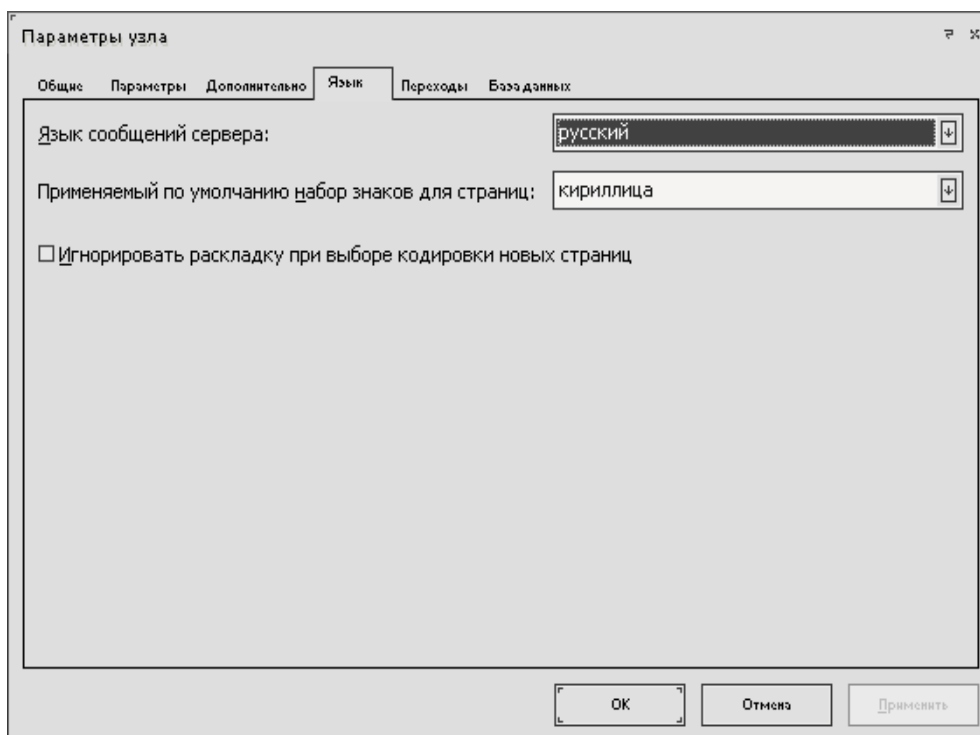


Рис. 14.12. Настройка кодировки языка сайта в окне *Настройка веб-узла*

5. Создайте страницы сайта с именами: *zagolov.htm*, *menu.htm*, *1.htm*, *2.htm*, *3.htm*. (Информационное наполнение страниц в данной лабораторной работе опускается и оставляется на последующую доработку созданного макета сайта.) С этой целью выполните следующие операции:

a) создайте новую страницу, щелкнув на кнопке *Новая страница*;

b) щелкнув правой кнопкой в области поля страницы, выберите в контекстном меню команду *Свойства страницы* (рис. 14.13);



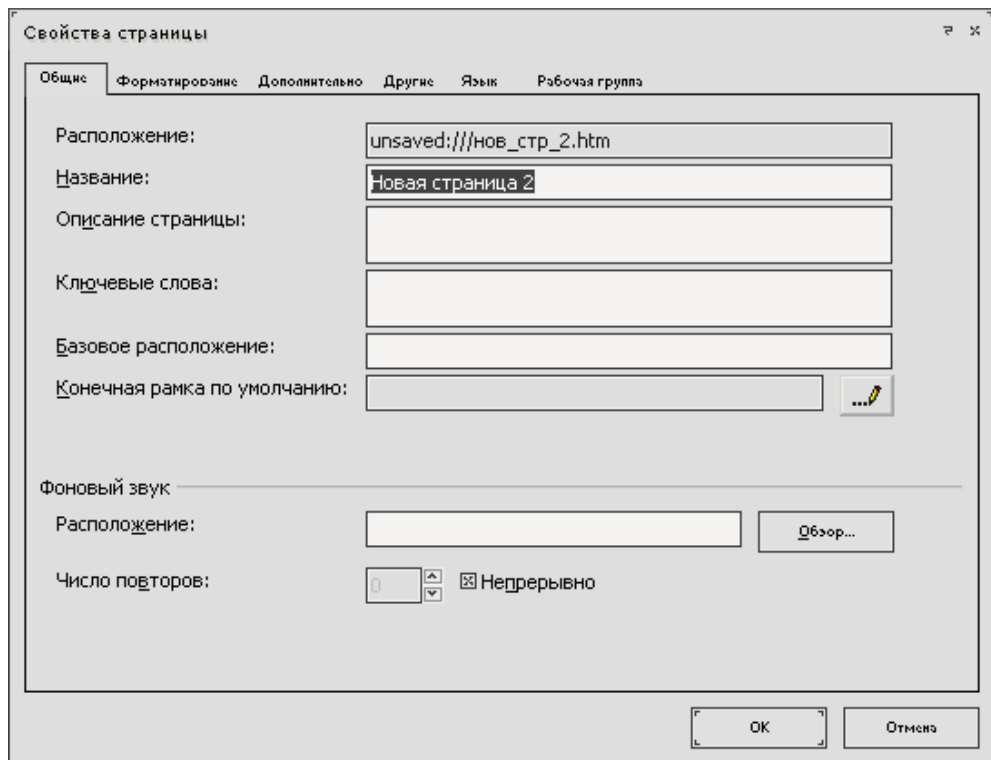


Рис. 14.13. Задание параметров страницы в окне Свойства страницы

с) в окне *Свойства* страницы на вкладке *Общие* введите текст заголовка страницы, перейдите на вкладку *Язык* и выберите следующие установки (рис. 14.14);

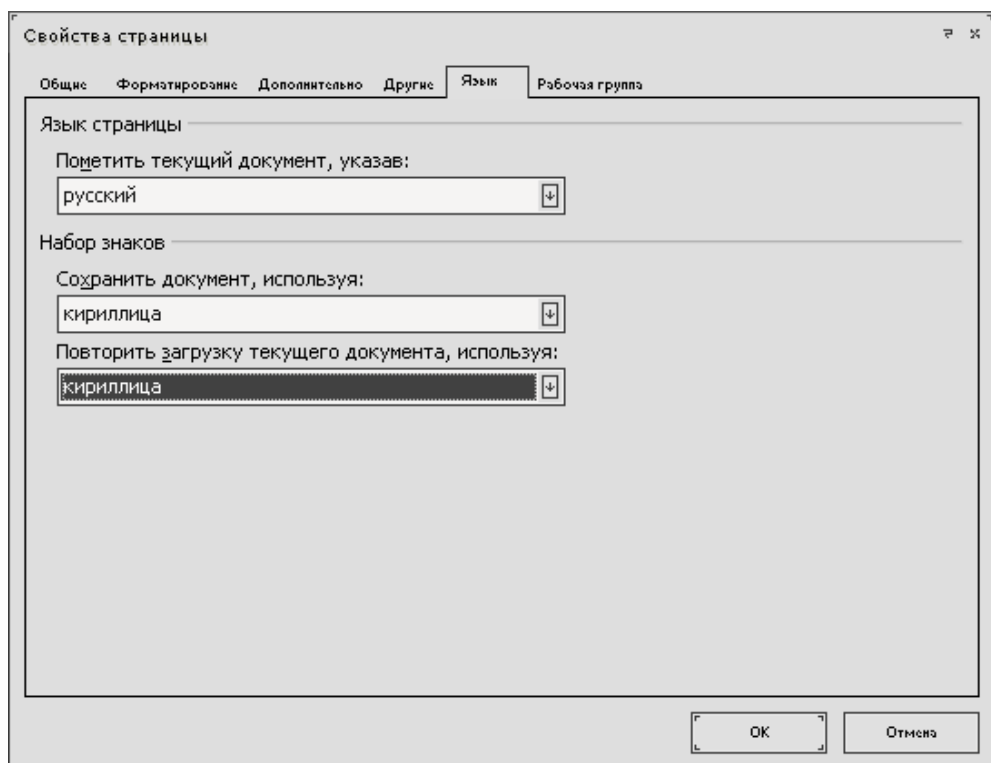


Рис. 14.14. Установка языка страницы и набора знаков

- d) пометьте текущий документ, указав: *русский*;
- e) сохраните документ, используя: *кириллица*;
- f) повторите загрузку текущего документа, используя *кириллица*;
- g) примените к страницам *zagolov.htm*, *menu.htm* тему для графического оформления: меню *Формат, Тема, Ветер* (рис. 14.15)

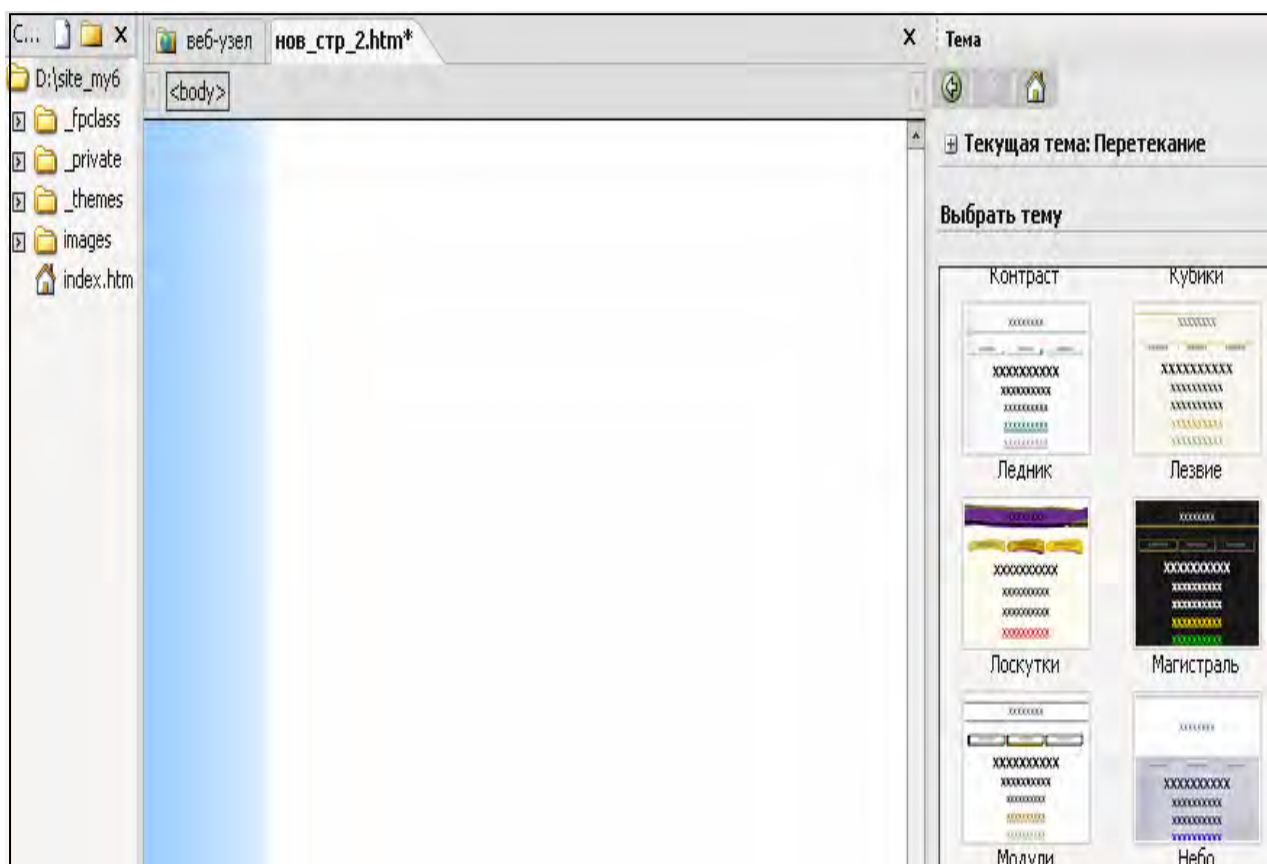


Рис. 14.15. Выбор темы страницы

- h) введите в область страницы соответствующий текстовый заголовок;
- i) сохраните страницу в папке сайта: меню *Файл, Сохранить как*, введите имя страницы;
- j) аналогично оформите и остальные веб-страницы (рис. 14.16).

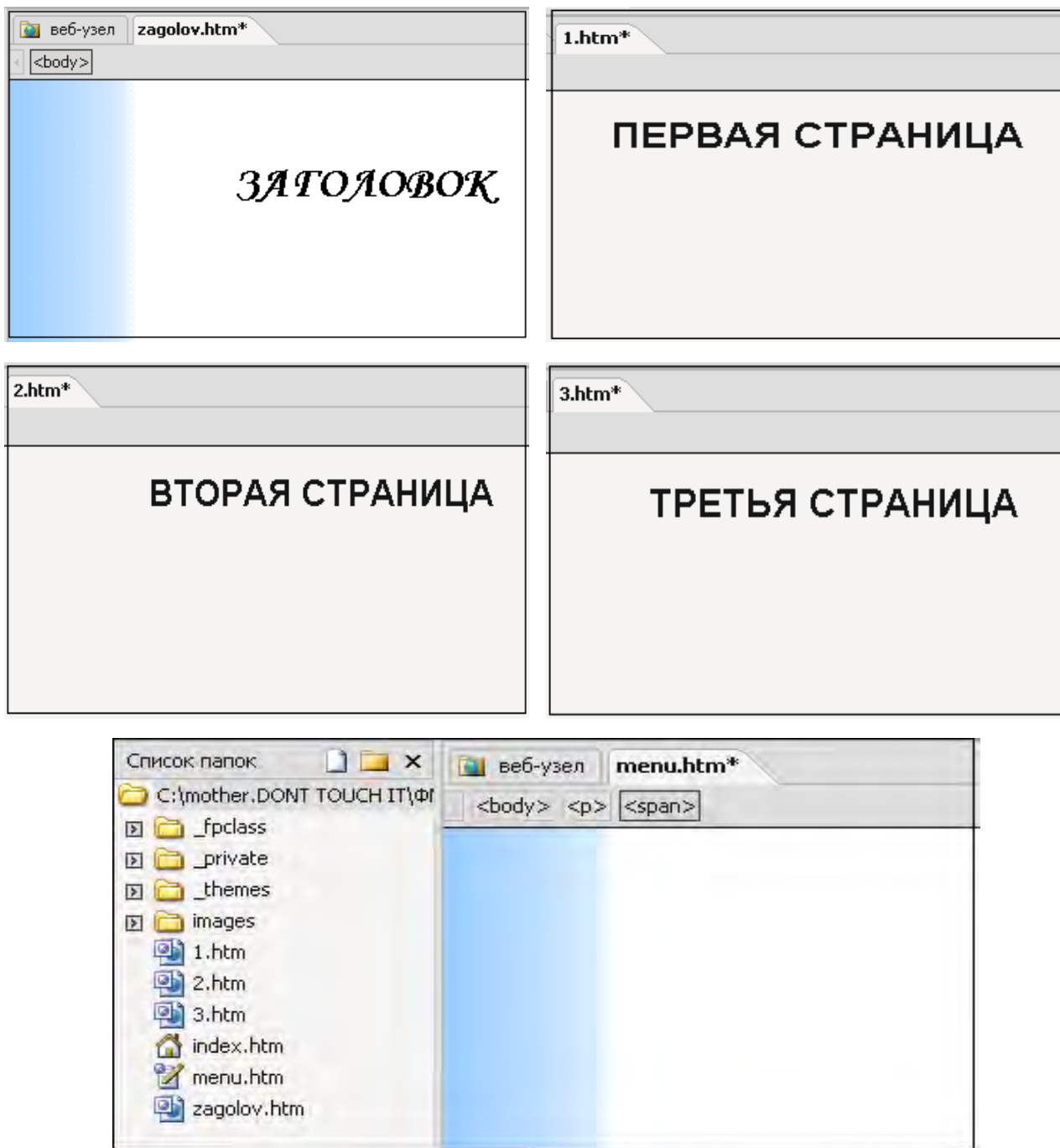


Рис. 14.16. Образцы заготовок страниц

6. Создайте главную страницу сайта с фреймовой структурой:
- выполните команду меню *Файл, Создать, Страница* или *Веб-узел*;
  - в панели *Создание веб-страницы* или узла в разделе *Создание с помощью шаблона* выберите *Шаблоны страниц* (рис. 14.17);

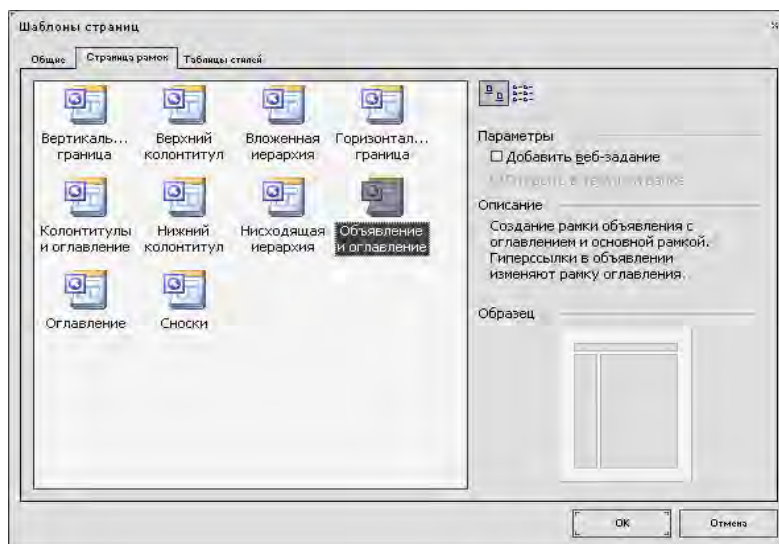


Рис. 14.17. Окно *Шаблоны страниц*

в) в открывшемся окне *Шаблоны страниц* перейдите на вкладку *Страница рамок* и выберите тип фреймовой структуры *Объявление и оглавление* (рис. 14.18);

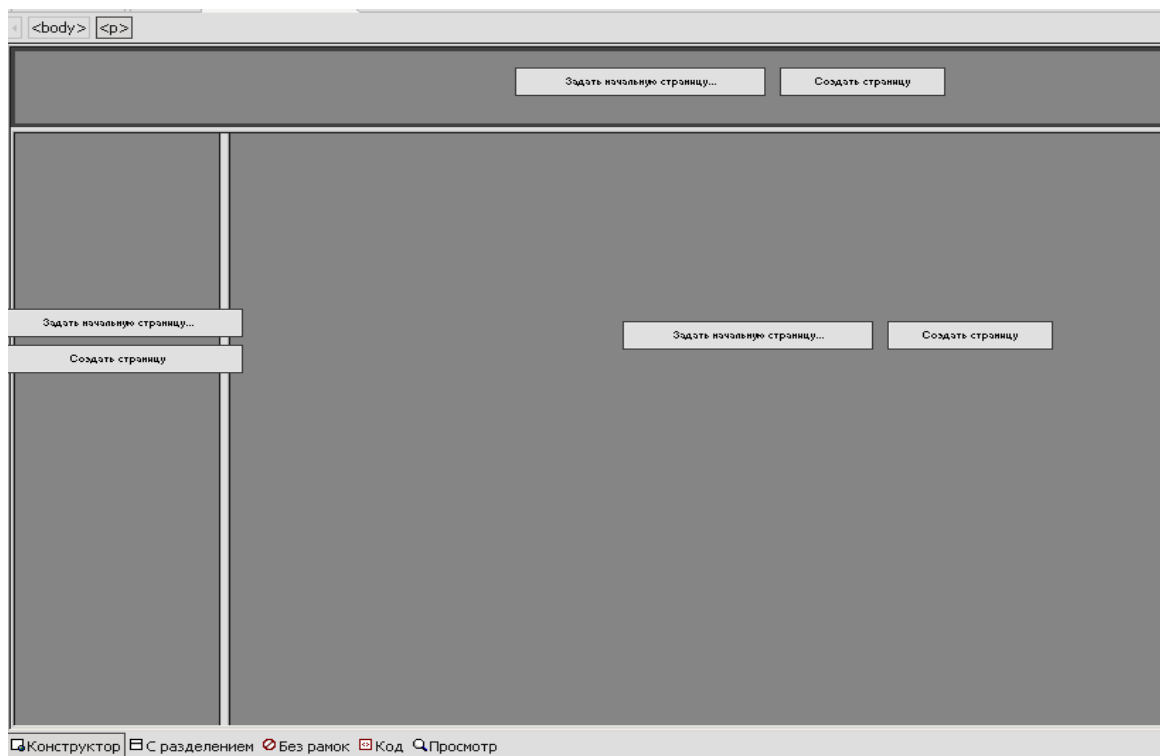


Рис. 14.18. Созданный редактором макет страницы

г) в окне отображения макета страницы (рис. 53.) щелкните на кнопке *Задать начальную страницу* в заголовочном фрейме; в списке файлов сайта выделите страницу *zagolov.htm* и щелкните на кнопке *ОК* (рис. 14.19);

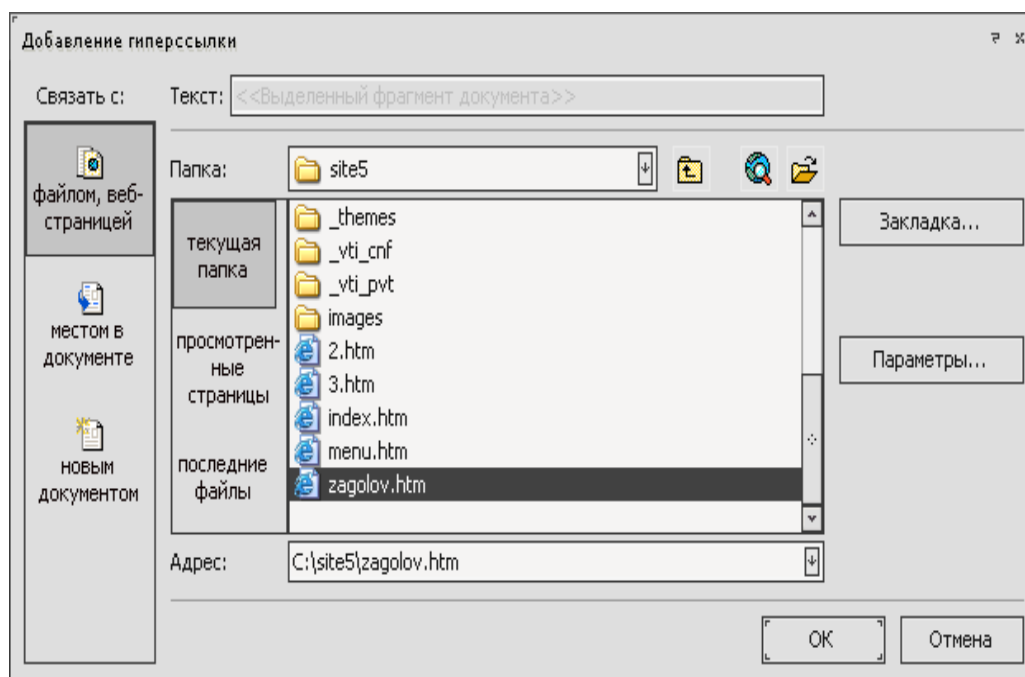


Рис. 14.19. Задание страницы, отображаемой во фрейме

д) в левом фрейме щелкните на кнопке *Задать начальную страницу*, в списке файлов сайта выделите страницу *menu.htm* и щелкните на кнопке **ОК**. страница *menu.htm*, открытая в левом фрейме, пока не содержит кнопки перехода на другие веб – страницы;

е) в правом фрейме щелкните на кнопке *Задать начальную страницу*, в списке файлов сайта выделите страницу *1.htm* и щелкните на кнопке **ОК**. Результат отражен на рис. 14.20.

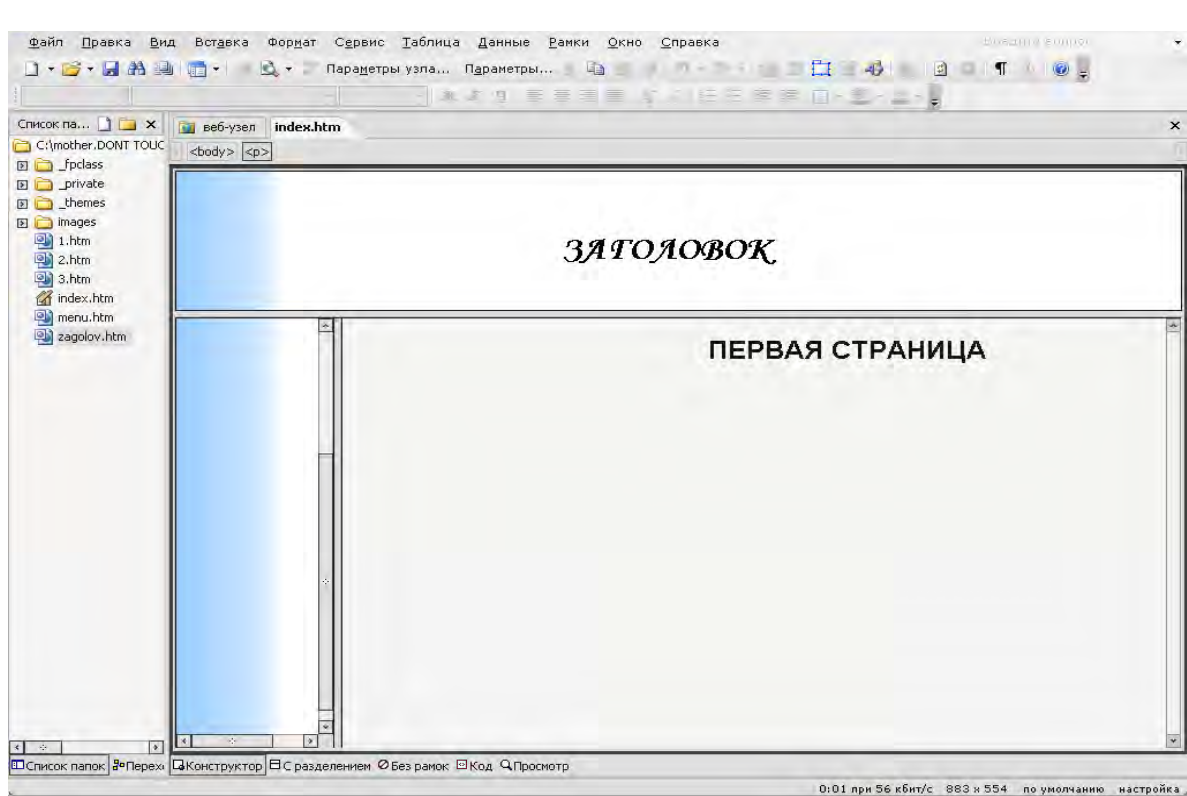


Рис. 14.20. Образец страницы index.htm

7. Если Вы хотите изменить вид границ фреймов, то, щелкая правой кнопкой по отдельным фреймам и выбирая в контекстном меню команду **Свойства рамки**, можно задать то или иное отображение границ фрейма на экране (рис. 14.21).

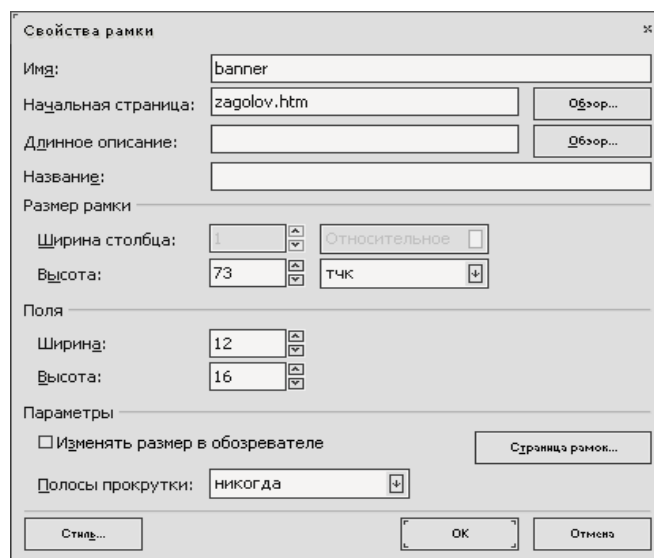


Рис. 14.21. Задание параметров фрейма в окне *Свойства рамки*

8. Продолжите работу над страницей *menu.htm* из страницы *index.htm*. Разместите кнопки с динамическими эффектами, осуществляющие навигацию по страницам сайта:

а) щелкните в области левого фрейма;

б) выполните команду меню *Вставка, Веб-компонент, Динамические эффекты, Меняющаяся кнопка* (рис. 14.22), щелкните на кнопке *Готово*.

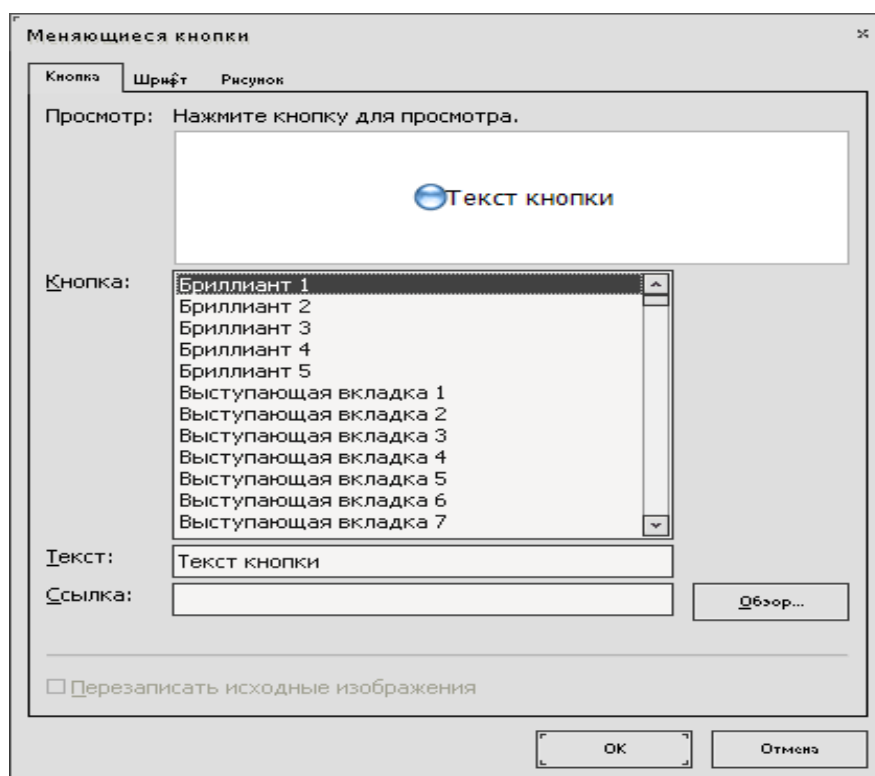


Рис. 14.22. Вставка компонента веб-узла

в) в окне *Свойства меняющейся кнопки* выполните следующие действия:

– введите текст на кнопке (используя кнопку *Шрифт*, можно изменить стиль надписи) (рис. 14.23);

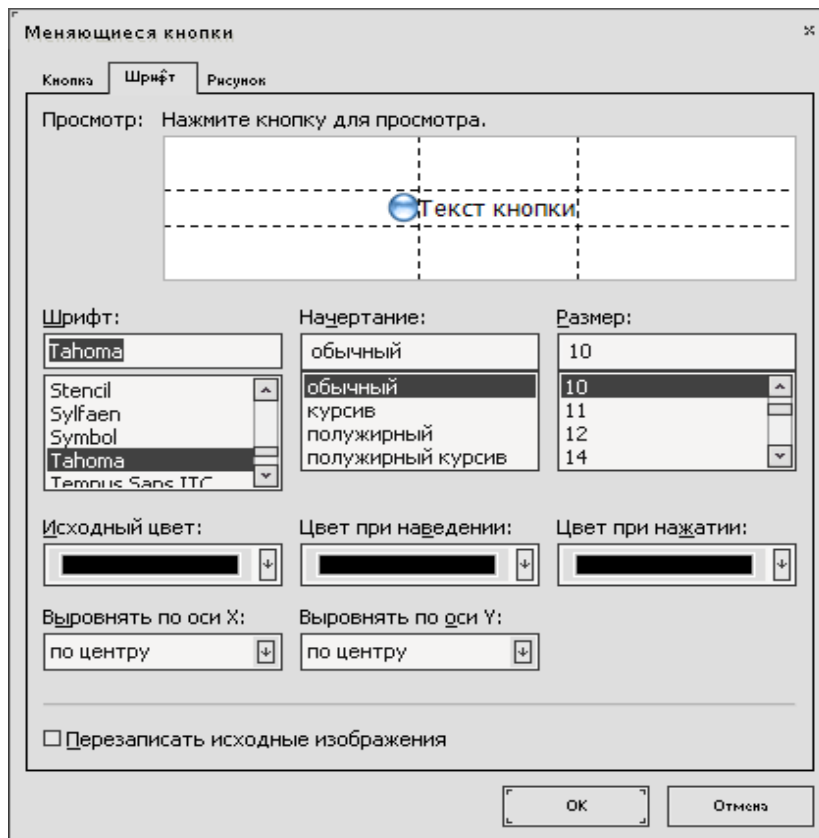


Рис. 14.23. Задание параметров меняющейся кнопки

– укажите файл перехода и фрейм, в котором будет открыт указанный файл, т.е. установите текстовый курсор в поле *Ссылка на:*, щелкните на кнопке *Обзор* в окне *Выбор гиперссылки для меняющейся кнопки* выделите имя страницы перехода в списке файлов страниц (рис. 14.24);

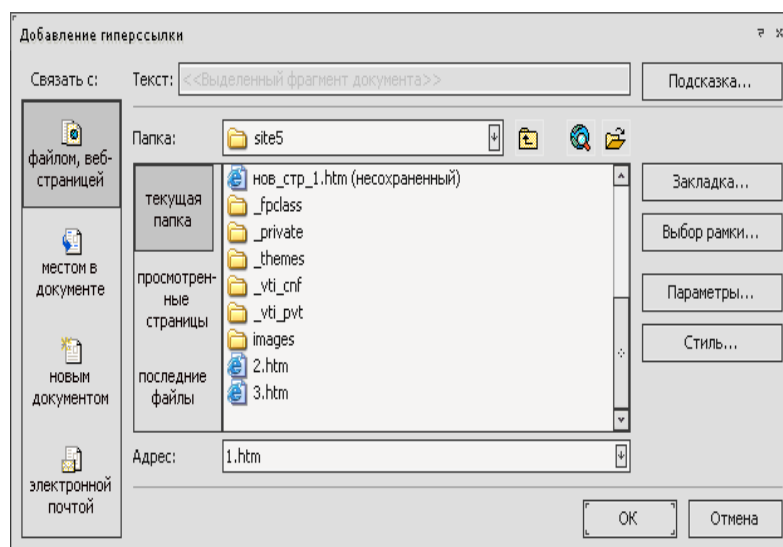


Рис. 14.24. Выбор гиперссылки для меняющейся кнопки



– Обзор в окне *Выбор гиперссылки для меняющейся кнопки* щелкните на кнопке *Выбор рамки* и укажите фрейм, в котором будет открываться страница (рис. 14.25); для этого выделите соответствующий фрейм щелчком мыши в разделе *Текущая страница рамок* и подтвердите выбор щелчками на экранных кнопках *ОК*;

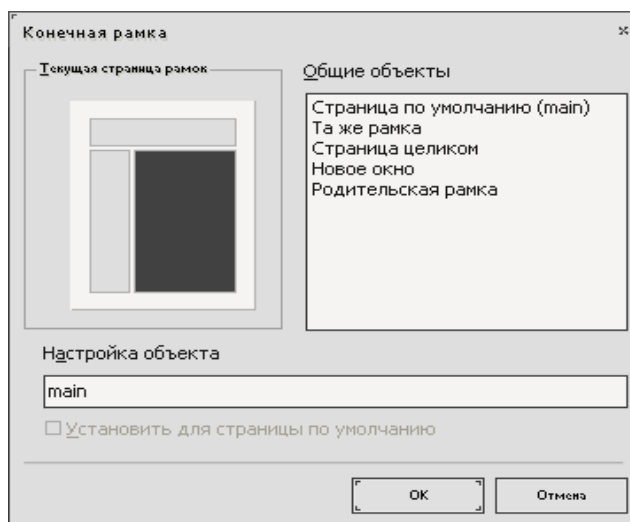


Рис. 14.25. Указание фрейма для открытия файла перехода

– Сохраните страницу *index.htm* на диске. Страница *menu.htm*, открытая в левом фрейме, содержит кнопки переходов по страницам сайта, которые будут открываться в правом фрейме (рис. 14.26).

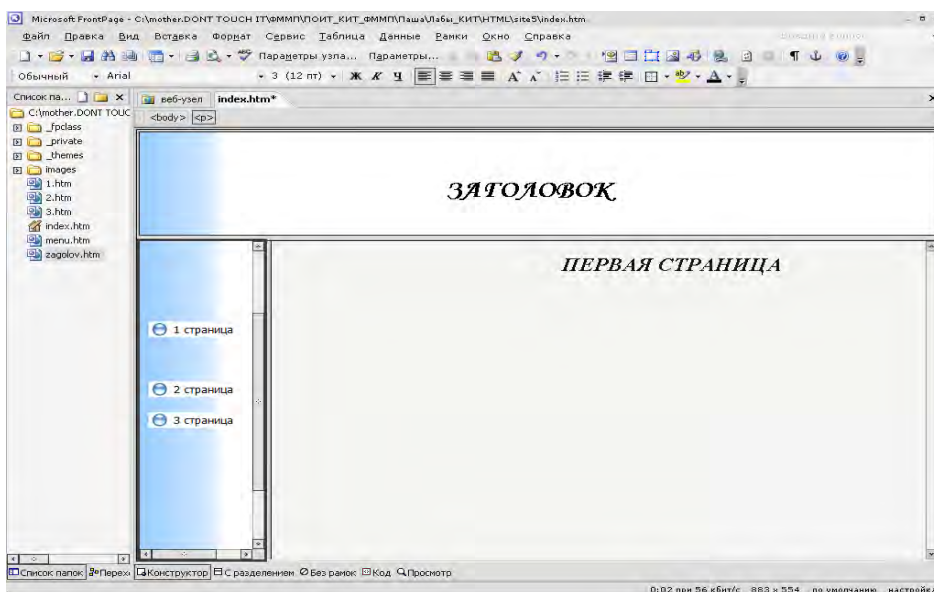


Рис. 14.26. Образец страницы *index.htm*

9. Просмотрите сайт в браузере, щелкнув на кнопке *Просмотр в браузере*.

#### **14.2. Контрольные вопросы**

1. Для чего предназначен редактор FrontPage?
2. Назовите основные элементы интерфейса редактора FrontPage?
3. Для чего используется *Тема* при оформлении веб-страницы?
4. Какие возможности по форматированию текста имеет FrontPage?

#### **14.3. Контрольные задания**

**Задание.** Создайте макет сайта с фреймовой структурой главной страницы. Разработайте удобную навигацию по страницам сайта. Разместите кнопки с динамическими эффектами, осуществляющие навигацию по страницам сайта.

##### **Вариант 1**

Создайте нисходящую иерархию данных. Основные гиперссылки вверху изменяют содержимое рамки в середине.

##### **Вариант 2**

Создайте рамку оглавления слева с гиперссылками для изменения страницы справа.

##### **Вариант 3**

Создайте рамку объявления с оглавлением и основной рамкой. Гиперссылки в объявлении изменяют рамку оглавления.

##### **Вариант 4**

Создайте колонтитул с кнопками переходов и основной рамкой снизу колонтитула. Гиперссылки в колонтитуле изменяют основную рамку.

##### **Вариант 5**

Создайте верхнего и нижнего колонтитулов для элементов перемещения. Гиперссылки в колонтитулах изменяют рамку оглавления.

##### **Вариант 6**

Создайте вложенную иерархию данных. Основные гиперссылки изменяют содержимое рамки справа.

## СОЗДАНИЕ ПАНЕЛИ ГИПЕРССЫЛОК ДЛЯ НАВИГАЦИИ ПО СТРАНИЦАМ САЙТА

**Цель работы:** изучить использование панели гиперссылок для осуществления навигации по страницам сайта.

### Методические указания

#### 1. Основные технологические этапы

1. Определите тему сайта (например, «Спецкурсы на ФММП»). Работая над темой сайта, хорошо продумайте его информационное содержание.

2. Создайте проект сайта, т. е. определите количество, взаимосвязь и тематику web-страниц. Проработайте структуру, общий дизайн сайта и содержание главной страницы (рис. 15.1).

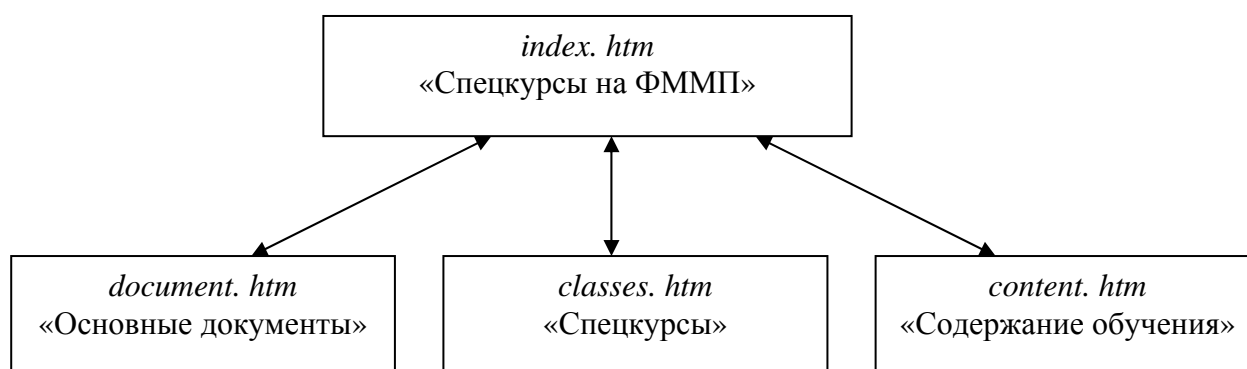


Рис. 15.1. Структура и общий дизайн разрабатываемого сайта

3. Создайте папку для хранения файлов сайта (например, Site), в которой соберите необходимый текстовый и графический материал.

4. Определите программное средство для создания сайта (например: Блокнот, Microsoft FrontPage и др.).

5. Создайте макет сайта (выполните макеты — заготовки страниц, содержащие элементы общего дизайна сайта: логотипы, фоновое оформление и

т. п.; обеспечьте взаимосвязь между страницами сайта с помощью кнопочных, текстовых или графических гиперссылок). Необходимо помнить, что хорошо разработанная навигация — важный фактор привлечения к сайту посетителей.

6. Выполните содержательное наполнение созданных макетов веб-страниц.

7. Протестируйте созданный сайт в наиболее распространенных браузерах (например: Internet Explorer, Netscape Navigator).

8. Устраните выявленные при тестировании неточности и ошибки.

9. Опубликуйте сайт в Интернете (например, можно разместить сайт на бесплатных хостингах: <http://www.narod.ru>, <http://holm.ru>, <http://hl.ru>).

10. Зарегистрируйте сайт в популярных поисковых системах (например: [www.yandex.ru](http://www.yandex.ru), [www.rambler.ru](http://www.rambler.ru), [www.aport.ru](http://www.aport.ru)).

Любой сайт имеет свою главную страницу, и работу над сайтом начинают с ее детальной разработки, хотя иногда полезно представить сайт сразу весь, как единое целое.

*Доступ к Web-странице* осуществляется при помощи браузеров, которые загружают файлы на компьютер пользователя, интерпретируют html-коды и отображают результаты. В html-кодах, как правило, содержатся указания для браузера — как необходимо отобразить текст, графику; воспроизводить или нет звуковое сопровождение; как вывести на экран ссылки на другие страницы

## **2. Создание Web-страницы в редакторе FrontPage**

Для облегчения подбора элементов оформления и цветовых схем используем готовый шаблон оформления – *тему*. С помощью меню *Формат* → *Тема* в появившемся диалоговом окне выберем, например, тему *Горизонт* и укажем *Применить как тему по умолчанию*. Тему можно применять к отдельным страницам, а также ко всему сайту. В последнем случае при создании каждой новой страницы тема устанавливается автоматически (рис. 15.2).

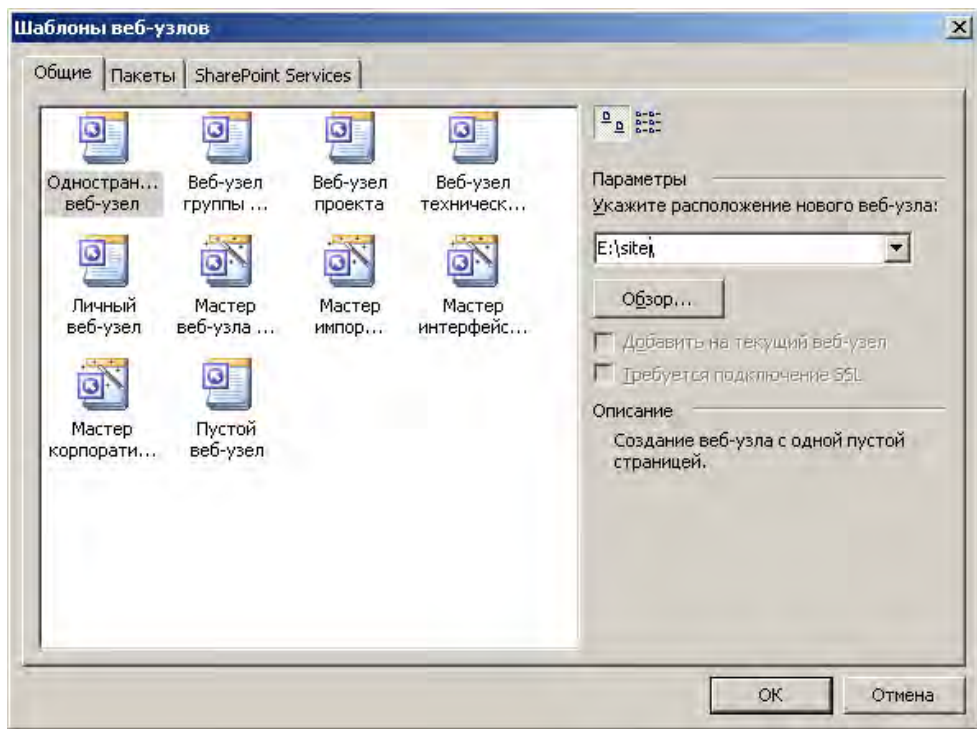


Рис. 15.2. Автоматическая установка темы новой страницы

### 3. Форматирование текста и размещение изображений в редакторе FrontPage

Форматирования текста в редакторе FrontPage осуществляется с помощью панели инструментов *Форматирование*, а также окон *Формат* (рис. 15.3) и *Абзац* (рис. 15.4).

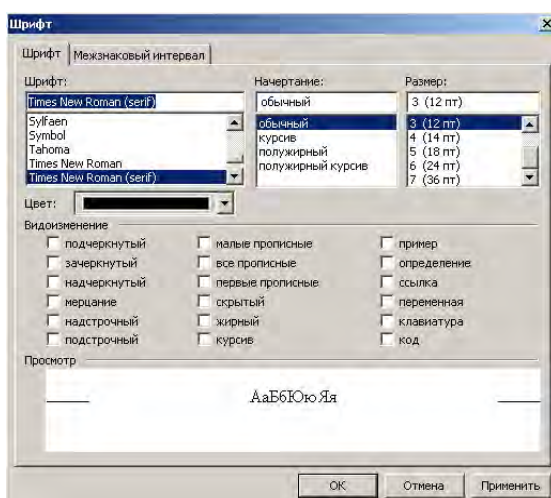


Рис. 15.3. Окно *Формат*

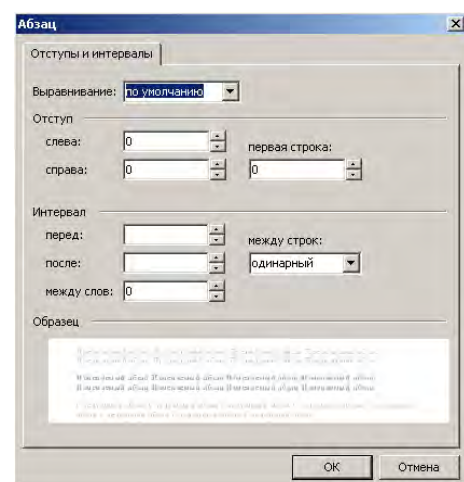


Рис. 15.4. Окно *Абзац*

Выделив текст в редакторе FrontPage (выполняется аналогично Word) в окне *Шрифт* можно выбрать его имя, начертание и указать размер, а в окне *Абзац* указать выравнивание, отступы и межстрочный интервал. Примеры форматированного текста представлены на рисунках 64 и 65. Важно помнить, что отображение веб-страницы на экране зависит от разрешения монитора и настройки браузера. Поэтому размер шрифта принято задавать в условных единицах от 1 до 7. Если размер шрифта не указан, то по умолчанию он принимается равным 3, что при установке в браузере *Размер шрифта* → *средний* соответствует 12 пунктам.

Для привлечения внимания к информации создается бегущая строка с помощью цепочки действий: *Вставка* → *Веб-компонент* → *Бегущая строка*. В окне *Свойства бегущей строки* (рис. 15.5) вводится текст, задается направление и скорость движения, время задержки и другие параметры.

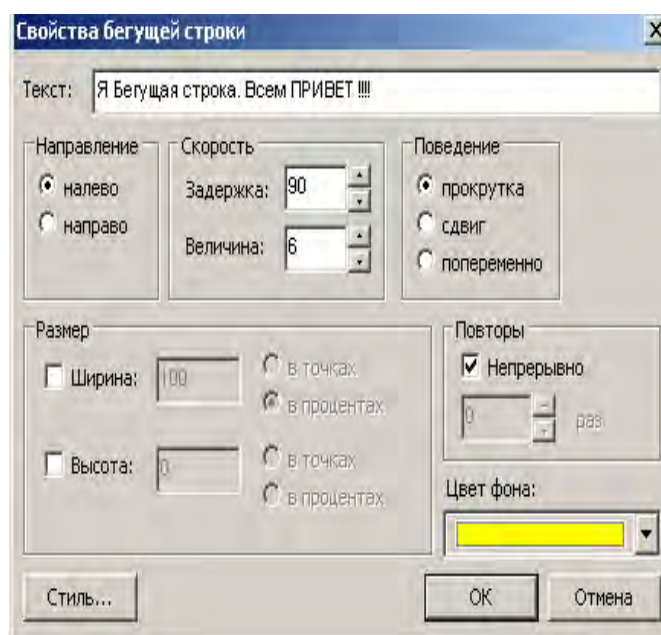


Рис. 15.5. Окно *Свойства бегущей строки*

#### 4. Размещение изображений

Размещение изображений на веб-странице производится аналогично тому, как это выполнялось в редакторе Word (команда меню *Вставка* → *Рисунок*.)

Параметры изображения задают на странице в окне *Положение*, которое активизируется с помощью меню *Формат* → *Положение* (рис. 15.6), а именно можно задать размер изображения (в пикселях или процентах), выравнивание, толщину границы, обтекание текстом и другие параметры.

Все изображения, которые мы видим на веб-страницах, хранятся в отдельных файлах, а на самой странице имеются лишь ссылки на соответствующие файлы.

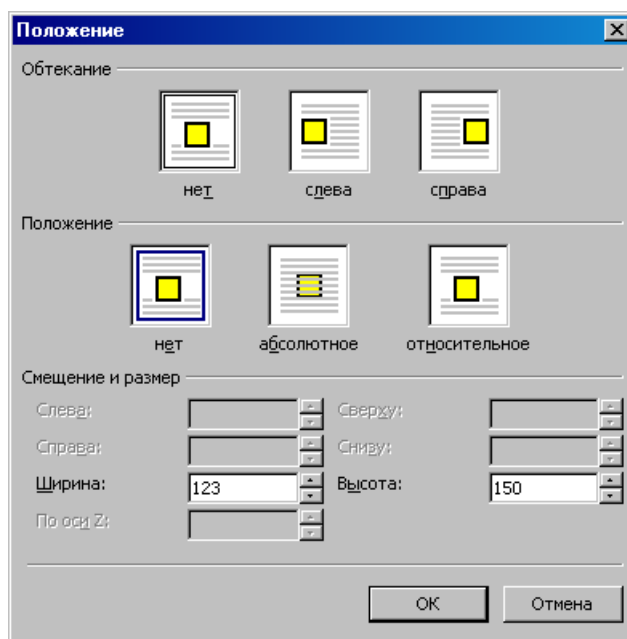


Рис. 15.6. Окно *Положение*

Программа FrontPage позволяет легко форматировать введенный текст: изменять тип заголовка, размеры и цвет шрифта, определять вид и начертание шрифта, формировать списки и т. д. Основные возможности форматирования доступны с помощью панели *Форматирование* или команд контекстном меню и выполняются аналогично текстовому процессору Microsoft Word.

Работа с графикой во FrontPage (вставка, выравнивание, изменение размера и т. п.) аналогична работе с графическими объектами в текстовом процессоре и осуществляется посредством команды *Рисунок* меню *Вставка*. Однако, при вставке графического изображения в HTML-документ возможны два варианта:

– выбранное изображение находится в папке сайта (в этом случае оно просто будет вставлено в документ);

– выбранное изображение находится вне папки сайта (FP предложит скопировать файл изображения в рабочую директорию).

Вставка таблицы в HTML-документ осуществляется с помощью команды меню *Таблица, Вставить, Таблица*.

Создание гиперссылок на страницах веб-узла является одним из важнейших этапов разработки сайта. Это могут быть гиперссылки на определенные элементы информации той же страницы, на другие страницы данного веб-узла или внешние сайты. Для создания текстовой гиперссылки используют команду меню *Вставка, Гиперссылка*.

Система FrontPage содержит готовые веб-компоненты (готовые программы, выполняемые при отображении страницы браузером), которые можно помещать на страницах сайта. Следует помнить о том, что некоторые веб-компоненты (например, панели гиперссылок) действуют только при размещении на странице, которая является составляющей сайта, созданного на основе FrontPage. Для полноценного функционирования вашего сайта его необходимо разместить на веб-сервере, поддерживающем пакет серверных расширений FrontPage Server Extensions 2003 (набор серверных программ, запускаемых на веб-сервере и расширяющих возможности сайта).

### ***15.1. Типовое задание***

***Пример 1.*** Создайте макет сайта, представленный на рис. 1.1. Разработайте удобную навигацию по страницам сайта, используя панель ссылок, основанную на структуре переходов (рис. 15.7).



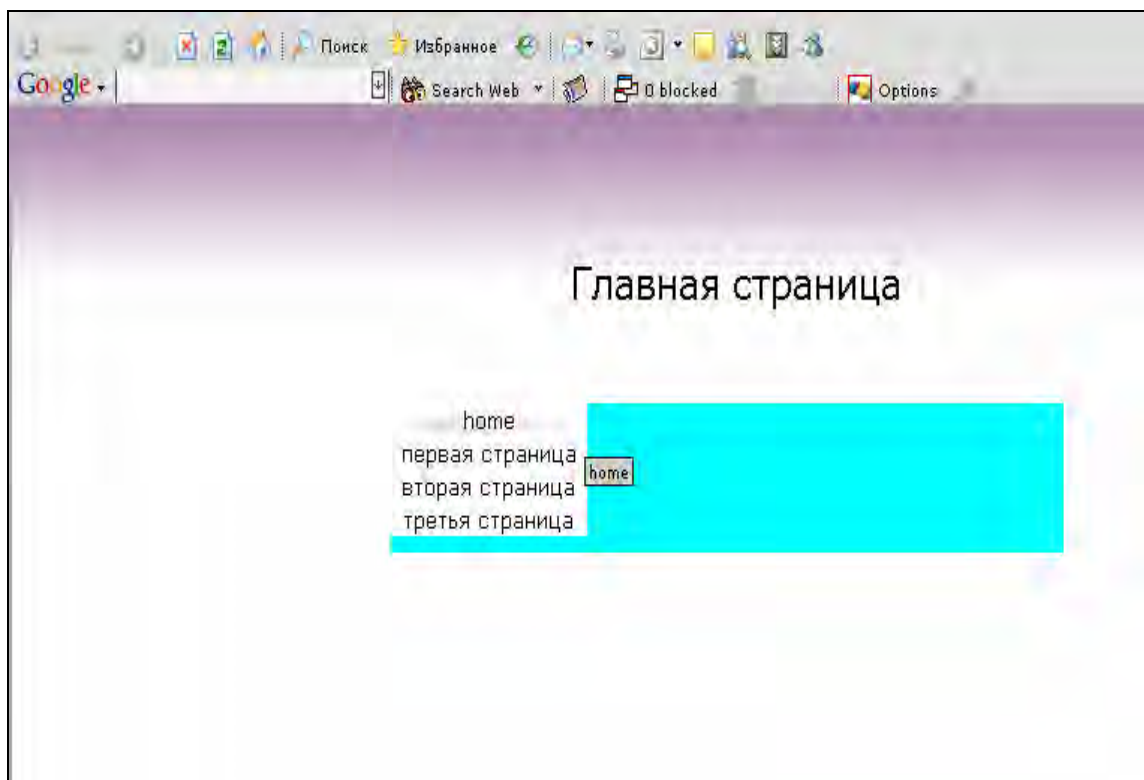


Рис. 15.7. Образец главной страницы сайта

### ***Порядок выполнения***

1. Создайте папку сайта с именем *site\_2*.
2. Откройте редактор Microsoft FrontPage.
3. Создайте сайт на основе папки *site\_2*.
4. Выполните настройку кодировки языка сайта.
5. Создайте страницы сайта с именами: *index.htm*, *1.htm*, *2.htm*, *3..*

Примените к страницам другую тему для графического оформления (*Формат, Тема, Перетекание*).

6. Создайте иерархическую диаграмму переходов по страницам сайта:

- на панели *Представления* щелкните на кнопке *Переходы* (рис. 15.8);
- отобразите иерархическую диаграмму переходов по страницам сайта методом перетаскивания веб – страницы в соответствующую позицию диаграммы (транспортируйте с помощью мыши пиктограмму веб – страницы из списка папок в поле диаграммы).

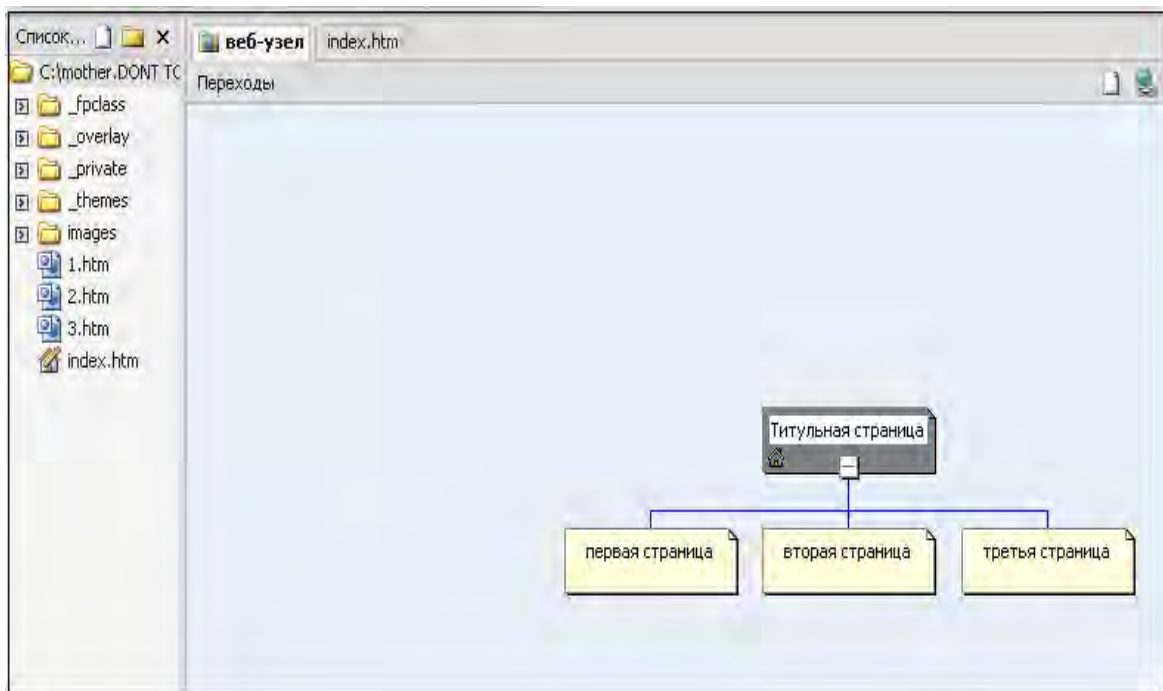


Рис. 15.8. Образец иерархической диаграммы переходов

7. Поместите на каждой странице объявление (баннер):
- отобразите страницу в режиме Страница;
  - выполните команду меню *Вставка, Объявления на странице*;
  - в окне *Свойства объявления на странице* (рис. 15.9) включите переключатель *Рисунок*, щелкните на кнопке *ОК*;

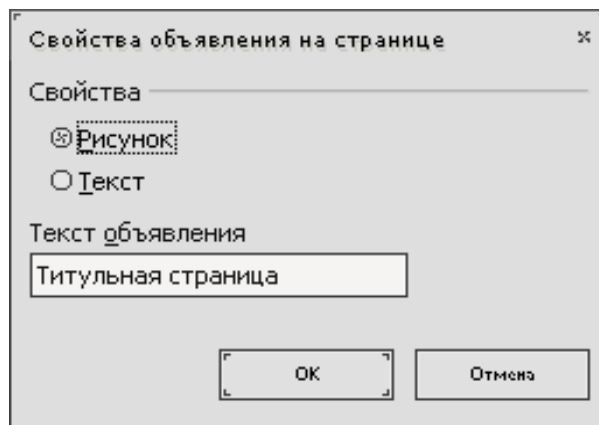


Рис. 15.9. Задание параметров объявления

- в текстовое поле *Текст объявления* введите соответствующий текст объявления и щелкните на кнопке *ОК*;
- выполните выравнивание объявления по центру страницы с помощью соответствующей кнопки на панели инструментов *Форматирование*;
- сохраните страницу в папке сайта, выполнив команду меню *Файл, Сохранить*.

*Примечание.* Для правильного отображения объявлений (баннеров) необходимо, чтобы страница была включена в иерархическую диаграмму сайта, к странице была применена тема (иначе объявление будет отображаться в виде обычного текста).

При неверном отображении текста объявления выполнить следующие действия:

- щелкнуть правой кнопкой мыши на объявлении;
- в контекстном меню выбрать команду *Свойства объявления*;
- проверить текст;
- проверить, включена ли опция *Рисунок*;
- проверить, включена ли текущая страница в иерархическую диаграмму сайта.

8. Создайте на главной странице сайта *index.htm* под объявлением пустую таблицу из одной строки и двух столбцов для жесткого размещения панели навигации и информационного содержания страницы:

- выполните команду меню *Таблица/ Вставить/ Таблица*. В окне *Вставка таблицы* укажите количество строк: 1, количество столбцов: 2;
- определите следующие параметры таблицы (рис. 15.10): в группе *Положение* установите *Выравнивание*: по центру, в группе *Границы* установите *Размер*: 0, для того чтобы границы таблицы были невидимыми;

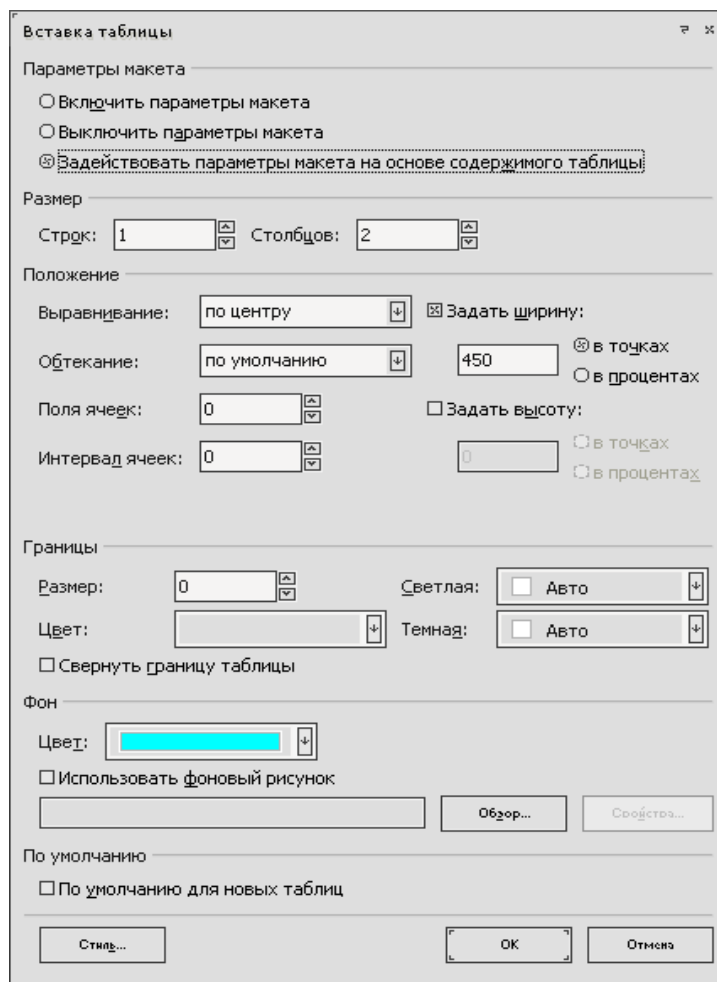


Рис. 15.10. Задание параметров таблицы

– выполните выравнивание содержимого ячеек таблицы по левому и верхнему краям:

- выделите ячейки таблицы;
- щелкните правой кнопкой мыши на выделенной области и в контекстном меню выберите команду *Свойства ячейки*;
- в появившемся окне *Свойства ячейки* в списке *Выровнять по горизонтали* укажите по левому краю, в списке *Выровнять по вертикали* укажите по верхнему краю (рис. 15.11).

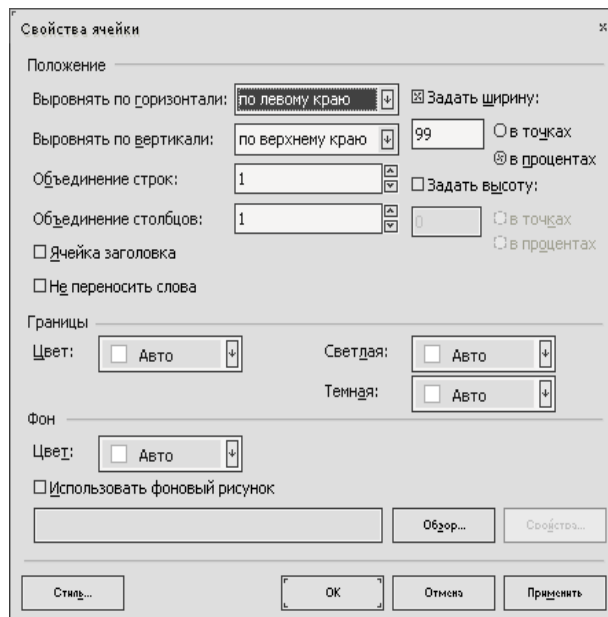


Рис. 15.11. Задание параметров ячеек таблицы в окне Свойства ячейки

9. Добавьте в левую ячейку таблицы *панель ссылок*:

- щелкните в левой ячейке таблицы;
- выберите в меню *Вставка, Веб-компонент, Панели ссылок*;
- в окне *Вставка компонента веб-узла* (рис. 15.12) выберите тип панели: *Панель, основанная на структуре переходов*, щелкните на кнопке *Далее*;
  - в следующем окне диалога укажите стиль панели *Использовать тему страницы* (рис. 15.13), щелкните на кнопке *Далее*;
  - выберите ориентацию панели *вертикальная* в диалоговом окне *Вставка компонента веб-узла*, щелкните на кнопке *Готово*;
  - в окне *Свойства панели ссылок* (рис. 15.14) на вкладке *Общие* задайте режим связи между страницами сайта: включите переключатель *Дочерние страницы домашней*, установите флажок *домашняя страница*;

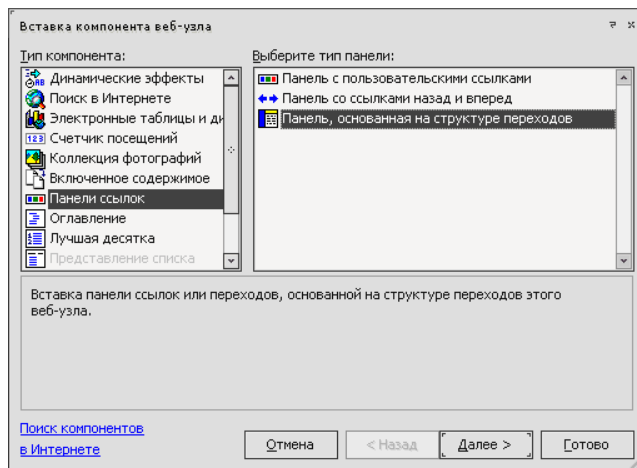


Рис. 15.12. Вставка панели переходов

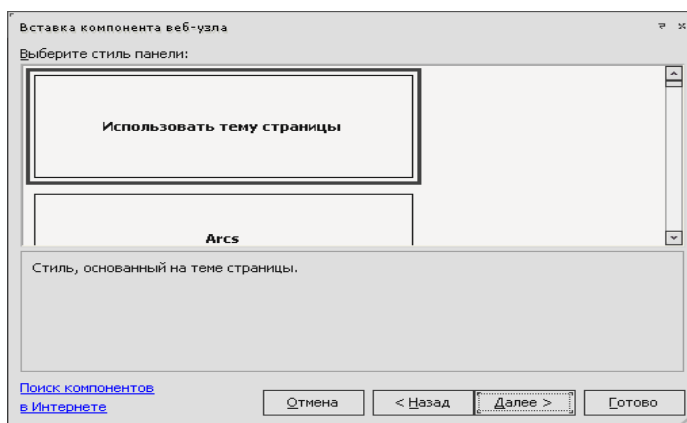


Рис. 15.13. Выбор стиля панели переходов в окне *Вставка компонента веб-узла*

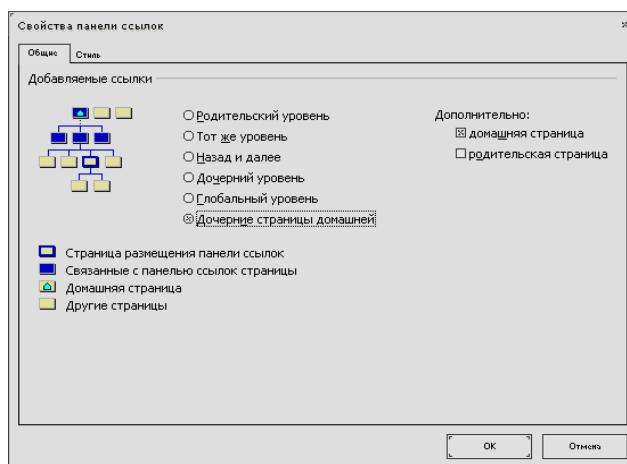


Рис. 15.14. Задание режима связи между страницами сайта в окне *Свойства панели ссылок*

10. Выровняйте ширину левой ячейки таблицы по ширине панели навигации перетаскиванием правой границы ячейки.

11. Сохраните страницу под именем *index.htm* на диске.

12. Скопируйте созданную на странице *index.htm* таблицу, с размещенной в левой ячейке панелью навигации, на все страницы сайта. Результатом должны быть веб-страницы, приведенные на рис. 15.15.

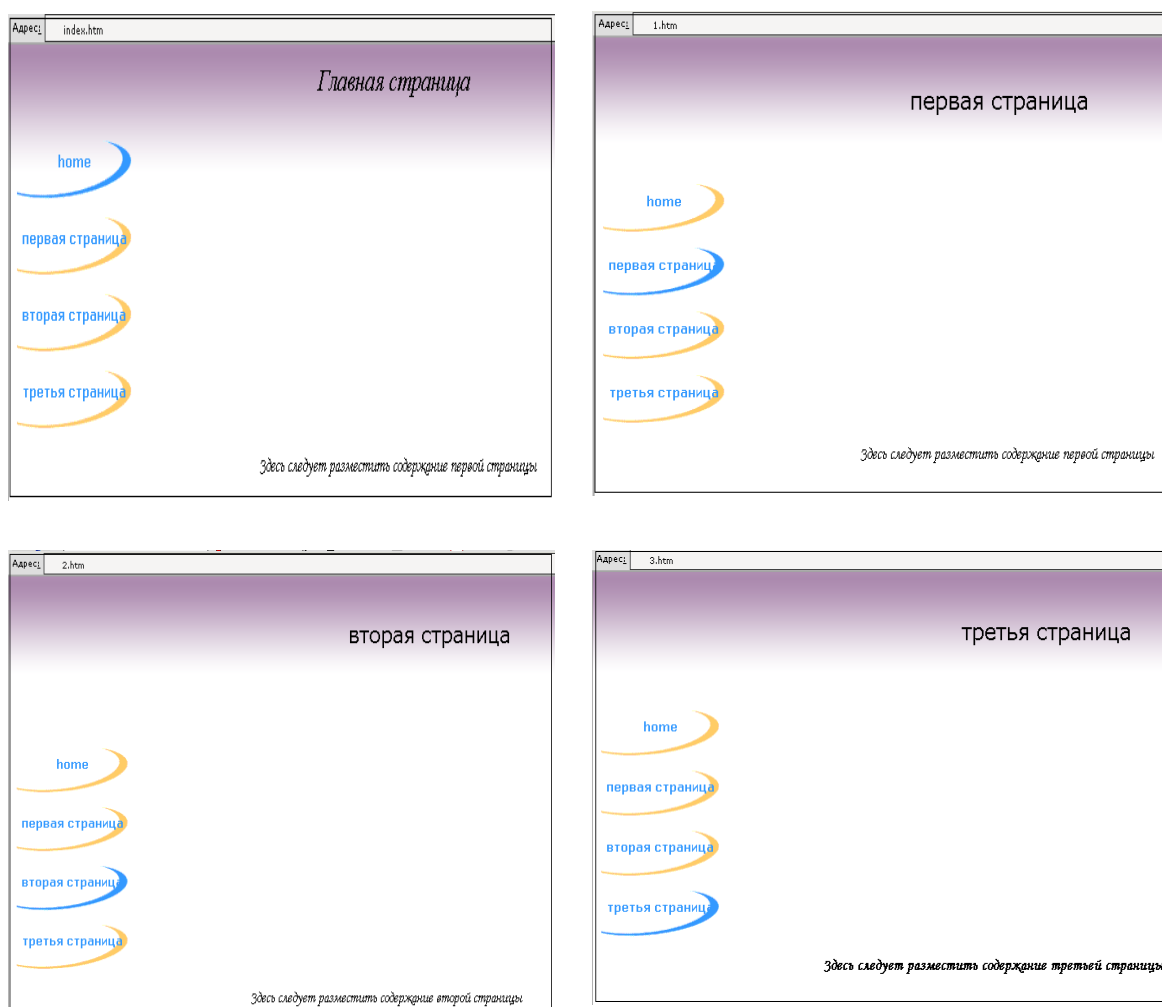


Рис. 15.15. Образцы страниц сайта

13. Просмотрите сайт в браузере, щелкнув на кнопке *Просмотр в браузере* основного меню программы FrontPage.

### **15.2. Контрольные вопросы**

1. Создание Web-страниц на основе стандартных шаблонов структуры страницы.
2. Гипертекстовые связи для перехода к Web-страницам.
3. Как вставить закладку на странице, создать гиперссылку к странице.
4. Порядок создания панели ссылок.
5. Как задается размер шрифта на веб-страницах?
6. Как создать бегущую строку?

### **15.3. Контрольные задания**

**Задание.** Используя панель ссылок, основанную на структуре переходов, разработайте удобную навигацию по страницам сайта. Создайте Web – сайт, используя соответствующие шаблоны веб-узлов *FrontPage*:

Номер варианта	Тема
1	Создать веб-узел рабочей группы, библиотекой общих документов, списком задач, списком контактов.
2	Создать веб-узел проекта, включающего список участников, календарный план, отчеты о состоянии, архив и форум обсуждений.
3	Создать веб-узел для служб технической поддержки.
4	Создать веб-узел для групп обсуждений с оглавлением и возможностью поиска по всему тексту.
5	Создать веб-узел, позволяющий подключиться к базе данных и просматривать, обновлять, удалять записи.
6	Создать корпоративный веб-узел для Интернета.

#### **Вариант 1**

Создайте Web-сайт дискуссий, используя соответствующий шаблон *FrontPage*.



### ***Вариант 2***

Создайте рамку оглавления слева с гиперссылками для изменения страницы справа.

### ***Вариант 3***

Создайте рамку объявления с оглавлением и основной рамкой. Гиперссылки в объявлении изменяют рамку оглавления.

### ***Вариант 4***

Создайте колонтитул с кнопками переходов и основной рамкой снизу колонтитула. Гиперссылки в колонтитуле изменяют основную рамку.

### ***Вариант 5***

Создайте верхнего и нижнего колонтитулов для элементов перемещения. Гиперссылки в колонтитулах изменяют рамку оглавления.

### ***Вариант 6***

Создайте вложенную иерархию данных. Основные гиперссылки изменяют содержимое рамки справа.

## Лабораторная работа № 16

### СОЗДАНИЕ WEB-СТРАНИЦ В РЕДАКТОРЕ MICROSOFT PUBLISHER

**Цель работы:** научиться создавать web-узел с использованием редакторов WYSIWYG (*Micrisift Publisher*) и публиковать, разработанный сайт на бесплатном сервере в сети Интернет

#### *Методические указания*

Web-узел включает в себя набор связанных между собой Web-страниц. Начальная страница сайта называется домашней (homepage). Связи между страницами задаются с помощью гиперссылок, которые могут быть выражены в виде текста, рисунков или кнопок. Щелчок гиперссылки левой кнопкой мыши позволяет перейти на другую Web-страницу, на которую указывает данная гиперссылка. Web-страница может содержать текст и рисунки, которые будут отображаться в окне браузера, и в отличие от печатной страницы, неограничена в размере.

*Microsoft Publisher* прежде всего используется для быстрого создания простых Web-узлов. Создание Web-страницы в *Microsoft Publisher* не требует знания HTML, специального языка, который используется при создании и отображении Web-страниц.

Проще всего создать Web-узел, воспользовавшись мастером, который выполнить все необходимые шаги. Щелкнув несколько раз кнопкой мыши, можно за считанные минуты создать привлекательный шаблон Web-сайта, а затем отредактировать его и дополнить, включая звуковое оформление, тексты и рисунки.

При разработке сайта не перегружайте Web-страницы графическими, динамическими, музыкальными, анимационными и другими объектами, элементы на странице не должны накладываться друг на друга, а между текстом и графикой всегда должно оставаться пространство. Следите за навигацией,

облегчающей работу с сайтом и привлекающей к нему посетителей. Фон, заголовки и общее расположение материала на экране должны соответствовать друг другу и не нарушать целостность восприятия.

*Макет* представляет собой прототип сайта, содержащий образцы текста и рисунков, которые необходимо заменить своими.

При работе не забывайте, что сначала *Web-узел* необходимо сохранять как файлы *Microsoft Publisher*. Не следует сохранять его как *Web-страницу* до тех пор, пока работа не будет закончена полностью. Связано это с тем, что, если файл *Publisher* сохранен как *Web-страница*, его уже нельзя редактировать. Созданный в *Publisher Web-узел* можно импортировать в *Microsoft FrontPage*.

## 16.1. Типовое задание

### Пример 1.

Создайте *Web-сайт* «Визитная карточка факультета», который будет рассказывать о самом факультете, об обучении, что интересного происходит на факультете кроме учебы с помощью редактора *Microsoft Publisher*. Объедините созданные страницы Вашего сайта гиперсвязью и протестируйте на работоспособность (рис 16.1).



Рис. 16.1. Оформление *Web-сайта* «Визитная карточка факультета»

### **Порядок выполнения**

1. Запустите *Microsoft Publisher*.
2. В *Области задач* в разделе *Начать с макета* выберите *Web-узлы и электронная почта/ Web-узлы*. В правую часть окна загрузятся макеты простых Web-узлов, на основе которых можно создать сайт.
3. Прокрутите содержащиеся в правой части окна макеты и дважды щелкните образец *Простой веб-узел. Фломастер* (рис. 16.2).

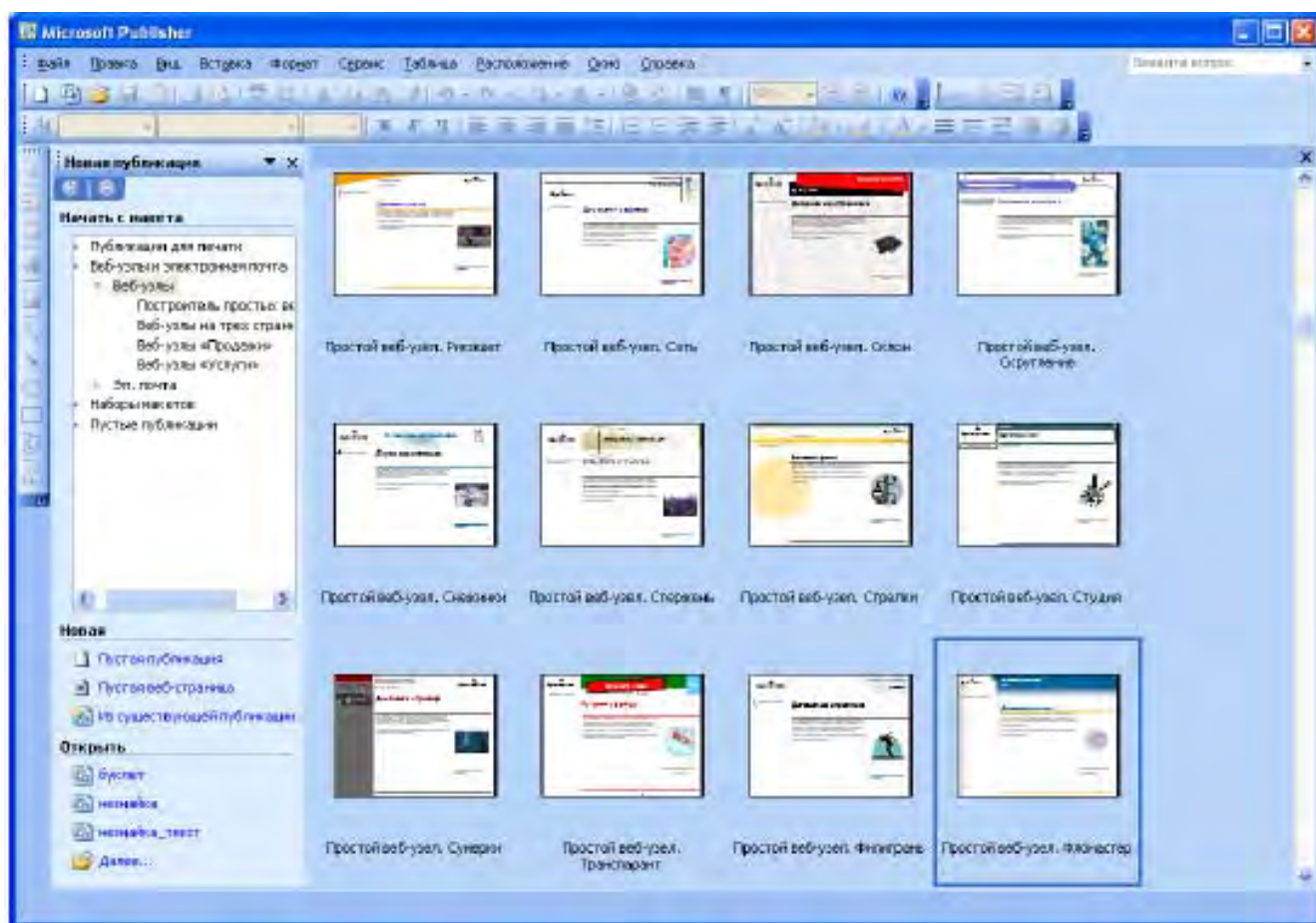


Рис. 16.2. Выбор образца макета *Простой веб-узел. Фломастер*

4. В диалоговом окне *Удобный построитель веб-узлов* установите флажки *Размещение сведений о профиле организации* и *Размещение сведений о способах связи с организацией*. Все остальное мало подходит (рис. 16.3).

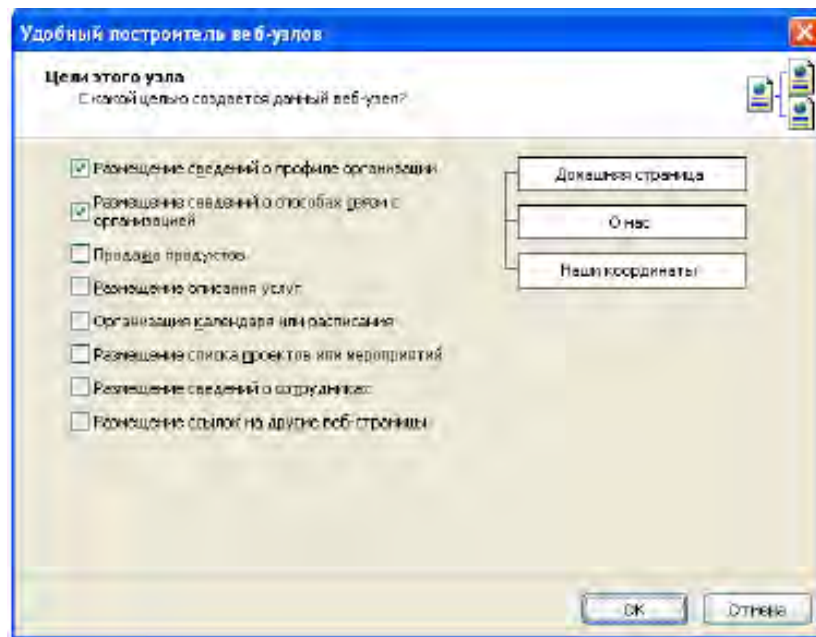


Рис. 16.3. Окно *Удобный построитель веб-узлов*

5. Щелкните *OK*.

В окне программы отобразится шаблон домашней страницы (рис. 16.4).

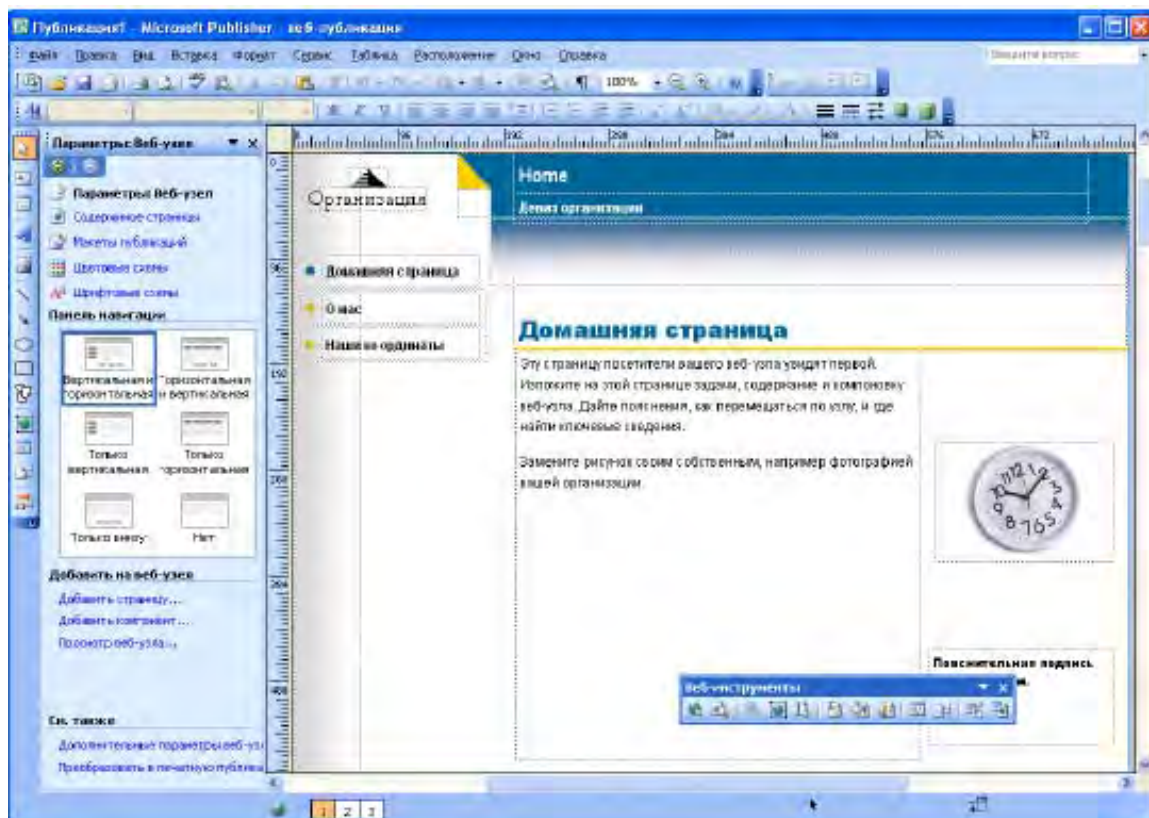


Рис. 16.4. Шаблон домашней страницы

## 1. Панель навигации

В *Области задач* отображается схема размещения панели навигации на страницах сайта.

Панель навигации представляет собой набор кнопок или текстовых гиперссылок (обычно это перечень основных Web-страниц сайта), которая отображается на каждой странице сайта. *Microsoft Publisher*, позволяет выбрать разные варианты размещения панели навигации на страницах сайта.

По умолчанию устанавливается вариант *Вертикальная и горизонтальная*, т. е. панель будет располагаться слева на странице и она же – вниз.

Во второй строке представлены единичные навигационные панели, т. е. навигационная панель будет на странице одна, и в зависимости от выбранного варианта располагается только слева на странице, в ее верхней части или только внизу. Последний вариант — страница без панели навигации (рис. 16.5).

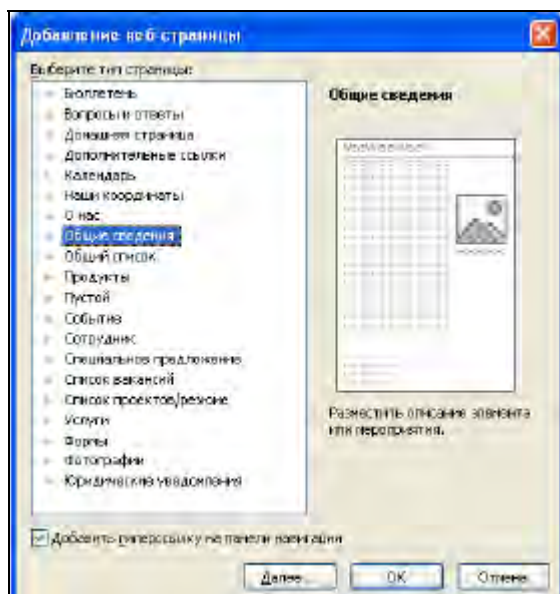


Рис. 16.5. Выбор типа страницы

## 2. Добавление в сайт Web-страниц

1. В *Области задач* после вариантов размещения панели навигации расположен список команд *Добавить на веб-узел*, щелкните *Добавить страницу*.

2. В открывшемся диалоговом окне *Добавление веб-страницы* в области *Выбрать тип страницы* щелкните *Общие сведения*. Обратите внимание, чтобы

был установлен флажок в окне *Добавить гиперссылку на панель навигации*.

3. Щелкните *OK*.

Теперь сайт состоит из четырех страниц.

### 3. Изменение внешнего вида кнопок

Для изменения внешнего вида панели навигации:

1. Выделите панель навигации.

2. Подведите указатель мыши к кнопке , расположенной под панелью навигации. Указатель примет вид . Щелкните кнопку.

3. В *Области задач* откроется раздел *Изменение Панели навигации*. В поле *Применить макет* установите флажок *Крупные кнопки*.

4. Используя полосу прокрутки, просмотрите представленный здесь же список макетов кнопок навигационной панели. Выберите понравившийся вариант и щелкните его левой кнопкой мыши (рис. 16.6).

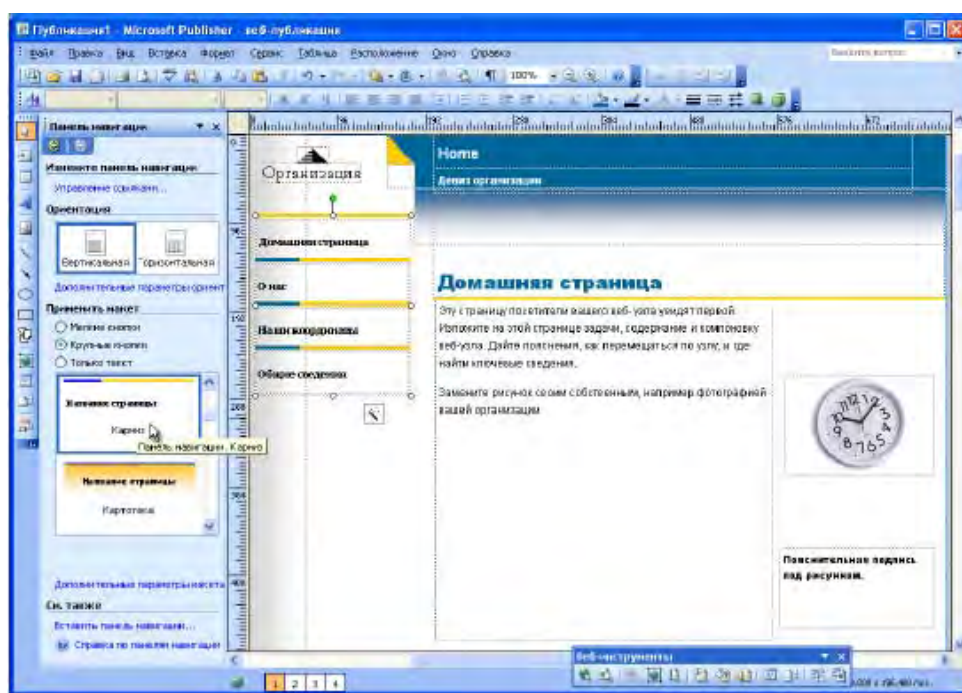


Рис. 16.6. Список макетов кнопок навигационной панели

*Примечание.* Для возврата в область задач *Содержимое страницы*, щелкнуть необходимое число раз кнопку *Назад* или раскрыть список с названиями разделов *Области задач* и выбрать нужный (рис. 16.7).

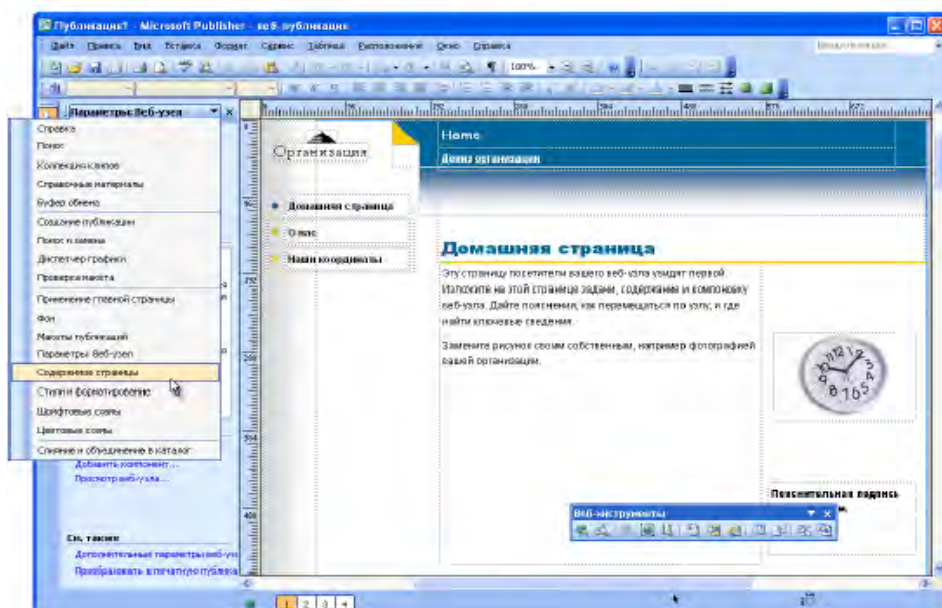


Рис. 16.7. Список с названиями разделов *Области задач*

#### 4. Замена названия кнопок панели навигации

1. Выделите текст первой кнопки панели навигации (рис 16.8).
2. Измените название кнопки на *Визитная карточка факультета*.
3. Измените название второй кнопки на *Мы учимся*.
4. Измените название третьей кнопки на *А что кроме учебы?*
5. Измените название четвертой кнопки на *Послесловие*.

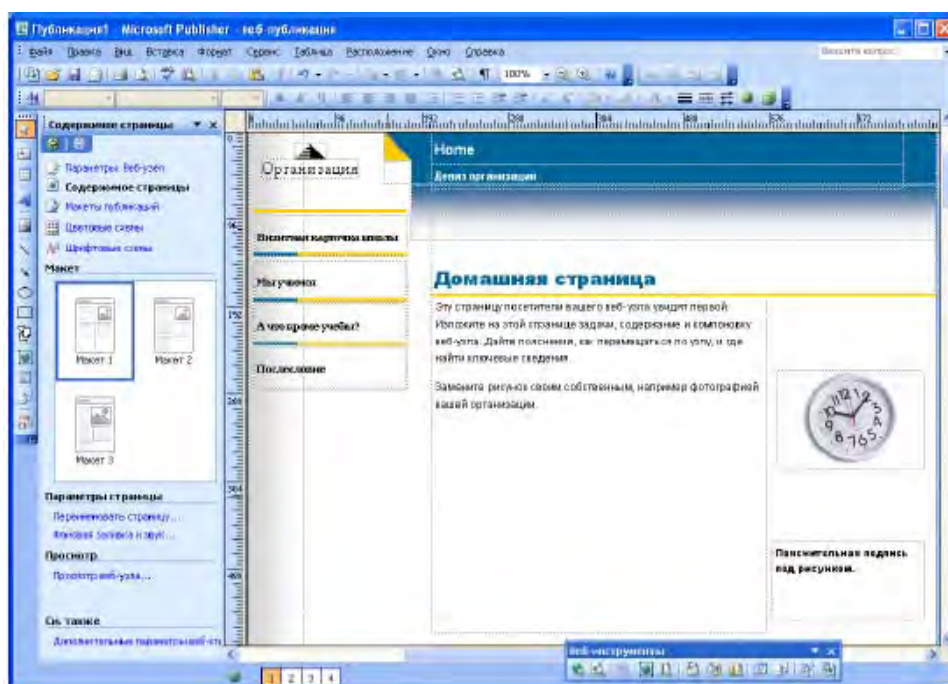


Рис. 16.8. Изменение названия кнопок



Просмотрите все страницы сайта и убедитесь, что название кнопок было автоматически изменено на всех навигационных панелях.

## 5. Выбор цветовой схемы

1. В Области задач щелкните *Цветовых схем*.
2. В открывшемся списке цветовых схем щелкните *Фьорд*. Выбранная цветовая схема будет применена для всех Web-страниц сайта (рис. 16.9).

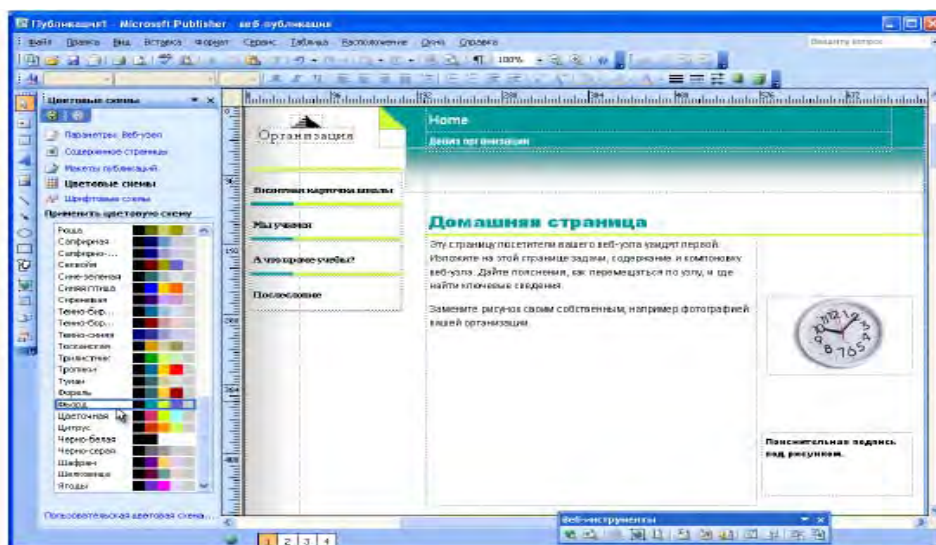


Рис. 16.9. Список цветовых схем

Для просмотра Web-страницы сайта, щелкните кнопку перехода по страницам **1** **2** **3** **4** в нижней части окна (рис. 16.10).

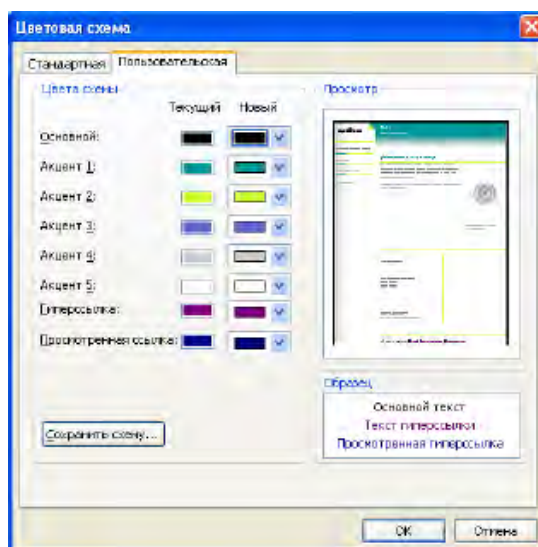


Рис. 16.10. Создание собственной цветовой схемы

Создать собственную цветовую схему можно, открыв диалоговое окно *Цветовая схема*/ вкладка *Пользовательская* (в нижней части *Области задач*), и сохранить ее, щелкнув кнопку *Сохранить схему*.

## 6. Вставка фона

1. Для установки фона страницы и добавления звуковых эффектов раскройте список названий областей задач и выберите в меню *Формат / Фон*.

2. В *Области задач* откроется раздел *Фон*. Щелкните в разделе *Применить фон* (рис. 16.11) подходящий вариант (для примера, *Акцент 1*), а затем – из предложенного списка вариантов фона выберите понравившийся (в данном случае цвет фона страницы выберите *14 % оттенок Акцент 1*, а вариант заливки страницы — однотонный).

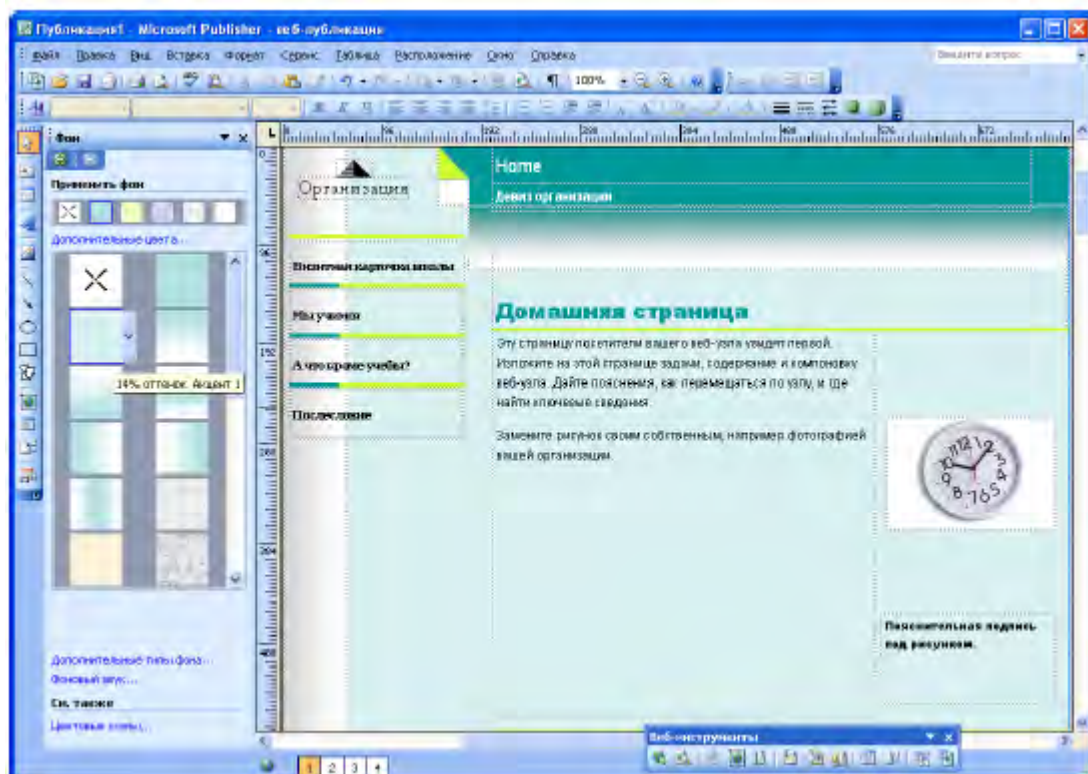


Рис. 16.11. Установка фона страницы

Выбранный вариант фона применяется только к текущей странице, все остальные страницы в данном нашем случае остались белыми. Если вас не

устраивают предложенные цветовые варианты, щелкните

### **Пример 2**

На *второй* странице сайта создайте вариант фона *Градиентная заливка* (по вертикали) (рис. 16.12). На *третьей* странице примените вариант фона *Заливка картинкой* (завитки). Для заливки *четвертой* страницы создайте собственный вариант фона.

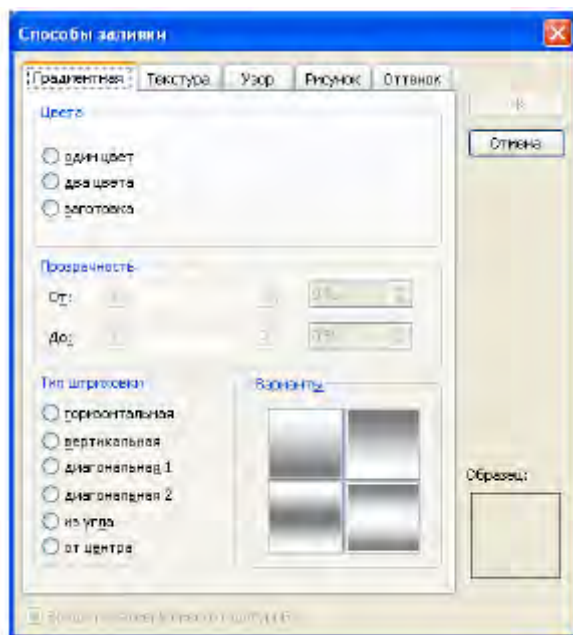


Рис. 16.12. Окно *Способы заливки*

### **Указания:**

1. В *Области задач* щелкните *Дополнительные типы фона*, откроется диалоговое окно *Способы заливки*;
2. Создайте свой цветовой вариант заливки, устанавливая различные параметры на вкладке *Градиентная*, *Текстура*, *Узор*, *Рисунок*, *Оттенок*;
3. После создания (или выбора из предложенных вариантов) подходящего фона щелкните *ОК*.

## **7. Вставка фонового звука**

1. Для включения в сайт фонового звука в *Области задач* щелкните *Фоновый звук*.

2. В открывшемся диалоговом окне *Параметры веб-страницы* (рис. 16.13) для выбора звука щелкните кнопку *Обзор*, если у вас есть записанный звук, найдите нужный файл, выделите его и щелкните *ОК*.

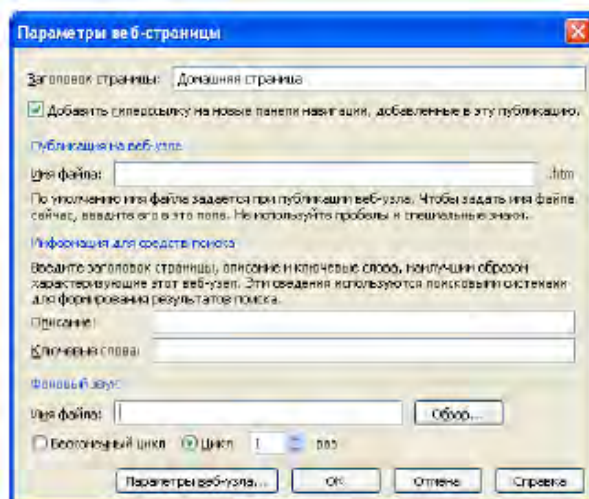


Рис. 16.13. Окно *Параметры веб-страницы*

3. Если у вас нет нужного файла, в *Publisher* есть коллекция звуков, которой вы можете воспользоваться. Для этого в диалоговом окне *Параметры веб-страницы* щелкните кнопку *Обзор*, по умолчанию откроется папка, содержащая коллекцию звуковых файлов программы (рис. 16.14). Выделите любой звуковой файл и щелкните кнопку *Открыть*.

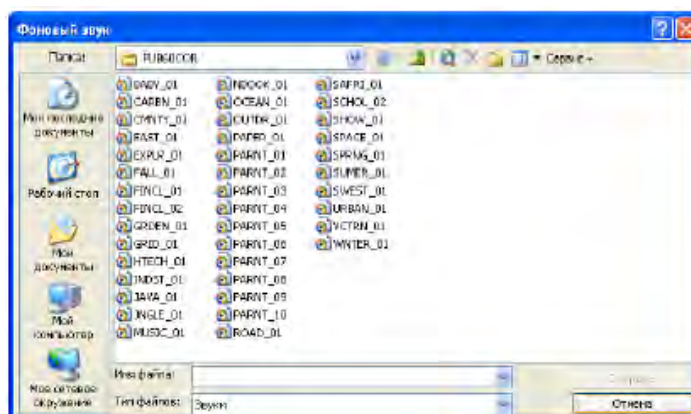


Рис. 16.14. Выбор звуковых файлов

4. Чтобы звуковой эффект был постоянным при просмотре данной Web-

страницы, в диалоговом окне *Параметры веб-страницы* установите переключатель в разделе *Фоновый звук* в положение *Бесконечный цикл*, а чтобы звук повторялся конкретное количество раз – в положение *Цикл* и в текстовом поле справа введите число повторов.

5. Щелкните *ОК*.

### **Пример 3**

Откройте вторую страницу сайта, установите фоновый звук, воспользовавшись файлом, предложенным преподавателем, расположенным на учебном диске. Установите количество повторов – 3. Можно воспользоваться файлами из собственной коллекции звуков или из коллекции программы.

## **8. Сохранение Web-узла**

1. В меню *Файл / Сохранить как*.

2. В поле *Папка* диалогового окна *Сохранить как* раскройте список дисков, выберите диск, а затем папку, куда предполагаете сохранить публикацию.

3. В поле *Имя файла* введите имя **web**. Программа автоматически установит расширение **.pub**.


4. Щелкните кнопку *Сохранить*.

Ввод содержания *Web-узла*. Если у вас есть опыт работы в *Microsoft Publisher*, добавьте на *Web-страницу* текст и рисунки. Замена автозаполнителей реальным содержанием, форматирование текста и изменение рамок осуществляется точно так же, как в других публикациях.

## **9. Работа с текстом**

**Пример 4.** Введите текст.

1. После запуска *Microsoft Publisher* откройте созданную в ходе выполнения упражнений публикацию с именем **web**.

2. Откройте первую страницу, щелкнув кнопку  на панели *Перемещение по страницам* в нижней части окна.

3. Щелкните левой кнопкой мыши в строке *Заголовок домашней страницы* и введите *Визитная карточка факультета*.

4. В рамке **Девиз** организации введите девиз.

5. Щелкните в главной текстовой рамке в центре страницы, удалите текст автозаполнителя и введите подзаголовок *Этапы большого пути*, далее наберите текст об истории создания факультета.

## 10. Форматирование текста

1. Выделите заголовок страницы *Визитная карточка школы*.

2. Отформатируйте заголовок: Times New Roman, 24, полужирный, выравнивание — по центру.

3. Для изменения цвета букв на панели инструментов **Форматирование** раскройте список *Цвет шрифта* (рис. 16.15) и из предложенной палитры выберите *Акцент 1*.

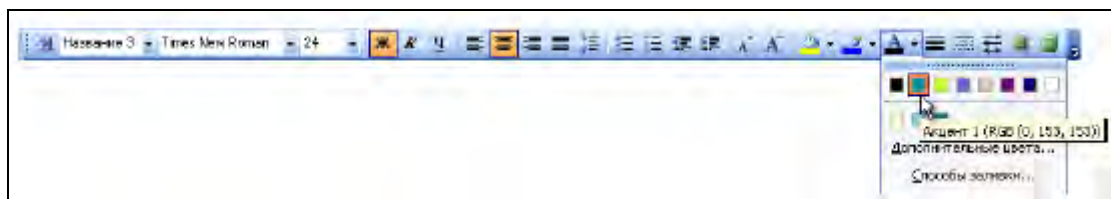


Рис. 16.15. Установка цвета шрифта

4. Щелкните на пустом месте публикации для отмены выделения.

5. Текст отобразится в выбранном цвете.

6. Сохраните публикацию.

### Пример 5

Отформатируйте текст с девизом: Times New Roman, 10, полужирный, курсив, белый, выравнивание — по центру. Отформатируйте подзаголовок *Этапы большого пути*: Times New Roman, 16, полужирный, цвет — Акцент 1, выравнивание — по центру. Отформатируйте основной текст: Times New Roman, 12, черный. Сохраните публикацию.

## 11. Вставка рисунка из Коллекции картинок

1. Удалите рисунки, которые расположены на Web-странице.

2. На панели инструментов **Рисование** щелкните кнопку **Рисунок** а затем кнопку **Коллекция картинок**.

3. В **Области задач** откроется раздел **Коллекция клипов**. В поле **Искать** введите ключевое слово поиска клипа (в данном случае — школа).

4. Щелкните кнопку **Начать**.

5. В **Области задач** откроется список с миниатюрными изображениями найденных по ключевому слову рисунков.

*Примечание.* Если ключевые слова заданы некорректно или с грамматическими ошибками, в **Области задач** появится надпись Клипы не найдены.

6. Щелкните нужный рисунок (в данном случае в качестве логотипа был выбран рисунок Академическая шапочка). Рисунок появится на странице.

## **12. Перемещение и изменение размеров рисунка**

1. Щелкните вставленный рисунок левой кнопкой мыши, по периметру появятся восемь маркеров.

2. Наведите указатель мыши на рисунок, когда указатель примет вид, нажмите левую кнопку мыши и, не отпуская ее, переместите рисунок в левый верхний угол страницы. Отпустите кнопку мыши.

3. Для изменения размеров рисунка подведите указатель мыши к угловому маркеру, нажмите левую кнопку мыши и перетащите указатель в нужном направлении.

4. Сохраните сайт как публикацию.

### ***Пример 6***

Разместите данный рисунок на всех страницах сайта как логотип.


1. В **Области задач** откройте раздел **Коллекция клипов**. В поле **Искать** введите ключевое слова лист, после загрузки миниатюрных изображений выберите нужный рисунок (листок) и разместите его на странице сайта согласно образцу.

2. Сохраните сайт как публикацию.

## **13. Вставка рисунков из файла**

Хотя Коллекция клипов содержит множество рисунков, это не означает,

что нужно ограничиваться стандартным набором изображений. При желании можно вставить в публикацию рисунки и фотографии, созданные в графических редакторах, отсканированные или найденные в Интернете.

1. Откройте третью страницу созданного в ходе выполнения упражнений сайта, для этого щелкните кнопку  на панели **Переход по страницам** в нижней части экрана.

2. Для вставки рисунка из файла в меню **Вставка** выберите *Рисунок\Из файла*.

3. В открывшемся диалоговом окне *Добавление рисунка* в раскрывающемся списке *Папка* выберите диск, а затем папку, содержащую файл с необходимым изображением.

4. Выделите файл *ris\_1.jpg* и щелкните кнопку *Вставить*. На странице сайта появится выбранный рисунок.

5. Измените размеры и разместите вставленный рисунок согласно образцу.

### **Пример 7**

1. Откройте третью страницу созданного сайта.

2. Вставьте и разместите еще один рисунок (*ris\_1.jpg*).

3. На четвертой странице разместите еще два рисунка:

. *Web\Рисунки\ris\_3.jpg*;

. *Web\Рисунки\ris\_4.jpg*;

. *Web\Рисунки\ris\_5.gif*.

4. Сохраните публикацию.

## **14. Настройка изображения**

В *Microsoft Publisher* есть возможность изменить настройки изображения, увеличить или уменьшить яркость и контрастность, обрезать или перекрасить рисунок.

1. При выделении рисунка должна появиться панель *Настройка изображения*, если ее нет, в меню *Вид/ Панели инструментов/Настройка изображения* (рис. 16.16).





Рис. 16.16. Окно *Настройка изображения*

2. Сохраните сайт как публикацию.

### **15. Дополнительно (создание гиперссылок)**

Гиперссылка представляет собой элемент страницы, щелчок на котором приводит к перемещению на другую Web-страницу, файл, сайт в Интернете или созданию сообщения на определенный адрес электронной почты. Гиперссылкой можно сделать любой размещенный на странице объект. Текстовые гиперссылки идентифицируются подчеркиванием и цветом. Графические гиперссылки не изменяют внешний вид, но их легко распознать по изменению вида указателя мыши.

### **16. Гиперссылки используются для:**

- перехода к файлу или Web-странице во внутренней сети или в Интернете;
- ссылки на файл или страницу, которые располагаются в данном сайте;
- отправки сообщения электронной почты;

При указании на гиперссылку (текст или картинку) указатель мыши принимает форму руки, показывая, что в этом месте можно сделать щелчок.

### **17. Текстовые гиперссылки**

*Текстовая гиперссылка* — это слово или фраза, для которой указан URL — адрес назначения.

1. Выделите текст (это может быть как слово, так и словосочетание).
2. В меню *Вставка* щелкните команду *Гиперссылка*, откроется диалоговое окно *Добавление гиперссылки*. (рис. 16.17).
3. В списке *Связать с...* щелкните файлом, веб-страницей.
4. В поле *Адрес* введите URL-адрес (например, <http://www.microsoft.com/rus>).

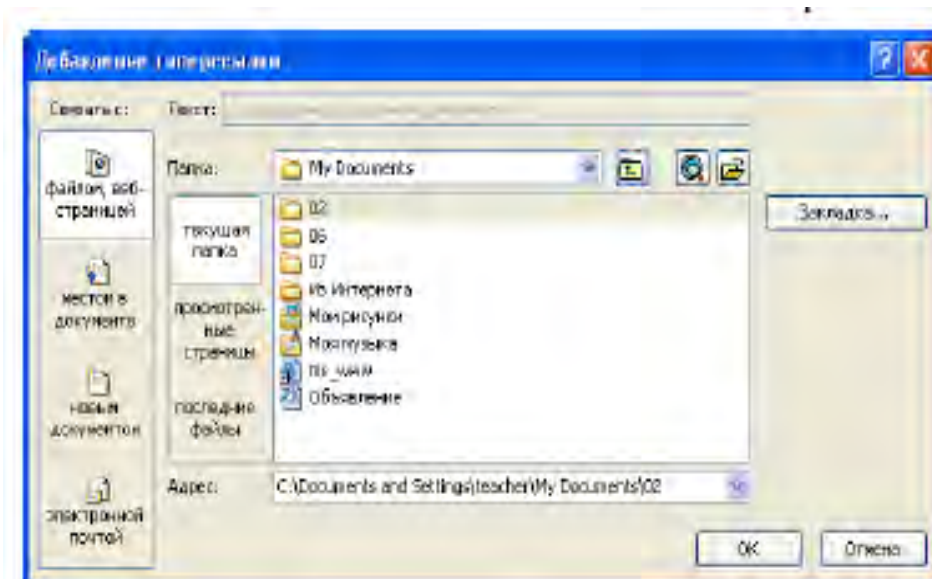


Рис. 16.17. Добавление гиперссылки

5. Щелкните **ОК**.

Слово или фраза, к которым присоединена гиперссылка, будут подчеркнуты и отобразятся другим цветом. Не забудьте, что просмотр работы гиперссылки возможен только в режиме *Предварительного просмотра веб-страницы*.

На данный момент созданный сайт содержит только гиперссылки навигационной панели.

**18. Создание гиперссылки на адрес электронной почты**

1. Запустите *Microsoft Publisher*.
2. Откройте созданный в ходе выполнения упражнений сайт.
3. Откройте четвертую страницу *Послесловие*.
4. Введите адрес электронной почты.
5. Выделите адрес электронной почты.
6. В меню *Вставка* щелкните *Гиперссылка*.
7. В открывшемся диалоговом окне *Добавление гиперссылки* в списке *Связать с* щелкните *электронной почтой*.
8. В открывшемся диалоговом окне в поле *Адрес эл.почты* введите адрес электронной почты.

9. Щелкните ОК. Теперь, когда посетитель Web-сайта щелкнет эту гиперссылку, программа электронной почты автоматически создаст шаблон электронного письма с введенным электронным адресом в строке *Кому*.

10. Сохраните сайт как публикацию.

### **19. Проверка макета**

*Microsoft Publisher* проверяет дизайн Web-страницы и открывает диалоговое окно в случае обнаружения следующих неточностей:

- текстовая рамка, внутри которой нет текста;
- переполнение текстовой рамки;
- рисунок, выходящий за края страницы;
- в случае наложения нескольких полей и преобразовании текста в графику;
- страница не связана гиперссылками с другими;
- слишком много времени требуется для загрузки страницы.

*Для проверки публикации на перечисленные неточности:*

1. В меню *Сервис* щелкните *Проверка макета*.
2. В *Области задач* откроется раздел *Проверка макет*, в списке которого будут перечислены элементы для исправления.

3. Для исправления выявленных программой неисправностей щелкните стрелку около элемента, а затем выполните одно из следующих действий.

а) Для перехода на страницу, где расположен элемент с ошибкой, выберите *Перейти* к этому элементу.

б) Чтобы автоматически исправить ошибку элемента, выберите *Исправить*: автоматически исправлять предложение.

с) Щелкните команду *Больше не выполнять эту проверку* для отключения проверки. Эта команда влияет на все встречающиеся в дальнейшем ошибки такого рода.

д) Щелкните команду *Объяснить*, чтобы открыть раздел справки, в котором более полно объясняется данная ошибка и предлагаются варианты ее исправления.

е) Для проверки орфографии макета в меню *Сервис* выберите *Правописание\Орфография*. Функция *Автозамены* применяется для автоматического обнаружения и исправления опечаток, орфографических и грамматических ошибок

## **20. Предварительный просмотр Web-страниц**

Созданные Web-страницы следует просмотреть в Web-браузере, чтобы убедиться, что они отображаются так, как было задумано, и проверить функционирование связей. В меню *Файл* выберите *Предварительный просмотр веб-страницы*. Web-узел будет открыт в браузере.

### **Необходимо убедиться в следующем:**

- текст виден и хорошо читается;
- графика на каждой странице видна и быстро загружается;
- гиперссылки действуют и ведут к соответствующим ресурсам;
- узел открывается и правильно функционирует на различных компьютерах;
- фоновый звук, если он есть, воспроизводится.

## **21. Размещение сайта в Интернете с помощью Microsoft Publisher**

Если вы собираетесь самостоятельно разместить свой сайт в Интернете, вам понадобится адрес для доступа к сайту, который можно получить, обратившись в соответствующую интернет-службу.

1. Откройте сайт, сохраненный как публикация (с расширением **.pub**).
2. В меню *Файл* щелкните *Сохранить как веб-страницу*.
3. В открывшемся диалоговом окне *Сохранить как веб-страницу* раскройте список *Папка* и выберите *Добавить/изменить адреса FTP* (рис 16.18).

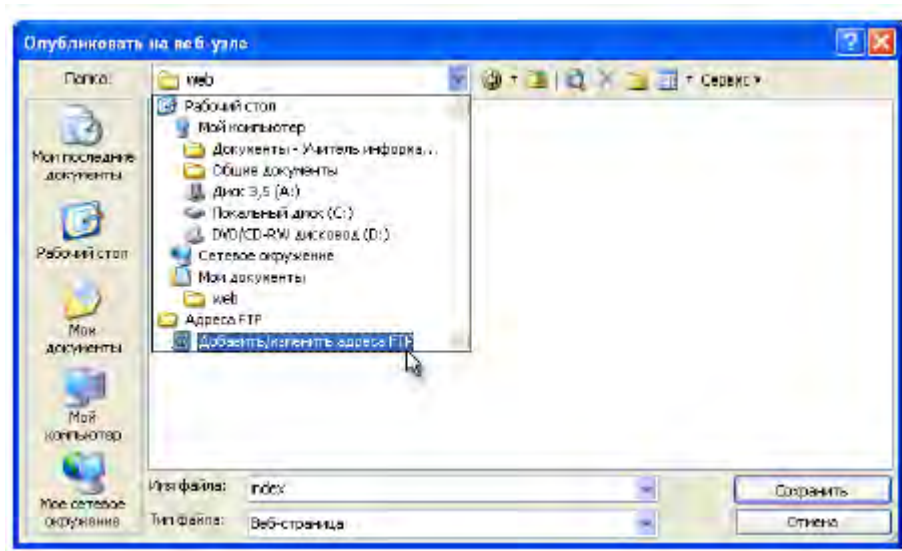


Рис. 16.18. Добавление адреса *FTP*

4. В открывшемся диалоговом окне в поле *Имя узла FTP* введите адрес FTP-сервера, на котором вам предоставили место (допустим, место предоставлено на бесплатном сервере **narod.ru**, а зарегистрированное имя пользователя — **sait**).

*Примечание.* В России есть несколько *Web-серверов*, предоставляющих бесплатные услуги по размещению сайтов, например <http://narod.ru>, <http://boom.ru>.

5. Установите флажок *Под именем* и в текстовое поле введите имя (рис. 16.19), под которым вы зарегистрировались на сервере.

6. В поле *Пароль* введите свой пароль.

7. Щелкните кнопку *Добавить*.

8. Щелкните *ОК*.

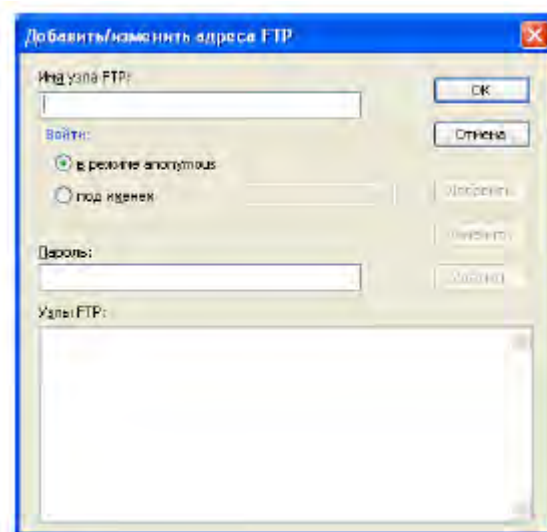


Рис. 16.19. Ввод имени узла

9. Вы вернулись к диалоговому окну *Сохранить как веб-страницу*. Вы можете переименовать сайт, введя в текстовое поле *Имя файла* новое имя.

Дважды щелкните левой кнопкой мыши новое местоположение сайта (в данном случае — строку ftp://sait.narod.ru).

10. Щелкните *ОК*.

11. После соединения с FTP-сервером и появления в текстовом поле *Папка* его имени щелкните кнопку *Сохранить*.

12. Откроется диалоговое окно *Сохранение*. Исчезновение этого окна будет означать, что загрузка всех рисунков, html-и других файлов из состава сайта завершена.

## ***16.2. Контрольные вопросы***

1. Назначение, основные возможности и состав *Micrisift Publisher* / *Элементы интерфейса, настройка рабочего места*.

2. Способы оздания Web-сайта *Micrisift Publisher*. Использование Мастеров Содержания и шаблонов Web-сайта.

3. Местонахождение и структура папок *Micrisift Publisher* для сохранения Web-сайта (по умолчанию).

4. Способы создания, добавления, удаления Web-страниц.

### 16.3. Контрольные задания

Размещение текста на других Web-страницах.

*Создание второй страницы*

1. Выделите и отформатируйте заголовок Мы учимся: Times New Roman, 24, полужирный, цвет — Акцент 1, выравнивание — по центру.
2. Введите подзаголовок Профили факультета: Times New Roman, 16, полужирный, цвет — Акцент 1, выравнивание — по центру.
3. Введите текст, приведенный на рис. 16.20.

<i>Естественнонаучное направление</i> •... •...	<i>Экономико-математическое</i> •... •...
---	---

Рис. 16.20. Примерное оформление текста

4. Отформатируйте текст: Times New Roman, 12, черный.
5. Введите и отформатируйте текст на третьей и четвертой страницах сайта согласно образцу. Сохраните сайт как публикацию.

## ТЕХНОЛОГИЯ ПРОЕКТИРОВАНИЯ WEB-СТРАНИЦ СРЕДСТВАМИ ОФИСНЫХ ПРИЛОЖЕНИЙ

**Цель работы:** создание Web-страницы в *MS Word*, преобразование документов в *MS Word* формат *HTML*. Редактирование и форматирование Web-страницы с помощью *MS Word*, гиперссылок. Создание закладок. Преобразование таблиц и диаграмм *MS Excel* в Web-страницы. Преобразование объектов баз данных *MS Access* в Web-страницы.

### Методические указания

#### 1. Создание Web-страниц средствами ТП *MS Word*

**Пример 1.** Создайте начальную страницу средствами *MS Word* Web-сайта. Сохраните ее под именем *index.html* в папке *site\html\rus*. Просмотрите начальную страницу Web-сайта в браузере *MS Internet Explorer*.

#### Порядок выполнения

1. Загрузите текстовый процессор *MS Word*, выполнив команду *Пуск/Программы/MS Word*.
2. Создайте начальную страницу Web-сайта, разместив текст, приведенный на рис. 17.1.

#### **Проектирование и разработка Web-сайта**

Создание Web-страниц средствами *MS Office*:

1. Создание Web-страниц с помощью *MS Word*
  - Преобразование таблиц и диаграмм *MS Excel* в Web-страницы
  - Преобразование объектов баз данных *MS Access* в Web-страницы
2. Создание интерактивных презентаций с помощью *MS PowerPoint*
3. Создание Web-страниц средствами языка *HTML*
4. Создание Web-страниц средствами *FrontPage*

Рис. 17.1. Текст, размещенный на начальной странице Web-сайта



3. Установите цвет заголовка начальной страницы – темно-красный, размер шрифта заголовка – 14.

4. Определите цвет фона Web-страницы: *Формат/Фон/ Способы заливки/градиентная/ заготовка/пергамент/ тип штриховки/из угла.*

5. Сохраните начальную страницу Web-сайта: *Файл / Сохранить / кнопка Обзор/* откройте папку *site/html/rus/index.html /OK.*

6. Закройте сохраненный файл *index.html.*

7. Просмотрите начальную страницу в браузере MS Internet Explorer: *Файл/Просмотр Web- страницы* или *Файл/Открыть/* кнопка *Обзор/* откройте папку *site/html/rus/index.html/* выберите *Тип файлов: Файлы HTML/* выберите файл *index.html /* кнопка *Открыть.*

### **17.1. Типовое задание**

**Пример 1.** Создайте новый документ и сохраните документ в формате *MS Word* под именем *word.doc* в папке *site\other*. Преобразуйте документ *MS Word* в формат HTML и сохраните HTML-документ под именем *word.html* в папке *site/html/rus*. Просмотрите Web-страницу в браузере MS Internet Explorer.

#### **Указания**

1. Создайте новый документ MS Word, разместив следующий текст:

#### **Создание Web – страниц с помощью MS Word**

MS Word позволяет:

- *создавать страницу с «чистого» листа;*
- *конвертировать уже имеющийся документ MS Word в Web страницу;*
- *редактировать и форматировать Web – страницу как обычный документ MS Word.*

2. Сохраните документ в формате *MS Word* под именем *word.doc* в папке *site\other*.

3. Установите цвет заголовка начальной страницы – коричневый, размер шрифта заголовка – 14.

4. Определите цвет фона Web-страницы: *Формат/Фон/Способы*

заливки/текстура/ полотно.

5. Сохраните документ в формате *HTML* в папке *site/html/rus/word.html/OK*.

6. Закройте сохраненный файл *word.html*.

7. Просмотрите Web-страницу в браузере MS Internet Explorer: *Файл/Просмотр Web-страницы* или *Файл/Открыть/* кнопка *Обзор/* откройте папку *site/html/rus/word.html /* выберите *Тип файлов: Файлы HTML/* выберите файл *word.html /* кнопка *Открыть*.

**Пример 2.** Отредактировать введенные данные начальной страницы, вставить бегущую строку, графические объекты. Результат работы сохранить в файле *word.html* и просмотреть в браузере *Internet Explorer*.

#### **Указания**

1. Вставьте после заголовка Web-страницы графический рисунок: *Вставка/ рисунок из файла/* откройте папку *C:\Programms Files\Microsoft Office\*

*Clipart\Lines* выберите файл *Осенние листья/* кнопка *Добавить* или осуществите поиск графический рисунок, расположенных на компьютере командой *Пуск/Поиск/ Файлы и папки/ \*.gif* (или *\*.jpg, \*.tif*) и выберите графические рисунки, соответствующие тематике начальной страницы.


2. В конце документа вставьте горизонтальную линию, после которой добавьте таблицу (число столбцов – 2, число строк – 1), в первую ячейку которой введите: *К началу документа*, во вторую – *К начальной странице*. Выровняйте содержимое ячеек по центру.

3. Сохраните начальную страницу Web-сайта: *Файл/Сохранить /site/html/rus/word.html/* кнопка *Сохранить/* в поле *Тип файла* щелкните по кнопке раскрывающегося списка и выберите: *Web-страница/OK*.

4. В конце документа вставьте бегущую строку, щелкнув по пиктограмме *Бегущая строка* на панели инструментов *Вид/ Панели инструментов/ Веб-компоненты*.

5. В диалоговом окне *Бегущая строка* введите текст бегущей строки, указав в ней свою фамилию, инициалы и номер телефона. Для текста бегущей

строки установите: *Движение* – *Насквозь*, *Цвет фона* – *Белый*, *Направление* – *Справа налево*. После установки параметров нажмите кнопку *ОК*.

6. Выполните переход в режим *Конструктора* , расположенной на панели инструментов *Веб-компоненты*.

7. Установите ширину бегущей строки равной *16см*, высоту – *0,60см*, выполнив команду *Формат/ Элемент управления/* вкладка *Размер*.

8. После установки необходимых параметров бегущей строки отмените режим *Конструктора*, щелкнув по пиктограмме *Выход из режима Конструктора*.

9. Закройте отредактированный файл *word.html*.

10. Открыв браузер *MS Internet Explorer*, просмотрите результат работы в окне браузера, задав команду *Вид/ Обновить*.

### **Пример 3.**

Продолжить редактирование Web-страницы, добавив ссылки:

- на файл *site/html/rus/word.doc*
- на начальную страницу Web-сайта, используя текст « *К начальной странице*»
- на заголовок Web-страницы, используя в качестве ссылки текст « *к началу документа*»,
- на документ *site/html/rus/word.html*.

Просмотрите измененную Web-страницу в *Internet Explorer* и проверьте работоспособность ссылок.

### **Указания**

1. С помощью панели задач *Windows* переключитесь в *MS Word*.
2. Вернувшись в *MS Word*, откройте файл ***site/html/rus/word.html***
3. Вставьте ссылку на файл, используя текст « *обычный документ Word*». Выделите текст/ *Вставка/гиперссылка/* в поле *Связать с .../новым документом* кнопка *Обзор.../* выберите файл *word.doc/ ОК*.
4. Сохраните файл: *Файл / Сохранить*.

5. Просмотрите измененную Web-страницу в *Internet Explorer* и проверьте работоспособность ссылки: с помощью панели задач Windows переключитесь в *Internet Explorer* / кнопка *Обновить*/щелчок мышью по ссылке/ вернитесь к предыдущему документу с помощью кнопки *Назад*.

6. С помощью панели задач Windows переключитесь в MS Word.

7. Вставьте в документ ссылку на начальную страницу Web-сайта, используя текст «*К начальной странице*», выделив текст *К начальной странице* /*Вставка/Гиперссылка* в поле *Связать с ../файлом, веб – страницей/* кнопка *Обзор.../* откройте папку *site/html/rus/index.html /OK*.

8. Сохраните Web- страницу: *Файл / Сохранить*.

9. Просмотрите измененную Web-страницу в *Internet Explorer*. Проверьте и работоспособность на начальную страницу Web-сайта : с помощью панели задач Windows переключитесь в *Internet Explorer* / кнопка *Обновить*/щелчок мышью по ссылке *К начальной странице* / вернитесь к предыдущему документу с помощью кнопки *Назад*

10. С помощью панели задач Windows переключитесь в MS Word.

11. Вставьте закладку на заголовок Web-страницы : выделите заголовок документа/*Вставка/ Закладка.../* в поле *Имя закладки* введите: *Заголовок/* кнопка *Добавить*.

12. Вставьте ссылку на заголовок Web-страницы, используя в качестве ссылки текст «*к началу документа*»: выделите текст *к началу документа* /*Вставка/Гиперссылка* в поле *Связать с ../местом в документе/* выберите закладку *Заголовок /OK*.

13. Сохраните Web-страницу: *Файл / Сохранить*.

14. Просмотрите измененную Web-страницу в *Internet Explorer*. Проверьте и работоспособность на начальную страницу Web-сайта : с помощью панели задач Windows переключитесь в *Internet Explorer* / кнопка *Обновить*/щелчок мышью по ссылке *К началу документа/* вернитесь к предыдущему документу с помощью кнопки *Назад*.

15. С помощью панели задач Windows переключитесь в MS Word.

16. Завершите работу с Web-страницей: *Файл/Заккрыть*.

17. Аналогично создайте ссылку на документ *site/html/rus/word.html*., используя в качестве ссылки текст «*Создание Web-страниц с помощью MS Word*»

## 2. Создание Web – страниц средствами ТП MS Excel

**Пример 1.** Представить результат работы фирмы в виде Web-страницы, в которой представлена таблица (см. Образец 1) и диаграмма. Сохраните документ в формате MS Excel под именем *excel.xls* в папке *site\other*. Преобразуйте документ MS Excel в формат HTML, сохранив под именем *excel.htm* и диаграмму под именем *Diagr.htm* в папке *site/html\rus*. Завершите работу с MS Excel. Просмотрите Web-страницу в браузере Internet Explorer

### Указания:


1. Загрузите MS Excel *Пуск/Программы/ MS Excel*.
2. Создайте таблицу, приведенную на (рис. 17.2), соблюдая показанное оформление.

Учет товаров					
Наименование товара	Количество товара	Цена товара	НДС товара	Итого цена с НДС	В том числе НДС
Продукты	203	9 100 000р.	1 820 000р.	10 920 000р.	2 216 760 000р.
Бытовая техника	102	6 000 000р.	1 200 000р.	7 200 000р.	734 400 000р.
Сантехника	45	72 000 000р.	14 400 000р.	86 400 000р.	3 888 000 000р.
Радиотовары	56	15 000 000р.	3 000 000р.	18 000 000р.	1 008 000 000р.
Продукты	32	5 000 000р.	1 000 000р.	6 000 000р.	192 000 000р.

Рис. 17.2. Схема учета товаров

### Примечания:

- Поле *НДС* рассчитывается как 20% от поля *Цена*.
- Поле *Итого цена с НДС* рассчитывается как сумма полей *Цена* и *НДС*.
- Поле *В том числе НДС* рассчитывается как произведение полей *Количество* и *НДС*.

3. С помощью мастера диаграмм по данным таблицы постройте диаграмму (выделите данные/ ). Тип диаграммы – точечная со значениями, соединенными сглаживающими линиями. В качестве названия диаграммы введите: *Учет товаров*. Отформатируйте диаграмму.

4. Сохраните документ в формате MS Excel: *Файл/ Сохранить/* откройте папку *site\other/* в поле *Имя файла* введите: *excel.xls / Сохранить*.

5. Для представления данных таблицы в виде Web-страницы выделите таблицу с полученными данными и преобразуйте ее в формат **html**, выполнив команду *Файл/ Сохранить как...*

6. В диалоговом окне *Сохранение документа* откройте свою папку, введите имя файла *excel.xls*, выберите тип файла *Web-страница*, установите переключатель в положение *Выделенное* (будет указан диапазон выделенных ячеек) и щелкните по кнопке *Опубликовать*.

7. В окне *Публикация Web- страницы* введите заголовок для *Web-страницы*, например *Перечень товаров*, щелкнув по кнопке *Изменить*.

8. Для публикации подготовленной таблицы нажмите кнопку *Опубликовать*.

9. Завершите работу с табличным процессором MS Excel.

10. Просмотрите созданную Web-страницу в браузере Internet Explorer.

**Пример 2.** Представить результат работы фирмы в виде Web-страницы, которая содержит диаграмму, представляющую графическое распределение НДС товара. Файл сохранить под именем *Diagr.htm* . Результат работы просмотрите в браузере Internet Explorer.

**Указание.** Создайте Web-страницу, содержащую диаграмму, по описанному ранее алгоритму

**Пример 3.** Открыть созданный ранее файл *index.html* в MS Word. Выделить текст *Преобразование таблиц и диаграмм Ms Excel в Web - страницы* и вставить гиперссылки на файлы *excel.xls* и *Diagr.htm*. Результат работы просмотрите в браузере Internet Explorer.

**Пример 4.** Отформатировать файл *excel.xls* и добавить в него ссылку на

начальную страницу *index.html*. Просмотреть измененную Web-страницу в браузере *Internet Explorer*.

**Пример 5.** Открыть файл *Diagr.htm* и добавить в него ссылку на начальную страницу *index.html*. Просмотреть измененную Web-страницу в браузере *Internet Explorer*.

### **3. Создание Web-страниц средствами СУБД MS Access**

**Пример 1.** Создать статическую Web-страницу для запроса *Информация о студентах*, выполнив экспорт данных в файл формата *.html*. Просмотрите созданную Web-страницу в браузере *Internet Explorer*

#### **Указания:**

1. Откройте базу данных *Студенты*. Выделите запрос *Информация о студентах* *Информация о студентах*.

2. Выполните команду *Файл/ Экспорт*. В диалоговом окне *Экспорт* в поле *Тип* файла из раскрывающегося списка выберите пункт *Документы HTML*, сохранив под именем *Страница о студентах*, нажав кнопку *Экспорт*.

3. Просмотрите Web-страницу в браузере *MS Internet Explorer*: *Файл/Открыть/* кнопка *Обзор/* откройте папку *site/html/rus/ Страница о студентах.html /* кнопка *Открыть*.

**Пример 2.** Создать страницу доступа к данным для таблицы *Сведения о дисциплинах* в режиме *Мастера страниц*

#### **Указания:**

1. В окне базы данных откройте объект *Страницы*.

2. Загрузите режим *Мастера создания страниц*, дважды щелкнув по строке *Создание страницы доступа к данным с помощью мастера*.

3. Из раскрывающегося списка в поле *Таблицы и запросы* выберите таблицу *Дисциплины*. Поместите поля таблицы *Дисциплины* в область *Выбранные поля*, щелкните по кнопке *Далее*.

4. Установите:

- сортировку по возрастанию;
- заголовок страницы – *Информация о дисциплинах*;


- переключатель *Изменить макет страницы*;
- флажок *Применить тему к странице* и щелкните *Готово*.

5. Просмотрите полученную страницу командой *Вид/ Просмотр страницы*.

6. Сохраните страницу в своей папке.

**Пример 3.** Создать страницу доступа к данным для таблицы в режиме *Конструктора* для запроса *Средний балл*.

**Указания:**

1. В окне базы данных откройте объект *Страницы*
2. Для создания страницы в режиме *Конструктора* щелкните по кнопке *Создать*, расположенной в окне базы данных.
3. В появившемся окне *Новая страница доступа к данным* выберите режим *Конструктор* и в качестве источника данных укажите запрос *Средний балл*.
4. В открывшемся режиме *Конструктора* создания страницы в область раздела *Название страницы* введите название страницы *Успеваемость студентов*.
5. Удалите область строки для ввода дополнительного текста, установив курсор под область названия страницы, нажав клавишу *Backspace*.
6. Откройте список полей, выбрав команду *Вид/Список полей*.
7. Разместите поля запроса *Средний балл* в главный раздел страницы.
8. Отформатируйте заголовок, элементы управления с помощью панели управления *Выравнивание и изменение размеров*.
9. Задайте тему для страницы командой *Формат/ Тема / радиус*.
10. Поместите рисунок командой *Вставка/Рисунок* и бегущую строку, выбрав на панели инструментов кнопку , в которую введите информацию о разработчике странице.
11. Просмотрите созданную страницу доступа к данным.

**Пример 4.** Создать страницу доступа к данным для таблицы в режиме *Конструктора* для таблицы *Информация*, содержащей гипертекстовые ссылки



на созданные ранее Web-страницы.

**Указания:**

1. В окне базы данных в списке *Объекты* выберите *Таблицы* и откройте данный объект.

2. Создайте структуру таблицы *Информация*, щелкнув по кнопке *Создать*. В появившемся диалоговом окне выберите режим *Конструктор*.

3. В бланке режима *Конструктора* в столбец *Имя поля* введите в нужной последовательности имена полей таблицы *Информация* установите тип данных см. табл. 17.1.

Таблица 17.1

Имя поля	Тип данных	Свойства таблицы	
		Подпись	Индексированное поле
Ссылки	Гиперссылка	Ссылки на объекты	Нет
Комментарий	Текстовый	Комментарий	Нет

4. Сохраните структуру таблицы, перейдите в режим таблицы (*Вид/Таблица*), наполните таблицу данными.

5. В поле *Ссылки* укажите адреса созданных Web-страниц, выполнив команду *Вставка/Гиперссылка*

6. Поле *Комментарий* содержит пояснение, какую страницу доступа к данным содержит адрес гиперссылки.

7. Сохраните таблицу *Информация*

**17.2. Контрольные вопросы**

1. Опишите алгоритм создания Web-страниц средствами текстового процессора MS Word.

2. Опишите алгоритм создания Web-страниц средствами табличного процессора MS Excel.

3. Опишите алгоритм создания Web – страниц средствами СУБД MS Access.

### **17.3. Контрольное задание**

#### **Задание**

- Средствами *MS Word*, *MS Excel*, *СУБД MS Access* разработать и создать структуру web-сайта **по вариантам**, используя типовые задания.

- Средствами *MS Excel* разработать Web- сайт, в котором представить результаты решения экономической задачи по **вариантам**.

- Средствами *СУБД MS Access* спроектировать следующие базы данных по **вариантам**, в которых необходимо:

– создать и наполнить таблицы данными, установив связи между ними;

– сконструировать запросы выбора, с вычисляемым полем;

– создать страницы доступа данных в режиме **Мастера страниц**;

– создать страницы данных в режиме **Конструктора**:

1. Организации (с полями *Номер организации, Название, Адрес, Телефон*);

2. Сотрудники (с полями *Фамилия, Должность, Адрес, Телефон*);

3. Ведомость удержаний (с полями *Фамилия, Аванс, Подоходный налог,*

*Прочие удержания*);

4. Продукция (с полями *Наименование, Сорт, Поставщик, Цена*);

5. Изделия (с полями *Изделие, Наименование, Цена, План выпуска,*

*Остатки на конец года*);

6. Спецодежда (с полями *Наименование, Сорт, Стоимость*).

Оцениваются содержание и дизайн фирмы.

## ЛИТЕРАТУРА

1. Белунцов, В. Новейший самоучитель по созданию Web – страниц / В. Белунцов – М.: ДЕСС КОМ, 2000. – 321 с.
2. Воген, Т. Мультимедиа: Практическое руководство / Т. Воген. – Минск: Асар, 1997. – 166 с.
3. Гончаров, А. HTML в примерах / А. Гончаров – СПб: Питер, 1997. – 201 с.
4. Интернет-технологии для пользователя: в 2 ч. Проектирование и разработка Web-узла: учебное пособие / И.В. Брезгунова [и др.]. – Минск: РИВШ, 2001. – 167 с.
5. Использование Internet / Дж. Хонникат [и др.]. – СПб: Издательский дом «Вильямс», 1998. – 150 с.
6. Петюшкин, А.В. Использование HTML / А.В. Петюшкин. – СПб: bvh-Петербург, 2003. – 78 с.
7. Попов, В.Б. Основы компьютерных технологий / В.Б. Попов. – М.: Финансы и статистика, 2002. – 168 с.
8. Пупцев, А.Е. Технология построения Web – страниц с помощью языка разметки гипертекста HTML / А.Е. Пупцев. – Минск: РИВШ, 2002. – 28 с.

## ПРИЛОЖЕНИЯ

### Приложение А

#### Структура отчета

Отчет о выполненной лабораторной работе должен включать:

1. Титульный лист.
2. Постановку задачи.
3. Описание хода выполнения работы (последовательность выполняемых действий, команд, тексты HTML-файлов и т.д.).
4. Краткие ответы на контрольные вопросы.
5. Выводы.
6. Список использованной литературы.

**Основные терминологические понятия**

**Язык HTML** (*HyperText Markup Language*) – язык разметки гипертекстовых документов.

**Гипертекст** – это метод представления текста, изображений, звука и видео, связанных друг с другом произвольной ассоциативной связью.

**Гиперссылка** – область документа, которая указывает на объекты различной природы: файлы, картинки, тексты, графику и др.

**Тег (дескриптор)** – стартовый или конечный элемент разметки, имеющий вид: < тег > ... < /тег >

**Атрибут** – параметр (свойство элементов разметки), имеющее стандартное имя с определенным набором значений.

**Ссылка** – это запись, связывающая между собой части документа, библиографические записи, классификационные деления или предметные рубрики.

**Код HTML** – набор конструкций языка HTML, т. е. совокупность тегов и атрибутов.

**HTML файл (страница)** – документ, построенный на основе языка HTML. Файлы HTML имеют расширение .htm или .html.

**Web-страница** – документ (файл), построенный в виде гипертекста и размещенный в World Wide Web.

**Сайт** – набор Web-страниц, принадлежащих, как правило, одному владельцу или группе владельцев, объединенных общими целями и задачами.

**Браузер** – программа для просмотра Web-страниц.

**HTTP** (*HyperText Transfer Protocol*) – протокол передачи гипертекста.

**URL** (*Uniform Resource Locator*) – универсальный указатель ресурса.

**FTP** (*File Transfer Protocol*) – протокол передачи файлов.

**TCP/IP** (*Transmission Control Protocol/Internet Protocol*) – протокол управления передачей/протокол Интернета.

**URL** (*Uniform Resource Locator*) – адрес некоторого объекта в Internet.

## Краткая справка по использованию языка HTML

Действие	Реализация
1	2
<b>Создание нового HTML-документа</b>	
Начало/конец документа	Теги <b>&lt;HTML&gt;...&lt;/HTML&gt;</b>
Служебная область документа	Содержится между тегами <b>&lt;HEAD&gt;...&lt;/HEAD&gt;</b>
Заголовок документа, который появляется в строке заголовка окна браузера	Содержится между тегами <b>&lt;TITLE&gt;...&lt;/TITLE&gt;</b> в служебной области документа
Служебная информация (кодировка, автор, ключевые слова и др.)	Атрибуты тега <b>&lt;META&gt;</b> , находящегося в служебной области документа
Основная область документа	Располагается между тегами <b>&lt;BODY&gt;...&lt;/BODY &gt;</b>
<b>Разбиение текста на строки и абзацы</b>	
Конец строки	Тег <b>&lt;BR&gt;</b>
Начало/конец абзаца	Теги <b>&lt;P&gt;...&lt;/P&gt;</b>
Горизонтальная линия	Тег <b>&lt;HR&gt;</b>
Предварительно отформатированный текст (например, разбитый на строки или с "лишними" пробелами)	Теги <b>&lt;PRE&gt;...&lt;/PRE&gt;</b>
<b>Создание заголовков</b>	
Заголовки от первого до шестого уровня	Соответственно теги <b>&lt;H1&gt;...&lt;/H1&gt;</b> , <b>&lt;H2&gt;...&lt;/H2&gt;</b> , ..., <b>&lt;H6&gt;...&lt;/H6&gt;</b>
<b>Выравнивание текста</b>	
Выравнивание текста по левому краю, по правому краю, по центру	1. Соответственно теги <b>&lt;LEFT&gt;...&lt;/LEFT&gt;</b> , <b>&lt;RIGHT&gt;...&lt;/RIGHT&gt;</b> , <b>&lt;CENTER&gt;...&lt;/CENTER&gt;</b> или: 2 Атрибуту <b>ALIGN</b> тега заголовка или абзаца присвоить соответственно значения <b>LEFT</b> , <b>RIGHT</b> , <b>CENTER</b>

1	2
<b>Форматирование шрифтов</b>	
Полужирный шрифт	Теги <B>...</B>
Курсив	Теги <EM>...</EM>
Подчеркнутый шрифт	Теги <i>...</i>
Зачеркнутый шрифт	Теги <S>...</S>
Шрифт пишущей машинки	Теги <TT>...</TT>
Нижние индексы	Теги <SUB>...</SUB>
Верхние индексы	Теги <SUP>...</SUP>
Мигающий текст	Теги <BLINK>...</BLINK> Не отображается <i>Internet Explorer</i>
<b>Размер шрифта</b>	
Размер шрифта по умолчанию	Тег <BASEFONT SIZE - (число от 1 до 7)>. Если не задано иное значение, по умолчанию используется шрифт размера 3
Изменение размеров шрифта отдельных фрагментов текста	Тег <FONT SIZE=>...</FONT> Примеры. 1. <FONT SIZE=5>.. „</FONT> Для фрагмента текста между тегами установлен шрифт размера 5. 2. <FONT SIZE=+2>., „</FONT> Для фрагмента текста между тегами установлен шрифт размера на 2 пункта больше, чем размер текущего шрифта
<b>Цвет шрифта</b>	
Цвет фона и шрифта для всего документа	Атрибуты <b>BG</b> <b>COLOR</b> и <b>TEXT</b> тега <BODY>. Цвет задается шестизначным шестнадцатеричным числом со структурой <b>RGB (Red-Green-Blue</b> , красный-зеленый-синий). Каждые 2 цифры обозначают присутствие соответствующего компонента в результирующем цвете. <b>Пример.</b> <BODY <b>BG</b> <b>COLOR</b> =#0GFF©0> Нет красного цвета, максимум зеленого цвета, нет синего цвета. Цвет фона – зеленый. Можно употреблять и названия цветов (см. пример 7)

1	2
Цвет шрифта для фрагмента текста	Атрибут <b>COLOR</b> тега <b>&lt;FONT&gt;</b>
<b>Гарнитура шрифта</b>	
Гарнитура шрифта для фрагмента текста	Атрибут <b>FACE</b> тега <b>&lt;FONT&gt;</b>
<b>Создание списков</b>	
Нумерованный список	Тег <b>&lt;OL&gt;...&lt;/OL&gt;</b>
Элемент нумерованного списка	Тег <b>&lt;LI&gt;</b>
Маркированный список	Тег <b>&lt;UL&gt;...&lt;/UL&gt;</b>
Элемент маркированного списка	Тег <b>&lt;LI&gt;</b>
Список определений	Тег <b>&lt;DL&gt;...&lt;/DL&gt;</b>
Термин в списке определений	Тег <b>&lt;DT&gt;</b>
Определение термина в списке определений	Тег <b>&lt;DD&gt;</b>
<b>Вставка и форматирование графических изображений</b>	
Вставка графических изображений	Тег <b>&lt;IMG SRC="URL или относительный адрес файла"&gt;</b>
Выравнивание изображения по левому, правому краям и по центру (текст обтекает рисунок)	Атрибуту <b>ALIGN</b> тега. <b>&lt;IMG&gt;</b> соответственно присвоить значения <b>LEFT, RIGHT, CENTER</b>
Отмена обтекания рисунка текстом и переход текста на новую строку там, где свободно правое поле, левое поле, оба поля.	Тег <b>&lt;BR&gt;</b> с атрибутом <b>CLEAR</b> ; последнему соответственно присвоить значения <b>RIGHT, LEFT, ALL</b>
Выравнивание текста относительно изображения: по верхнему краю, по нижнему краю, посередине.	Атрибуту <b>ALIGN</b> тега <b>&lt;IMG&gt;</b> соответственно присвоить значения <b>TOP, BOTTOM, MIDDLE</b>
Размер пустых полей над/под изображением и слева/справа от него в пикселах	Атрибуты <b>VSPACE</b> и <b>HSPACE</b> тега <b>&lt;IMG&gt;</b>
Вставка графических изображений низким разрешением (для увеличения скорости передачи информации)	
Задание вертикальных и горизонтальных размеров изображения в пикселах	Атрибуты <b>HEIGHT</b> и <b>WIDTH</b> тега <b>&lt;IMG&gt;</b> Не отображается в Netscape Navigator
Текст вместо изображения (если графика не отображается или браузер не поддерживает графику)	Атрибут <b>ALT</b> тега <b>&lt;IMG&gt;</b>



1	2
<b>Создание и форматирование таблиц</b>	
Создание таблицы	Теги <b>&lt;TABLE&gt;...&lt;/TABLE&gt;</b>
Выделение каждой ячейки и таблицы в целом рамкой; задание ширины рамки в пикселах	Атрибут <b>BORDER</b> тега <b>&lt;TABLE&gt;</b>
Имя таблицы	Вводится между тегами <b>&lt;CAPTION&gt;...&lt;/CAPTION&gt;</b>
Расположение имени над или под таблицей	Атрибуту <b>ALIGN</b> тега <b>&lt;CAPTION&gt;</b> присвоить значения <b>TOP, BOTTOM</b>
Строка таблицы	Вводится между тегами <b>&lt;TR&gt;...&lt;/TR&gt;</b>
Заголовки столбцов	Вводится между тегами <b>&lt;TH&gt;...&lt;/TH&gt;</b>
Данные для каждой ячейки	Вводится между тегами <b>&lt;TD&gt;...&lt;/TD&gt;</b>
Объединение нескольких соседних ячеек по горизонтали или по вертикали	Атрибуты <b>COLSPAN</b> или <b>ROWSPAN</b> тега <b>&lt;TB&gt;</b> или <b>&lt;TH&gt;</b>
Горизонтальное выравнивание соответственно в строках, заголовках столбцов или ячейках по левому краю, по правому краю, посередине	Атрибуту <b>ALIGN</b> тегов <b>&lt;TR&gt;</b> , <b>&lt;TH&gt;</b> , <b>&lt;TD&gt;</b> соответственно присвоить значения <b>LEFT, RIGHT, CENTER</b>
Вертикальное выравнивание в заголовках столбцов или ячейках по верхнему краю, по нижнему краю, посередине	Атрибуту <b>VALIGN</b> тегов <b>&lt;TH&gt;</b> и <b>&lt;TD&gt;</b> соответственно присвоить значения <b>TOP, BOTTOM, MIDDLE</b>
Цвет таблицы, строк или ячеек	Атрибут <b>BGCOLOR</b> соответственно тегов <b>&lt;TABLE&gt;</b> , <b>&lt;TR&gt;</b> , <b>&lt;TD&gt;</b>
Ширина рамки вокруг каждой ячейки в пикселах	Атрибут <b>CELLSPACING</b> тега <b>&lt;TABLE&gt;</b>
Ширина таблицы (ячейки) в процентах от ширины экрана (таблицы) или в пикселах	Атрибут <b>WIDTH</b> тега <b>&lt;TABLE&gt;</b> или <b>&lt;TD&gt;</b>
Величина пустых полей между содержимым ячейки и ее рамкой в пикселах	Атрибут <b>CELLPADDING</b> тега <b>&lt;TABLE&gt;</b>
Цвет всей рамки (рамка выглядит плоской)	Атрибут <b>BORDERCOLOR</b> тега <b>&lt;TABLE&gt;</b> (в тег <b>&lt;TABLE&gt;</b> должен быть помещен атрибут <b>BORDER</b> )

1	2
Цвет светлой части рамки	Атрибут <b>BORDERCOLORLIGHT</b> тега <b>&lt;TABLE&gt;</b> (в тег <b>&lt;TABLE&gt;</b> должен быть помещен атрибут <b>BORDER</b> ) Не отображается в Netscape Navigator
Цвет темной части рамки	Атрибут <b>BORDERCOLORDARK</b> тега <b>&lt;TABLE&gt;</b> (в тег <b>&lt;TABLE&gt;</b> должен быть помещен атрибут <b>BORDER</b> ) Не отображается в Netscape Navigator
Дизайн внешней рамки таблицы	Атрибут <b>&lt;TABLE&gt;</b> тега <b>&lt;TABLE&gt;</b> (в тег <b>&lt;TABLE&gt;</b> быть помещен атрибут <b>BORDER</b> ) Не отображается в Netscape Navigator
Отображать все четыре стороны внешней рамки таблицы	Атрибуту <b>FRAME</b> тега <b>&lt;TABLE&gt;</b> присвоить значение <b>BOX</b>
Отображать только верхнюю часть внешней рамки таблицы	Атрибуту <b>FRAME</b> тега <b>&lt;TABLE&gt;</b> присвоить значение <b>ABOVE</b>
Отображать вертикальные части внешней рамки таблицы слева и справа	Атрибуту <b>FRAME</b> тега <b>&lt;TABLE&gt;</b> присвоить значение <b>VSIDES</b>
Отображать только правую часть внешней рамки таблицы	Атрибуту <b>FRAME</b> тега <b>&lt;TABLE&gt;</b> присвоить значение <b>RHS</b>
Дизайн разделительных линий между строками и столбцами	Атрибут <b>RULES</b> тега <b>&lt;TABLE&gt;</b> (в тег <b>&lt;TABLE&gt;</b> должен быть помещен атрибут <b>BORDER</b> ) Не отображается в Netscape Navigator
Отображать все горизонтальные и вертикальные линии между строками и столбцами	Атрибуту <b>RULES</b> тега <b>&lt;TABLE&gt;</b> присвоить значение <b>ALL</b>

## Список основных параметров для задания стилей

### *Цвет и фон*

#### **background**

Параметр позволяет установить одновременно до пяти атрибутов стиля фона. Значения могут идти в любом порядке, браузер сам определит, какое из них соответствует нужному атрибуту.

#### **background-attachment**

Параметр `background-attachment` устанавливает, будет ли прокручиваться фоновое изображение вместе с содержимым элемента. Изображение может быть зафиксировано и оставаться неподвижным, либо перемещаться совместно с документом.

#### **background-color**

Устанавливает фоновый цвет элемента.

#### **background-image**

Устанавливает фоновое изображение для элемента. Если одновременно для элемента задан цвет фона, он будет показан, пока фоновая картинка не загрузится полностью.

#### **background-position**

Задаёт начальное положение фонового изображения, установленного с помощью параметра `background-image`.

#### **background-repeat**

Определяет, как будет повторяться фоновое изображение, установленное с помощью параметра `background-image`, и по какой оси. Можно установить повторение рисунка только по горизонтали, по вертикали или в обе стороны.

#### **color**

Определяет цвет текста элемента.

#### **opacity**

Определяет уровень прозрачности элемента веб-страницы. При частичной или полной прозрачности через элемент проступает фоновый рисунок или другие элементы, расположенные ниже полупрозрачного объекта.

## *Границы*

### **border**

Параметр позволяет одновременно установить толщину, стиль и цвет рамки вокруг элемента. Значения могут идти в любом порядке, разделяясь пробелом, браузер сам определит, какое из них соответствует нужному атрибуту.

### **border-bottom**

Параметр позволяет одновременно установить толщину, стиль и цвет границы внизу элемента. Значения могут идти в любом порядке, разделяясь пробелом, браузер сам определит, какое из них соответствует нужному атрибуту.

### **border-bottom-color**

Устанавливает цвет границы внизу элемента.

### **border-bottom-style**

Устанавливает стиль границы внизу элемента.

### **border-bottom-width**

Задаёт толщину границы внизу элемента.

### **border-color**

Устанавливает цвет границы на разных сторонах элемента. Параметр позволяет задать цвет границы сразу на всех сторонах элемента или определить цвет границы только на указанных сторонах.

### **border-left**

Параметр позволяет одновременно установить толщину, стиль и цвет левой границы элемента. Значения могут идти в любом порядке, разделяясь пробелом, браузер сам определит, какое из них соответствует нужному атрибуту.

### **border-left-color**

Задаёт цвет границы слева от элемента.

### **border-left-style**

Устанавливает стиль границы слева от элемента.

### **border-left-width**

Задаёт толщину границы слева от элемента.

## **border-right**

Параметр позволяет одновременно установить толщину, стиль и цвет правой границы элемента. Значения могут идти в любом порядке, разделяясь пробелом, браузер сам определит, какое из них соответствует нужному атрибуту.

## **border-right-color**

Задаёт цвет границы справа от элемента. Значение цвета может задаваться по его названию, шестнадцатеричному значению, либо с помощью RGB.

## **border-right-style**

Устанавливает стиль границы справа от элемента.

## **border-right-width**

Устанавливает толщину границы справа от элемента.

## **border-style**

Устанавливает стиль рамки вокруг элемента. Допустимо задавать индивидуальные стили для разных сторон элемента.

## **border-top**

Одновременно задаёт толщину, стиль и цвет границы сверху элемента.

## **border-top-color**

Задаёт цвет границы сверху элемента.

## **border-top-style**

Устанавливает стиль границы сверху элемента.

## **border-top-width**

Устанавливает толщину границы сверху элемента.

## **border-width**

Задаёт толщину границы одновременно на всех сторонах элемента или индивидуально для каждой стороны. Способ изменения толщины зависит от числа аргументов.

## **outline**

Универсальный атрибут, одновременно устанавливающий цвет, стиль и толщину внешней границы на всех четырёх сторонах элемента. В отличие от линии, задаваемой через border, атрибут outline не влияет на положение блока и его ширину.

**outline-color**

Задает цвет внешней границы элемента. В отличие от линии, задаваемой через border, линия через атрибут outline отображается вокруг элемента, не влияя на ширину блока или его положение.

**outline-style**

Задает стиль внешней границы элемента.

**outline-width**

Задает толщину внешней границы элемента.

## *Шрифт*

**font**

Универсальный параметр, который позволяет установить одновременно несколько атрибутов стиля шрифта.

**font-family**

Устанавливает семейство шрифта, которое будет использоваться для оформления текста содержимого. Список шрифтов может включать одно или несколько названий, разделенных запятой.

**font-size**

Определяет размер шрифта элемента, который может быть установлен несколькими способами.

**font-style**

Определяет начертание шрифта — обычное, курсивное или наклонное. Когда для текста установлено курсивное или наклонное начертание, браузер обращается к системе для поиска подходящего шрифта. Если заданный шрифт не найден, браузер использует специальный алгоритм для имитации нужного вида текста.

**font-variant**

Определяет, как нужно представлять строчные буквы, делать их все прописными уменьшенного размера или оставить без изменений. Такой способ изменения символов называется капителью.

**font-weight**

Устанавливает насыщенность шрифта. Значение устанавливается от 100 до 900 с шагом 100. Также допустимо использование ключевых слов.

## *Текст*

### **direction**

Атрибуты `unicode-bidi` и `direction` задают, как должен располагаться текст используемого языка — слева направо или справа налево.

### **letter-spacing**

Определяет интервал между символами в пределах элемента. Браузеры обычно устанавливают расстояние между символами, исходя из типа и вида шрифта, его размеров и настроек операционной системы. Чтобы изменить это значение и применяется данный атрибут.

### **line-height**

Устанавливает интерлиньяж (межстрочный интервал) текста, отсчет ведется от базовой линии шрифта. При обычных обстоятельствах расстояние между строками зависит от вида и размера шрифта и определяется браузером автоматически.

### **text-align**

Определяет горизонтальное выравнивание текста в пределах элемента.

### **text-decoration**

Добавляет оформление текста в виде его подчеркивания, перечеркивания, линии над текстом и мигания. Одновременно можно применить более одного стиля, перечисляя значения через пробел.

### **text-indent**

Устанавливает величину отступа первой строки блока текста (например, для параграфа `P`). Воздействия на все остальные строки не оказывается.

### **text-transform**

Управляет преобразованием текста элемента в заглавные или прописные символы. Когда значение отлично от `none`, регистр исходного текста будет изменен.

### **unicode-bidi**

Атрибуты `unicode-bidi` и `direction` задают, как должен располагаться текст используемого языка — слева направо или справа налево.

### **vertical-align**

Выравнивает элемент по вертикали относительно своего родителя или окружающего текста.

## **white-space**

Параметр `white-space` устанавливает, как отображать пробелы между словами.

## **word-spacing**

Устанавливает интервал между словами. Если установлен параметр выравнивания `justify`, то атрибут `word-spacing` не действует, поскольку интервал между словами будет установлен принудительно.

## *Списки*

### **list-style**

Атрибут, позволяющий одновременно задать стиль маркера, его положение, а также изображение, которое будет использоваться в качестве маркера.

### **list-style-image**

Устанавливает адрес изображения, которое служит в качестве маркера списка. Этот атрибут наследуется, поэтому для отдельных элементов списка для восстановления маркера используется значение `none`.

### **list-style-position**

Определяет, как будет размещаться маркер относительно текста. Имеется два значения: `outside` — маркер вынесен за границу элемента списка и `inside` — маркер обтекает текст.

### **list-style-type**

Изменяет вид маркера для каждого элемента списка. Этот атрибут используется только в случае, когда значение свойства `list-style-image` установлено как `none`. Маркеры различаются для маркированного списка (тег `UL`) и нумерованного (тег `OL`).

## *Форматирование*

### **clear**

Параметр устанавливает, с какой стороны элемента запрещено его обтекание другими элементами. Если установлено обтекание элемента с помощью параметра `float`, свойство `clear` отменяет его действие для указанных сторон.

### **clip**

Параметр `clip` определяет область позиционированного элемента, в которой будет показано его содержимое. Все, что не помещается в эту область,



будет обрезано и становится невидимым. На данный момент единственно доступная форма области — прямоугольник.

### **display**

Многоцелевой атрибут, который определяет, как элемент должен быть показан в документе.

### **float**

Определяет, по какой стороне будет выравниваться элемент, при этом остальные элементы будут обтекать его с других сторон.

### **height**

Устанавливает высоту блочных или заменяемых элементов (к ним, например, относится тег `IMG`). Высота не включает толщину границ вокруг элемента, значение отступов и полей.

### **max-height**

Устанавливает максимальную высоту элемента. Значение высоты элемента будет вычисляться в зависимости от установленных параметров `height`, `max-height` и `min-height`.

### **max-width**

Устанавливает максимальную ширину элемента.

### **min-height**

Задаёт минимальную высоту элемента. Значение высоты элемента будет вычисляться в зависимости от установленных параметров `height`, `max-height` и `min-height`.

### **min-width**

Задаёт минимальную ширину элемента. Если окно браузера достигает заданной минимальной ширины элемента, то его ширина остается неизменной и появляется горизонтальная полоса прокрутки.

### **overflow**

Свойство `overflow` управляет отображением содержания блочного элемента, если оно целиком не помещается и выходит за область заданных размеров.

**visibility**

Предназначен для отображения или скрытия элемента, включая рамку вокруг него и фон. При скрытии элемента, хотя он и становится не виден, место, которое элемент занимает, остается за ним.

**width**

Устанавливает ширину блочных или заменяемых элементов. Ширина не включает толщину границ вокруг элемента, значение отступов и полей.

### *Позиционирование*

**bottom**

Устанавливает положение нижнего края содержимого элемента без учета толщины рамок и отступов. Отсчет координат зависит от параметра `position`, он обычно принимает значение `relative` (относительное положение) или `absolute` (абсолютное положение).

**left**

Для позиционированного элемента определяет расстояние от левого края родительского элемента, не включая отступ, поле и ширину рамки, до левого края дочернего элемента. Отсчет координат зависит от значения параметра `position`.

**position**

Устанавливает способ позиционирования элемента относительно окна браузера или других объектов на веб-странице.

**right**

Для позиционированного элемента определяет расстояние от правого края родительского элемента, не включая отступ, поле и ширину рамки, до правого края дочернего элемента.

**top**

Для позиционированного элемента определяет расстояние от верхнего края родительского элемента, не включая отступ, поле и ширину рамки, до верхнего края дочернего элемента.

**z-index**

Управляет размещением элементов по z-оси, что позволяет накладывать элементы друг на друга.

## *Отступы*

### **margin**

Устанавливает величину отступа от каждого края элемента. Отступом является пространство от границы текущего элемента до внутренней границы его родительского элемента.

### **margin-bottom**

Устанавливает величину отступа от нижнего края элемента.

### **margin-left**

Устанавливает величину отступа от левого края элемента.

### **margin-right**

Задаёт величину отступа от правого края элемента.

### **margin-top**

Устанавливает величину отступа от верхнего края элемента.

## *Поля*

### **padding**

Устанавливает значение полей вокруг содержимого элемента. Полем называется расстояние от внутреннего края рамки элемента до воображаемого прямоугольника, ограничивающего его содержимое.

### **padding-bottom**

Устанавливает значение поля от нижнего края содержимого элемента.

### **padding-left**

Устанавливает значение поля от левого края содержимого элемента.

### **padding-right**

Задаёт значение поля от правого края содержимого элемента.

### **padding-top**

Задаёт величину поля от верхнего края содержимого элемента.

## *Таблицы*

### **border-collapse**

Устанавливает, как отображать границы вокруг ячеек таблицы. Этот параметр играет роль, когда для ячеек установлена рамка, тогда в месте стыка ячеек получится линия двойной толщины. Добавление значения collapse

заставляет браузер анализировать подобные места в таблице и убирать в ней двойные линии.

### **border-spacing**

Задаёт расстояние между границами ячеек в таблице. Атрибут `border-spacing` не работает в случае, когда для таблицы установлен параметр `border-collapse` со значением `collapse`.

### **caption-side**

Определяет положение заголовка таблицы, который задаётся с помощью тега `CAPTION`, относительно самой таблицы. Параметр `caption-side` выводит заголовок до или после таблицы, а выравнивание текста по правому или левому краю устанавливается через атрибут `text-align`.

### **empty-cells**

Параметр задаёт отображение границ и фона в ячейке, если в ней нет видимого содержания. При одновременном добавлении к таблице параметра `border-collapse` со значением `collapse`, атрибут `empty-cells` игнорируется.

### **table-layout**

Определяет, как браузер должен вычислять высоту и ширину ячеек таблицы, основываясь на её содержимом.

## *Интерфейс*

### **cursor**

Устанавливает форму курсора, когда он находится в пределах элемента. Вид курсора зависит от операционной системы и установленных параметров.

## *Важность*

### **important**

Играет роль в том случае, когда пользователи подключают свою собственную таблицу стилей. Если возникает противоречие, когда стиль автора страницы и пользователя для одного и того же элемента не совпадает, то `important` позволяет повысить приоритет стелевого параметра.

## *Псевдоэлементы*

### **after**

Псевдоэлемент, который используется для вывода желаемого контента после элемента, к которому он добавляется. Псевдоэлемент `after` работает совместно с атрибутом `content`.

## **before**

Псевдоэлемент `before` применяется для отображения желаемого контента до элемента, к которому он добавляется. `before` работает совместно с атрибутом `content`.

## **first-letter**

Псевдоэлемент `first-letter` определяет стиль первого символа в тексте элемента, к которому добавляется. К этому псевдоэлементу могут применяться только стилевые атрибуты, связанные со свойствами шрифта, полями, отступами, границами, цветом и фоном.

## *Псевдоклассы*

### **active**

Псевдокласс `active` определяет стиль для активной ссылки. Активной ссылка становится при нажатии на нее.

### **first-child**

Псевдокласс `first-child` добавляет стилевое оформление к первому дочернему элементу селектора, который расположен в дереве элементов документа.

### **focus**

Псевдокласс `focus` определяет стиль для элемента, получающего фокус. Например, им может быть текстовое поле формы, в которое устанавливается курсор.

### **hover**

Определяет стиль элемента при наведении на него курсора мыши.

### **link**

Псевдокласс `link` применяется к ссылкам, которые еще не посещались пользователем, и задает для них стилевое оформление.

### **visited**

Псевдокласс `visited` применяется к ссылкам, уже посещенным пользователем, и задает для них стилевое оформление.

## Контент

### content

Атрибут content позволяет вставлять генерируемое содержание в текст веб-страницы, которое первоначально в тексте отсутствует.

### counter-increment

Предназначен для увеличения значения счетчика приращений, который задается параметром counter-reset. Такой счетчик подсчитывает количество отображений элементов на странице и может выводиться с помощью свойства content и псевдоэлементов after и before. Это позволяет создавать списки (в том числе многоуровневые), в которых нумерация и вид задаются через стили.

### counter-reset

Устанавливает идентификатор, в котором будет храниться счетчик отображений определенного элемента, а также начальное значение счетчика. Такой счетчик может выводиться с помощью свойства content и псевдоэлементов after и before.

### quotes

Устанавливает вид кавычек, который применяется в тексте документа.

## Описание

Определяет, как будет повторяться фоновое изображение, установленное с помощью параметра **background-image**, и по какой оси. Можно установить повторение рисунка только по горизонтали, по вертикали или в обе стороны.

## Синтаксис

*background-repeat: no-repeat | repeat | repeat-x | repeat-y*

## Аргументы

### no-repeat

Устанавливает одно фоновое изображение в элементе без его повторений, положение которого определяется атрибутом **background-position** (по умолчанию в левом верхнем углу).

### repeat

- Фоновое изображение повторяется по горизонтали и вертикали **repeat-x**.
- Фоновый рисунок повторяется только по горизонтали **repeat-y**.

- Фоновый рисунок повторяется только по вертикали **before**.

### Описание

Псевдоэлемент **before** применяется для отображения желаемого контента до элемента, к которому он добавляется. Псевдоэлемент **before** работает совместно с атрибутом **content**.

Для **before** характерны следующие особенности.

1. При добавлении **before** к блочному элементу, значение свойства **display** может быть только: **block, inline, none, marker**. Все остальные значения будут трактоваться как **block**.
2. При добавлении **before** к встроенному элементу, значение **display** ограничено аргументами **inline** и **none**. Все остальное будет восприниматься как **inline**.
3. Стилиевые свойства **before** наследуются от элемента, к которому он добавляется.

### Синтаксис

Элемент: `before { content: "текст" }`

### Аргумент

Нет.