

## Лабораторная работа № 4. Блоковая модель

Цель: познакомиться с блоковой моделью, изучить способы размещения элементов на web-странице.

### Краткая теория

#### Границы

Стиль границ HTML-элемента (свойство border-style значения solid, dashed, dotted, double).

Цвет границы (свойство border-color).

Толщина границы (свойство border-width значения в пикселях или thin, medium, thick).

Задание стилей для отдельных сторон, названия сторон перечислены на рисунке (рис.1.):



Рис.1. Стили и названия сторон блоковой модели

Способ быстрого задания стилей границ.

Пример1. Border-style:dashed double solid groove – к верхней границе будет применено dashed, к правой double, к нижней solid, а к левой groove.

Пример2. Border-style:dashed double –верхняя и нижняя граница будут оформлены как dashed, а левая и правая граница как double.

Сокращенная форма записи объединяет все свойства оформления границ в одном свойстве border. Порядок следования свойств: border-width, border-style, border-color (можно пропускать неиспользуемые свойства). Пример: border:1px solid.

**Отступы** (величина в пикселях (px), сантиметрах (cm), процентах (%), em):

- внутренние (свойство padding);
- внешние (свойства margin);
- величина отступа может быть задана отдельно для каждой стороны элемента.

Краткая форма записи: padding: 60px 20px 40px 50px; margin: 30px 50px.

## Отображение элементов

Существуют два способа отображения элементов: `visibility:hidden` и `display:none`).

## Размещение элементов

Местоположение элементов задается с помощью следующих CSS свойств:

- `top` – величина смещения текущего элемента от верхнего края родительского элемента;
- `bottom` – от нижнего края родительского элемента;
- `left` – от левого края родительского элемента;
- `right` – от правого края родительского элемента.

Описанные выше свойства позиционирования не вступят в силу пока Вы не зададите способ размещения. В CSS существуют 4 различных способа размещения элементов:

- статическое (`position:static`) действует по умолчанию, элементы всегда отображаются там, где они были объявлены (свойства `top`, `bottom`, `left` и `right` не работают со статичными элементами);
- фиксированное (`position:fixed`) элементы не изменяют своего местоположения даже при прокрутке окна браузера (свойства `top`, `bottom`, `left`, `right` работают);
- относительное (`position:relative`);
- абсолютное (`position:absolute`).

Наложение элементов (свойство `z-index` значение – любое число, может быть отрицательное).

## Выравнивание элементов.

По центру с помощью `margin`. Блочные элементы могут быть выровнены по центру установкой `margin` с левой и правой стороны значения `auto`.

С помощью свойств позиционирования

С помощью свойства `float`. Элемент, выровненный с помощью `float`, будет прижат к левой или правой границе родительского элемента (в зависимости от заданного значения) и заставит следующие за ним элементы "обтекать" его с противоположной стороны. Очистить элементы от `float` можно командой `clear:both`.

## Задания к лабораторной работе № 4

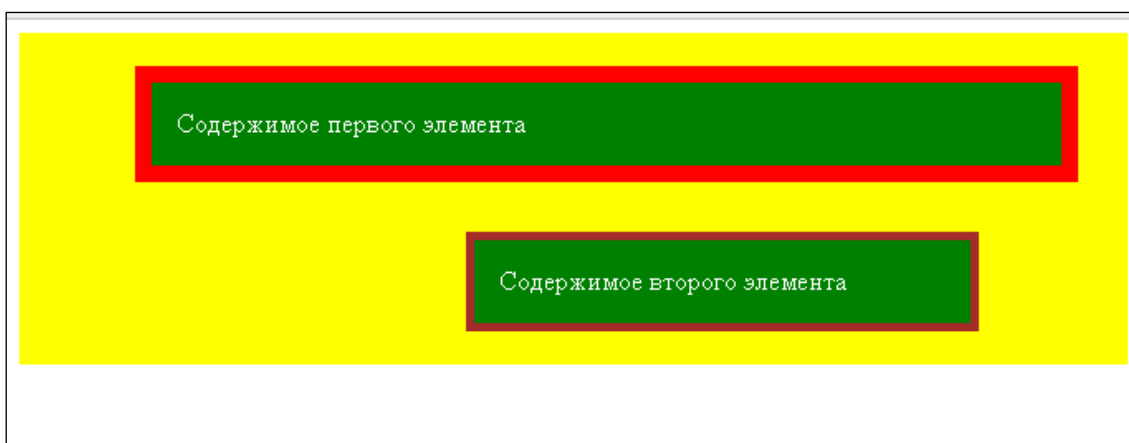
**Задание 1.** Для предварительного знакомства с блочной моделью внимательно посмотрите следующий HTML документ и его отображение браузером.

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

```

#wrap {
margin:0px;
padding:20px;
height:160px;
background-color:yellow;
}
.ex1 {
border:10px red solid;
margin-left:50px;
margin-right:10px;
padding:15px;
background-color:green;
color:white;
}
.ex2 {
border:5px brown solid;
margin-top:30px;
margin-left:250px;
margin-right:70px;
padding:15px;
background-color:green;
color:white;
}
</style>
</head> <body>
<div id='wrap'>
<div class='ex1'>Содержимое первого элемента</div>
<div class='ex2'>Содержимое второго элемента</div>
</div>
</body> </html>

```



На примере изучите внутренние и внешние отступы элементов. Измените цвет элементов, размер элементов, величину отступов и сохраните как свою работу.

**Задание 2.** Оформите элементы согласно их описанию. Стиль границ, цвет и толщину можно придумать свои.

```

<html>
<head>
<style type='text/css'>

```

```

#par1
{
/* Пишите код здесь */
}
#par2
{
/* Пишите код здесь */
}
#par3
{
/* Пишите код здесь */
}
#par4
{
/* Пишите код здесь */
}
#par5
{
/* Пишите код здесь */
}
#par6
{
/* Пишите код здесь */
}
#par7
{
/* Пишите код здесь */
}
</style>
</head>
<body>
<p id='par1'>Я имею сплошную границу коричневого цвета.</p>
<p id='par2'>Я имею сплошную границу красного цвета слева и
справа.</p>
<p id='par3'>Я имею сплошную границу розового цвета сверху, и
пунктирную границу голубого цвета слева, справа и снизу.</p>
<p id='par4'>Я имею границу оранжевого цвета толщиной 1
пиксель.</p>
<p id='par5'>Я имею границу красного цвета толщиной 2 пикселя
сверху и снизу и пунктирную границу серого цвета слева и
справа.</p>
<p id='par6'>Я имею двойную сплошную границу #1435AD цвета.
</p>
<p id='par7'>Я имею пунктирную границу #FF8100 цвета и
толщиной 1 пиксель сверху и сплошную границу #0B6124 цвета
толщиной 2 пикселя снизу. </p>
</body>
</html>

```

**Задание 3.** Узнайте кодовое слово, используя свойство `visibility:hidden`, затем свойство `display:none`). Объясните разницу.

```

<html> <head>
<style type='text/css'>
span

```

```

{
font-size:1.6em;
font-family:verdana;
color:green;
}
/* Пишите код здесь */
</style>
</head>
<body>
<b>
<p>Для того, чтобы узнать кодовое слово скройте элементы со
следующими id: code, winter, sky, frog, forest, cloud, hide.
</p></b><br />
<hr />
<span id='code'>Б</span><span id='mirror'>А</span><span
id='winter'>ДЖ</span><span id='grass'>Л</span><span
id='sky'>МЖ</span><span id='bright'>Т</span><span
id='forest'>Ь</span><span id='sun'>А</span><span
id='frog'>ЗП</span><span id='cloud'>ШН</span><span
id='true'>Й</span><span id='hide'>РО</span>
<hr />
</body> </html>

```

**Задание 4.** Поработайте с элементами div на своей второй web-странице. При помощи свойств margin, border, padding задайте для них различный цвет и толщину отступа, границы полей так, чтобы эти составляющие модели блоковых элементов стали видны. Поэкспериментируйте с разными способами их размещения, не менее 3 вариантов, сохраните в файлах.

На второй web-странице создайте список следующего вида, где элементы списка являются ссылками на страницы с выполненными заданиями:

- задание 1
  - задание 2
  - задание 3
- 1) задание 4.1
  - 2) задание 4.2
  - 3) задание 4.3