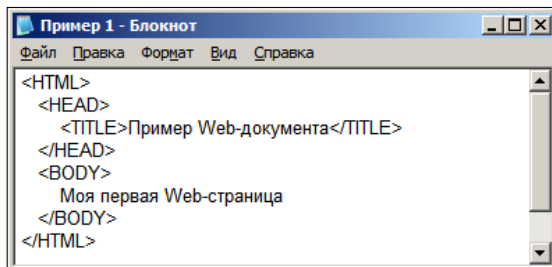


Лабораторная работа № 12. СОЗДАНИЕ ВЕБ-САЙТА СРЕДСТВАМИ HTML

Цель: ознакомиться с основами языка HTML и веб-технологиями Microsoft Excel; научиться вводить HTML-код страницы, применять веб-технологии Microsoft Excel, создавать Web-сайты.

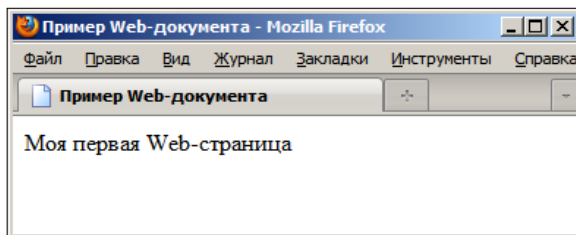
Основные понятия

Создание Web-документа. HTML-документы могут быть созданы при помощи текстового редактора, специализированного Web-редактора. В учебном процессе можно использовать **Блокнот** (Notepad), простой текстовый редактор, входящий в состав стандартных приложений Windows или Web-редактор – hеbs. При сохранении документа в Блокноте необходимо указать тип файла *htm*, (например, *Пример1.htm*). Web-редактор сам добавит тип файла – *htm*. Для просмотра документа можно использовать браузеры Mozilla, Opera, Google Chrome или Internet Explorer. На рисунках представлен вид Web-документа *Пример 1.htm* в Блокноте – HTML-код или источник, и в браузере Mozilla Firefox.



```
Пример 1 - Блокнот
Файл  Правка  Формат  Вид  Справка
<HTML>
<HEAD>
<TITLE>Пример Web-документа</TITLE>
</HEAD>
<BODY>
Моя первая Web-страница
</BODY>
</HTML>
```

HTML-код документа



Интерпретация кода браузером Mozilla Firefox

Основные положения языка HTML. Все теги HTML начинаются с «<» (левой угловой скобки) и заканчиваются символом «>» (правой угловой скобки). Как правило, существует стартовый тег и завершающий тег. Для примера приведем теги заголовка, определяющие текст, находящийся внутри стартового и завершающего тега и описывающий название документа:

```
<TITLE> Название документа </TITLE>
```

Завершающий тег выглядит так же, как и стартовый тег, но отличается от него прямым слэшем перед текстом внутри угловых скобок. В данном примере тег <TITLE> сообщает браузеру об использовании формата заголовка, а тег </TITLE> – о завершении текста заголовка.

Некоторые теги, такие, как <P> (тег, определяющий абзац), не требуют завершающего тега, но его использование придает исходному тексту документа улучшенную читаемость и структурируемость.

HTML не реагирует на регистр символов, описывающих тег, и приведенный ранее пример может выглядеть следующим образом:

```
<title> Название документа </title>
```

Дополнительные пробелы, символы табуляции и возврата каретки (клавиша <Enter>), добавленные в исходный текст HTML-документа для его лучшей читаемости, будут проигнорированы браузером при интерпретации документа.

Структура документа. Типовые элементы.

<HTML>. Когда WEB-браузер получает документ, он определяет, как документ должен быть интерпретирован. Самый первый тег, который встречается в документе, должен быть тегом <HTML>. Данный тег сообщает браузеру, что документ написан с использованием языка HTML. Минимальный HTML-документ будет выглядеть так:

```
<HTML> ...тело документа... </HTML>
```

<HEAD> (заголовочная часть документа). Тег заголовочной части документа должен быть использован сразу после тега <HTML> и более нигде в теле документа. Данный тег содержит общее описание документа. Избегайте размещать какой-либо текст внутри тега <HEAD>. Стартовый тег <HEAD> помещается непосредственно перед тегом <TITLE> и другими тегами <META>, <STYLE>, <LINK>, <SCRIPT> описывающими документ, а завершающий тег </HEAD> размещается сразу после окончания описания документа.

Например:

```
<HTML>  
<HEAD>  
<TITLE> Список сотрудников </TITLE>  
</HEAD>  
...  
</HTML>
```

Технически, стартовые и завершающие теги `<HTML>`, `<HEAD>` и `<BODY>` необязательны, но рекомендуется их использовать, поскольку они позволяют браузеру определять заголовочную и смысловую часть документа.

`<TITLE>` (заголовок или название документа). Большинство браузеров отображают содержимое тега `<TITLE>` в заголовке окна, содержащего документ и в файле закладок, если он поддерживается браузером. Заголовок, ограниченный тегами `<TITLE>` и `</TITLE>`, размещается внутри `<HEAD>`-тегов, как показано выше.

`<BODY>` (тело документа). Теги тела документа идентифицируют отображаемые в окне компоненты HTML-документа. Тело документа может содержать ссылки на другие документы, текст, графику и другую форматированную информацию. Тело документа должно находиться между тегами `<BODY>` и `</BODY>`. Это та часть документа, которая отображается как текстовая и графическая (смысловая) информация документа.

```
<HTML>  
<HEAD> ... </HEAD>  
<BODY>  
Содержание документа (контент)  
</BODY>  
</HTML>
```

Структурное форматирование текста.

`<Hn>` (уровни заголовков). При вводе HTML-кода текстовой части документа, в ней можно выделить следующие структурные части: обычный текст, заголовки частей текста, заголовки более высокого уровня и т.д. Большинство браузеров поддерживает интерпретацию шести уровней заголовков, определяя каждому из них собственный стиль. Заголовок самого верхнего уровня имеет признак «1». Синтаксис заголовка уровня 1 следующий:

```
<H1> Заголовок первого уровня </H1>
```

Заголовки всех поддерживаемых уровней могут быть представлены в общем случае так:

`<Hn>` Заголовок n-го уровня `</Hn>`

где *n* – цифра от 1 до 6, определяющая уровень заголовка.

Дополнительные параметры тега `<Hn>`:

`<H1 align= left | center | right>`

позволяют выравнивать заголовок по левому краю, центру и по правому краю окна, соответственно.

`<P>` (тег абзаца). В отличие от большинства текстовых процессоров, в HTML-документе обычно игнорируются символы возврата каретки. Физический разрыв абзаца может находиться в любом месте исходного текста документа (HTML-кода), для удобства его читаемости. Однако браузер разделяет абзацы только при наличии тега `<P>`.

Дополнительные параметры тега `<P>`:

`<P align=left | center | right | justify>`

позволяют выравнивать абзац по левому краю, центру, правому краю и по ширине окна, соответственно.

`
` (разрыв строки). Сообщает браузеру о разрыве строки. Наилучший пример использования данного тега – форматированный адрес или любая другая последовательность строк, где браузер должен отображать их одну под другой. Например:

Алексей Ярцев `
`
Дмитровское шоссе, `
`
д.9Б, офис 326 `
`

Центрирование элементов документа. Вы можете центрировать все элементы документа в окне браузера. Для этого нужно использовать тег `<CENTER>`. Все элементы между тегами `<CENTER>` и `</CENTER>` будут находиться в центре окна.

Горизонтальная линия. Используя тег `<HR>`, вы можете разделить текст горизонтальной чертой. Синтаксис тега:

`<HR size=number width=number | percent align=left | center | right noshade>`

Параметры тега:

`size` – толщина линии в пикселах.

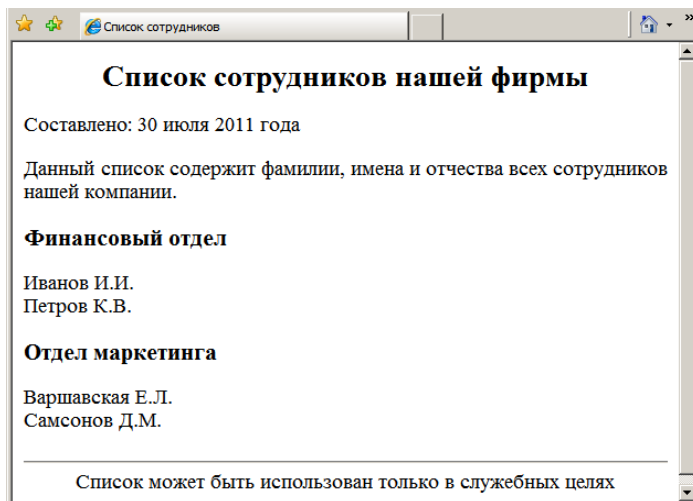
`width` – ширина линии в пикселах или процентах от ширины окна браузера.

align – расположение на экране (слева | по центру | справа).

noshade – по умолчанию линия представлена в 3D виде с тенью, noshade позволяет представить линию просто однотонной темной полосой.

color – цвет линии.

Рассмотрим несколько более подробный пример, собранный из рассмотренных элементов.



Web-страница в браузере Internet Explorer

Примечание. Заголовок «Список сотрудников» не отображен браузером как часть документа. Он появится в заголовке окна (вкладки) браузера.

Введите предлагаемый html-код.

```
<HTML>
<HEAD> <TITLE> Список сотрудников </TITLE></HEAD>
<BODY>
  <H2 align=center> Список сотрудников нашей фирмы </H2>
  Составлено: 30 июля 2011 года
  <P align=justify>Данный список содержит фамилии, имена и от-
  чества всех сотрудников нашей компании.
<H3>Финансовый отдел</H3>
  Иванов И.И.<BR>
  Петров К.В.
<H3>Отдел маркетинга</H3>
  Варшавская Е.Л.<BR>
```

```
    Самсонов Д.М.<P>
<HR>
<CENTER>
    Список может быть использован только в служебных целях
</CENTER>
</BODY> </HTML>
```

Сохраните документ в файле *Пример 2.htm* в папке *ЛабРаb.№12*. Откройте документ в браузере.

Структурирование текста с помощью списков. Существует три основных вида списков в HTML-документе: нумерованный, маркированный и список описаний (определений). Вы можете создавать вложенные списки, используя, различные теги списков или повторяя одни внутри других. Для этого просто необходимо разместить одну пару тегов (стартовый и завершающий) внутри другой. Будут ли элементы вложенного списка иметь те же маркеры, обозначающие элемент списка – зависит от браузера.

Нумерованные списки. В нумерованном списке браузер автоматически вставляет номера элементов по порядку. Если вы удалите один или несколько элементов списка, то оставшиеся номера автоматически будут пересчитаны. Нумерованный список начинается стартовым тегом `` и завершается тегом ``. Каждый элемент списка начинается с тега ``. Например:

```
<OL>
  <LI> Программирование
  <LI> Алгоритмизация
  <LI> Проектирование
</OL>
```

Маркированные списки. Для маркированных списков браузер обычно использует маркеры для пометки элемента списка. Вид маркера, как правило, настраивает пользователь. Маркированный список начинается стартовым тегом `` и завершается тегом ``. Каждый элемент списка начинается с тега ``. Например:

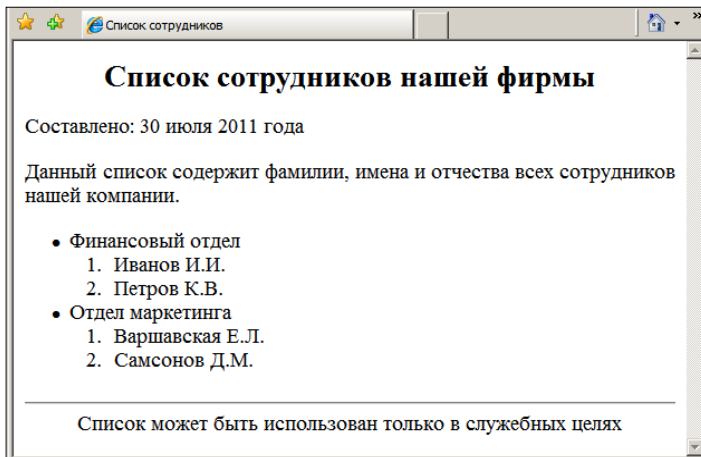
```
<UL>
  <LI> Программирование
  <LI> Алгоритмизация
  <LI> Проектирование
</UL>
```

- | | |
|---------------------|--------------------|
| 1. Программирование | • Программирование |
| 2. Алгоритмизация | • Алгоритмизация |
| 3. Проектирование | • Проектирование |

Нумерованный и маркированный списки

Многоуровневые списки. Многоуровневые списки формируются из вложенных нумерованных (маркированных) списков. Рассмотрим пример вложенных списков:

```
<UL>
<LI> Финансовый отдел
  <OL>
    <LI> Иванов И.И.
    <LI> Петров К.В.
  </OL>
<LI> Отдел маркетинга
  <OL>
    <LI> Варшавская Е.Л.
    <LI> Самсонов Д.М.
  </OL>
</UL>
```



Многоуровневый список в HTML

Физическое и логическое форматирование текста.

Добавление стилей в HTML-документ. HTML позволяет использовать различные стили шрифтов для выделения текстовой информации в документах. Вот короткий список стилей, поддерживаемых большинством браузеров:

Тег – визуальное выделение фрагмента текста, отображается жирным начертанием;

Тег <I> – выделяет часть текста курсивом;

Тег <U> – делает текст подчеркнутым.

Вы можете комбинировать различные виды стилей, например жирный и наклонный.

Комбинирование стилей позволяет вам отображать в одной строке несколько элементов различными стилями, например:

```
<B>Жизнь</B> - <I>это <B>песня!</B></I>
```

Дополнительные стили: <big> – большой, <small> – маленький, <sub> – подстрочник (нижний индекс), <sup> – надстрочник (верхний индекс).

FONT и его атрибуты. Свойства шрифта можно определять с помощью тега , для которого предусмотрены следующие атрибуты:

size – размер шрифта;

color – цвет шрифта, определяемый именем или кодовым значением (см. раздел «Управление цветом шрифта и фона страницы» данной работы).

face – гарнитура шрифта или список допустимых шрифтов.

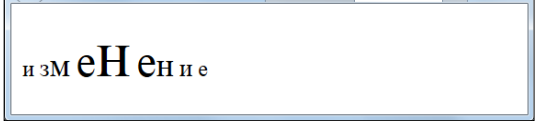
Размер символов. Вы можете изменять размер шрифта при помощи тега:

```
<FONT size=+ | – | n>
```

Шрифт может иметь размер от 1 до 7. Вы можете прямо указать размер шрифта цифрой, или указать смещение относительно базового значения (по умолчанию – 3) в положительную или отрицательную сторону: или .

Например:

```
<P>и  
<FONT size=+1>3</FONT><FONT size=+2>м</FONT>  
<FONT size=+3>е</FONT><FONT size=+4>н</FONT>  
<FONT size=+3>е</FONT><FONT size=+2>н</FONT>  
<FONT size=+1>и</FONT> е</P>
```



Изменение размера текста

Шрифт (гарнитура). Текст отображается указанным шрифтом.

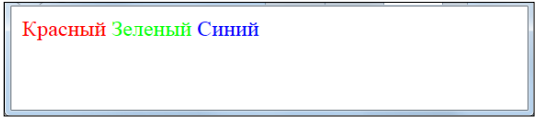
```
<FONT face=Arial>Используется шрифт Arial</FONT>
```

Цвет символов. Вы можете изменить цвет шрифта при помощи тега:

```
<FONT color=«цвет»>
```

Цвет указывается непосредственно вводом значения (красный – red), в RGB-формате (Red-Green-Blue, Красный-Зеленый-Синий) посредством указания размерности каждой компоненты цвета в шестнадцатиричном формате (красный – #FF0000) или десятичном (красный – RGB(255,0,0)): , , .

```
<FONT color=red>Красный</FONT>  
<FONT color=#00FF00>Зеленый</FONT>  
<FONT color=rgb(0,0,255)>Синий</FONT>
```



Изменение цвета текста

Допустим, вы хотите, чтобы фрагмент текста был показан шрифтом Arial, имеющим оливковый цвет и размер 16 пунктов. Для этого введите:

```
<FONT face=«Arial» size=16 color=«olive»>Выделенный текст</FONT>
```

Задание стандартных цветов. Многие HTML-авторы любят использовать заранее predetermined цвета фона документа, обычно текста и ссылок. Чтобы задать эти цвета, необходимо включить в тег <BODY> дополнительные параметры:

```
<BODY bgcolor=«#XXXXXX» text=«#XXXXXX» link= «#XXXXXX»>
```

где каждый из параметров определяет цвет определенного элемента.

Опишем эти параметры:

bgcolor – цвет фона документа;
text – цвет простого текста документа;
link, vlink, alink – цвет ссылок.

Например:

```
<BODY bgcolor="#000000" text="#FFFFFF" link="#9690CC">
```

Данная строка определяет черный цвет фона документа, белый текст и серебристые ссылки.

Таблица стандартных цветов

Цвет	Код	Название		Цвет	Код	Название	
черный	#000000	black		фиолетовый	#FF00FF	magenta	
белый	#FFFFFF	white		бирюзовый	#00FFFF	cyan	
красный	#FF0000	red		желтый	#FFFF00	yellow	
зеленый	#00FF00	lime		золотой	#FFD800	gold	
синий	#0000FF	blue		оранжевый	#FFA500	orange	
серый	#808080	gray		коричневый	#A82828	brown	

Бегущая строка <MARQUEE>

Вставленный между тегами текст появится из-за правого края программы просмотра и, пройдя через всю страницу, скроется за левым краем.

```
<MARQUEE>Текст бегущей строки</MARQUEE>
```

Атрибуты тега MARQUEE:

width – ширина окна бегущей строки;

height – высота окна бегущей строки;

bgcolor – цвет фона окна бегущей строки;

direction – направление движения: left (слева), right (справа);

behavior – поведение: scroll (прокрутить текст n-раз и исчезнуть), slide (прокрутить текст n-раз и остаться);

loop – число раз прокрутки. Если число не указано или указано число 1, то прокрутка будет бесконечной;

scrollamount – количество пикселей, на которое строка смещается за 1 шаг;

scrolldelay –задержка в миллисекундах перед каждым шагом прокрутки;

align – выравнивание строки в своем окне: top (верх), middle (центр), bottom (низ).

Определение величины шрифта бегущей строки:

```
<FONT size =n><MARQUEE> Текст </MARQUEE> </FONT>
```

Определение фона бегущей строки:

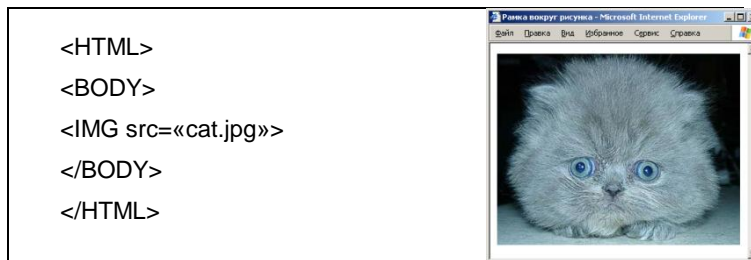
```
<MARQUEE bgcolor=«цвет»> Текст </MARQUEE>
```

Вставка графического файла в Web-документ. Изображение (рисунок) можно поместить в любое место Web-страницы, в текст, таблицу, заголовок с помощью элемента IMG (image), состоящего из одного тега и атрибута src=«имя файла». Этот атрибут является обязательным. Важно, чтобы тег находился между тегами <BODY>...</BODY>. Например,

```
<IMG src=«cat.jpg»>
```

Подразумевается, что файл изображения находится в той же папке, что и создаваемый файл HTML. Если все графические файлы находятся в этой директории, тег будет иметь вид:

```
<IMG src=«Имя файла.тип»>
```



Если графический файл находится в папке IMAGES с тем же уровнем вложения, что и текущая папка, атрибут src имеет шаблон: src=«../IMAGES/Имя файла.тип».

Если файл рисунка – на другом диске, например, на диске D в папке IMAGES, тогда элемент будет иметь вид:

```
<IMG src=«FILE://D:\IMAGES/Имя файла.тип»>
```

Гипертекстовые ссылки.

Важнейшим достоинством языка HTML является возможность создания гиперссылок. С помощью гиперссылок устанавливаются связи с другими местами данного документа (внутристраничные связи), с другими страницами данного сервера (внутрисистемные), с другими WWW-серверами (межсистемные), а также с файл-серверами и почто-

выми серверами (межпротокольные). Именно гиперссылки совместно с линиями связи и серверами участвуют в создании WWW. Для определения местоположения отдельного ресурса сети используется указатель, который называется URL (Uniform Resource Locator). URL – это универсальный указатель ресурсов, адрес который описывает путь к конкретному документу.

Гипертекстовые ссылки являются ключевым компонентом, делающим WEB привлекательным для пользователей. Добавляя гипертекстовые ссылки (далее – ссылки), вы делаете набор документов связанным и структурированным, что позволяет пользователю получать необходимую ему информацию быстро и удобно.

Вставка ссылок в документ.

Текстовые и графические ссылки. Для вставки ссылок в документ используется элемент A, который определяется парным тегом `<A ...>...`, с обязательным атрибутом href. Между начальным и конечным тегами располагается указатель ссылки.

`<A ...>указатель ссылки`

Указатель отображается в документе при просмотре его в браузере. В зависимости от вида указателя ссылки могут быть **текстовыми** (слово, словосочетание, предложение, блок текста), **графическими** (рисунок) и **смешанными**. Например:

Текстовая ссылка:

```
<IMG src=...ФБУ.jpg>  
<A href=...baa.by/facultet/fbu>  
    Факультет бухгалтерского учета  
</A>
```

Графическая ссылка:

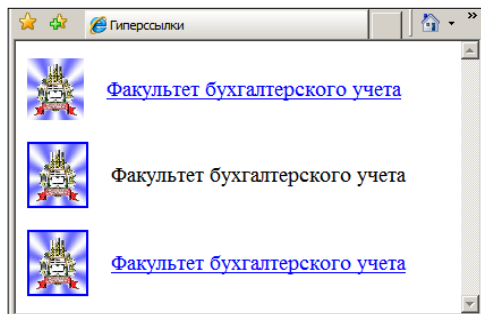
```
<A href=...baa.by/facultet/fbu>  
    <IMG src=...ФБУ.jpg>  
</A>  
Факультет бухгалтерского учета
```

Смешанная ссылка 1:

```
<A href=...baa.by/facultet/fbu>  
    <IMG src=...ФБУ.jpg> Факультет бухгалтерского учета  
</A>
```

Смешанная ссылка 2:

```
<A href=...baa.by/facultet/fbu>  
  <IMG src=...ФБУ.jpg>  
</A>  
<A href=...baa.by/facultet/fbu>  
  Факультет бухгалтерского учета  
</A>
```



Текстовые и графические ссылки

Внутренние и внешние ссылки. Для организации перехода в указанное место текущего документа применяют **внутренние ссылки**.

Внешние ссылки устанавливают связь с другими страницами данного сервера (внутрисистемные), с другими WWW-серверами (межсистемные), а также с файл-серверами и почтовыми серверами (межпротокольные).

Внешние ссылки могут быть **относительными** или **абсолютными**.

Предположим, что на вашем компьютере есть папка *Сайт1*, а в ней три файла *index.htm*, *page1.htm*, *page2.htm*, переход по ссылкам со страницы *index.htm* можно организовать следующим образом:

```
<A href=page1.htm>Страница 1</A>  
<A href=page2.htm>Страница 2</A>
```

Усложним задачу, в папке *Сайт1*, находятся вложенные папки *Страницы* и *Рисунки*. Файлы *page1.htm*, *page2.htm* хранятся в папке *Страницы*, а файл *ФБУ.jpg* в папке *Рисунки*, переход по ссылкам и загрузка рисунка организуется следующим образом:

```
<A href=Страницы/page1.htm>Страница 1</A>  
<A href=Страницы/page2.htm>Страница 2</A>  
<A href=Рисунки/ФБУ.jpg>Рисунок</A>
```

Web-технологии Microsoft Excel. Microsoft Excel наряду с сохранением файлов в обычном для него формате, *.xlsx, предоставляет возможность сохранения отдельных объектов: диапазона ячеек, графиков и диаграмм в формате *.htm. Данная функция позволяет пользователю, без знания основ разработки Web-страниц, быстро создавать страницы с результатами обработки данных в табличном процессоре Microsoft Excel.

Для выполнения данной операции необходимо:

- 1) выделить объект (диапазон ячеек или диаграмму);
- 2) выполнить команды **Сохранить как** ➤ **Другие форматы**;
- 3) в окне диалога **Сохранение документа**, выбрать папку для сохранения документа, ввести имя файла, выбрать тип файла: **Веб-страница**;
- 4) установить переключатель **Сохранить выделенное**, нажать кнопку **<Изменить>**, для ввода заголовка страницы. В окне диалога **Задать заголовок**, ввести текст определяющий содержание таблицы и нажать кнопку **<ОК>**;
- 6) нажать кнопку **<Сохранить>**. В диалоговом окне **Публикация веб-страницы** проверить введенные параметры и нажать кнопку **<Опубликовать>**.

Практическая работа

Задание 1

Создайте веб-страницу, содержащую основные теги, которые определяют структуру веб-страницы и элементы содержательной части: бегущую строку, заголовки, абзацы, списки и т. д. (рис. 1, стр. 16).

Технология выполнения

1. Все файлы должны быть сохранены в папке *ЛабРаb№12*. Это – *index.htm, page1.htm, page2.htm, page3.htm, page4.htm, fon.jpg, img.jpg*.
2. Создайте веб-страницу, сохраните ее в файле *index.htm*.
3. Введите теги определяющие структуру страницы: **<HTML>**, **<HEAD>**, **<TITLE>**, **<BODY>**.
4. Между парой тегов **<TITLE>** введите название документа «СППР в Microsoft Excel».
5. В область **BODY** введите бегущую строку «ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ», используйте парный тег **<MARQUEE>**.
6. Введите текст «МАТЕМАТИЧЕСКОЕ ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ», оформите его как заголовок 2-го уровня, парный тег **<H2>**. Введите текст «Транспортная задача», оформите как заголовок

2-го уровня. Введите текст «Задание» и оформите его как заголовок 3-го уровня.

7. Введите текст «Разработать математическую модель, создать компьютерную модель (электронную таблицу) и найти оптимальное решение задачи.». Для создания отступа – «красной строки» используйте конструкцию – неразрывный пробел, введите пять элементов. Оформите текст как абзац, тег <P>.

8. Введите заголовок 3-го уровня – «Технология выполнения».

9. Создайте нумерованный список (парный тег) с элементами: «Постановка задачи», «Математическая модель», «Компьютерная модель», «Поиск оптимального решения». Каждый элемент определяется тегом .

10. Создайте маркированный список (парный тег) с элементами: «Целевая функция», «Система ограничений», «Граничные условия». Поместите его после элемента «Математическая модель». Таким образом, в HTML формируются многоуровневые списки.

11. Вставьте заголовок 3-го уровня «Выполнение задания».

12. Вставьте нумерованный список с элементами: «Постановка задачи», «Исходные данные», «Математическая модель» и «Результат».

13. Добавьте разделительную линию – <HR> и <ADDRESS> парный тег, содержащий данные о разработчике сайта: «Задание выполнил магистрант: Фамилия И.О.».

Задание 2

Выполните оформление веб-страницы (рис. 2, стр. 17).

1. Для оформления фона страницы, используйте фоновый рисунок *fon.jpg* из папки *Дополнительный материал*. Скопируйте рисунок в папку *ЛабРаб№12*. Добавьте атрибут `background` в тег <BODY> и укажите ссылку на файл *fon.jpg*. <BODY background=fon.jpg>.

2. Выполните оформление бегущей строки. Установите высоту области – 60; цвет фона – белый. Формат текста бегущей строки: шрифт – Arial; цвет – красный; размер – 6; полужирное начертание.

3. Для заголовков 2-го уровня, установите шрифт – Arial и выравнивание по центру.

4. Для текста определенного как абзац установите выравнивание по ширине. Ключевые слова в тексте выделите красным цветом, кроме этого «найти оптимальное решение задачи» – выделите полужирным курсивом.

5. Для разделительной линии установите красный цвет и толщину – 4.

6. Адрес выровняйте по центру окна (парный тег <CENTER>). Раз-

мер обычного текста в документе увеличьте на +1.

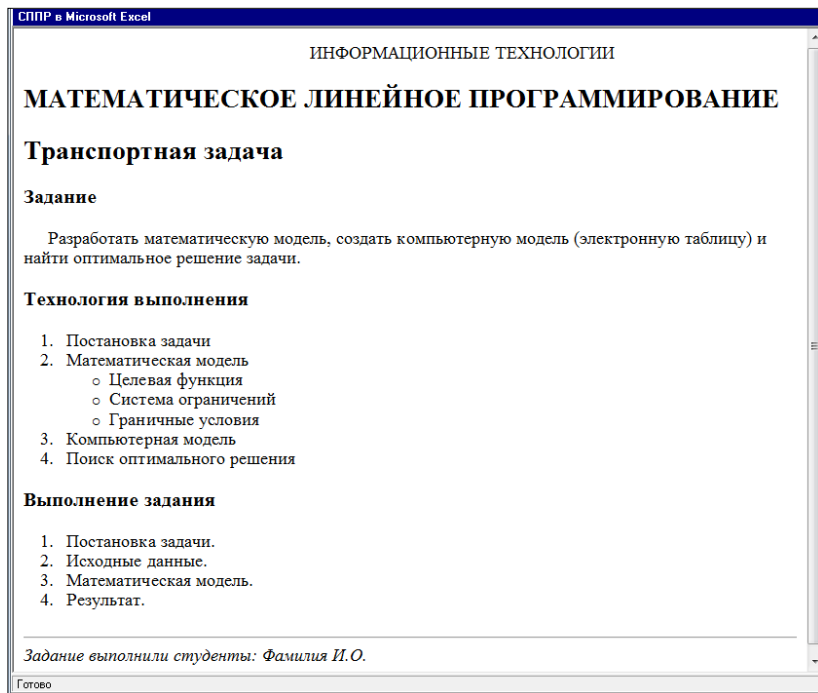


Рис.1. Образец страницы index.htm

Задание 3. Создайте страницу *page1.htm* по образцу (рис. 3, стр.17).

1. Создайте новый документ, сохраните в файле *page1*.
2. Установите цвет фона страницы. Добавьте атрибут `bgcolor` в тег `<BODY>` и выберите цвет.
3. Введите заголовок 3-го уровня «ПОСТАНОВКА ЗАДАЧИ». Установите выравнивание заголовка по центру окна и шрифт текста Arial.
4. Введите абзац текста, установите выравнивание текста по ширине страницы. Добавьте отступы (красная строка). Увеличьте размер символов на +1.



Рис. 2. Вид страницы index.htm после форматирования



Рис. 3. Образец страницы page1.htm

Задание 4. Создайте страницу *page2.htm* (рис.4).

Исходные данные				
Затраты на перевозку единицы груза, спрос и запас				
Поставщики	Потребители			Запас
	A	B	C	
I	7	6	4	120
II	3	8	5	100
III	2	3	7	80
Спрос	90	90	120	

Рис. 4. Образец страницы page2.htm

1. Скопируйте рабочую книгу *Задание 1.xlsx* из папки *Дополнительный материал* в папку *ЛабРаб№12*. Откройте рабочую книгу.

2. Выделите диапазон ячеек, в котором содержатся данные для решения задачи – **A1 : E7**. Выполните команды **Сохранить как» Другие форматы**. В диалоговом окне **Сохранение документа** в поле **Имя файла:** введите *page2*, в списке **Тип файла:** выберите **Веб-страница**. В группе переключателей **Сохранить:** выберите **выделенное: \$A\$1:\$E\$7**. В области **Заголовок**, нажмите кнопку изменить и введите заголовок «ИСХОДНЫЕ ДАННЫЕ». Нажмите кнопку **<Сохранить>**.

3. Проверьте наличие файла *page2.htm* в папке *ЛабРаб№12*.

Задание 5

Создайте страницу *page3.htm* (рис.5).

Математическая модель	
МАТЕМАТИЧЕСКАЯ МОДЕЛЬ	
$7x_{11} + 6x_{12} + 4x_{13} + 3x_{21} + 8x_{22} + 5x_{23} + 2x_{31} + 3x_{32} + 7x_{33} \rightarrow \min.$	
$\begin{cases} x_{11} + x_{12} + x_{13} = 120; \\ x_{21} + x_{22} + x_{23} = 100; \\ x_{31} + x_{32} + x_{33} = 80. \end{cases}$	$\begin{cases} x_{11} + x_{21} + x_{31} = 90; \\ x_{12} + x_{22} + x_{32} = 90; \\ x_{31} + x_{23} + x_{33} = 120. \end{cases}$
$x_{ij} \geq 0, \quad i = \overline{1, 3}, \quad j = \overline{1, 3}.$	

Рис. 5. Образец страницы page3.htm

1. Загрузите текстовый процессор Microsoft Word.
2. В редакторе формул введите математическую модель задачи:

$$7x_{11} + 6x_{12} + 4x_{13} + 3x_{21} + 8x_{22} + 5x_{23} + 2x_{31} + 3x_{32} + 7x_{33} \rightarrow \min$$

$$\begin{cases} x_{11} + x_{12} + x_{13} = 120; \\ x_{21} + x_{22} + x_{23} = 100; \\ x_{31} + x_{32} + x_{33} = 80. \end{cases} \quad \begin{cases} x_{11} + x_{21} + x_{31} = 90; \\ x_{12} + x_{22} + x_{32} = 90; \\ x_{13} + x_{23} + x_{33} = 80. \end{cases}$$

$$x_{ij} \geq 0, \quad i = \overline{1, 3}, \quad j = \overline{1, 3}.$$

3. С помощью стандартного приложения Windows **Ножницы** выделите математическую модель и сохраните в файле *img.jpg* в папке *ЛабРаб№12*. В случае, когда работа в Microsoft Word вызывает проблемы, можете скопировать рисунок *img.jpg* из папки *Дополнительный материал* в папку *ЛабРаб№12*.

4. Создайте новую страницу – *page2.htm*.
5. Введите заголовок 2 уровня «МАТЕМАТИЧЕСКАЯ МОДЕЛЬ».
6. Вставьте рисунок *img.jpg*. ``.

Задание 6

Создайте страницу *page4.htm* (рис. 5).

1. Создайте страницу, применив веб-технологии Microsoft Excel по аналогии с заданием 4. При этом следует выделить диапазон ячеек с результатом решения задачи и ввести заголовок «РЕЗУЛЬТАТ».

Поставщики	Потребители			Вывоз
	A	B	C	
I	0	10	110	120
II	90	0	10	100
III	0	80	0	80
Доставка	90	90	120	1060

Рис. 5. Образец страницы *page4.html*

Задание 7. Создайте веб-сайт с главной страницей *index.htm* и связанными страницами *page1.htm*, *page2.htm*, *page3.htm*, *page4.htm* (рис. 6).



Рис. 6. Главная страница сайта

1. Откройте страницу *index.htm* в редакторе. В разделе выполнение задания, для каждого элемента списка, добавьте ссылки с текстовым указателем – » **Смотреть далее** для перехода на соответствующие страницы сайта. Например, переход на страницу *page1.htm* – «Постановка задачи» оформляется в следующем виде:

```
<LI>Постановка задачи.  
<A href=«page1.htm» &raquo; &quot; Смотреть далее</A>  
</LI>
```

где конструкция » выводит правые русские кавычки.

2. В тег <BODY> добавьте атрибуты изменяющие цвет гиперссылок: активная ссылка – красный цвет, просмотренные и не просмотренные ссылки – черный.

3. Для создания ссылок без подчеркивания в область <HEAD> введите описание стиля:

```
<STYLE> A {text-decoration: none} </STYLE>
```

4. Просмотрите все страницы сайта в браузерах, установленных на вашем компьютере.

Контрольные вопросы

1. Опишите, как сохранить документ, созданный в текстовом редакторе Блокнот, в формате HTML.

2. Укажите основные тэги, необходимые для создания документа на языке HTML.

3. Как указать название HTML-документа?

4. Чем отличается название HTML-документа от имени соответствующего файла?

5. Перечислите параметры тэга <BODY>.

6. Как задаются цвета на языке HTML?

7. Опишите основные приемы форматирования HTML-документов.

8. Как определяется абзац на языке HTML?

9. Перечислите значения атрибута ALIGN для тэга абзаца.

10. Для чего используется тэг
?

11. Как задаются заголовки в HTML-документах?

12. Перечислите тэги для создания нумерованных, маркированных списков и списков определений.

13. Каким тэгом задается разделительная линия?

14. Перечислите атрибуты разделительной линии.

15. Как создать гиперссылку на объект текущего Web-узла?

16. Чем отличается относительный URL-адрес от абсолютного адреса?

17. Как задается абсолютный URL-адрес?

20. Как создать гиперссылку на графическом изображении?