

Как из макета создать работающий веб-сайт?

Доброго времени суток, уважаемые читатели! В этом руководстве, подробно, шаг за шагом показано, как можно сделать верстку сайта из макета. Здесь приведены основные сведения о том, как это делать. Эта статья не претендует на истину в последней инстанции, но здесь изложены основные принципы, того, как можно сверстать свой сайт с нуля. Прежде чем начнём, скажу, чтобы шаг за шагом следить за выполнением этого урока, Вы можете скачать работающий пример в конце статьи. Итак, погнали!

Некоторые дополнения к статье:

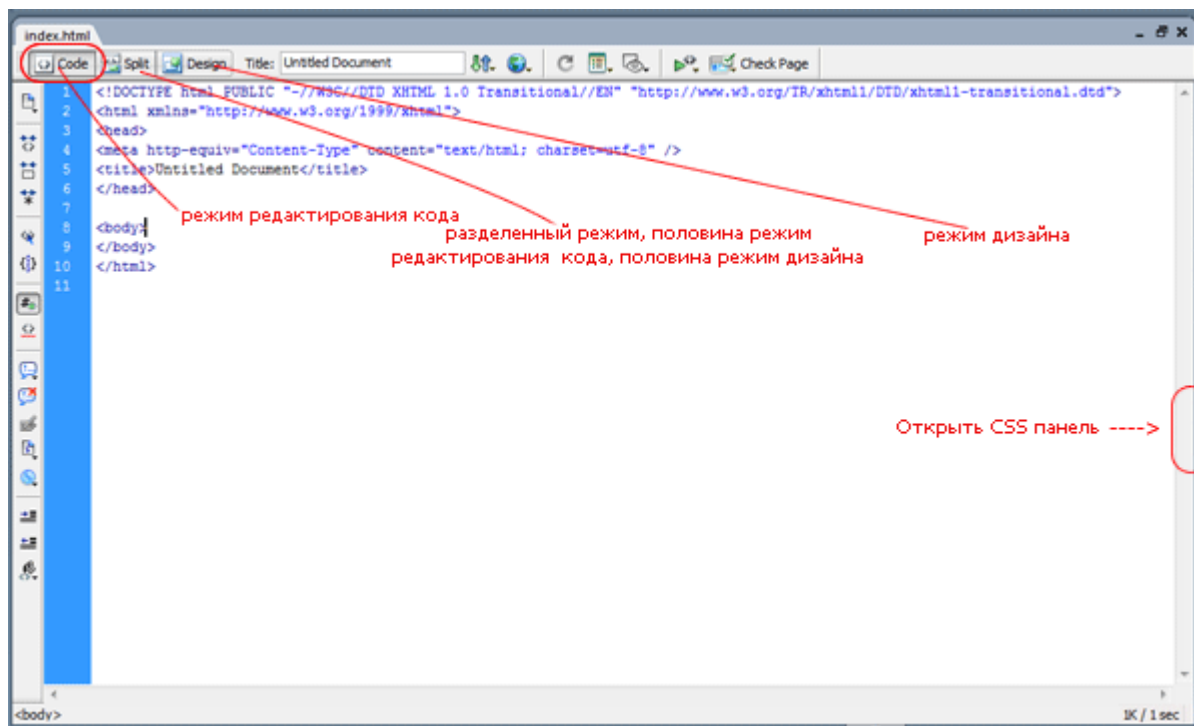
- *Лучше не делать фон сайта картинкой, а сделать то же самое с помощью CSS.*
- *Не нужно делать навигацию с помощью картинок, поисковики этого не любят. Лучше подключить шрифты.*
- *Закругленные углы лучше сделать при помощи свойства CSS – `border-radius`.*
- *В конце статьи приведен бесплатный, дополнительный материал, который поможет быстро оптимизировать ваш сайт и , используя последние достижения в этой области.*

Создайте папку на Вашем рабочем столе и назовите её, как пожелаете, внутри папки создайте другую папку и назовите её «images». Откройте Блокнот откройте пункт меню «Файл > Сохранить как» введите «styles.css» убедитесь, что имя файла включает расширение «.css». Откройте редактор Dreamweaver и выберите [HTML](#) как показано на рисунке выше.

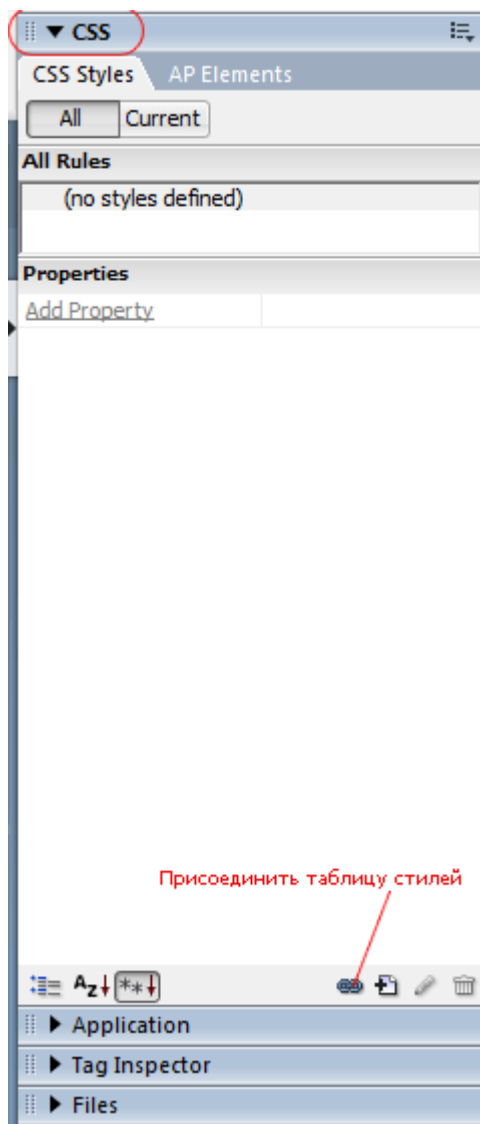
Сейчас Вы должны создать пустой HTML-документ в редакторе Dreamweaver, перейдите «Файл > Сохранить как» и сохраните файл как «index.html». В редакторе Dreamweaver Вам нужно перейти в

режим кодирования и открыть CSS панель. CSS панель находится справа, если Вы видите её нажмите на стрелочку.

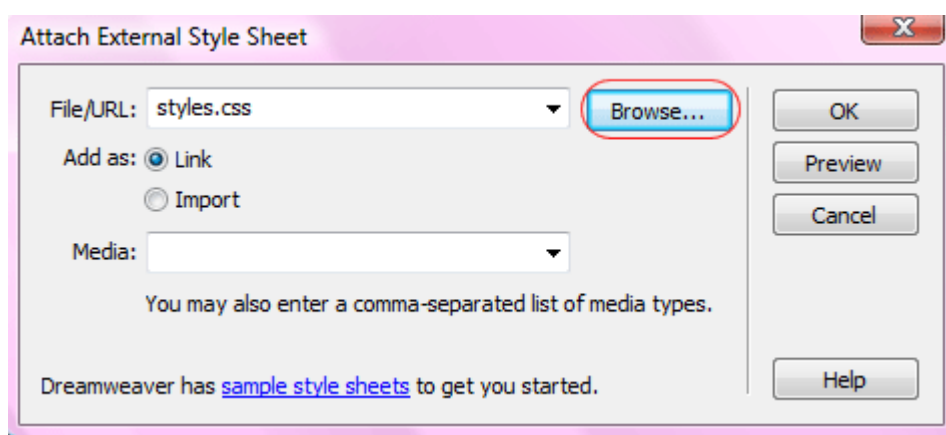
В Dreamweaver есть три режима: редактирования кода, дизайна и разделенный режим, включающий два предыдущих. Режим кодирования показывает HTML-код документа, который открыт в данный момент. Разделенный режим включает оба режима, режим редактирования кода и режим дизайна, и занимает половину рабочей области редактора. Режим дизайна показывает, как выглядит страница, **но** этот вид не точно отображает то, что будет в окне браузера. Поэтому я редко использую этот режим, а проверяю дизайн в IE и [Firefox](#). Вы можете открыть ваш макет в браузере, нажав клавишу F12.



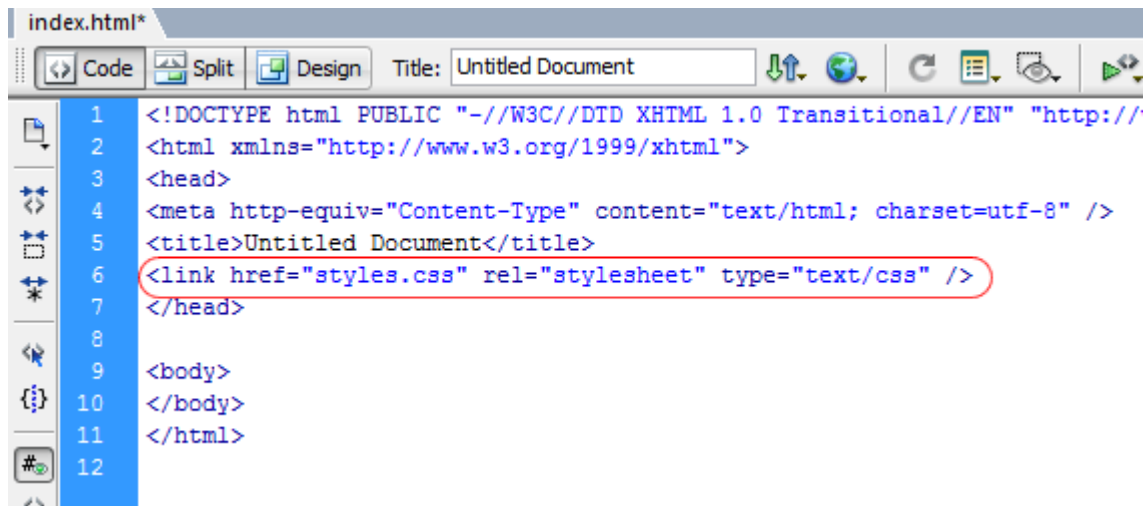
Если CSS панель открыта, то нажмите на кнопку внизу панели «Присоединить таблицу стилей».



Нажмите кнопку Обзор (Browse) в окне диалога найдите таблицу стилей, которую Вы создали на первом шаге (файл styles.css).



Заметьте, что после этого в Вашем коде HTML-файла index.html появилась новая строка.



```
index.html*
Code Split Design Title: Untitled Document
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5 <title>Untitled Document</title>
6 <link href="styles.css" rel="stylesheet" type="text/css" />
7 </head>
8
9 <body>
10 </body>
11 </html>
12
```

Эта строка кода очень важна, так как она говорит HTML-файлу, что искать стили для него нужно в этом файле (styles.css). Не правда ли, что большинство тех, кто читает эту статью новички? Хорошо, я создам мой шаблон с самого начала до конца, пункт за пунктом. Мне легче делать всё по-своему, но не бойтесь экспериментировать.

Давайте начнём с того, что заложим первое основание нашей веб-страницы. Нам нужны такие блоки как: контейнер, заголовок, панель навигации и логотип. Контейнер – это место, в котором мы будем хранить наш макет, заголовок это заголовок панели навигации. Достаточно просто, не правда ли?! Все эти элементы будут «завёрнуты» в тэги div и будут выглядеть следующим образом:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Untitled Document</title>
<link href="/img_articles/16940/styles.css" rel="stylesheet"
type="text/css" />
</head>

<body>
<!--CONTAINER-->
```

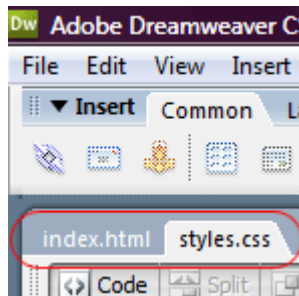
```
<div id="container">
  <!--HEADER BACKGROUND-->
  <div id="header">
    <!--MENU BAR-->
    <div id="nav">
    </div>
    <!--MENU BAR END-->
    <!--HEADER LOGO-->
    <div id="logo">
    </div>
    <!--HEADER LOGO END-->

    </div>
  <!--HEADER BACKGROUND END-->
</div>
<!--CONTAINER END-->
</body>
</html>
```

Давайте разберем код в деталях. Первый div – это контейнер, который содержит веб-сайт. Всё на сайте будет сверстано внутри этого тэга. Тэг div заголовка вмещает только фоновое изображение позади панели меню, Вы это увидите, когда добавите стили. Так как мы хотим, поместить фоновое изображение за меню и логотип, нам нужно разместить эти пункты внутри тэга div заголовка. Как видно, тэг div заголовка заканчивается в нижней части HTML-файла с контейнером.

Когда мы формируем содержание нашего сайта мы должны поместить его после закрывающего тэга </div> заголовка, но перед закрывающим тэгом </div> контейнера. Теперь давайте настроим стили наших тэгов div.

Откройте ваш файл styles.css в Dreamweaver, используя вкладки для переключения между этими двумя файлами.

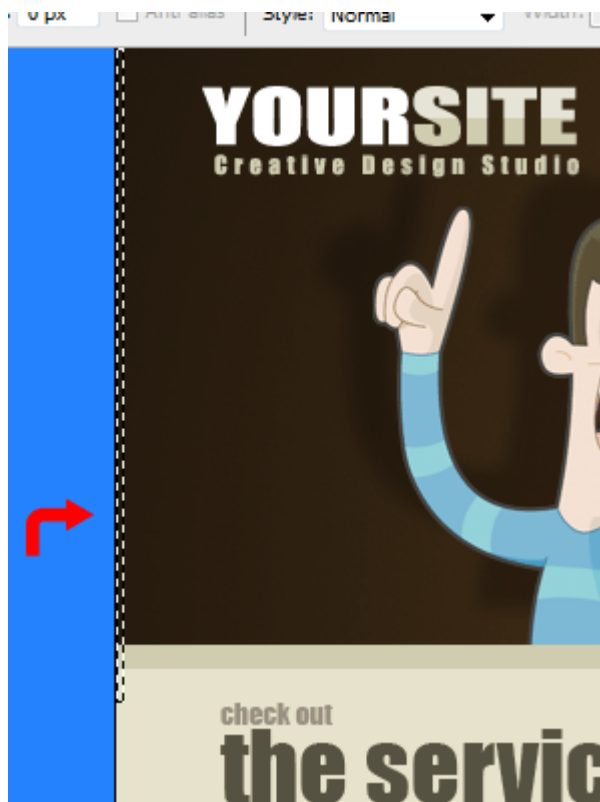


В первую часть кода CSS мы должны добавить стиль тела (body) документа, он выглядит следующим образом:

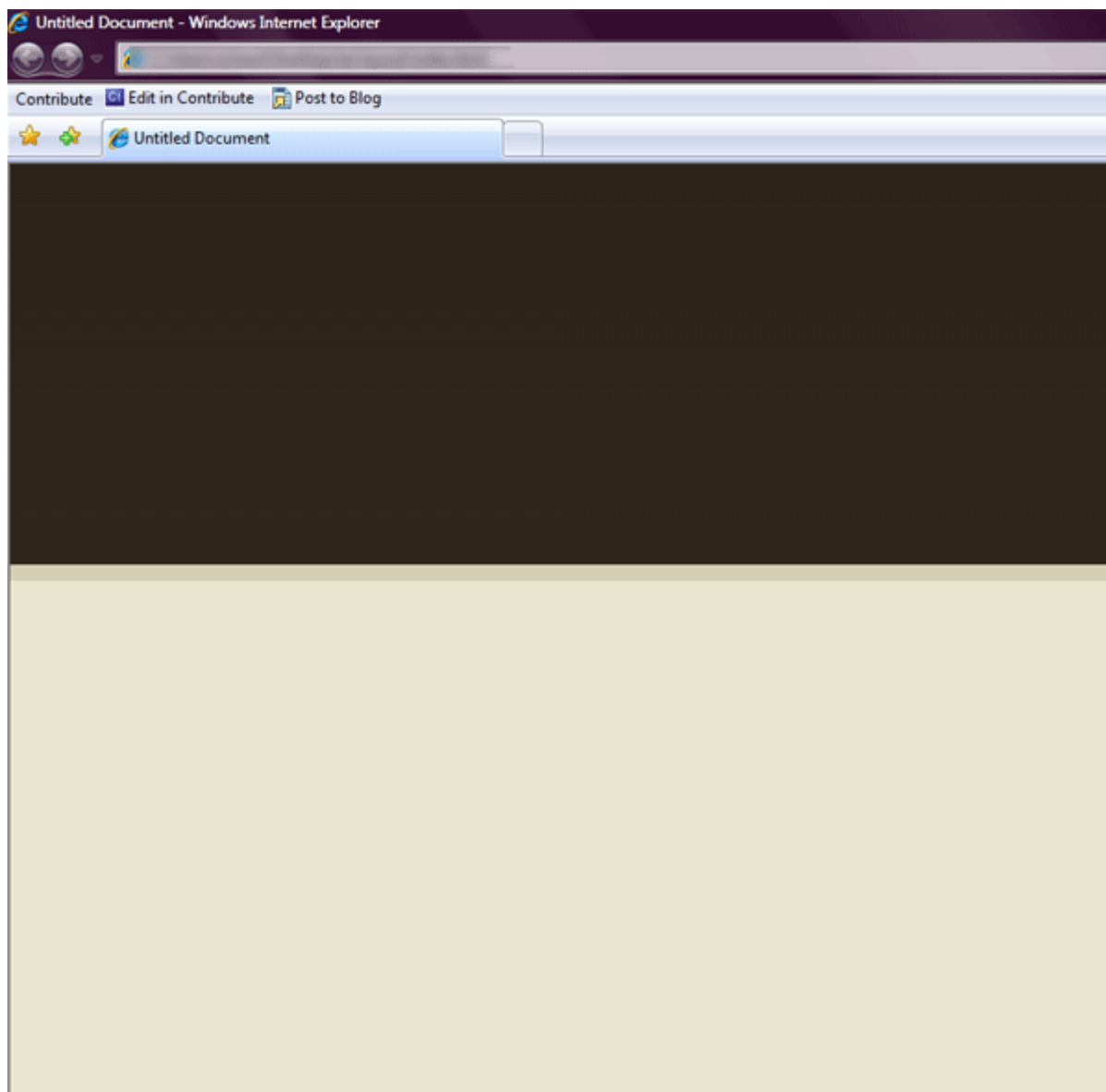
```
*{
    padding:0;
    margin:0;
}
body {
    background-image:url(images/bg.png);
    background-repeat:repeat-x;
    background-color:#eae5cf;
    padding:0;
    margin:0;
}
```

Мы не хотим расширять края сайта, поэтому мы поставим свойства «padding» и «margin» в 0, нам нужно добавить шрифт, который мы будем использовать на всём сайте. У нас есть фоновое изображение, которое является задним фоном, я покажу вам, как создать его за минуту.

Мы хотим, чтобы фоновое изображение повторялось по оси X, то есть по горизонтали. Цвет фона связаны с фоном, в файле PSD. Чтобы создать фоновое изображение заголовка, выделите в Вашем PSD файле полосу в 1 или 2 пикселя. Убедитесь, что все слои сведены, **но не сохраняйте PSD файл**, просто скопируйте и вставьте фрагмент в новый документ, затем отмените все изменения в файле PSD. Сохраните изображение, которое вы скопировали, как «bg.PNG» в папке изображений.



Вернитесь к своему HTML-файлу и нажмите F12, Вы должны увидеть в браузере Ваш файл с фоновым изображением.



Теперь создадим наш контейнер и фоновое изображение для заголовка. Первым мы создадим фоновое изображение для заголовка. В PSD файле скройте все элементы заголовка, затем выделите заднюю часть заголовка (убедитесь опять, что Вы свели слои в PSD файле, но не забудьте вернуть все изменения). Скопируйте и вставьте заголовок в новый документ и сохраните как «header_bg.png» в папку изображений.



Теперь добавьте в CSS, стили для нашего контейнера и тэга заголовка.

```
#container {  
width:900px;  
padding:0;  
margin:auto;  
}  
/* HEADER BACKGROUND IMAGE */  
#header{  
                                background-repeat:no-repeat;  
background-image: url(images/header_bg.png);  
height: 298px;  
width: 900px;  
float:left;  
margin:auto;  
}  
/* HEADER BACKGROUND IMAGE END */
```

Мой PSD файл шириной 900px, поэтому мы будем использовать контейнер 900px, и опять мы не будем использовать свойство «padding», но установим «margin» в «auto», это центр нашей веб-страницы в браузере. Для тэга div заголовка мы должны добавить изображение «header_bg.png». Мы не хотим, чтобы фон заголовка повторялся, а был фиксированным.

Вам необходимо проверить ширину и высоту изображения, которое вы сохранили, мое 900px ширина и 298px высота.

Ваше может быть другим, поэтому настройте изображение соответствующим образом. Сохраните CSS файл, а затем вернитесь

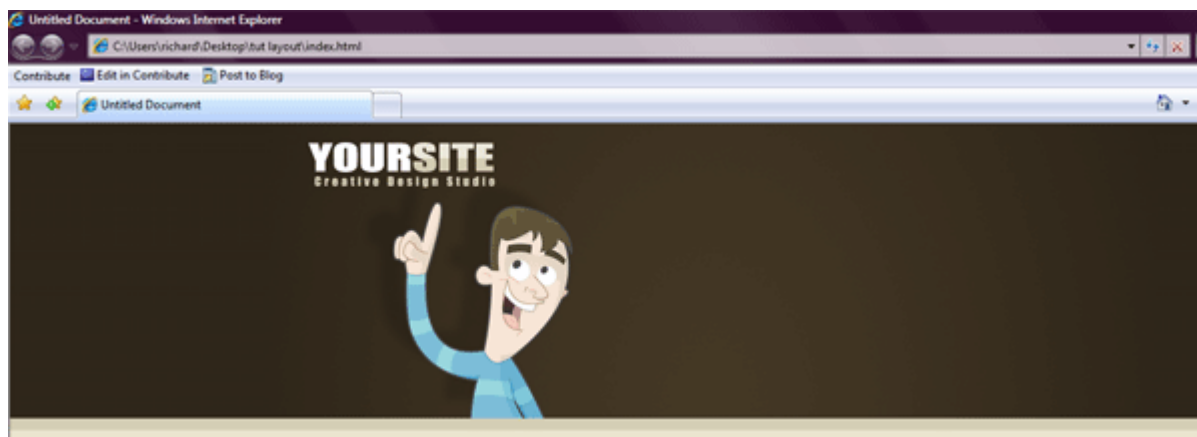
в ваш HTML файл нажмите F12 и посмотрите. Вы должны увидеть свое фоновое изображение в заголовке. Теперь давайте добавим наш логотип, который будет названием сайта. Вернитесь к PSD файлу и скопируйте Ваш логотип в новый документ и сохраните как «logo.png».



Давайте добавим стили CSS для изображения логотипа.

```
/* HEADER LOGO */  
#logo {  
background-repeat:no-repeat;  
background-image: url(images/logo.png);  
height: 298px;  
width: 315px;  
float:left;  
}  
/* HEADER LOGO END*/
```

Опять почти такой же результат, отличаются ширина и высота. Они соответствуют изображению, которое вы только что сохранили (logo.png). Сохраните файл CSS, перейдите в HTML и нажмите F12. Вот как он выглядит в браузере:



Пришло время для вёрстки навигации. В тэге `div` навигации нужно добавить меню, которое выглядит следующим образом:

```
<!--MENU BAR-->
<div id="nav">
<ul class="menu">
<li id="about"><a
href="/img_articles/16940/#"><span>#</span></a></li>
<li id="portfolio"><a
href="/img_articles/16940/#"><span>#</span></a></li>
<li id="clients"><a
href="/img_articles/16940/#"><span>#</span></a></li>
<li id="contact"><a
href="/img_articles/16940/#"><span>#</span></a></li>
</ul>
</div>
<!--MENU BAR END-->
```

В новой части кода «`ul class=»menu»` означает неупорядоченный список. Все элементы навигации имеют «`li id»` идентификатор списка. Ссылки имеют ID, поскольку для навигации мы не будем использовать обычный текст, а будем использовать изображения. Зайдем в PSD-файл и вставим каждый пункт навигации и сохраним как отдельные изображения. Нужно обозначить их в соответствии с назначением изображения. В целом у Вас должно быть 4 изображения. Кроме того, при сохранении файлов PNG, убедитесь, что изображения имеют прозрачный фон.

CSS-код для навигации следующий:

```
/* NAVIGATION */
#nav{
width:480px;
padding-top:20px;
float:right;
height:40px;
margin-right: 40px;
}
.menu ul{
padding:0;
margin:0;
list-style-type:none;
list-style:none;
display:inline;
}
.menu li{
padding-right:10px;
margin:0 5px 0 0;
height:40px;
list-style-type:none;
list-style:none;
display:inline;
float:left;
}
span{
display:none;
}
#about a{
background:url(images/about.png);
width:107px;
height:39px;
display:block;
}
#portfolio a{

background:url(images/portfolio.png);
width:106px;
height:39px;
display:block;
}
#clients a{
```

```
background:url(images/clients.png);
width:98px;
height:39px;
display:block;
}
#contact a{
background:url(images/contact.png);
width:109px;
height:39px;
display:block;
}
/* NAVIGATION END*/
```

Навигационный тэг `div` содержит в себе изображения навигации. Теперь нам нужно добавить ширину, она должна соответствовать размеру всех изображений вместе, например: если каждое изображение `200px`, то это равно `200px x 4 = ширине окна`.

Затем нам нужно добавить заполнение (`padding`) для верхней части окна так, чтобы оно не прилипла к правой верхней части окна браузера. Высота панели навигации соответствует высоте изображений и они все должны быть одинаковыми. Нам нужно сдвинуться правее, поэтому добавим `40` пикселей вправо, используя свойство `margin`.

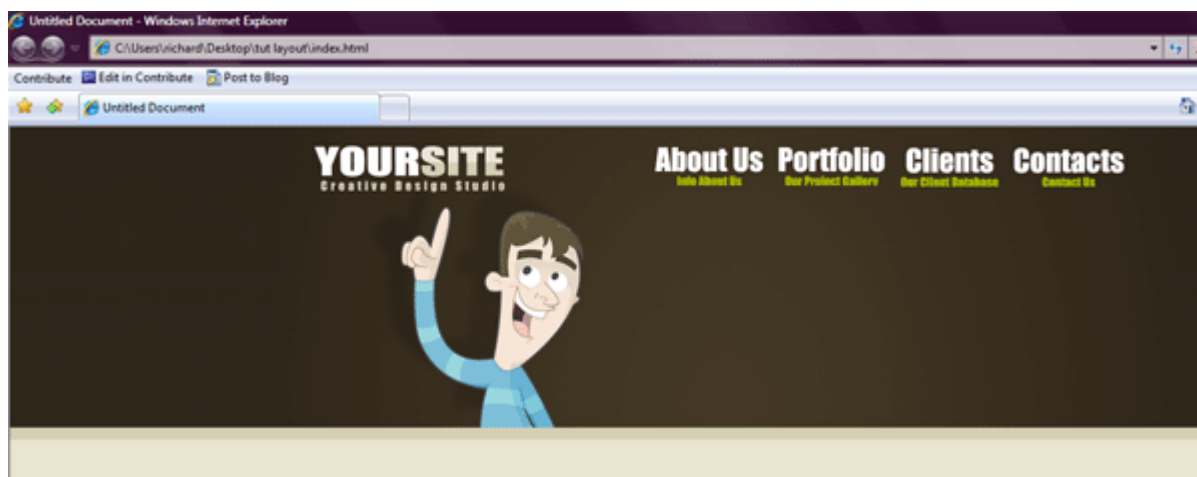
Теперь следующая часть кода относится к `ul` и `li` тэгам. Тэги `li` используются с идентификаторами `id`. Название класса в CSS таблице начинается с точки (`.`), а название идентификатора с решётки (`#`). Разница между классом и идентификатором в том, что класс может быть использован снова и снова, а идентификатор используется один раз. Наиболее важное свойство для `ul` и `li` это `list-style` оно позволяет удалить маркер из списка и поместить вместо него изображение.

Стиль `span` также важен, он позволяет добавлять стиль в текст, или манипулировать им с помощью, например, языка JavaScript. Далее идет код для изображений навигации, как вы можете видеть, что каждый идентификатор тэга `li` имеет свой

стиль, а названия пунктов простые и понятные. Для каждого пункта есть своя ссылка на фоновое изображение (background:url), которое мы создали ранее.

Ширина и высота соответствует размерам каждого изображения.

Свойство «display:block» означает то, что «элемент будет отображаться как элемент уровня блока, с разрывом строки перед элементом и после элемента». Сохраните файл CSS, перейдите в ваш HTML-файл и нажмите F12.



Следующим важным моментом является создание текста приветствия в панели навигации. Нам нужно добавить новый HTML-код для этой части:

```
<!--HEADER LOGO END-->
<!--FEATURED TEXT-->
<div id="featured">
<h1>Welcome...</h1>
<p>SOME TEXT HERE</p>
</div>
<!--FEATURED TEXT END-->
</div>
<!--HEADER BACKGROUND END-->
```

Тэг h1 означает, что это заголовок первого уровня. Вы можете использовать столько этих заголовков, сколько пожелаете. Теги p являются тэгами абзацев, где будет расположен наш текст. Таким образом, этот кусок кода содержит 3 элемента: div тэг с идентификатором «featured», тэг заголовка h1 и тег абзаца p.

Код CSS выглядит следующим образом:

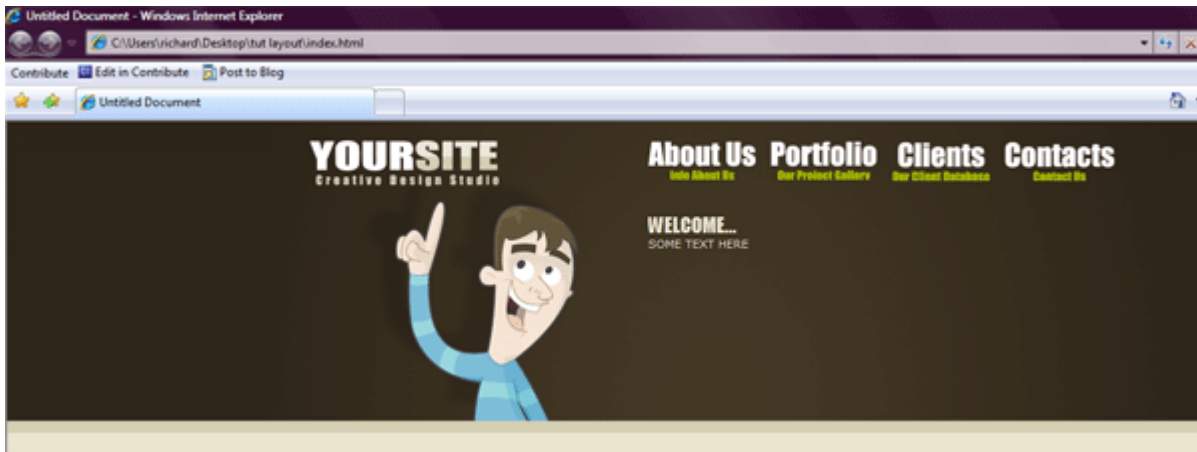
```
/* FEATURED*/
#featured {
height: 200px;
width: 480px;

font-size: 12px;
color: #eae5cf;
margin-top: 90px;
margin-left: 380px;
}
h1 {
padding:0;
margin:0;
font-size:20px;
color:#eae5cf;
text-transform:uppercase;

font-weight: normal;
}
p {
padding:0;
margin:0;
font-size:11px;
color:#eae5cf;

}
/* FEATURED END*/
```

Высота и ширина «#featured» соответствует ширине и высоте текста, который будет в нём расположен. Свойств «margin-top» сдвигает блок текста вниз на 90px и «margin-left» сдвигает его вправо на 380x. Другие свойства относятся к тексту, Вы можете установить какой угодно шрифт, цвет и размер текста. Ваш HTML файл должен выглядеть также как здесь:



Далее нам нужно создать область для содержимого или контента сайта. Эта область будет вмещать один привлекательный заголовок и шесть блоков контента, которые могли бы растягиваться. Первым нужно сделать привлекательный заголовок. Перейдите в PSD файл и скопируйте заголовок, а затем вставьте его в новый документ и сохраните его в папке изображений.



Перейдите в HTML документ и добавьте этот код перед закрывающим тэгом `</div>` контейнера.

```
<!--HEADER BACKGROUND END-->
<!--CONTENT LEFT-->
<div id="content">
<h2></h2>
</div>
<!--CONTENT LEFT END-->
</div>
<!--CONTAINER END-->
```

Тэг `div` с идентификатором «`content`» представляет собой большой блок, в который будет помещён блок контента и блок верхней и нижней панелей, которые будут находиться снизу и сверху. Сейчас там находится только тэг `h2` для нашего привлекательного заголовка. Добавьте этот CSS-код для тэгов `div` и `h2`:

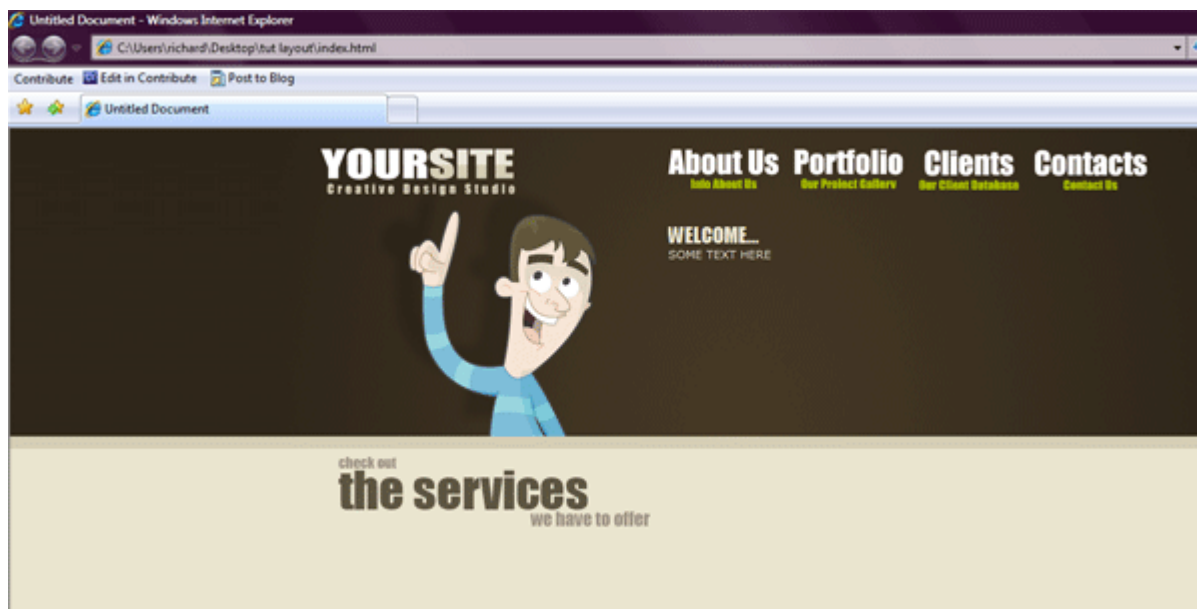
```
/* CONTENT*/
```



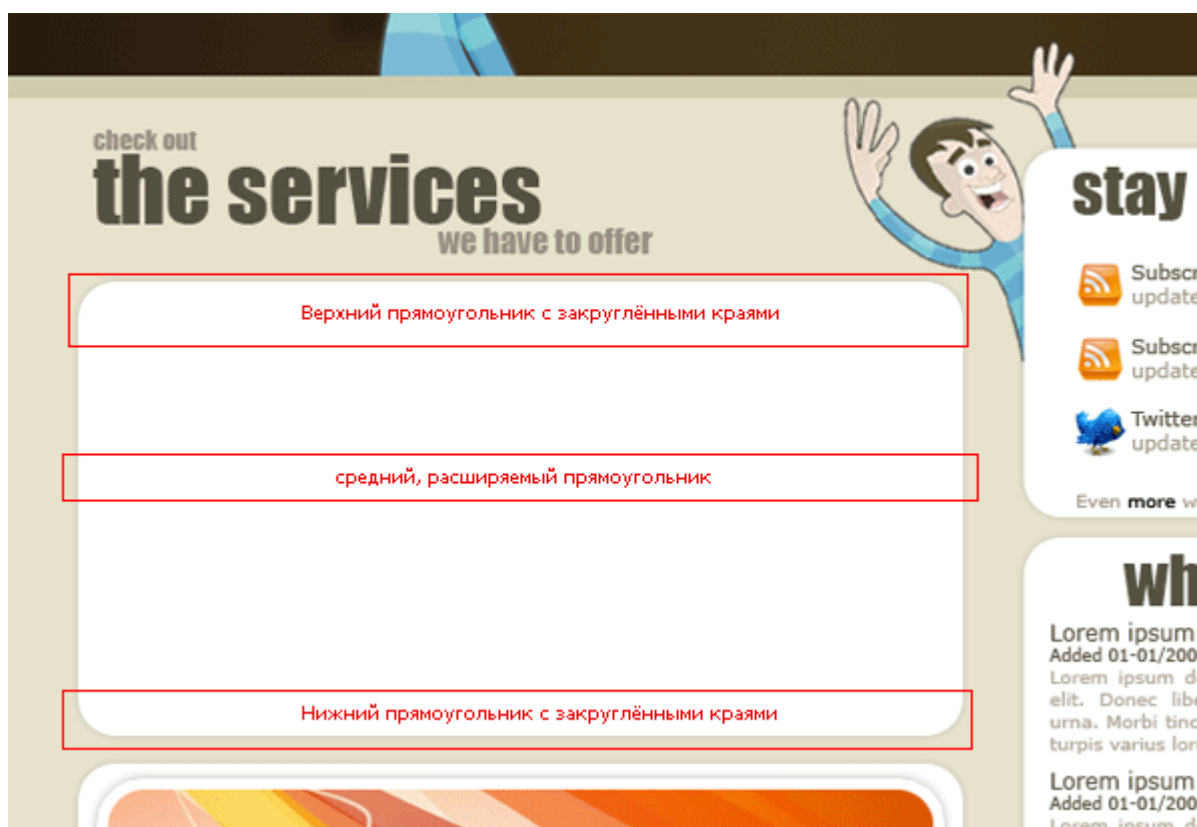
```
#content {
width:800px;
padding-top: 20px;
margin:auto;
float : left;
}
h2 {
background:url(images/h2.png);
width:302px;
height:67px;
margin-bottom:20px;
margin-left: 10px;
}
/* CONTENT END*/
```

Блок с содержимым должен быть шириной 800px. Разрыв между блоком контента и блоком верхней и нижней панелей будет сильно увеличен. Мы установили свойство «padding» в 20px по высоте в тэге div с идентификатором «content», чтобы он не приклеялся снизу заголовка. Затем мы установили свойство «margin» в «auto» поэтому всё, внутри тэга div с идентификатором content будет выравниваться по центру.

Стиль h2 для нашего заголовка имеет фоновое изображение «h2.png». Ширина и высота должна быть той же, что и у изображения. Наконец, мы добавили свойству «margin-bottom: 20px», чтобы блок контента не прилипал снизу к нашему привлекательному заголовку, а свойство margin-left сдвигает привлекательный заголовок вправо. Посмотрите HTML-файл, он должен выглядеть так:

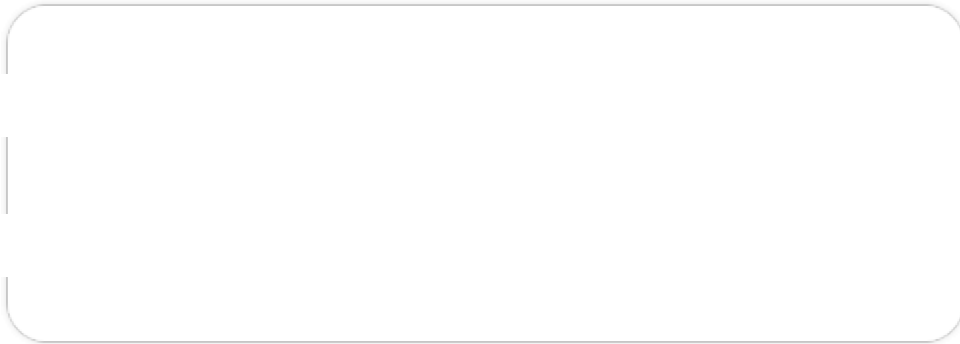


Теперь поговорим о самом блоке контента. Блок контента окружён сверху и снизу прямоугольниками с закруглёнными краями, таким образом мы имеем три элемента для каждого блока контента. Верхний прямоугольник с закруглёнными краями, средний маленький кусочек, который может расширяться, в зависимости от размера контента и нижний прямоугольник с закруглёнными краями.



Разделим эти части контентных блоков на три разных файла и

сохраним их как png-файлы с прозрачным фоном в папке с изображениями.



Добавим HTML-код для блоков контента. Так как эти контентные блоки будут повторяться, нам нужно использовать классы. Помните я говорил, что идентификаторы должны использоваться только один раз. Вот теперь давайте рассмотрим замечательный пример. Внутри контента div между тэгом h2 и закрывающим тэгом </div> Вам нужно добавить следующий код:

```
<div class="content_box">
```

```
<div class="content_top">  
<h3>Content Box One</h3>  
</div>
```

```
<div class="content_main">  
<p>some text</p>  
</div>
```

```
<div class="content_bottom">  
<p>Posted on 04/01/2009</p>  
</div>
```

Первый класс «content_box» – это блок, в котором будет находиться наш контентный блок, итак это будет контентный блок внутри блока. Также есть класс «content_top» это верхняя часть контентного блока. Здесь я хочу добавить заголовок, поэтому я могу использовать тэг h3.

Класс «content_main» это расширяющаяся часть нашего блока, поэтому этот блок должен подстраиваться под входящий в

него текст и изображения. В конце класс «content_bottom» Вы уже догадались, что это подвал нашего блока. Тэг p является подвалом контентного блока, который я добавил. Вот CSS код для перечисленных выше элементов:

```
/* CONTENT BOX*/
.content_box {
width:485px;
float:left;
margin:auto;
}
.content_top{
background:url(images/content_top.png);
background-repeat:no-repeat;
width:485px;
height:38px;
}
.content_top h3{
color: #5C5741;

font-size: 24px;
font-weight: normal;
padding-top: 6px;
padding-left: 20px;
}
.content_main{
background:url(images/content.png);
background-repeat:repeat-y;
width:485px;
padding-top: 6px;
padding-left: 20px;
padding-right: 0px;
}
.content_main p {

color: #A09484;
font-size: 11px;
padding-right: 30px;
}
.content_bottom {
background:url(images/content_bottom.png);
```

```
background-repeat:no-repeat;
width:485px;
height:36px;
margin-bottom: 10px;
}
.content_bottom p {

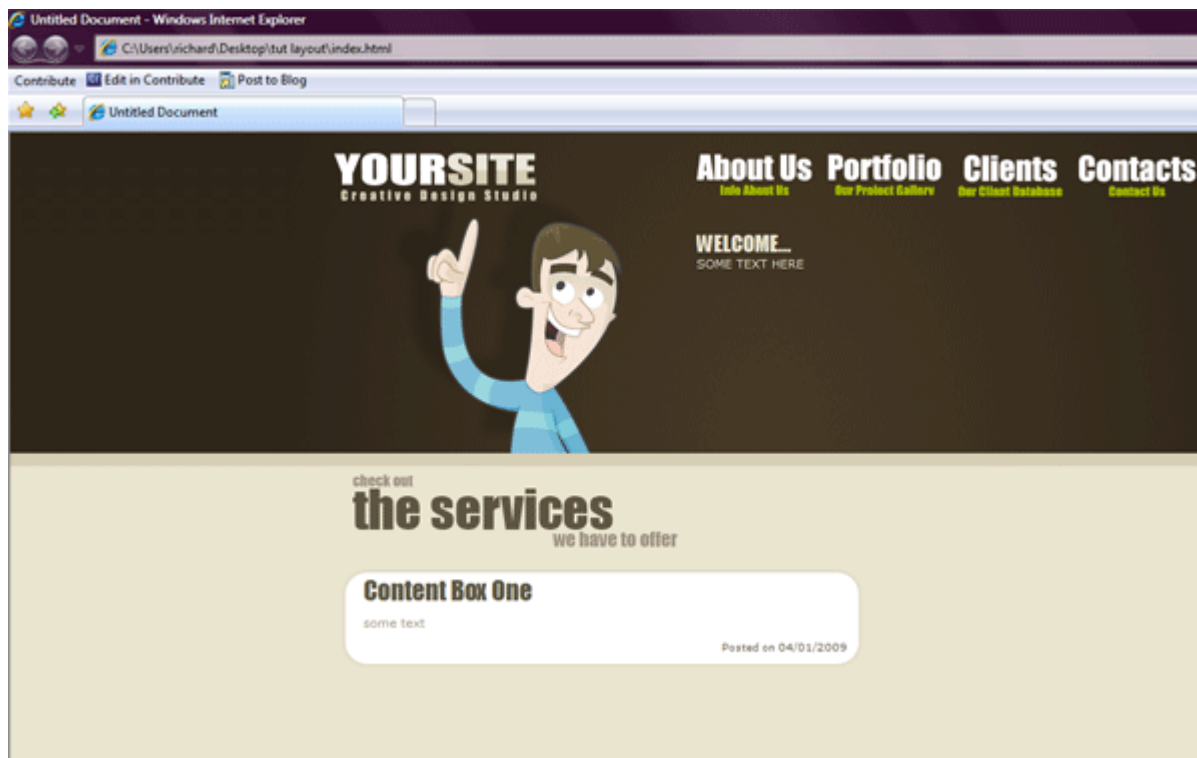
color: #5C5741;
font-size: 9px;
float: right;
padding-top: 10px;
padding-right: 15px;
padding-left: 15px;
padding-bottom: 10px;
}
/* CONTENT BOX END*/
```

Класс «content_box» имеет свойство «width:485px» этого достаточно, чтобы вмещать изображения. Мы хотим, чтобы изображения находились слева, поэтому ставим свойство «float» в «left», мы также хотим, чтобы контентные блоки находились ровно в центре, поэтому ставим свойство «margin» в «auto». Следующий набор классов описывает контентные блоки и их классы содержат изображения, которые формируют контентный блок.

Второй класс «content_top» он выводит верхний прямоугольник с закруглёнными краями, поэтому мы выставляем реальные размеры его фонового изображения, оно не будет повторяться. Класс «content_bottom», точно такой же, но только отличается свойством «margin-bottom» равным 10px. Я добавил этот разрыв, чтобы блоки не приклеивались друг к другу снизу.

Класс «content_main» вмещает в себя основной контент, сейчас он чуть-чуть отличается от других классов, так как изображение, которое мы создали в [Photoshop](#) будет повторяться по оси y или по вертикали, так же заметьте, что класс имеет величину в свойстве «width», так как ширина должна быть одинаковой.

Здесь нет высоты, потому что мы хотим расширять блок, но если мы установим высоту никакого расширения не произойдёт, так как блок будет фиксированным по вертикали. Мы установили здесь свойство «padding», так как нам нужно, чтобы текст находился внутри блока, был сдвинут слева и сверху внутрь. Если Вы сохраните Ваши файлы HTML и CSS нажмите F12 и в браузере посмотрите что получилось, Вы должны увидеть Ваш контентный блок.



Чтобы добавить больше контентных блоков, Вы можете продублировать HTML-код и соответствующие классы. Три дополнительных контентных блока будут выглядеть следующим образом.

```
<!--CONTENT LEFT-->
<div id="content">
<h2></h2>
<div class="content_box">
<div class="content_top">
<h3>Content Box One</h3>
</div>
<div class="content_main">
<p>some text</p>
```

```
</div>
<div class="content_bottom">
<p>Posted on 04/01/2009</p>
</div>
<div class="content_box">
<div class="content_top">
<h3>Content Box Two</h3>
</div>
<div class="content_main">
<p>some text</p>
</div>
<div class="content_bottom">
<p>Posted on 04/01/2009</p>
</div>
<div class="content_box">
<div class="content_top">
<h3>Content Box Three</h3>
</div>
<div class="content_main">
<p>some text</p>
</div>
<div class="content_bottom">
<p>Posted on 04/01/2009</p>
</div>
</div>
</div>
</div>
</div>
<!--CONTENT LEFT END-->
```

Теперь поговорим о правой боковой панели, создается она точно так же, как и главные контентные блоки. Включает три изображения для классов «sidebar_top», «sidebar_main» и «sidebar_bottom». Откройте PSD-файл и создайте эти изображения. Разметка та же, за исключением классов тэгов div. Давайте подготовим наш HTML-файл. Разметка должна выглядеть так:

```
<!--SIDEBAR RIGHT-->
<div id="sidebar">
<div class="sidebar_top">
<h4>stay connected...</h4>
```

```
</div>
<div class="sidebar_main">
<p>some text here</p>
</div>
<div class="sidebar_bottom">
</div>
</div>
<!--SIDEBAR END-->
```

Начальный тэг div – это и есть боковая панель, которая вмещает в себя блоки боковой панели. Первый класс – это верхняя часть боковой панели. Тэг h4 – это заголовок четвертого уровня для нашего блока. Класс «sidebar_main» – это наша расширяемая область и последний класс «sidebar_bottom», который является подвалом нашей боковой панели. Файл CSS выглядит следующим образом.

```
/* SIDEBAR*/
#sidebar {
width:305px;
float: right;
margin-top : 3px;
}
.sidebar_top {
background:url(images/sb_top.png);
background-repeat:no-repeat;
width:305px;
height:40px;
}
.sidebar_top h4 {

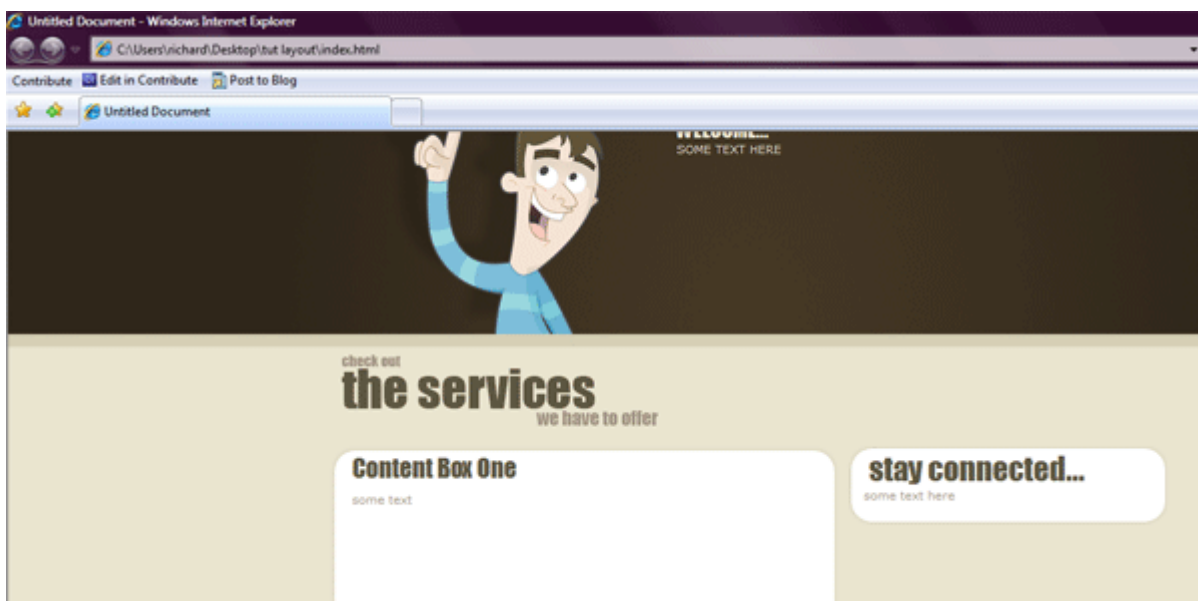
font-size: 30px;
font-weight: normal;
color: #5C5741;
padding-top:2px;
padding-left:20px;
}
.sidebar_main {
background:url(images/sb_main.png);
background-repeat:repeat-y;
width:305px;
```



```
padding-left:15px;
}
.sidebar_main p {

color: #A09484;
font-size:11px;
padding-right: 10px;
}
.sidebar_bottom {
background:url(images/sb_bottom.png);
background-repeat:no-repeat;
width:305px;
height:22px;
margin-bottom: 10px;
}
/* SIDEBAR END*/
```

Здесь два простых стиля тэга div, ширина нашего блока, в котором размещается контент равна ширине изображения в 305px. Нам нужно иметь разрыв справа от нашей боковой панели «float:right». Классы «sidebar_top» и «sidebar_bottom» похожи на классы главного контентного блока, а «sidebar_bottom» имеет разрыв в 10px свойство «margin». Класс «sidebar_main» такой же как и предыдущие, но не имеет высоты, так как его фоновое изображение должно повторяться, как у главного контентного блока. Когда Вы добавите код просмотрите его, нажав F12. Это мой:



Скопируйте классы, чтобы добавить больше блоков на Вашу боковую панель, код для трех блоков выглядит так:

```
<!--SIDEBAR RIGHT-->
<div id="sidebar">

<div class="sidebar_top">
<h4>stay connected...</h4>
</div>
<div class="sidebar_main">
<p>some text</p>
</div>
<div class="sidebar_bottom">
</div>

<div class="sidebar_top">
<h4>whats new...</h4>
</div>
<div class="sidebar_main">
<p>some text</p>
</div>

<div class="sidebar_bottom">
</div>

<div class="sidebar_top">
<h4>monthly poll...!</h4>
</div>
<div class="sidebar_main">
<p>some text</p>
</div>
<div class="sidebar_bottom">
</div>
</div>
<!--SIDEBAR RIGHT END-->
```

Теперь пришло время заняться подвалом, тэг div подвала должен располагаться за тэгом div контейнера. Подвал не будет фиксированным, как заголовок, а будет повторяться вдоль оси x. Код подвала следующий:

```
<!--FOOTER-->
```

```
<div id="footer">
<p>Copyright &amp;amp;amp;amp;amp;amp;copy; Yoursite.com | All
Rights Reserved<br />
DO NOT REMOVE THE LINK BELOW
<br />
Designed                               By                               <a
href="http://www.hv-designs.co.uk">HV-Designs.co.uk</a></p>
</div>
<!--FOOTER END-->
```

Код достаточно прост, тэг div с абзацем p для текста. CSS-код для подвала выглядит так:

```
/* FOOTER*/
#footer {
height:101px;

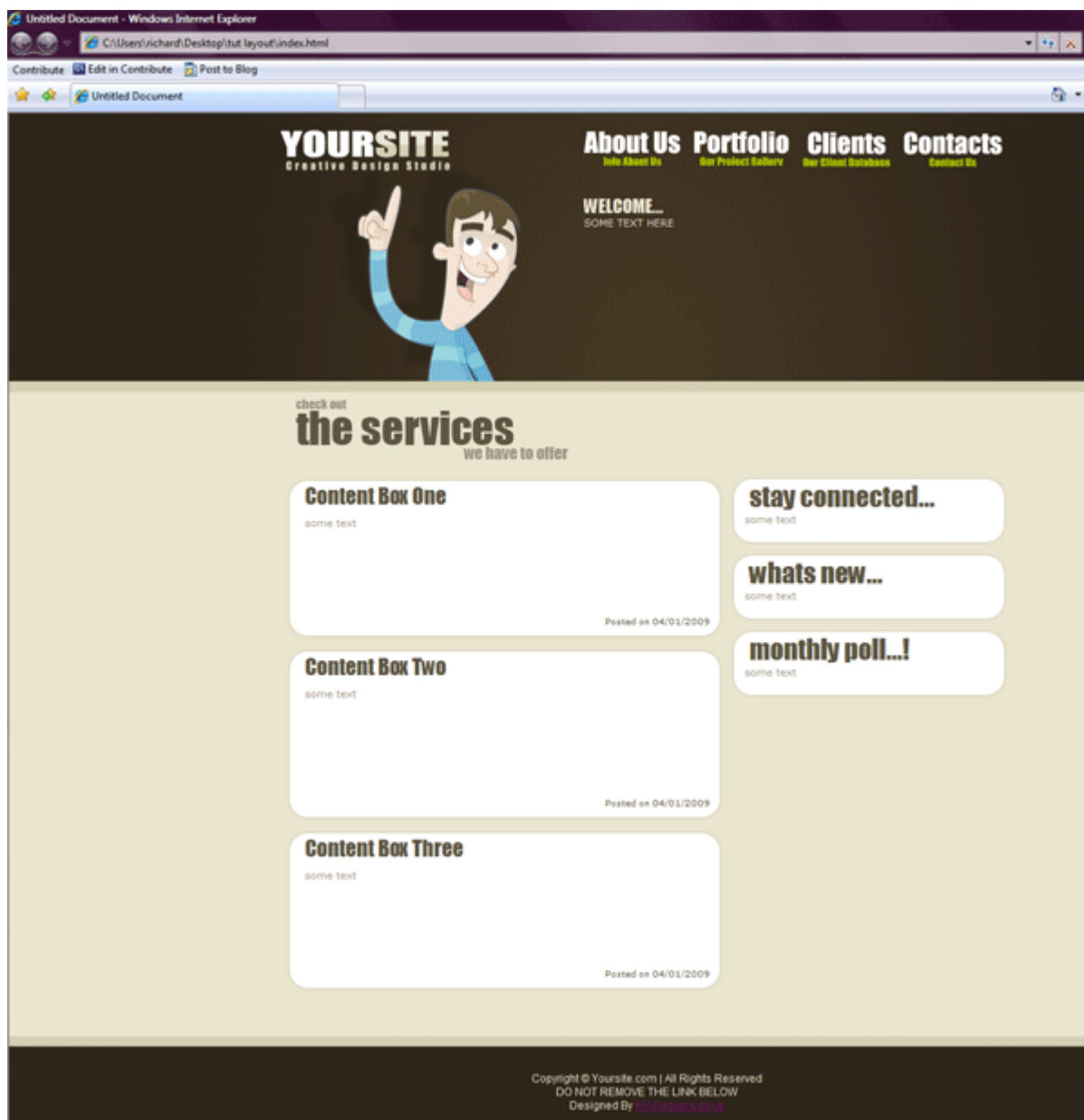
background-image:url(images/footer.png);
background-repeat:repeat-x;
background-color:#2F2519;
clear:both;
}
#footer p {

font-size: 12px;
color: #EAE5CF;
margin: auto;
text-align: center;
padding-top: 20px;
}
/* FOOTER END*/
```

Тэг подвала div имеет высоту 101px, такую же как и высота изображения, но не имеет ширину, так как мы хотим расширять его по оси x, «background-repeat:repeat-x». Свойство clear:both очень важно, так как мы не хотим, чтобы какие-либо элементы были размещены по обе стороны заголовка. Абзац p подвала не нуждается в объяснении, он объясняет себя сам.

Вот и всё, только ещё можно сказать о дополнительных стилях, например, стили для ссылок, заголовков или цвета для шрифтов. Что касается ссылок на страницы вашего сайта, то Вы

можете прочитать, как сделать их более привлекательными для посетителей Вашего сайта, прочитав статью о [настройке ссылок сайта в удобные для восприятия человеком](#).



Спасибо за то, что прочитали эту статью.

Источник: www.webdesign.org



Пожалуйста, [Войдите](#) или [Зарегистрируйтесь](#) чтобы получить ссылку на скачивание.

Этот блог читают уже много людей
- читай и Ты!

Да, Я тоже хочу читать статьи!