

## Лабораторная работа

### Виды верстки: плоская, табличная, блочная. Изучение особенностей табличной и блочной верстки.

#### Цель

Ознакомиться с различными способами верстки. Получить навыки работы в них.

#### Плоская верстка

С плоской версткой все предельно просто. Если Вы в качестве базовых элементов страницы не используете ни таблицы ни блоки, то это и есть плоская верстка. Сейчас сайты так не делают, поэтому идем дальше.

#### Табличная верстка

Достоинства и недостатки табличной верстки:


Основные достоинства таблиц:	Недостатки таблиц:
<ul style="list-style-type: none"><li>• Простое создание колонок</li><li>• "Резиновая" верстка</li><li>• "Склейка" изображений</li><li>• Однозначная трактовка браузерами</li></ul>	<ul style="list-style-type: none"><li>• Долгая загрузка</li><li>• Громоздкий код</li><li>• Плохая индексация поисковиками</li><li>• Нет разделения содержимого и оформления</li><li>• Несоответствие стандартам</li><li>• Невозможность конвертации в мобильную версию</li></ul>

#### Пример табличной верстки

Вот что мы хотим получить:

логотип		
Меню		
<a href="#">menu1</a>	<a href="#">menu1</a>	<a href="#">menu1</a>
Субменю		
<ul style="list-style-type: none"><li>• <a href="#">Ссылка</a></li><li>• <a href="#">Ссылка</a></li><li>• <a href="#">Ссылка</a></li></ul>	контент-верх	суб-контент-верх
текст	контент-низ	суб-контент-низ
подвал		

Рис. 6.1. Пример табличной верстки

 Впрочем, у Вас уже достаточно знаний, чтобы **выполнить** **данный** **пример** **самостоятельно!**  
По макету с рис. 6.1. сверстать страницу табличной версткой.



*Важно!*

Применение табличной верстки Целесообразно в том случае, если

- Высота колонок должна быть одинаковой
- Макет должен занимать всю высоту окна браузера, независимо от объема информации
- Нет времени на сложную верстку

### Блочная верстка

Слой — это элемент веб-страницы, созданный с помощью тега `<div>`, к которому применяется стилевое оформление. Благодаря этому тегу HTML-код распадается на ряд четких наглядных блоков, за счет чего верстка слоями называется также блочной версткой. Код при этом получается более компактным, чем при табличной верстке, к тому же поисковые системы его лучше индексируют.

Сейчас я Вам продемонстрирую как разметить страницу с рис. 6.2.

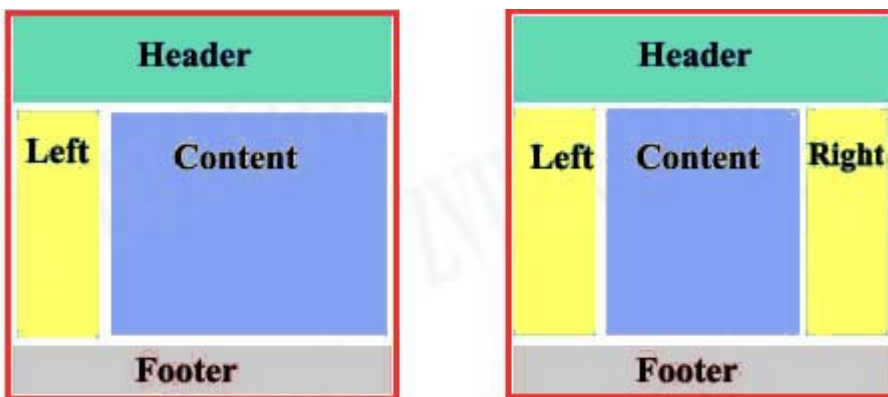


Рис. 6.2. Пример для

блочной разметки

На рис 6.2. представлены варианты для резиновой верстки с двумя и тремя колонками. Давайте выполним разметку двух колонок по шагам:

#### *Шаг 1*

Сначала, как обычно, выполним разметку HTML

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01  
Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">  
<html>  
<head>  
<meta http-equiv="Content-Type" content="text/html;  
charset=windows-1251">  
<title>Блочная верстка</title>  
<link href="style_1.css" rel="stylesheet" type="text/css">  
</head>  
<body>  
  <div id="container">
```

```

        <div id="header">
            <h1>Header</h1>
        </div>

        <div id="left">
            <h3>left Content</h3>
        </div>

        <div id="content">
            <h1>Main Content </h1>
            <p>Lorem ipsum dolor sit amet, consectetur
adipiscing elit.
            Praesent aliquam, justo convallis luctus
rutrum, erat nulla fermentum
            Integer turpis arcu, pellentesque eget, cursus et,
fermentum ut, sapien.
            Fusce metus mi, eleifend sollicitudin, molestie
id, varius et, nibh.
            Donec nec libero.</p>
            <h1>Content</h1>
            <p>Lorem ipsum dolor sit amet, consectetur
adipiscing elit.</p>
            <p>Quisque ornare risus quis ligula Phasellus
tristique
            purus a augue condimentum adipiscing.
Aenean sagittis.
            Etiam leo pede, rhoncus venenatis, tristique in,
vulputate at, odio.</p>
        </div>
        <div id="footer">
            <p><strong>Footer</strong></p>
        </div>
    </div>
</body>
</html>

```

Обратите внимание на присутствие базового (главного) контейнера, который уже содержит все остальные. Наличие этого главного контейнера позволяет регулировать общие параметры страницы, что значительно облегчает верстку слоями.

Результат получится вот такой:

# Header

left Content

# Main Content

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent aliquam, justo convallis luctus rutrum, erat nulla fermentum Integer turpis arcu, pellentesque eget, cursus et, fermentum ut, sapien. Fusce metus mi, eleifend sollicitudin, molestie id, varius et, nibh. Donec nec libero.

# Content

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Quisque ornare risus quis ligula. Phasellus tristique purus a augue condimentum adipiscing. Aenean sagittis. Etiam leo pede, rhoncus venenatis, tristique in, vulputate at, odio.

Footer

Рис.

6.3. Пример для блочной разметки. Шаг 1.

Оно и понятно, ведь стиль-то CSS у нас не написан. Итак, займемся стилями.

## Шаг 2

```
/* CSS Document */
body, html {
margin:0px; /*Это хорошая практика обнулять поля и
отсуны, т.к.
различные браузеры их по разному воспринимают. */
padding:0px;
text-align:center; /*Выравниваем макет (содержимое
элемента body
будет посередине) по центру в старых версиях браузеров
*/
}
#container{
margin:0 auto; /*выравниваем макет по центру в
современных браузерах */
width:650px;
}
/*Здесь пишем стили для шапки сайта */
#header{
background-color:#64dab2;
}

/*Здесь пишем стили для левой колонки сайта */
#left{
background-color:#ffff66;
}

/*Здесь пишем стили для блока контента */
```

```

#content{
background-color:#83a0f3;
}
#content h1 {
margin:0px; /* Обнуляем отступы для заголовка первого
уровня,
находящегося в блоке контента.*/
}
/*Здесь пишем стили для подвала сайта */
#footer{
background-color:#ccc;
}

```

После применения данного стиля style\_1.css результат будет следующим:



Рис.

6.4. Пример для блочной разметки. Шаг 2. Первоначальное стилевое оформление

Теперь необходимо блоки Left и Content поставить рядом, как на [рис. 6.2](#).

Меняем ширину блоку Left на 150px и устанавливаем для него обтекание:

```

#left{
...
width:150px; /*ширина колонки */
float:left; /*обязательное выравнивание по левому краю,
с включением обтекания*/
}

```

Смотрим результат, получаем не совсем то, что хотели (рис. 6.4. а):

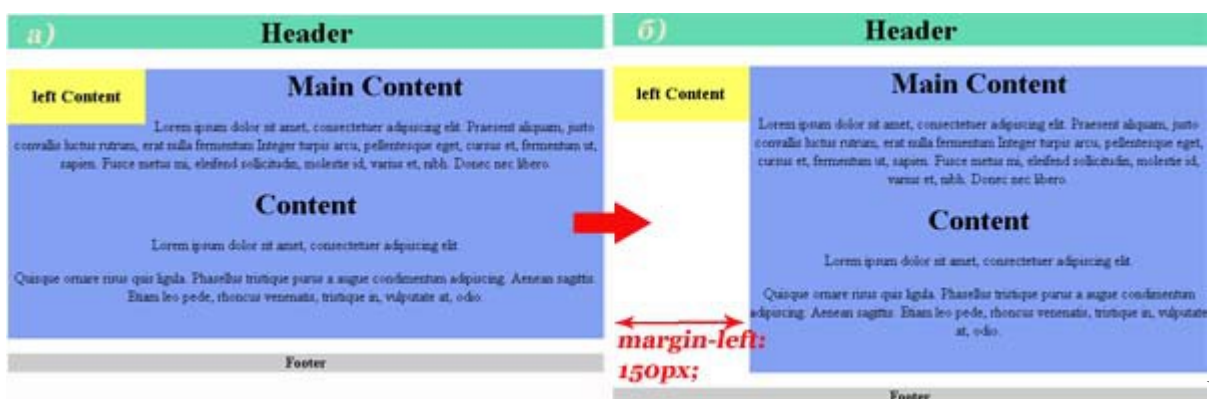


Рис.

6.4. Шаг 2. Обтекание блоков

Для того, чтобы из рис. 6.4.а) получить рис 6.4. б) необходимо сдвинуть блок Content вправо на ширину блока Left:

```
#content{
  background-color:#83a0f3;
  margin-left:150px;
}
```

Тоже самое можно сделать иначе, установив блоку Left `width:20%` и `float:left;`, а блоку Content `width:80%` и `float:right;` (см. пример в лекциях)

При этом не забудьте поставить для блока footer значение `clear:both;`, иначе при увеличении количества текста в предыдущих блоках может получиться седующий эффект:

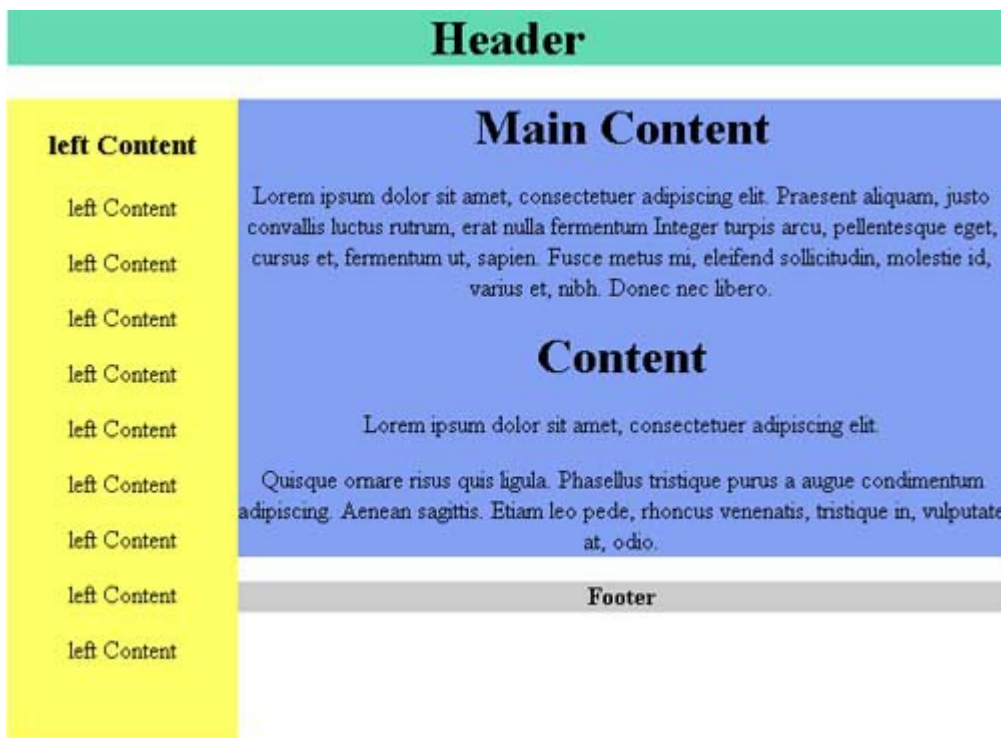


Рис. 6.5.

Необходима отмена обтекания блоков



Итак, **Ваша**

**задача**

По макету с рис. 6.2 (б) разработать трехколонный блочный макет, используя в качестве базы двухколонный и растянуть содержимое на всю ширину страницы.

И последнее: Проверьте свой трехколоночный макет: хорошо ли он минимизируется? Не скажут ли блоки? Тем, кто забыл о свойстве `min-width` рекомендую установить его для блока Content и снова проверить минимизацию Вашего макета.



*Важно!*

При верстке слоями учитывайте **блочную модель**, и помните, что, когда Вы задаете ширину блока `width`, а затем добавляете ему границы `border` и отступы `margin`, `padding`, то к величине блока добавляются значения границ и отступов и блок расширяется:



с. 6.6. Инспектирование блочной модели в FireBug.

Центрировать картинку в резиновом дизайне можно, например, так:

```
img {display: block; width: auto; height: auto; margin: 0 auto; }
```