

Создание списков

Списки – средство структурирования документа

Типы списков

- Неупорядоченный (маркированный)
 - Список элементов меню
 - Список элементов каталогов
- Упорядоченный (нумерованный)
- Список определений

Неупорядоченные списки

``

`` Первый пункт списка

`` Второй пункт списка

`` Третий пункт списка

``

`` Первый подпункт

`` Второй подпункт

``

``

`<LI TYPE=тип>` - маркер элемента списка

`<UL TYPE="disk" | "circle" | "square">`-весь список

Упорядоченные списки

```
<OL TYPE="I" START="2">  
  <LI> Пункт два  
  <LI> Пункт три  
  <LI> Пункт четыре  
</OL>
```

```
<OL TYPE="A" START="2">  
  <LI> Пункт, озаглавленный буквой В.  
  <LI VALUE="6"> Пункт, озаглавленный буквой F.  
  <LI> Пункт, озаглавленный буквой G.  
</OL>
```

Параметры:

START - определяет первое число, с которого начинается нумерация пунктов. (только целые числа). **VALUE** – изменение нумерации внутри списка

TYPE - определяет стиль нумерации пунктов. Может иметь значения:

"A" - заглавные буквы A, B, C ...

"a" - строчные буквы a, b, c ...

"I" - большие римские числа I, II, III ...

"i" - маленькие римские числа i, ii, iii ...

"1" - арабские числа 1, 2, 3 ...

По умолчанию используется **TYPE="1"**.

- Первый пункт списка
- Второй пункт списка
- Третий пункт списка
 - Первый подпункт
 - Второй подпункт

- II. Пункт два
- III. Пункт три
- IV. Пункт четыре

- V. Пункт, озаглавленный буквой В.
- F. Пункт, озаглавленный буквой F.
- G. Пункт, озаглавленный буквой G.

Списки определений

Список определений - ряд терминов и их соответствующая расшифровка. Определение термина обычно размещается после его наименования.

Формат:

```
<DL>
<DT> Термин-1
<DD> Определение-1
<DT> Термин-2
<DD> Определение-2
</DL>
```

Например:

```
<DL>
<DT> NCSA
<DD> NCSA (National Center for
      Supercomputing Applications).
<DT> CTC
<DD> CTC (Cornell Theory Center).
</DL>
```

В окне браузера это будет выглядеть так:

```
NCSA
  NCSA (National Center for
  Supercomputing Applications).
CTC
  CTC (Cornell Theory Center).
```

Примечание: В документе не может быть подряд несколько элементов <DD>.

Если список очень большой, можно использовать дополнительный атрибут COMPACT в рамках элемента <DL>. Таким образом вы указываете браузеру, что он должен использовать самый маленький шрифт для вывода списка. Например:

```
<DL COMPACT>
```

```
<DT> GOOD
```

```
<DD> Enjoyable, desirable, pleasant, happy, etc., in compact format
```

```
<DT> Book
```

```
<DD> A literary or scientific work, in compact format.
```

```
</DL>
```

В окне браузера это будет выглядеть так:

GOOD

Enjoyable, desirable, pleasant, happy, etc., in compact format

Book

A literary or scientific work, in compact format.

Элементы <DIR>, <MENU>

(Список с короткими названиями пунктов, список пунктов)

Для того, чтобы вывести список с короткими названиями пунктов (менее 20 символов), можно использовать тег <DIR>. Обычно такой список выводится в колонках по 24 символа. Данный элемент поддерживает атрибут COMPACT. Например:

```
<DIR>  
<LI> A-F  
<LI> G-M  
<LI> N-S  
<LI> T-Z  
</DIR>
```

В окне браузера это будет выглядеть так:

- A-F
- G-M
- N-S
- T-Z

Используя элемент <DIR> можно вывести список пунктов по колонкам. Для вывода каждого пункта в отдельной строке можно использовать элемент <MENU>:

```
<MENU>  
<LI> Первый пункт меню  
<LI> Второй пункт меню  
<LI> Третий пункт меню  
</MENU>
```

В окне браузера это будет выглядеть так:

- Первый пункт меню
- Второй пункт меню
- Третий пункт меню

Вложенные списки

Один список может быть вложен в другой

```
<ol>
  <li>Документы:
    <ul>
      <li>Паспорт</li>
      <li>Билеты</li>
      <li>Гостиница - бронь</li>
    </ul>
  </li>
  <li>Предметы личной гигиены:
    <ul>
      <li>Зубная щетка</li>
      <li>Паста</li>
      <li>Мыло</li>
    </ul>
  </li>
</ol>
```

1. Документы:
 - Паспорт
 - Билеты
 - Гостиница - бронь
2. Предметы личной гигиены:
 - Зубная щетка
 - Паста
 - Мыло

Таблицы

Элемент <TABLE>

Используется для создания таблиц в HTML-документах. Таблица начинается с необязательного заголовка, за которым следует одна или более строк. Каждая из этих строк состоит из одной либо нескольких ячеек, представляющих заголовки и ячейки с данными. Таблица может содержать параграфы, списки, заголовки, формы, рисунки и отформатированный текст.

По умолчанию браузер выводит текст по центру в ячейке заголовка таблицы, а в ячейках с данными - с выравниванием по левому краю.

Данный элемент поддерживает атрибуты: ALIGN, BORDER, CELLPADDING, CELLSPACING, WIDTH.

Таблицы в HTML формируются построчно. Сначала с помощью элемента TR необходимо создать ряд таблицы, в который затем элементом TD помещаются ячейки.

Элемент <TH>

Используется для определения ячеек заголовка таблицы. Обычно заголовков браузер выводит более крупным и жирным шрифтом нежели данные.

Данный элемент поддерживает атрибуты: ALIGN, COLSPAN, HEIGHT, NOWRAP, VALIGN, WIDTH .

Элемент <TD>

Используется для определения атрибутов отдельных ячеек таблицы. Данный элемент поддерживает атрибуты: ALIGN, COLSPAN, HEIGHT, NOWRAP, VALIGN, WIDTH .

Элемент <TR>

Используется для формирования строк в таблице. С его помощью устанавливается вертикальное и горизонтальное выравнивание содержимого ячеек по всей строке. Можно также запретить перенос слов в любой строке таблицы.

Данный элемент поддерживает атрибуты: ALIGN, VALIGN.

Элемент <CAPTION>

Используется при форматировании наименований таблиц. Поддерживает атрибут ALIGN.

Например:

```
<CAPTION ALIGN=CENTER> Таблица расходов </CAPTION>
```

Таблицы

```
<TABLE BORDER>  
  <TR>  
    <TD>A1</TD> <TD>B1</TD> <TD>C1</TD>  
  </TR>  
  <TR>  
    <TD>A2</TD> <TD>B2</TD> <TD>C2</TD>  
  </TR>  
</TABLE>
```

A1	B1	C1
A2	B2	C2

Параметры тегов TR, TH, TD

- **ALIGN** - определяет способ горизонтального выравнивания содержимого ячейки. Возможные значения: **left**, **center**, **right**. По умолчанию способ выравнивания определяется значением параметра ALIGN элемента TR. Если же и он не задан, то для TD выполняется выравнивание по левому краю, а для TH - центрирование.
- **VALIGN** - определяет способ вертикального выравнивания содержимого ячейки. Возможные значения: **top**, **bottom**, **middle**. По умолчанию происходит выравнивание по центру (*VALIGN="middle"*), если значение этого параметра не было задано ранее в элементе [TR](#).
- **WIDTH** - определяет ширину ячейки. Ширина задается в пикселах или в процентном отношении к ширине таблицы.
- **HEIGHT** - определяет высоту ячейки. Высота задается в пикселах или в процентном отношении к ширине таблицы.
- **COLSPAN** - определяет количество столбцов, на которые простирается данная ячейка.
- **ROWSPAN** - определяет количество рядов, на которые простирается данная ячейка.
- **NOWRAP** - блокирует автоматический перенос слов в пределах текущей ячейки.
- **BGCOLOR** - определяет цвет фона ячейки.
- **BACKGROUND** - заполняет ячейку фоновым рисунком. Необходимо указать URL рисунка.

Слияние ячеек

```
<TABLE BORDER>
```

```
<TR>
```

```
<TH ROWSPAN=2>HDD</TH>
```

```
<TD>WD Caviar 3.1Gb</TD><TD ALIGN="right">85$</TD>
```

```
</TR>
```

```
<TR>
```

```
<TD>Quantum FB ST 6.4Gb</TD><TD ALIGN="right">110$</TD>
```

```
</TR>
```

```
</TABLE>
```

HDD	WD Caviar 30 Gb	85\$
	Quantum 40 Gb	110\$

Слияние ячеек

```
<TABLE BORDER>  
  <TR>  
    <TH COLSPAN=2>Video</TH>  
  </TR>  
  <TR>  
    <TD>Matrox G400</TD><TD ALIGN="right">115$</TD>  
  </TR>  
  <TR>  
    <TD>Voodoo III</TD><TD ALIGN="right">129.95$</TD>  
  </TR>  
</TABLE>
```

Video	
Matrox G400	115\$
Voodoo III	129.95\$

Фреймы

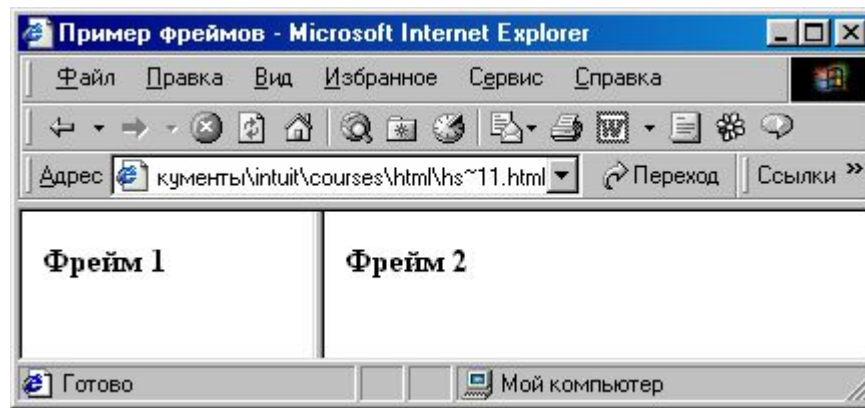
Вводя тег `<FRAME>`, дизайнер HTML-страницы разделяет экран браузера на части.

Задание фреймовой структуры

Для начала мы должны представить себе общий вид страницы – где расположить *фреймы* и какого они будут размера. Затем можно подумать об их содержании. Ниже приводится код простой фреймовой *структуры* с использованием тега `<FRAMESET>`.

Обратите внимание: страница с фреймовой *структурой* не содержит тега `<BODY>`.

```
<HTML>
<HEAD>
<TITLE>Пример фреймов</TITLE>
</HEAD> <FRAMESET COLS="25%, 75%">
<FRAME SRC="menu.html"> <FRAME SRC="main.html"
NAME="main">
</FRAMESET>
</HTML>
```



Определения атрибутов `rows = length-list`

Этот атрибут специфицирует выкладку горизонтальных рамок. Значение представляет собой список длин, разделенных запятыми. Если атрибут не задан, значение по умолчанию равно 100%.

`cols = length-list`

Этот атрибут специфицирует выкладку вертикальных рамок. Значение представляет собой список длин, разделенных запятыми. Если атрибут не задан, значение по умолчанию равно 100%.

Все рамки предполагаются прямоугольными. Установка атрибута `rows` определяет число рамок по горизонтали, а атрибут `cols` задает число рамок по вертикали. Таким образом, задается сетка рамок. Если атрибут `rows` не задан, каждая колонка занимает всю длину страницы. Если атрибут `cols` не задан, каждый ряд занимает всю ширину страницы. Если не заданы оба атрибута, на странице присутствует одна рамка, занимающая всю страницу.

Размер может задаваться в пикселях (абсолютно), в процентах от размеров экрана или в относительных длинах в форме i^* , где i - целое число. Когда заданы оба атрибута, агент пользователя сначала выделяет размеры, заданные абсолютно, затем оставшуюся часть делит в соответствии с определенными долями. Значение $*$ эквивалентно 1^* . Отображение страницы осуществляется слева направо и сверху вниз. Пример (экран делится на две равные части, верхнюю и нижнюю):

```
<frameset rows="50%, 50%">  
... остальная часть определения ...  
</frameset>
```

В следующем примере создается три колонки: вторая имеет фиксированную ширину в 250 пикселей (что удобно для картинки известного размера). Первая получает 25% оставшегося пространства, а третья - 75%.

```
<frameset cols="1*,250,3*">  
... остальная часть определения ...  
</frameset>
```

В следующем примере создается решетка 2x3

```
<frameset rows="30%,70%" cols="33%,34%,33%">  
... остальная часть определения ...  
</frameset>
```

Задание содержимого фрейма

Тег `<FRAME>` определяет внешний вид и поведение *фрейма*. Этот тег не имеет закрывающего тега, поскольку в нем ничего не содержится. Вся суть тега `<FRAME>` в его атрибутах. Их шесть: `NAME`, `MARGINWIDTH`, `MARGINHEIGHT`, `SCROLLING`, `NORESIZE` и `SRC`.

Атрибут `NAME`

Если вы хотите, чтобы при щелчке мышью на ссылке соответствующая страница отображалась в определенном *фрейме*, необходимо указать этот *фрейм*, чтобы страница "знала", что куда загружать. Например, можно записать такую строку:

```
<FRAME SRC="URL" NAME="имя_окна">
```

Атрибут SRC

Атрибут SRC применяется в теге FRAME при разработке фреймовой *структуры* для того, чтобы определить, какая страница появится в том или ином кадре. Если вы зададите атрибут SRC не для всех *фреймов*, у вас возникнут проблемы. Даже если страницы, отображаемые во *фрейме*, выбираются в соседнем *фрейме*, вы должны по крайней мере задать для каждого *фрейма* начальную страницу. Если вы не укажете начальную страницу и URL, *фрейм* окажется пустым, а результаты могут быть самыми неожиданными.

Атрибут TARGET

Определяет окно(фрейм), на которое указывает гипертекстовая ссылка. Этот параметр используется только совместно с параметром HREF.

```
<A HREF=«новый URL» TARGET=«имя_окна»> текст ссылки  
</A>
```

В качестве значения необходимо задать либо имя одного из существующих фреймов, либо одно из следующих зарезервированных имен:

_self - указывает, что определенный в параметре HREF документ должен отображаться в текущем фрейме;

_parent - указывает, что документ должен отображаться во фрейме-родителе текущего фрейма. Иначе говоря, `_parent` ссылается на окно, содержащее FRAMESET, включающий текущий фрейм;

_top - указывает, что документ должен отображаться в окне-родителе всей текущей фреймовой структуры;

_blank - указывает, что документ должен отображаться в новом окне.

Навигационные карты

Карты, обрабатываемые на клиентской машине браузером, создаются с помощью элементов MAP и AREA представляют собой гиперссылки, оформленные в виде активных областей на изображении. Применить созданную карту к изображению можно вызвав ее по имени с помощью атрибута USEMAP элемента IMG.

MAP

Создает новую навигационную карту. Между начальным и конечным тегами содержит один или несколько элементов AREA, определяющих навигационные области карты.

Атрибуты

NAME - единственный и обязательный атрибут. Определяет имя навигационной карты, уникальное для данного документа. Используется для вызова карты с помощью атрибута USEMAP элемента IMG.

AREA

Создает область карты, определенной с помощью элемента MAP. Элемент должен располагаться между начальным и конечным тегами элемента MAP. Не имеет конечного тега.

Атрибуты

SHAPE - определяет форму навигационной области.

Возможные значения:

rect - прямоугольник;

circle - окружность;

poly - многоугольник.

COORDS - определяет координаты навигационной области на карте. Используется в паре с атрибутом SHAPE.

Способы задания координат для разных типов областей:
SHAPE="rect" COORDS="левый x, верхний y, правый x,
нижний y";

SHAPE="circle" COORDS="центр x, центр y, радиус";

SHAPE="poly" COORDS="x1,y1,x2,y2,x3,y3,...";

HREF - определяет область как гипертекстовую ссылку.
Значение задается в виде ссылки (URL).

TARGET - определяет окно (фрейм), на которое указывает гипертекстовая ссылка. Этот атрибут используется только совместно с атрибутом HREF.

NOHREF - определяет область как неактивную (невосприимчивую к нажатию). Данный атрибут противоположен атрибуту HREF и используется для отмены действия последнего.

ALT - определяет альтернативный текст-подсказку для данной области.

Пример:

Создаем карту с именем ImageMap

```
<MAP NAME="ImageMap">  
<AREA href="something.html" SHAPE="rect"  
COORDS="0,0,70,140" ALT="Левая половинка">  
<AREA href="anything.html" SHAPE="rect"  
COORDS="71,0,140,140" ALT="Правая половинка">  
</MAP>  
<!-- Создали. -->  
<BODY>  
<!-- Связываем ее с изображением -->  
<IMG src="/img/block.gif" USEMAP="#ImageMap"  
HEIGHT="140" WIDTH="140" BORDER="0">
```

...

В данном примере создано квадратное изображение размером 140x140 пикселей, левая часть которого является ссылкой на файл *something.html*, а правая - на файл *anything.html*.