



Возможности, структура и основные элементы языка HTML

Литература

1. HTML5 Standard. Режим доступа <http://www.w3.org/TR/2014/REC-html5-20141028/> свободный. Англ. яз.
2. Learn HTML. Режим доступа <http://www.w3schools.com/html/default.asp> свободный. Англ.яз.
3. Фримен Э., Фримен Э. Изучаем HTML, XHTML и CSS. СПб.: Питер, 2014
4. Гоше Х. HTML5. Для профессионалов. СПб: Питер, 2014
5. Хоган Б. HTML5 и CSS3. Веб-разработка по стандартам нового поколения. СПб.: Питер, 2014.
6. Макфарленд Д. Большая книга CSS. СПб.: Питер, 2014.
7. Квинт И. Создаем сайты с помощью HTML, XHTML и CSS. СПб.: Питер, 2014.
8. Дакетт Д. HTML и CSS. Разработка и дизайн веб-сайтов. М.: Эксмо, 2013.

Основные принципы работы WWW

1. Отсутствие централизованных органов управления и контроля – все могут использовать информацию, открыто представленную в **WWW**, и публиковать там свои данные. Базовый элемент www – **web-страница**.
2. Универсальность, стандартность, аппаратная независимость протоколов обмена данными и динамическое согласование форматов документов:

HTTP (Hypertext Transfer Protocol) – протокол прикладного уровня для передачи гипертекста. Стандартный протокол web-связи, чаще всего используется при обмене информацией между браузером сервером. Центральным объектом в HTTP является ресурс, на который указывает URI в запросе клиента.

URI (Uniform Resource Identifier) - унифицированный идентификатор ресурса. Представляет собой последовательность символов, идентифицирующая абстрактный или физический ресурс.

URI = URL + URN, где URL (Uniform Resource Locator) - это часть URI, которая, определяет адрес хоста сетевого ресурса (для несетевых ресурсов эта часть может опускаться), URN (Uniform Resource Name) - это часть URI которая определяет имя ресурса на хосте в локальном пространстве имён (и, соответственно, в определённом контексте).

HTML (HyperText Markup Language) – стандартный язык для разметки контента, используемый для создания Web-страниц, и поддерживаемый всеми Интернет-браузерами.

Общие сведения о языке HTML

Язык HTML (Hyper Text Markup Language – язык разметки гипертекстов) – стандартный язык разметки документов во Всемирной паутине. Большинство веб-страниц создаются при помощи языка HTML. Язык HTML интерпретируется браузером и отображается в виде документа, в удобной для человека форме.

HTML представляет собой коллекцию управляющих символов – **тегов** (или **дескрипторов**), с помощью которых можно добавлять и форматировать элементы документа. Для настройки внешнего вида и особенностей функционирования **элемента Web-страницы** должны быть установлены необходимые **атрибуты** данного элемента.

Работа со страницей HTML

- Необходима инструментальная среда (абсолютное большинство инструментальных сред позволяет создать HTML-код);
- Просмотр исходного HTML-кода возможен в любом браузере;
- Код страницы следует сохранять в файле с расширением `.htm` или `.html`.

Структура HTML-документа

Каждый HTML-документ начинается с тега `<html>` и заканчивается тегом `</html>`. Это тэг является **корневым элементом** HTML-документа.

До корневого элемента может находиться **пролог (декларация)** HTML-документа.

HTML-документы состоят из двух разделов:

- **заголовка** (содержит установки глобальных параметров Web-страницы);
- **основного раздела** (содержит текст и элементы страницы, отображаемые в окне браузера).

Данные между
тегами

`<head>` и `</head>`

Данные между
тегами

`<body>` и `</body>`

Перечисленные теги – **ОБЯЗАТЕЛЬНЫЕ** (они должны присутствовать во всех **HTML-документах**).

Структура HTML-документа

Стандартная структура HTML-документа:

<html>

<head>

<title>Заголовок страницы ***</title>***

параметры страницы ...

</head>

<body>

текст страницы ...

</body>

</html>

Парные и непарные теги HTML

● Парные теги.

Парные теги (контейнеры) состоят из двух частей – открывающего и закрывающего тегов. Открывающий тег обозначается как и одиночный – `<тег>`, а в закрывающем используют слеш – `</тег>` (например, `<html> ... </html>`).

Парные теги могут быть вложенными.

● Непарные теги.

Используются независимо от других тегов (например, `<meta>` или `
` – тег разрыва строки).

Рекомендуется перед закрывающей угловой скобкой непарного тега ставить прямой слеш (например: `<area />`, `<link />`, `<hr />`, `<meta />`, ``, `
`, `<input />`, `<col />`).

Вложенность тегов HTML и комментарии

- В SGML и основанных на нем языках (HTML версий 2-5, языки семейства XML и т.д.) элементы должны быть строго вложенными друг в друга, а «перехлест» элементов недопустим:
 - неправильно: `<a>Имя Фамилия`
 - правильно : `<a>Имя Фамилия`
- HTML позволяет вставлять в тело документа **комментарии**, которые сохраняются при передаче документа по сети, но не отображаются браузером.
 - ***Синтаксис комментария:***
`<!-- Это комментарий -->`

Блочные и строчные элементы

В HTML 4.01 элементы делились на блочные и строчные (в CSS это формальное деление осталось)

Блочные элементы характеризуются тем, что занимают всю доступную ширину контейнера, высота элемента определяется его содержимым, и он всегда начинается с новой строки. Используются для форматирования целых блоков текста (например, `<blockquote>`, `<div>`, `<h1>` - `<h6>`, `<p>`, `<pre>`).

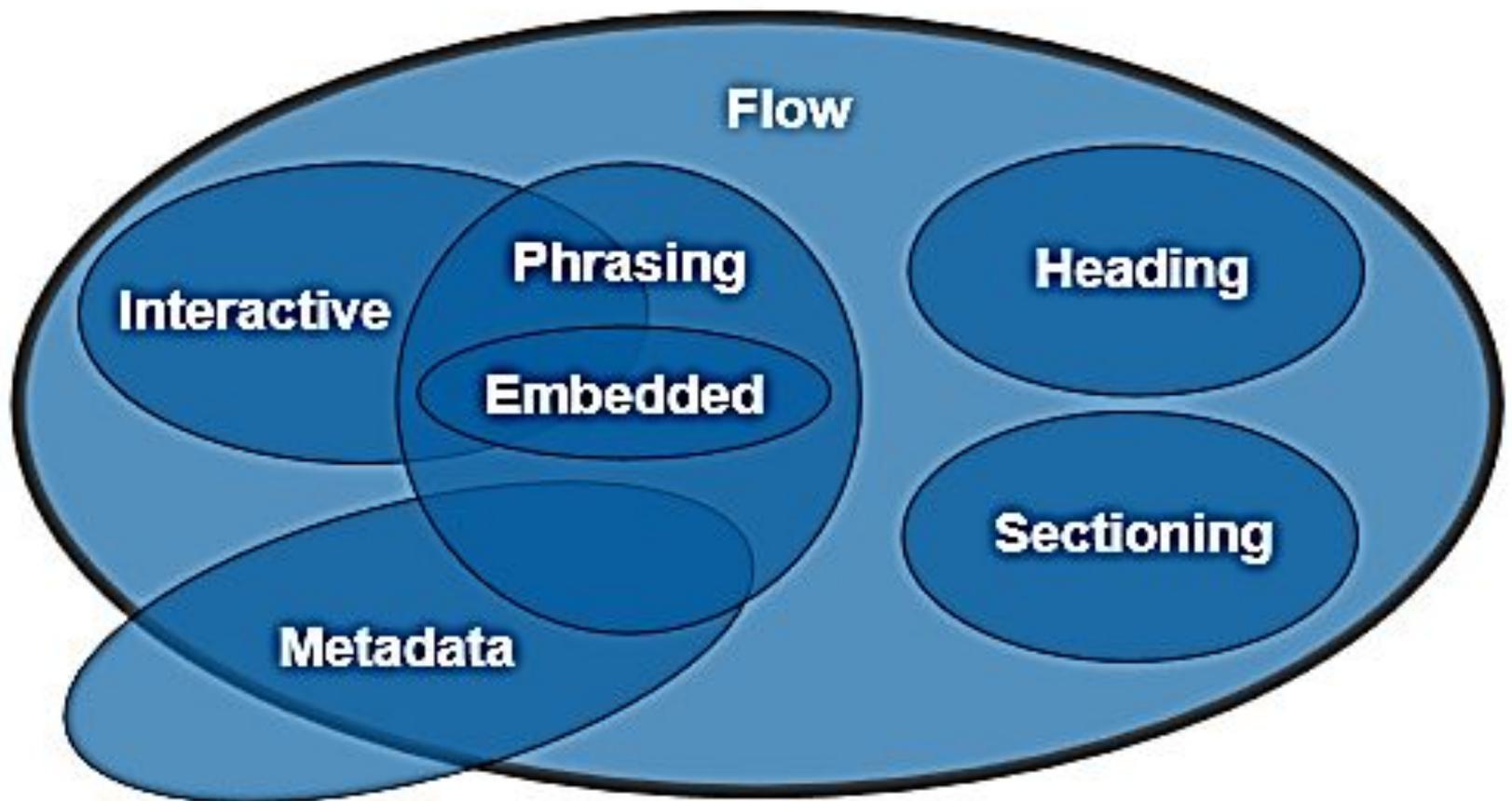
Строчные элементы – это такие элементы, которые являются непосредственной частью другого элемента, например, текстового абзаца. В основном они используются для изменения вида текста или его логического выделения при форматировании отдельных слов и символов (например, `<a>`, `<i>`, `` и т.п.)

Классификация элементов HTML5

The following broad categories are used in specification w3.org:

- Metadata content (link, style, title);
- Flow content (a, br, button, h1-h1, div);
- Sectioning content (article, aside, nav, section);
- Heading content (h1 - h6);
- Phrasing content (a, em, i, small, span);
- Embedded content (audio, img);
- Interactive content (a, audio, button).

Классификация элементов HTML5



Атрибуты элементов HTML

Атрибуты представляют собой описание особенностей элемента. Особенности атрибутов HTML:

- Атрибуты могут быть только в открывающем тэге;
- Атрибутов может быть несколько, разделяются между собой пробелами;
- Могут относиться к одной из трех групп: универсальные, уникальные или атрибуты событий;
- За атрибутом может быть закреплен перечень значений.

```
<img src = "myImage" alt = "Flower">
```

Форматирование текста в HTML-документе

- **Создание нового абзаца.**

Пара блочных тегов `<p> ... </p>`.

Создается объект абзаца.

По умолчанию абзац выравнивается влево;

слова автоматически переносятся на новую строку, когда текст достигает края окна браузера.

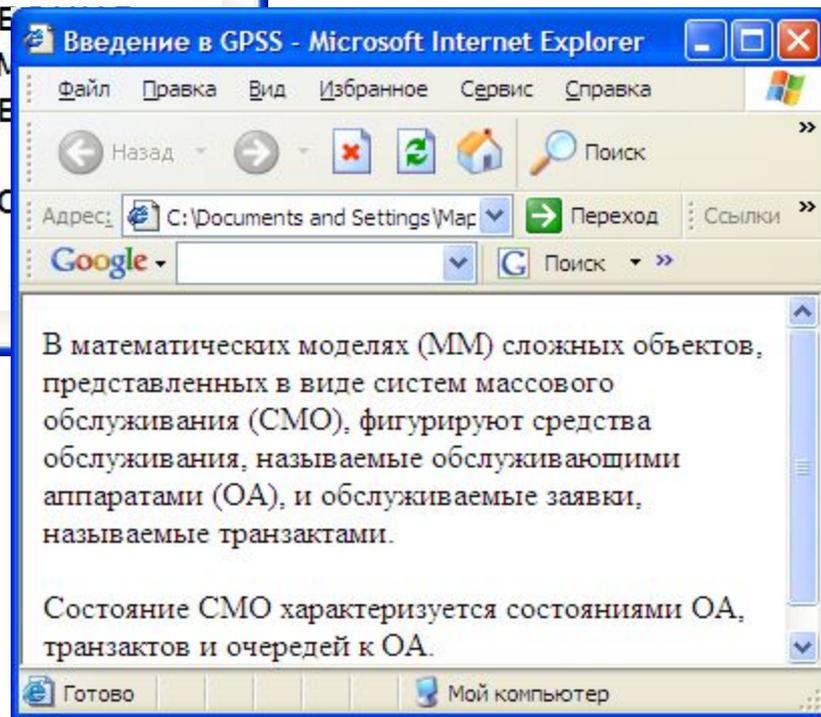
Форматирование текста в HTML-документе

Пример:

Начало абзаца

```
<TITLE>Введение в GPSS</TITLE>
<P>В математических моделях (ММ)
сложных объектов, представленных
в виде систем массового обслуживания
(СМО), фигурируют средства обслужи
называемые обслуживающими аппаратам
(OA), и обслуживаемые заявки, назыв
транзактами
</P>
Состояние СМО характеризуется состо
OA, транзактов и очередей к OA.
```

Конец абзаца



Форматирование текста в HTML-документе

- **Разрыв строки.**

Непарный тег **
**.

Текст, следующий за **
, будет показан с новой строки (при этом **НЕ будет создан новый абзац и **НЕ** будет добавлена пустая строка как в случае **<p> ...</p>**).

Часто используется для

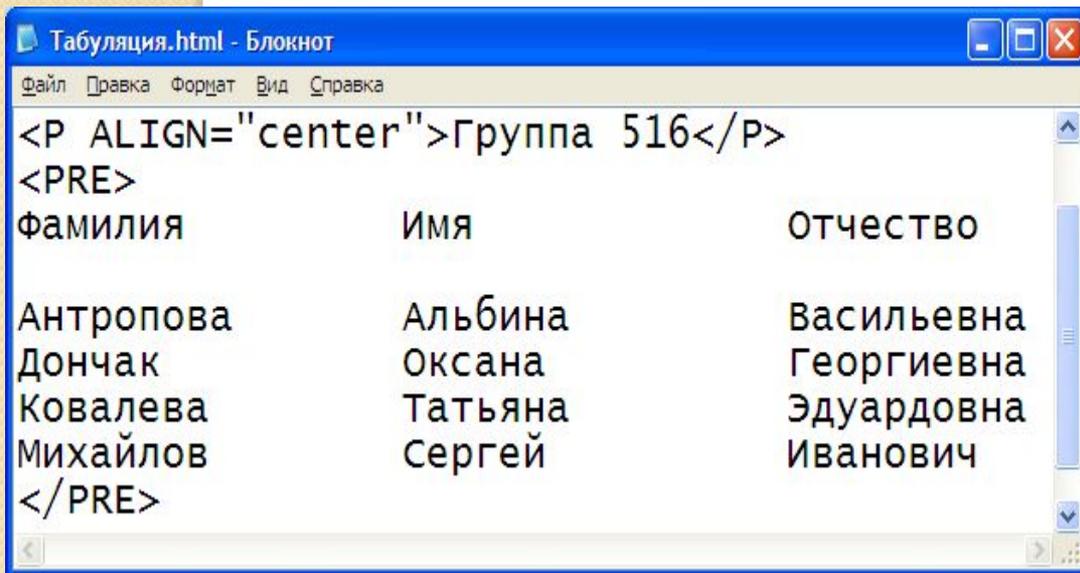
- добавления рисунка в текст с новой строки;
- увеличения отступа между абзацами.

Форматирование текста в HTML-документе

- **Управление пробелами и отступами.**

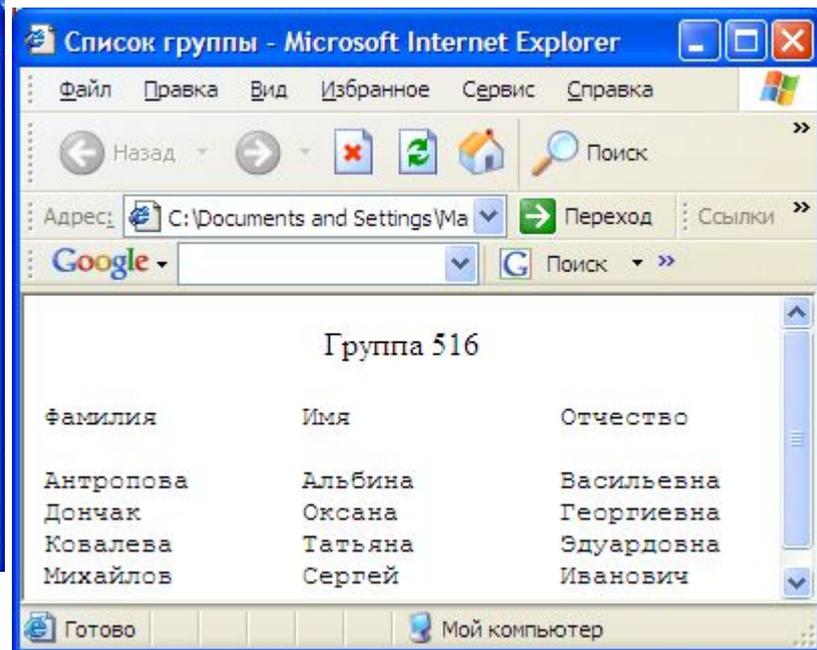
- Пара тегов `<pre> ... </pre>`.

В тексте, выделенном с помощью этой пары, будут учитываться все символы пробелов и табуляции, внесенные средствами текстового редактора.



Табуляция.html - Блокнот

```
Файл Правка Формат Вид Справка
<P ALIGN="center">Группа 516</P>
<PRE>
Фамилия          Имя          Отчество
Антропова        Альбина      Васильевна
Дончак           Оксана       Георгиевна
Ковалева         Татьяна      Эдуардовна
Михайлов         Сергей       Иванович
</PRE>
```



Форматирование текста в HTML-документе

● **Текстовые заголовки:**

Теги, устанавливающие заголовки шести уровней от `<h1> ... </h1>` до `<h6> ... </h6>`.

Шрифты первых трех уровней больше размера основного шрифта страницы;

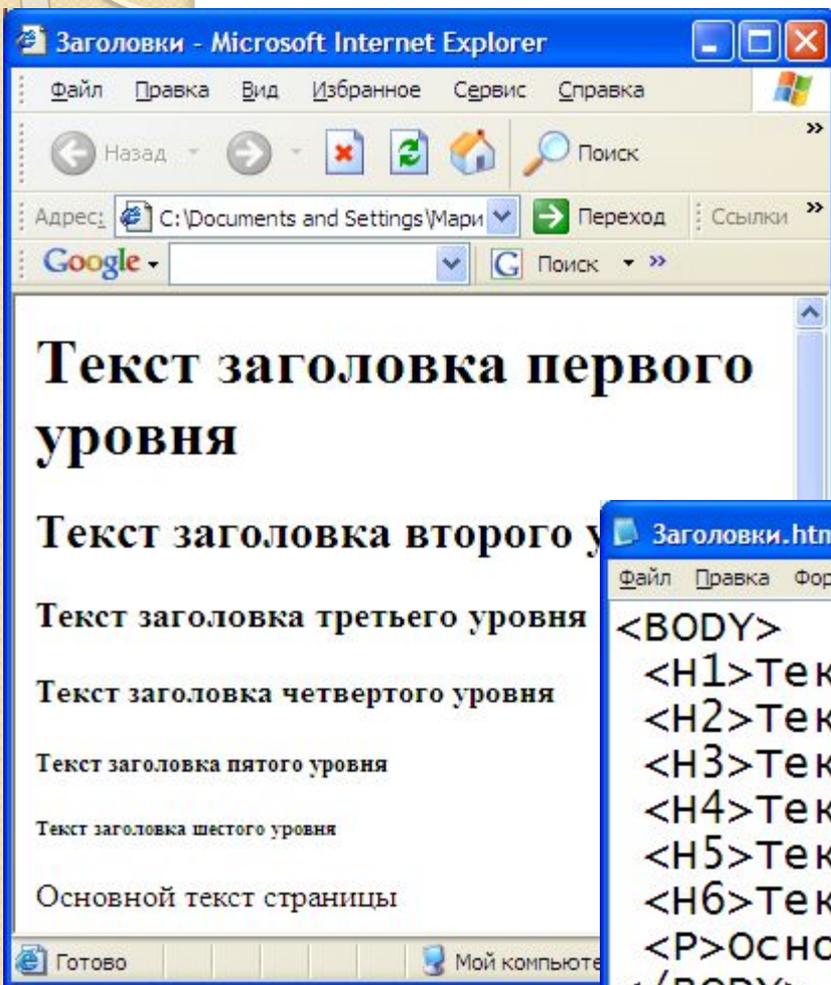
размер шрифта 4-го уровня совпадает с размером основного шрифта;

размер шрифта 5-го и 6-го уровней меньше размера основного шрифта (можно использовать для примечаний или сносок).

Для многих поисковых роботов (*Google, Rambler, Яндекс*) при индексировании Web-страниц текст заголовков имеет более высокий приоритет.

Форматирование текста в HTML-документе

Пример:



Заголовки.html - Блокнот

Файл Правка Формат Вид Справка

```
<BODY>
  <H1>Текст заголовка первого уровня</H1>
  <H2>Текст заголовка второго уровня</H2>
  <H3>Текст заголовка третьего уровня</H3>
  <H4>Текст заголовка четвертого уровня</H4>
  <H5>Текст заголовка пятого уровня</H5>
  <H6>Текст заголовка шестого уровня</H6>
  <P>Основной текст страницы</P>
</BODY>
```

Форматирование текста в HTML-документе

■ **Специальные символы.**

Для добавления символов, зарезервированных в качестве служебных в языке HTML, или отсутствующих на клавиатуре используется числовой или именной код.

Коды некоторых специальных СИМВОЛОВ:

| Символ | Числовой код | Именной код |
|------------|--------------|-------------|
| " | " | " |
| & | & | & |
| < | < | < |
| > | > | > |
| € (евро) | ¢ | ¢ |
| § | § | § |
| © | © | © |
| « | « | « |
| ® | ® | ® |
| ° (градус) | ° | ° |
| ± | ± | ± |
| » | » | » |
| ÷ | ÷ | ÷ |

Форматирование текста в HTML-документе

- **Создание списков в HTML-документе:**
 - Сортированная информация (нумерованный).
 - Несортированная информация (маркированный).
 - Термины и определения

Форматирование текста в HTML-документе

- **Нумерованные списки:**

Используется комбинация двух пар тегов:

** ... ** – установка начала и конца нумерованного списка;

** ... ** – выделение отдельных пунктов списка.

Форматирование текста в HTML-документе

- Атрибут ***type*** устанавливает тип нумерованного списка.

| Значение атрибута TYPE | Тип нумерации пунктов списка |
|------------------------|--------------------------------------|
| '1' | 1, 2, 3, 4, ... (задан по умолчанию) |
| 'i' | i, ii, iii, iv, ... |
| 'I' | I, II, III, IV, ... |
| 'a' | a, b, c, d, ... |
| 'A' | A, B, C, D, ... |

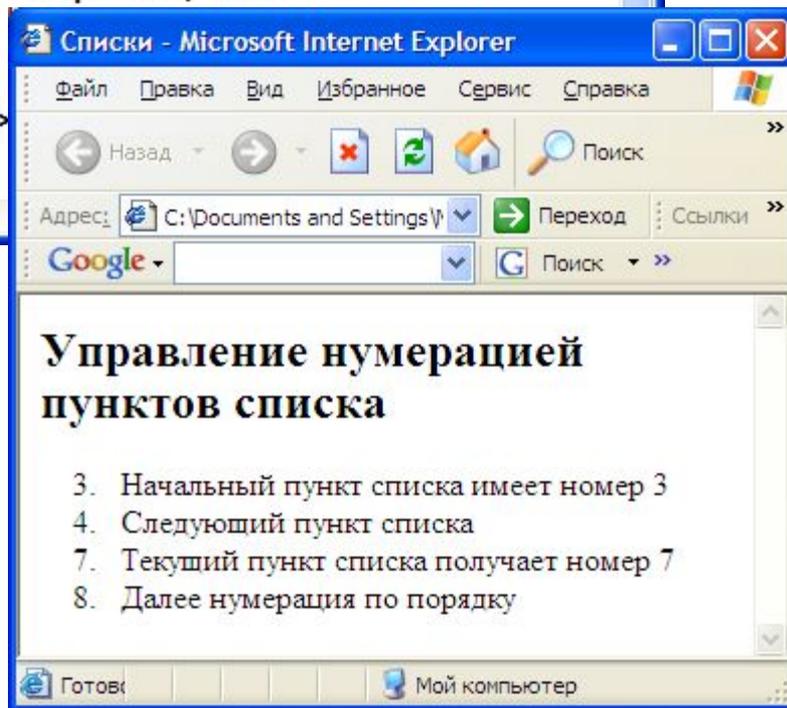
- Атрибут ***start*** устанавливает начальный номер пунктов списка.
- Атрибут ***value*** присваивает номер текущему пункту списка (например, в случае пропуска некоторых номеров).

Форматирование текста в HTML-документе

Пример:

Модифицированный нумерованный список:

```
Списки2.html - Блокнот
Файл Правка Формат Вид Справка
<h2>Управление нумерацией пунктов списка</h2>
<ol start='3'>
  <li>Начальный пункт списка имеет номер 3</li>
  <li>Следующий пункт списка</li>
  <li value='7'>Текущий пункт списка
  <li>далее нумерация по порядку</li>
</ol>
```



Форматирование текста в HTML-документе

• **Маркированные списки.**

Используется комбинация двух пар тегов:

` ... ` – установка начала и конца маркированного списка;

` ... ` – выделение отдельных пунктов списка.

Маркировка по умолчанию – черные кружочки.

Изменение типа маркера – с помощью атрибута ***type***.

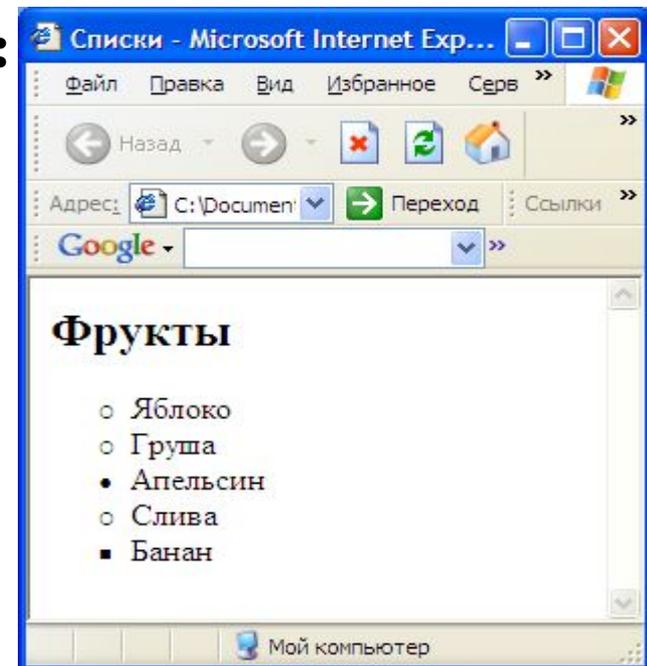
- установка атрибута в теге `` применяет тип маркера ко всему списку;
- установка атрибута в теге `` изменяет маркер текущего пункта списка.

Форматирование текста в HTML-документе

| Значение атрибута TYPE | Тип маркера |
|------------------------|------------------------|
| 'disc' | ● (задан по умолчанию) |
| 'circle' | ○ |
| 'square' | ■ |

Создание маркированного списка:

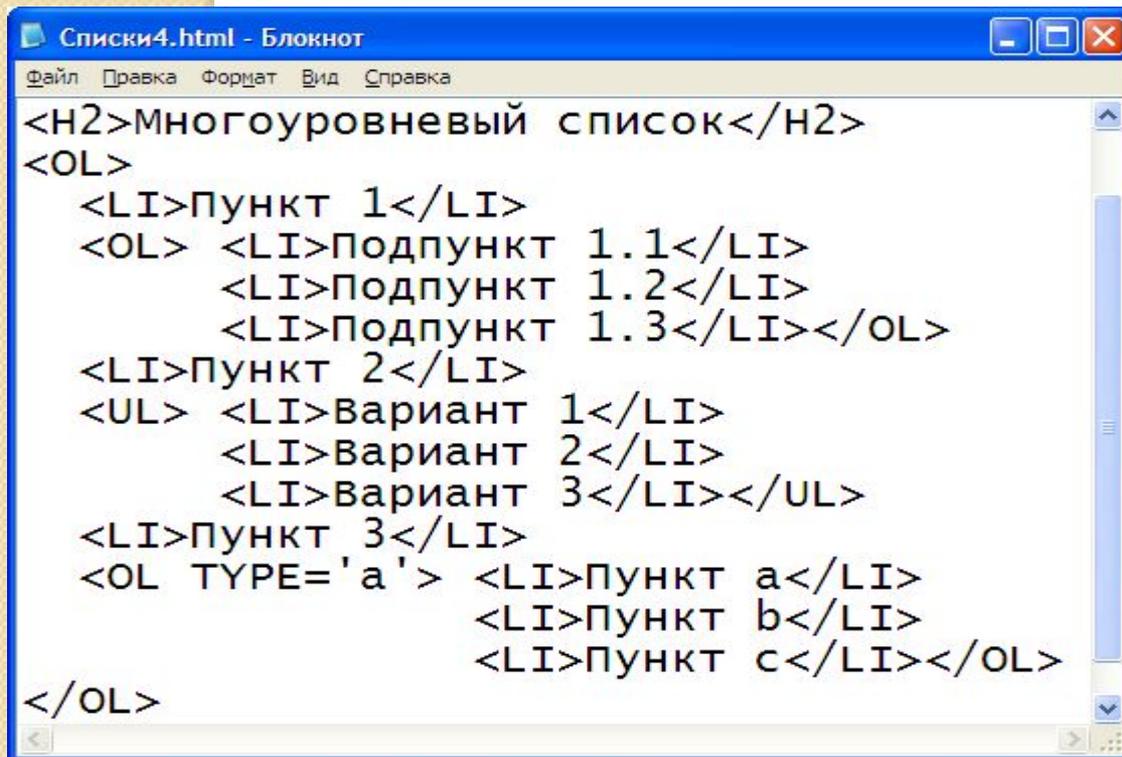
```
Списки3.html - Блокнот
Файл Правка Формат Вид Справка
<H2>Фрукты</H2>
<UL TYPE='circle'>
  <LI>Яблоко</LI>
  <LI>Груша</LI>
  <LI TYPE='disc'>Апельсин</LI>
  <LI>Слива</LI>
  <LI TYPE='square'>Банан</LI>
</UL>
```



Форматирование текста в HTML-документе

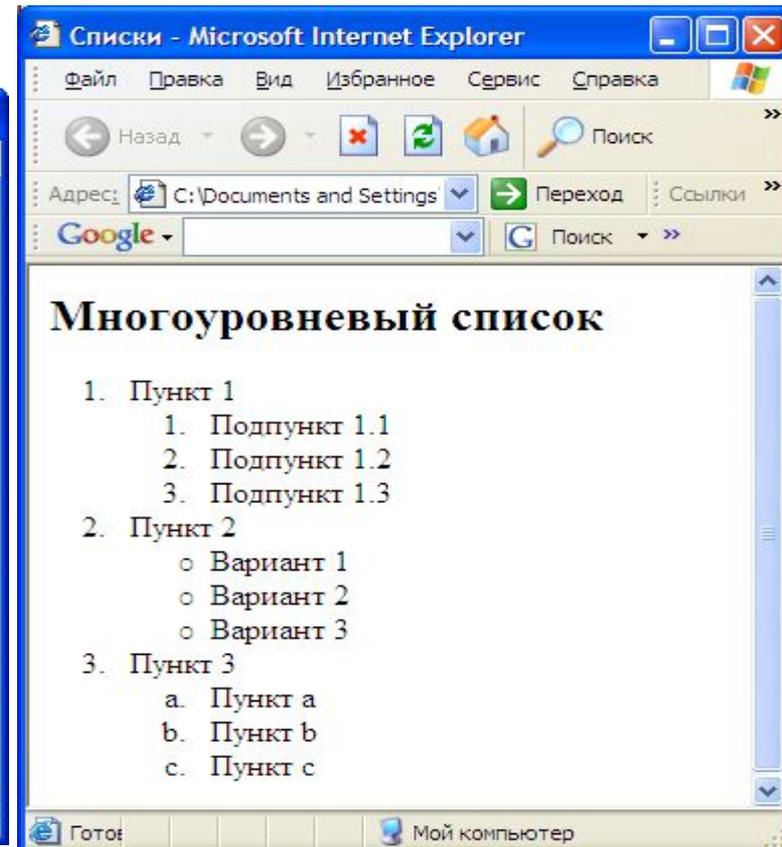
- **Многоуровневые списки.**

Для создания – вставка в список нового списка после пункта, требующего детализации:



```
Списки4.html - Блокнот
Файл  Правка  Формат  Вид  Справка

<H2>Многоуровневый список</H2>
<OL>
  <LI>пункт 1</LI>
  <OL> <LI>Подпункт 1.1</LI>
    <LI>Подпункт 1.2</LI>
    <LI>Подпункт 1.3</LI></OL>
  <LI>пункт 2</LI>
  <UL> <LI>Вариант 1</LI>
    <LI>Вариант 2</LI>
    <LI>Вариант 3</LI></UL>
  <LI>пункт 3</LI>
  <OL TYPE='a'> <LI>пункт a</LI>
    <LI>пункт b</LI>
    <LI>пункт c</LI></OL>
</OL>
```



Форматирование текста в HTML-документе

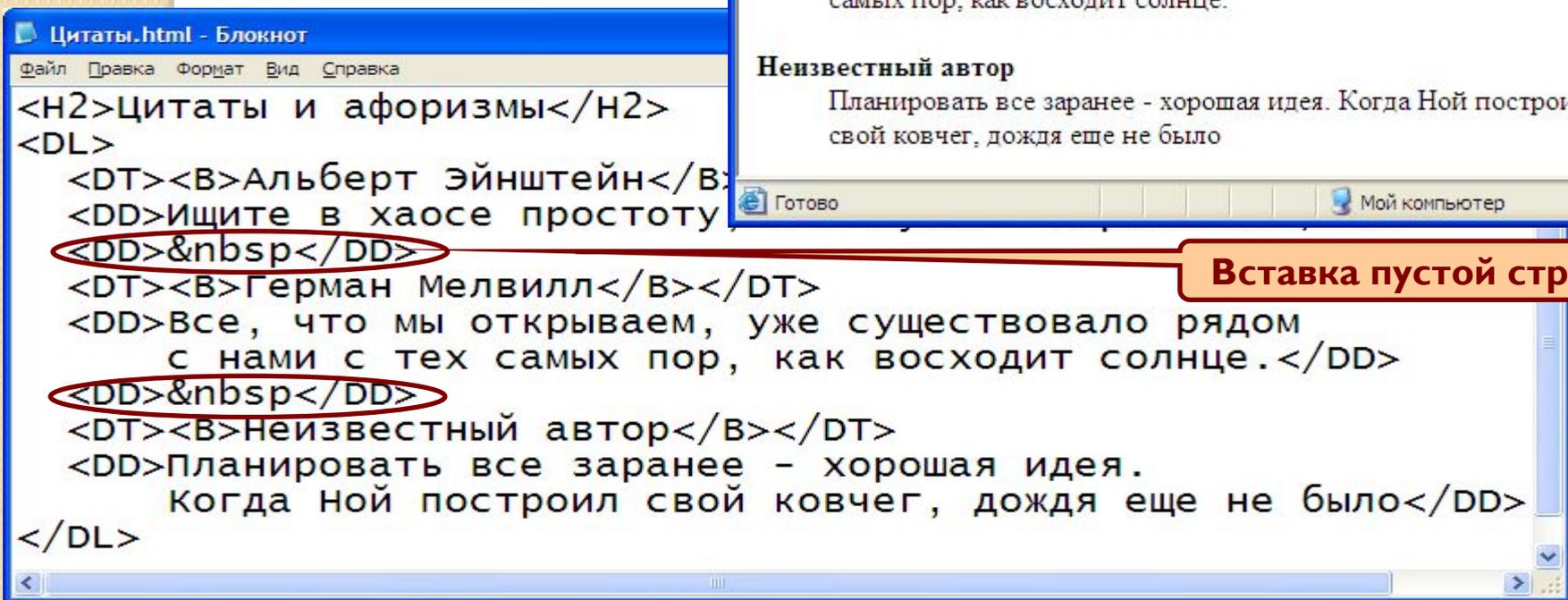
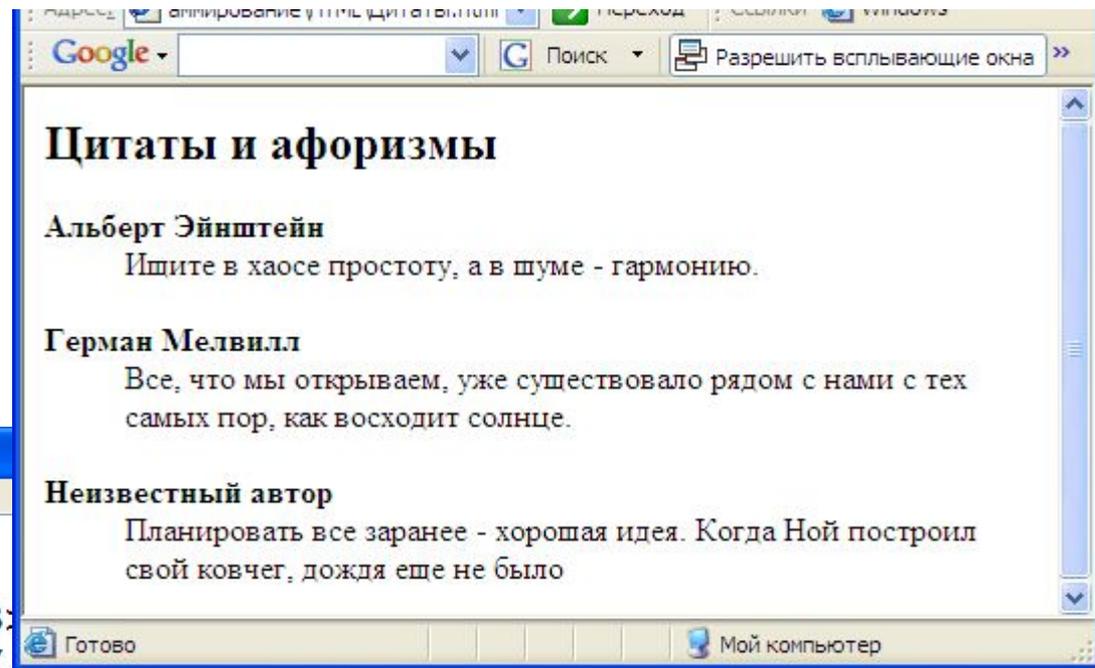
● **Создание отступов текста с помощью элементов списка:**

Список определений предназначен для управления отступами от левого поля страницы. В нем используются теги:

- **<dl> ... </dl>** – начало и окончание списка определений;
- **<dt> ... </dt>** – заголовок термина;
- **<dd> ... </dd>** – определение термина.

Форматирование текста в HTML-документе

Пример:



Вставка пустой строки

Гиперссылки HTML

Гиперссылки – одни из наиболее важных элементов *Web*-страниц. Могут быть 3 видов: служебные (<link>), графические (<area>), обычные для перехода между документами (<a>).

Атрибут **href** является общим у всех гиперссылок и определяет целевой URI ресурс.

Пример гиперссылок:

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

```
<a href = "http://www.yandex.ru">Ссылка на Яндекс</a>
```

```
<area href="URL" />
```

Гиперссылки HTML

Пример:

```
<html>
  <head>
    <title>This ia my test page</title>
    <link rel="stylesheet" type="text/css" href="test.css">
  </head>

  <body>
    <a href = "http://www.yandex.ru">This is hyperlink to Yandex</a><br>
    <a href = "http://w3.org"><img src = "hyperlink.jpg"></a>
  </body>
</html>
```

Основные параметры обычных гиперссылок

- Цвета ссылок по умолчанию:
 - неотработанная – синий,
 - активная – красный,
 - отработанная – фиолетовый.

Можно изменить атрибутами элемента `<body>` - `link`, `alink`, `vlink` соответственно.

- Способ открытия гиперссылки по умолчанию - в текущем окне. Можно изменить атрибутом `target` в элементе `<a>`:
 - `_blank` - загружает страницу в новое окно браузера.
 - `_self` - загружает страницу в текущее окно (по умолчанию).
- В атрибуте `title` можно назначить текст подсказки при наведении курсора на ссылку;
- В разделе `<head>` можно задать базовое поведение гиперссылок в элементе `<base target>`:
`<head> <base target = “_blank”> </head>`

Внутренние ссылки документа (метка, якорь, anchor).

Старый вариант создания:

- **Гиперссылки на закладки в тексте страницы.**

Закладка в тексте документа создается с помощью тега `<a>` с атрибутом *name* вместо *href*.

Например:

`Раздел I`

Создание закладки
с именем chapter I

`Перейти к разделу I`

Создание гиперссылки
на закладку chapter I

Переход по этой гиперссылке приведет не к открытию новой страницы, а к прокручиванию текущего документа до закладки.

Внутренние ссылки документа (метка, якорь, anchor).

Новый вариант создания:

- Создаем метку на раздел документа:

```
<a href="#chapter 1">Перейти к разделу 1</a>
```

- Получаем метку внутри структурного элемента HTML с помощью атрибута id, не создавая дополнительный элемент <name>:

```
<h1 id = "chapter 1">Chapter 1</h1>
```

Связывание Web-страниц с помощью гиперссылок

Можно ссылаться на закладки в других документах.
Имя файла отделяется от имени закладки символом
#.

Например:

**** К разделу I ****

Закладка с именем
chapter1 в документе
book.html

Как не надо делать: mailto

- **Гиперссылки на адреса электронной почты.**

В результате щелчка на такой гиперссылке на компьютере пользователя будет запущено приложение клиента электронной почты, установленное по умолчанию.

В поле **Кому** (*Outlook Express*) автоматически вводится адрес электронной почты, заданный в гиперссылке.

Для создания такой гиперссылки используется служебное слово **mailto:**

Текст гиперссылки



Работа с таблицами в HTML

Создание таблицы

Создание таблицы – парный тег

`<table> ... </table>`.

Создает объект таблицы (пустая таблица, не содержащая ячеек) в том месте текста, где он добавлен в код *HTML*.

Добавление в таблицу строк и ячеек:

- **`<tr> ... </tr>`** – добавление новой строки;
- **`<th> ... </th>`** – добавление ячейки заголовка;
- **`<td> ... </td>`** – добавление обычной ячейки.

Создание таблицы

Создание
простейшей
таблицы: 4 строки
(одна –
заголовки), 2
столбца).

**Строки
комментариев**

```
Таблицы1.html - Блокнот
Файл  Правка  Формат  Вид  Справка
<H2>Расписание на понедельник</H2>
<BR>
<!--Создание объекта таблицы-->
<TABLE>
<!--Создание новой строки-->
<TR>
  <!--Создание 2 ячеек заголовков-->
  <TH>&nbsp;</TH><TH>&nbsp;</TH>
</TR>
<!--Создание новой строки-->
<TR>
  <!--Создание 2 обычных ячеек-->
  <TD>&nbsp;</TD><TD>&nbsp;</TD>
</TR>
<!--Создание новой строки-->
<TR>
  <!--Создание 2 обычных ячеек-->
  <TD>&nbsp;</TD><TD>&nbsp;</TD>
</TR><!--Создание новой строки-->
<TR>
  <!--Создание 2 обычных ячеек-->
  <TD>&nbsp;</TD><TD>&nbsp;</TD>
</TR>
</TABLE>
```

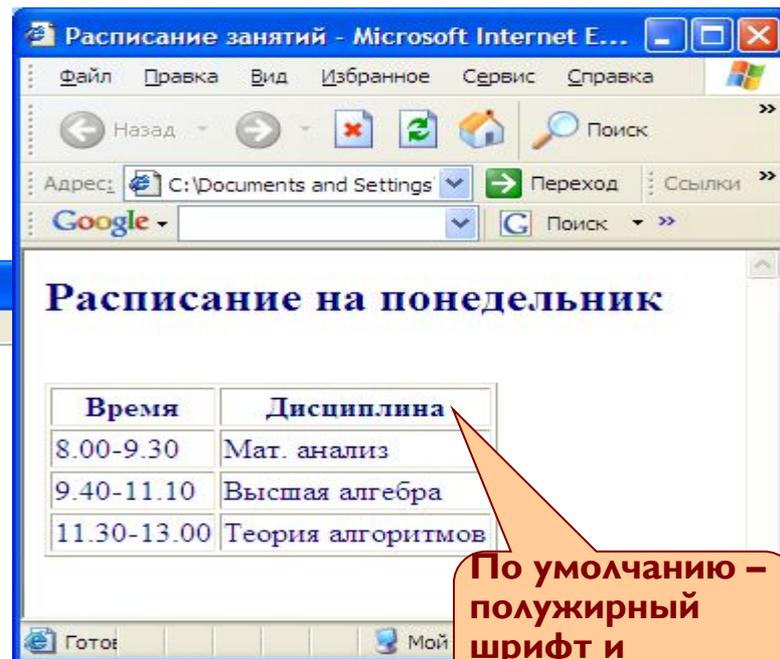
Создание таблицы

Пример.

Заполнение ячеек данными:

```
Таблицы1.html - Блокнот
Файл Правка Формат Вид Справка
<BODY BGCOLOR=white TEXT=navy>
<H2>Расписание на понедельник</H2>
<BR>
<TABLE BORDER='1'>
<TR>
    <TH>Время</TH><TH>Дисциплина</TH>
</TR>
<!--Создание новой строки-->
<TR>
    <TD>8.00-9.30</TD><TD>Мат. анализ</TD>
</TR>
<TR>
    <TD>9.40-11.10</TD><TD>Высшая алгебра</TD>
</TR>
<TR>
    <TD>11.30-13.00</TD><TD>Теория алгоритмов</TD>
</TR>
</TABLE>
```

Отображение границ

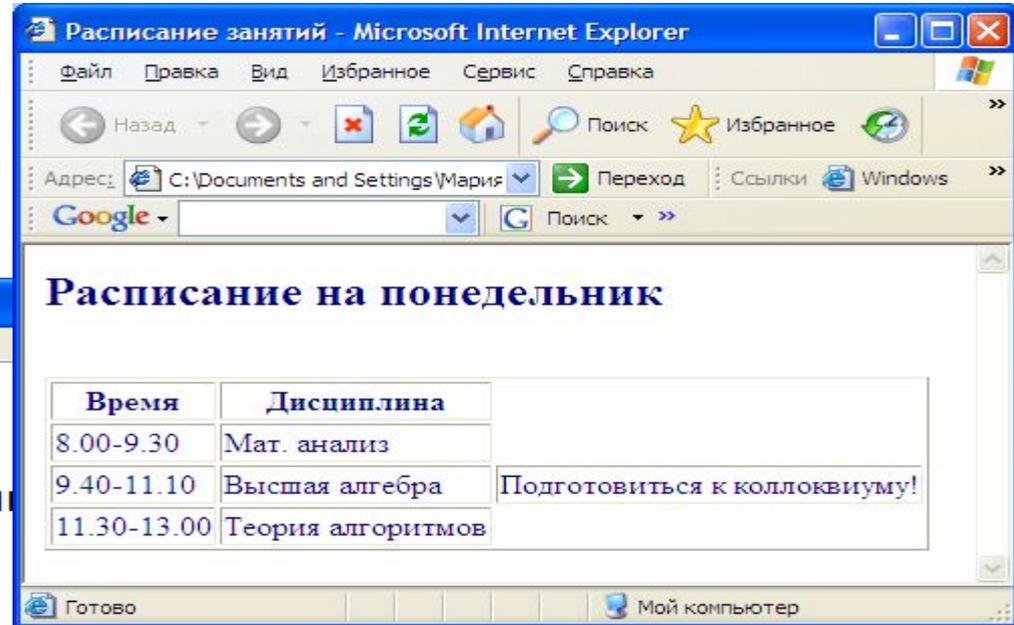


По умолчанию – полужирный шрифт и выравнивание по центру

Создание таблицы

Строки таблицы не обязательно должны содержать одинаковое число ячеек.

```
Таблицы1.html - Блокнот
Файл Правка Формат Вид Справка
<BR>
<TABLE BORDER='1'>
<TR>
    <TH>Время</TH><TH>Дисциплина</TH>
</TR>
<!--Создание новой строки-->
<TR>
    <TD>8.00-9.30</TD><TD>Мат. анализ</TD>
</TR>
<TR>
    <TD>9.40-11.10</TD><TD>Высшая алгебра</TD>
    <TD>Подготовиться к коллоквиуму!</TD>
</TR>
<TR>
    <TD>11.30-13.00</TD><TD>Теория алгоритмов</TD>
</TR>
</TABLE>
```



Создание таблицы

Добавление заголовков таблицы

Использование тегов `<h1>` - `<h6>` может создать проблемы с выравниванием заголовка относительно таблицы.

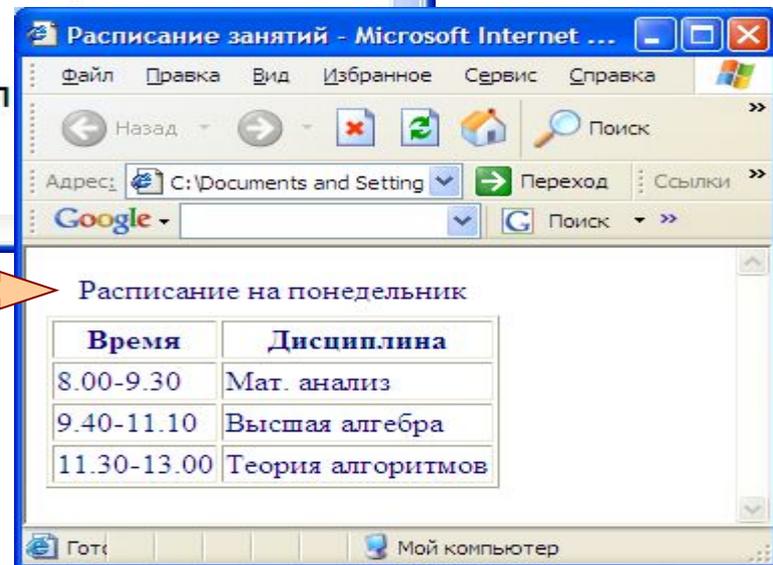
Тег `<caption> ... </caption>` создает заголовок непосредственно в таблице.

Выравнивание по умолчанию – по центру таблицы.

Для изменения – атрибут `align` этого тега.

Создание таблицы

```
Таблицы2.html - Блокнот
Файл Правка Формат Вид Справка
<TABLE BORDER='1'>
<CAPTION>Расписание на понедельник</CAPTION>
<TR>
  <TH>Время</TH><TH>Дисциплина</TH>
</TR>
<!--Создание новой строки-->
<TR>
  <TD>8.00-9.30</TD><TD>Мат. анализ</TD>
</TR>
<TR>
  <TD>9.40-11.10</TD><TD>Высшая алгебра</TD>
</TR>
<TR>
  <TD>11.30-13.00</TD><TD>Теория ал
</TR>
</TABLE>
```



Создание таблицы

- **Форматирование текста в ячейках таблицы.**

- Выравнивание текста.

По горизонтали – атрибут ***align***:

- ***left*** – влево;
- ***center*** – по центру;
- ***right*** – вправо.

По вертикали – атрибут ***valign***:

- ***top*** – вверх;
- ***middle*** – по центру;
- ***bottom*** – вниз.

Создание таблицы

- **Группирование и объединение ячеек.**

Используется для создания ячейки, занимающей несколько столбцов и/или строк.

Для этого в тегах `<td>` или `<th>` устанавливаются параметры:

- ***colspan*** – число столбцов, занятых ячейкой;
- ***rowspan*** – число строк, занятых ячейкой.

Следует учитывать: при объединении n столбцов (строк) в текущей строке (текущем столбце) остается на $n-1$ ячеек меньше.

Создание таблицы

- **Группирование ячеек.**

Позволяет форматировать группу ячеек как одно целое.

- Группирование столбцов.

Используются парные теги:

- **<colgroup>** – структурная группа столбцов; применяется для разбивки таблицы на столбцы разных типов (например: столбцы заголовков и столбцы данных).
- **<col>** – неструктурная группа столбцов; применяется для произвольного группирования столбцов таблицы, имеющих общий формат данных.

Помещаются в код между тегами **<table>** и созданием первой строки с помощью **<tr>**.

Создание таблицы

- Группирование строк.

Используются парные теги:

- **<thead>** – создает группу строк заголовков. Этот тег можно использовать в коде таблицы только один раз.
- **<tbody>** – используется для создания одной или нескольких групп строк.
- **<tfoot>** – создает группу строк нижнего колонтитула таблицы. Этот тег можно использовать в коде таблицы только один раз.

Создание таблицы

- **Установка размеров таблицы.**

Атрибуты ***width*** (ширина) и ***height*** (высота) тега **`<table>`**.

Задаются в пикселях или в процентах от размеров *окна браузера* (установка одного атрибута в процентах не приводит к пропорциональному изменению второго атрибута).

- **Установка размеров отдельных ячеек.**

Атрибуты ***width*** (ширина) и ***height*** (высота) тегов **`<td>`** или **`<th>`**.

Задаются в пикселях или в процентах от размеров *таблицы* (установка одного атрибута в процентах приведет к пропорциональному изменению второго атрибута).



HTML-формы

Форма HTML

Форма - один из важных элементов любого HTML-документа, предназначенная для обмена данными между пользователем и сервером. Позволяет организовать пользовательский интерфейс web-приложения для отправки данных (пользователем) на сервер.

Также с помощью клиентских скриптов можно получить доступ к любому элементу формы, изменять его и применять по своему усмотрению.

Создание формы

- Форма создается с помощью тэга `<form>...</form>`, внутри которого размещаются элементы обработки и/или передачи информации.
- Элемент `<form>` имеет атрибуты, определяющие способ представления и обработки данных;
- Элемент `<form>` - блочный, внутри него могут быть любые элементы, кроме другой формы;
- На одной странице могут быть несколько форм; при этом формы не могут быть вложены друг в друга (см. пункт выше);
- Информация из формы передается браузером на сервер, где, как правило, обрабатывается серверной технологией (PHP, ASP.NET и т.п.).

Основные атрибуты <form>

```
<form action = "file.php" method = "POST"
```

```
enctype =
```

```
"multipart/form-data">
```

```
<!-- Элементы внутри формы -->
```

```
</form>
```

Атрибуты являются обязательными; если их значения не указаны – браузер подставляет значения по умолчанию.

- Action – определяет адрес документа, обрабатывающего данные из формы (по умолчанию – текущий документ).
- Method – определяет используемый метод отправки данных (GET или POST; по умолчанию - GET).
- Enctype – определяет способ кодирования данных (по умолчанию - application/x-www-form-urlencoded).

Элементы формы:

однострочное текстовое поле

Однострочное текстовое поле предназначено для ввода одной строки символов с помощью клавиатуры. В HTML 4.01:

```
<input type = "text" ... />  
<input type = "password" ... />
```

Возможные атрибуты:

- Name – обязательный атрибут, предназначенный для идентификации поля обработчиком;
- Value – начальный текст, отображаемый в поле;
- Size – размер (ширина) поля, определяемая числом символов (password - звездочек) моноширинного шрифта.
- Maxlength – максимальное количество вводимых символов.

Элементы формы: однострочное текстовое поле

```
<html>
  <head>
    <title>Example</title>
  </head>
  <body>
    <form action="file.php" method = "POST" enctype = "multipart/form-data">
      <p>Login: <input type = "text" name = "login" maxlength = "10"></p>
      <p>Password: <input type = "password" name = "psw" maxlength = "5"></p>
      <!-- Другие элементы формы -->
    </form>
  </body>
</html>
```

Login:

Password:

Элементы формы:

многострочное текстовое поле

Многострочное текстовое поле предназначено для создания текстовой области, состоящей из нескольких строк.

<textarea атрибуты>

Текст

</textarea>

Основные атрибуты:

Name – имя поля;

Rows – высота поля в строках текста;

Cols – ширина поля;

Maxlength – максимально возможное количество символов;

Элементы формы: многострочное текстовое поле

```
<form action="file.php" method = "POST" enctype = "multipart/form-data">
  <h2>This is a text area:</h2>
  <p>
    <textarea name = "MyText" cols = "20" rows = "5">
      This is start value, you can change it
    </textarea>
  </p>

  <!-- Другие элементы формы -->
</form>
```

This is a text area:

This is start value,
you can change it

Элементы формы:

многострочное текстовое поле

Поле может находиться в состояниях «блокировано» (disabled) и «только для чтения» (readonly).

Текст внутри заблокированного поля нельзя выделить и добавить, также содержимое такого поля не отправляется формой на сервер.

Текст внутри поля для чтения доступен для копирования, но его нельзя отредактировать.

```
<textarea name = "MyText" disabled>...</textarea>
```

```
<textarea name = "MyText" readonly>...</readonly>
```

Элементы формы:

многострочное текстовое поле

Спецификация HTML5 не определяет вид поля и текста в подобных состояниях, поэтому браузеры по-разному его отображают:

Internet Explorer

Opera

Chrome

Firefox

Элементы формы:

КНОПКИ

<input type = “button”...>

<input type = “submit”...>

<input type = “reset”...>

Button – простая кнопка; для нее пишется обработчик;

Submit – передача данных из формы обработчику;

Reset – кнопка сброса значений формы к первоначальным значениям.

Атрибуты кнопок:

Name – имя кнопки (для обработчика);

Value - значение кнопки (надпись).

Элементы формы:

КНОПКИ

```
<form action="file.php" method = "POST" enctype = "multipart/form-data">
  <p>Login: <input type = "text" name = "login" maxlength = "10"></p>
  <p>Password: <input type = "password" name = "psw" maxlength = "5"></p>
  <p><input type = "submit" name = "submit" value = "send!"></p>
</form>
```

Login:

Password:

```
<form>
  <button name = "myButton">
    <img src = "dog_small.jpg" title = "This is a button too">
  </button>
</form>
```



This is a button too

Элементы формы:

переключатели (radio)

Переключатели позволяют выбрать единственный вариант ответа из нескольких предложенных.

```
<input type="radio" ...>
```

Атрибуты:

Name – имя **группы** переключателей (должно быть одинаковым у всех элементов группы);

Checked – определяет выбранный по умолчанию элемент;

Value – определяет значение, которое будет оправлено обработчику (следовательно, должно быть уникальным для каждого элемента группы).

Элементы формы: переключатели (radio)

```
<form action="file.php" method = "POST" enctype = "multipart/form-data">
  <p><input type = "radio" name = "MyRadio" value = "First">First Variant</p>
  <p><input type = "radio" name = "MyRadio" value = "Second">Second Variant</p>
  <p><input type = "radio" name = "MyRadio" value = "Third">Third Variant</p>
  <p><input type = "submit" name = "submit">
</form>
```

- First Variant
- Second Variant
- Third Variant

Отправить

Элементы формы: флажки (checkbox)

Флажки используются для выбора любого количества вариантов из предложенного списка.

```
<input type="checkbox" ...>
```

Атрибуты аналогичны атрибутам radio.

Элементы формы: флажки (checkbox)

```
<form action="file.php" method = "POST" enctype = "multipart/form-data">
  <p><input type = "checkbox" name = "MyCheckBox" value = "First">First Variant</p>
  <p><input type = "checkbox" name = "MyCheckBox" value = "Second">Second Variant</p>
  <p><input type = "checkbox" name = "MyCheckBox" value = "Third">Third Variant</p>
  <p><input type = "submit" name = "submit">
</form>
```

First Variant

Second Variant

Third Variant

Отправить

Элементы формы:

поле со списком

Позволяет организовать выпадающее меню.

```
<select атрибуты>
```

```
  <option атрибуты>Пункт 1</option>
```

```
<option атрибуты>Пункт 2</option>
```

```
</select>
```

Select – контейнер для элементов списка;

Option – элементы списка.

Элементы формы:

поле со списком

Атрибуты элемента <select>

Name – имя элемента;

Size – высота списка (количество одновременно видимых элементов);

Multiple – возможность множественного выбора (с использованием клавиш Ctrl и Shift).

Элементы формы:

поле со списком

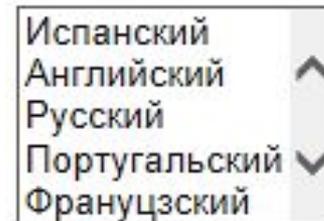
Основные атрибуты элемента <option>

Selected – определяет выделенный элемент списка;

Value – определяет отправляемое на сервер значение.

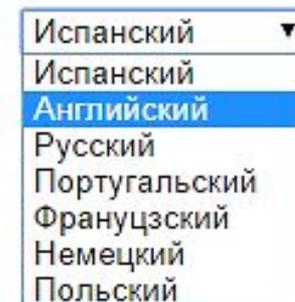
Элементы формы: поле со списком

```
<form action="file.php" method = "POST" enctype = "multipart/form-data">  
  <select name = "languages" size = "5">  
    <option value = "Sp">Испанский</option>  
    <option value = "En">Английский</option>  
    <option value = "Rus">Русский</option>  
    <option value = "Pr">Португальский</option>  
    <option value = "Fr">Французский</option>  
    <option value = "Gr">Немецкий</option>  
    <option value = "Pl">Польский</option>  
  </select>  
</form>
```



| | |
|---------------|---|
| Испанский | |
| Английский | ▲ |
| Русский | |
| Португальский | ▼ |
| Французский | |

```
<form action="file.php" method = "POST" enctype = "multipart/form-data">  
  <select name = "languages" size = "1">  
    <option value = "Sp">Испанский</option>  
    <option value = "En">Английский</option>  
    <option value = "Rus">Русский</option>  
    <option value = "Pr">Португальский</option>  
    <option value = "Fr">Французский</option>  
    <option value = "Gr">Немецкий</option>  
    <option value = "Pl">Польский</option>  
  </select>  
</form>
```



| | |
|---------------|---|
| Испанский | ▼ |
| Испанский | |
| Английский | |
| Русский | |
| Португальский | |
| Французский | |
| Немецкий | |
| Польский | |

Элементы формы: поле со списком

Группировка элементов поля со списком:

```
<form action="file.php" method = "POST" enctype = "multipart/form-data">  
  <select name = "languages" size = "1">  
    <optgroup label = "Романская группа">  
      <option value = "Sp">Испанский</option>  
      <option value = "Pr">Португальский</option>  
      <option value = "Fr">Французский</option>  
    </optgroup>  
    <optgroup label = "Германская группа">  
      <option value = "En">Английский</option>  
      <option value = "Gr">Немецкий</option>  
    </optgroup>  
    <optgroup label = "Славянская группа">  
      <option value = "Rus">Русский</option>  
      <option value = "Pl">Польский</option>  
    </optgroup>  
  </select>  
</form>
```

| | |
|-------------------|---|
| Испанский | ▼ |
| Романская группа | |
| Испанский | |
| Португальский | |
| Французский | |
| Германская группа | |
| Английский | |
| Немецкий | |
| Славянская группа | |
| Русский | |
| Польский | |

Элементы формы:

скрытое поле

Возникают ситуации, когда требуется передать в форме некоторые промежуточные данные, которые носят технический характер и должны быть скрыты от пользователя. Для этой цели применяется скрытое поле, которое не отображается на странице.

<input type = “hidden”...>

Атрибуты

Name – имя поля;

Value – передаваемая на обработку информация.

Элементы формы:

поле для загрузки прикрепленных файлов

Для вызова программной утилиты выбора загружаемого файла используется поле

```
<input type = "file" name = "...">
```

Атрибуты

Name – определяет имя элемента;

Accepted – определяет фильтр на типы файлов;

Size – определяет ширину поля;

Multiple – позволяет загружать одновременно несколько файлов.

По-разному отображается в различных браузерах.

Элементы формы: поле для загрузки прикрепленных файлов

```
<form action="file.php" method = "POST" enctype = "multipart/form-data">  
  <input type = "file">  
</form>
```

Выберите файл Файл не выбран

```
<form action="file.php" method = "POST" enctype = "multipart/form-data">  
  <input type = "file" name = "files" multiple>  
</form>
```

Обзор...

Элементы формы:

поле для загрузки прикрепленных файлов

Некоторые допустимые значения атрибута accept:

| Значение | Описание |
|----------------------|------------------------------------|
| image/jpeg | Только файлы в формате JPEG. |
| image/jpeg,image/png | Только файлы в формате JPEG и PNG. |
| image/* | Любые графические файлы. |
| image/*,video/* | Любые графические и видеофайлы. |

```
<input type="file" name="img" multiple  
accept="image/*,image/jpeg">
```

Элементы формы: некоторые возможности HTML 5

<input type="email" ... >

<input type="url" ... >

<input type="color" ... >

<input type="number" ... >

<input type="date" ... >

<input type="datetime" ... >

Группировка элементов форм

При создании сложной формы можно группировать некоторые элементы форм между собой, такое группирование удобно для пользователя и позволяет визуально отделить один логический блок от другого.

```
<fieldset>
```

```
  <legend>Заголовок</legend>
```

```
  <!-- Elements -->
```

```
</fieldset>
```

Группировка элементов форм

```
<form>
  <fieldset>
    <legend>Group</legend>
    <p>Login: <input name="login"></p>
    <p>Password: <input type="password" name="pass"></p>
    <p><input type="submit" value="Enter"></p>
  </fieldset>
</form>
```

Group

Login:

Password: