

# Блочная верстка сайта



Урок 18 - 19

# Исходный макет



- Header (шапка или заголовок);
- Навигационное меню;
- Боковая панель;
- Основной блок с контентом;
- Footer (ноги или подвал);

# Результат

## Название сайта

header (шапка сайта)

Главная

Навигация 1

Навигация 2

Навигация 3

### Левая панель

Здесь может быть какой-нибудь текст. Здесь может быть какой-нибудь текст.

## Основной контент страницы

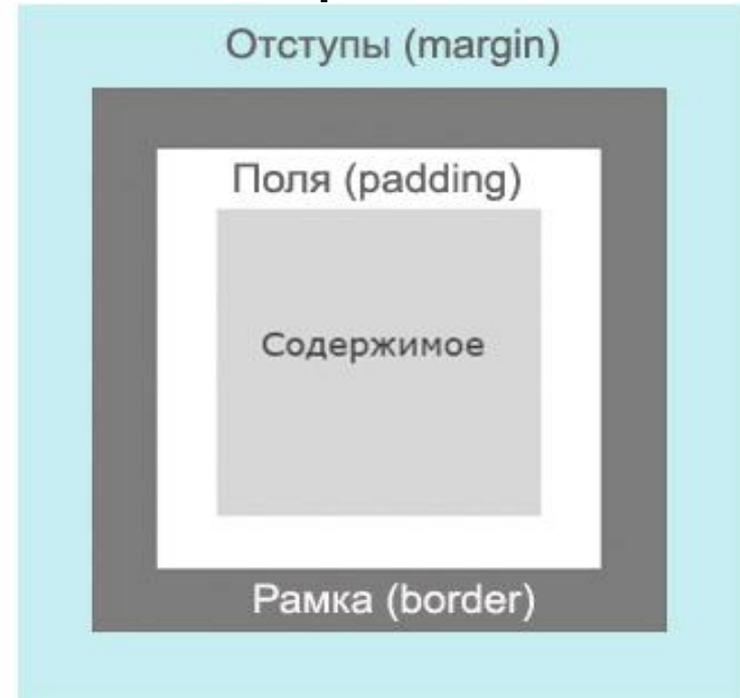
Здесь может быть какой-нибудь текст. Здесь может быть какой-нибудь текст.

# Верстка блоками DIV

**Рамка (border)** – это контур, для которого можно задать такие характеристики как толщина, цвет и тип (пунктирная, сплошная, точечная).

**Поля (padding)** – отделяют содержимое блока от его рамки, чтобы текст, например, не был «впритык» к стенкам блока.

**Отступы (margin)** – это пустое пространство между различными блоками, позволяющее на заданном расстоянии расположить два блока относительно друг друга.



# Практическая работа 1



Урок 18 - 19

# Атрибуты **relative** и **absolute**

Иногда бывает необходимо разместить какой-то блок в строго заданном положении относительно родительского.

## **Задание 1.**

Создайте html-документ, в теле которого разместите код, как на рисунке:

```
<body>
  <div id="parent">
    <div id="child">
      Это белый квадрат - дочерний элемент для черного квадрата
    </div>
  </div>
</body>
```

# Атрибуты **relative** и **absolute**

Предположим, мы хотим разместить дочерний блок в 20 пикселях левее и ниже верхнего левого угла родительского блока. Создайте таблицу стилей css:

```
#parent {  
  margin: 50px 100px;  
  background: #000;  
  width: 500px;  
  height: 500px;  
}
```

```
#child {  
  position: absolute;  
  top: 20px;  
  left: 20px;  
  width: 200px;  
  height: 200px;  
  background: #fff;  
  border: 1px solid #000;  
}
```

Сохраните и оцените результат в браузере:

Это белый квадрат -  
дочерний элемент для  
черного квадрата



**Не совсем тот результат!**

Наш дочерний элемент ушел не к маме, а к дедушке (т.е. к BODY)!

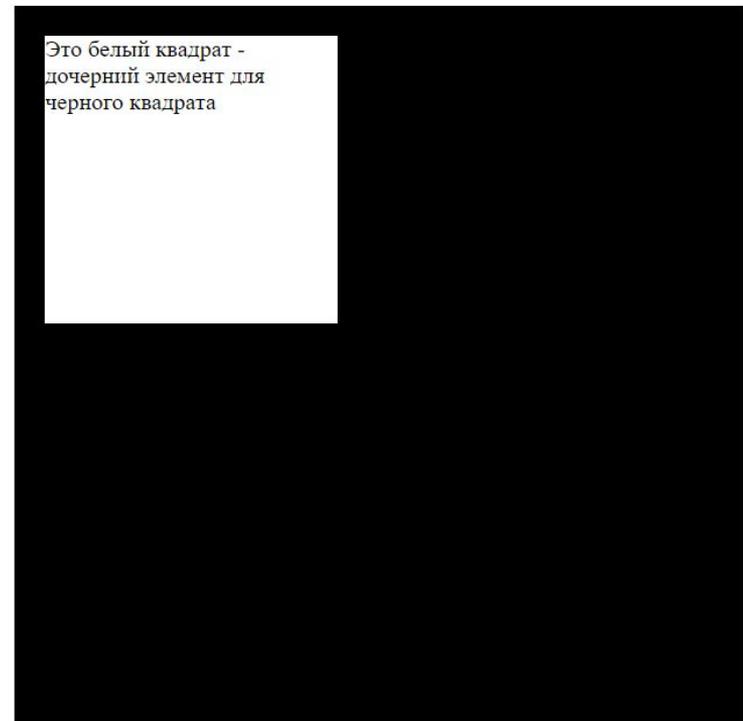
# Атрибуты **relative** и **absolute**

Проблема решается довольно просто: родителю дополнительно задаётся **position: relative;**

Дополните css:

```
1 #parent {
2   position: relative;
3   margin: 50px 100px;
4   background: #000;
5   width: 500px;
6   height: 500px;
7 }
8
9 #child {
10  position: absolute;
11  top: 20px;
12  left: 20px;
13  width: 200px;
14  height: 200px;
15  background: #fff;
16  border: 1px solid #000;
17 }
```

Сохраните и оцените результат в браузере:



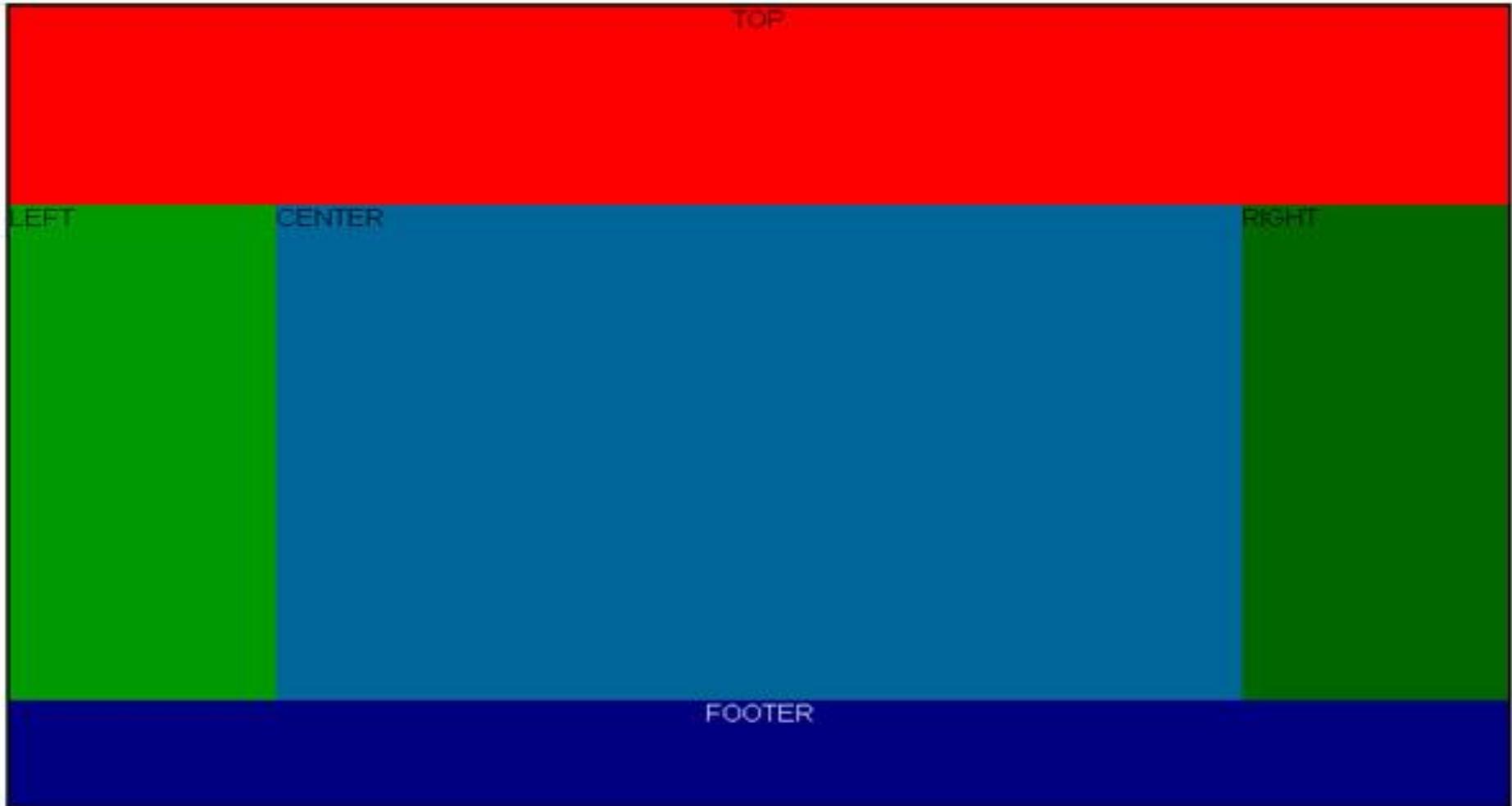
# Практическая работа 2



Урок 18 - 19

# Самостоятельная работа

Создать web-страницу, внешний вид которой изображен ниже на рисунке:



# Шаг 1

Контейнер будет содержать в себе пять блоков:

- **TOP** – шапка сайта, обычно содержит логотип компании, название, заголовки и слоганы, поиск, навигацию;
- **LEFT** и **RIGHT** – левая и правая колонки, обычно содержат рекламу, навигацию, рассылку, новости и т. д.:
- **CENTER** – содержит основной текст страницы;
- **FOOTER** – подвал сайта, содержащий копирайты и другие сведения о сайте, возможно, дублировать навигацию, что является признаком хорошего тона.

# Шаг 1

В файле `style.css` описываем параметры всего документа:

```
1  /*Задаем параметры всего документа*/
2  *{
3      margin: 0; /*Отступы*/
4      padding: 0; /*Поля*/
5      border: 0; /*Рамка*/
6  }
7
8  body{
9      font-family: "Arial", Courier; /*Шрифт документа*/
10     color: #000000; /*Цвет шрифта*/
11     padding: 0; /*поля*/
12     background: #ffffff; /*Фон документа белый*/
13 }
```

# Шаг 1

Работаем в index.html:

1

```
<body>
  <div id="container">

  </div>
</body>
</html>
```

2

```
<body>
  <div id="container">
    <div id="top">TOP</div>

  </div>
</body>
</html>
```

# Шаг 1

В файле `style.css` описываем параметры контейнера:

```
15 #container {
16     width: 90%; /*Ширина контейнера*/
17     margin: 0 auto; /*Отступы сверху и снизу - 0,
18     справа и слева - автоматически*/
19     border: 2px #000 solid; /*Рамка*/
20 }
```

# ЭТОТ знак говорит, что элемент является уникальным атрибутом и используется в HTML документе в теге `div` один раз.

# Шаг 1

В файле `style.css` описываем параметры `top`:

```
22  #top {  
23      width: 100%; /*Ширина шапки*/  
24      height: 140px; /*Высота шапки*/  
25      background: #ff0000; /*Фон шапки*/  
26      text-align: center; /*Выравнивание текста*/  
27  }
```

## Шаг 2

Работаем в index.html – добавим еще 4 блока в контейнер:

3

```
<body>
  <div id="container">

    <div id="top">TOP</div>
    <div id="left_bar">LEFT</div>
    <div id="main">MAIN</div>
    <div id="right_bar">RIGHT</div>
    <div class="clear"></div>

  </div>
</body>
</html>
```

## Шаг 2

Следующим этапом верстки будет расположение трех блоков последовательно по горизонтали. Для этого используем **float** – это позволит обтекать другим элементам наш блок справа или слева.

Добавим в `css`-документ следующий код:

# Шаг 2

```
29 #left_bar{
30     background: #009900;
31     width: 30%;
32     height: 350px;
33     float: left;
34 }
35 #main{
36     position: relative;
37     background: #006699;
38     width: 40%;
39     height: 350px;
40     float: left;
41 }
42 #right_bar{
43     background: #006600;
44     width: 30%;
45     height: 350px;
46     float: right;
47 }
48 .clear{
49     clear: both; /*Запрет обтекания с обеих сторон*/
50 }
```

# Шаг 3

Блок footer добавьте самостоятельно

```
<body>
  <div id="container">

    <div id="top">TOP</div>
    <div id="left_bar">LEFT</div>
    <div id="main">MAIN</div>
    <div id="right_bar">RIGHT</div>
    <div class="clear"></div>
    <div id="footer">FOOTER</div>

  </div>
</body>
</html>
```

4

```
52 #footer{
53     width: 100%; /*Ширина подвала*/
54     height: 140px; /*Высота подвала*/
55     background: #ff00ff; /*Фон подвала*/
56     text-align: center; /*Выравнивание подвала*/
57 }
```

# Самостоятельная работа 2

Создать web-страницу, внешний вид которой изображен ниже на рисунке:



# CSS справочник

<http://html-css-tegs.ru/>



## HTML теги:

Теги для работы с текстом

Работа со ссылками

Работа со списками

Работа с изображениями

Работа с таблицами

Работа с формами

Работа с документами

Другие теги

## CSS свойства:

Свойства текста

text-indent  
text-align  
text-decoration  
text-shadow  
letter-spacing  
line-height  
word-spacing  
text-transform  
white-space  
direction  
color

## CSS свойства шрифтов.

Свойство **font-family** определяет название шрифта - Tahoma, Arial, Times New Roman и другие. Можно указывать несколько шрифтов. Если у пользователя не будет первого шрифта, то применится второй и т. д.

### Значения:

**Tahoma** - названия шрифта.

### Пример:

```
.p { font-family: Times New Roman, Times, serif; }
```

Свойство **font-style** определяет стиль шрифта.

### Значения:

**normal** - обычный.

**italic, oblique** - курсивный.

### Пример:

```
.p { font-style : italic; }
```