

Лабораторная работа № 11

Тема работы: «Подготовка psd-макета к верстке»

1. Цель работы

Формирование умений верстки html-страницы в соответствии с psd-макетом.

2. Задание

Сверстать HTML-страницу в соответствии с psd-макетом, используя средства HTML и CSS.

3. Оснащение работы

ПК, текстовый редактор, браузер.

4. Основные теоретические сведения

Преимущество данной верстки состоит в том, что код получается более «чистым», а значит быстрее загружается и легче изменяется под специфические нужды.

В качестве примера выбран бесплатный psd шаблон Corporate Blue от студии Pscklaboratory, рисунок 11.1.



Рисунок 11.1 – PSD-макет Corporate Blue

Структура файлов представлена на рисунке 11.2.

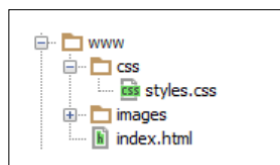


Рисунок 11.2 – Структура файлов

Сперва создается простая структуру файлов для документов. Далее необходимо создать папку с названием проекта, например Whitesquare. В ней

создаем пустой файл index.html. В папке проекта создаем папку css с пустым файлом styles.css. В папке проекта создаем пустую папку images.

Общие изображения

На данном этапе нужно нарезать и сохранить только общие изображения, которые будут на всех страницах сайта и не относятся к контенту. В данном случае это будет светло-серый фон страницы, фон заголовка, пустое изображение, два логотипа и кнопки социальных сетей.

Логотипы можно сохранить следующим образом:

```
/images/logo.png  
/images/footer-logo.png
```

В качестве пустых картинок из макета использовано однопиксельное серое изображение.

Повторяющиеся фоновые изображения необходимо вырезать минимальным кусочком достаточным для образования полного изображения повторением по вертикали и горизонтали.

```
/images/bg.png  
/images/h1-bg.png
```

Иконки социальных сетей с одинаковыми размерами удобно сохранить в один файл и использовать как спрайты для более быстрой загрузки. Для этого можно склеить картинки вручную в Photoshop, а можно сначала нарезать по одной, а затем склеить с помощью специально сервиса, например: <http://ru.spritegen.website-performance.org>. В итоге получится два файла:

```
/images/social.png  
/images/social-small.png
```

Общее правило при именовании изображений заключается в том, что мелкие и простые картинки, такие, как иконки, логотипы и так далее сохраняются в формате png, а фотографии в формате jpg.

Основные стили

На данном этапе желательно перенести все визуальные стили из дизайна в CSS, которые будут применяться по умолчанию для каждого тега.

Основной цвет фона примерно соответствует цвету #f8f8f8. Он будет показан в случае, если фоновая картинка не загрузится. Наверху страницы находится серая дизайнерская полоска. Применим ее через свойство border для body.

Основным шрифтом является тот шрифт, которым написан текст в области контента. Чтобы узнать его стили нужно выделить его в Photoshop'е и посмотреть свойства шрифта. В данном случае это Tahoma 12px с цветом #8f8f8f. Так же в этом макете параграфы имеют увеличенные отступы.

Прописываем все эти стили в styles.css:

```
body {
  color: #8f8f8f;
  font: 12px Tahoma, sans-serif;
  background-color: #f8f8f8;
  border-top: 5px solid #7e7e7e;
  margin: 0;
}
input[type="text"] {
  background-color: #f3f3f3;
  border: 1px solid #e7e7e7;
  height: 30px;
  color: #b2b2b2;
  padding: 0 10px;
  vertical-align: top;
}
button {
  color: #fff;
  background-color: #29c5e6;
  border: none;
  height: 32px;
  font-family: 'Oswald', sans-serif;
}
p {
  margin: 20px 0;
}
```

В дальнейшем все стили будут описаны в этом же файле.

Каркас HTML

Запишем в index.html следующий код:

```
<!doctype html>
<html>
<head>
  <meta http-equiv="Content-type" content="text/html; charset=utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=Edge">
  <title>Whitesquare</title>
  <link rel="stylesheet" href="css/styles.css" type="text/css">
  <link rel="stylesheet"
href="http://fonts.googleapis.com/css?family=Oswald:400,300" type="text/css">

  <!--[if lt IE 9]>
  <script
src="http://html5shiv.googlecode.com/svn/trunk/html5.js"></script>
```

```

<![endif]-->
</head>
<body>
</body>
</html>

```

Здесь указываем, что используем разметку HTML5, кодировку utf-8, страница называется Whitesquare. Также подключаем файл стилей и внешний файл со стилями шрифтов.

В последнем блоке в секции head подключаем специальный скрипт, который позволяет поддерживать Html5 теги в браузерах Internet Explorer меньше 9 версии. Мета-тег X-UA-Compatible сообщает, что в случае использования браузера Internet Explorer, он должен отразить сайт самым современным способом.

Весь html код в дальнейшем будет относиться к этому же файлу, поэтому специально указывать куда прописывать html код автор больше не будет.

Макет

Сайт состоит из двух колонок: основного контента и сайдбара. Над ними находится шапка (header), в которой располагаются три горизонтальных блока: логотип с поиском, меню и название страницы. В самом низу под колонками располагается серый горизонтальный блок футера (footer). Вышеописанная структура макета представлена на рисунке 11.3.

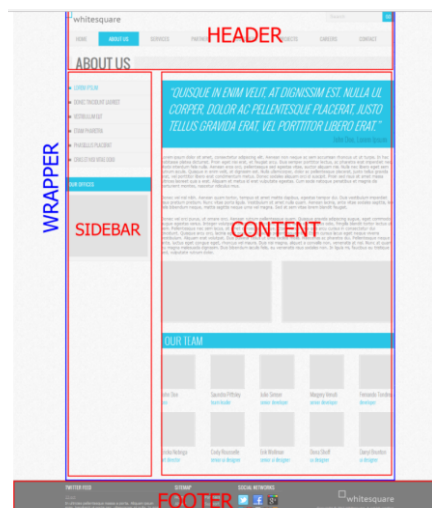


Рисунок 11.3 – Внешний вид макета

Опишем это в теге body:

```

<body>
  <div id="wrapper">
    <header></header>
    <nav></nav>
    <div id="heading"></div>

```

```

        <aside></aside>
        <section></section>
    </div>
    <footer></footer>
</body>

```

Wrapper используется для объединения блоков и их выравнивания по центру страницы.

Затем укажем стили блоков:

```

#wrapper {
    max-width: 960px;
    margin: auto;
}
header {
    padding: 20px 0;
}

```

Логотип

Вставляем логотип в тег header:

```

<header>
    <a href="/"><img src="" alt="Whitesquare logo"></a>
</header>

```

Дополнительных стилей не требуется.

Поиск

Вставляем форму поиска в тег header:

```

<header>
...
<form name="search" action="#" method="get">
    <input type="text" name="q" placeholder="Search"><button
type="submit">GO</button>
</form>
</header>

```

И стили выравнивания по правому краю для нее:

```

form[name="search"] {
    float: right;
}

```

Меню

Для отображения меню необходимо создать список со ссылками внутри тега nav:

```
<nav>
  <ul class="top-menu">
    <li><a href="/home/">HOME</a></li>
    <li class="active">ABOUT US</li>
    <li><a href="/services/">SERVICES</a></li>
    <li><a href="/partners/">PARTNERS</a></li>
    <li><a href="/customers/">CUSTOMERS</a></li>
    <li><a href="/projects/">PROJECTS</a></li>
    <li><a href="/careers/">CAREERS</a></li>
    <li><a href="/contact/">CONTACT</a></li>
  </ul>
</nav>
```

CSS стили для него будут следующие:

```
nav a {
  text-decoration: none;
}

nav ul {
  margin: 0;
  padding: 0;
}

nav li {
  list-style-position: inside;
  font: 14px 'Oswald', sans-serif;
  padding: 10px;
}

.top-menu li {
  display: inline-block;
  padding: 10px 30px;
  margin: 0;
}

.top-menu li.active {
  background: #29c5e6;
  color: #fff;
}

.top-menu a {
```

```
color: #b2b2b2;  
}
```

Здесь описано, что для всех навигаций ссылки не будут иметь подчеркивания, убрали стандартные отступы для элементов списка, отобразили список горизонтально и указали нужные цвета и шрифт.

Заголовок страницы

Заголовок страницы помещается в div с идентификатором heading:

```
<div id="heading">  
  <h1>ABOUT US</h1>  
</div>
```

Заголовок имеет следующие стили:

```
#heading {  
  background: transparent url(../images/h1-bg.png);  
  margin: 30px 0;  
  padding-left: 20px;  
}  
  
h1 {  
  display: inline-block;  
  color: #7e7e7e;  
  font: 40px/40px 'Oswald', sans-serif;  
  background: url(../images/bg.png);  
  margin: 0;  
  padding: 0 10px;  
}
```

Рисуем серую полоску фоном на div'е, и в нее вкладываем инлайновый h1 с нужным шрифтом и фоном цвета страницы, чтобы создалось впечатление прозрачного фона для h1.

Колонки

Для того, чтобы создать колонки страницы нужно прописать следующие стили:

```
aside {  
  float: left;  
  width: 250px;  
}  
section {  
  margin-left: 280px;  
  padding-bottom: 50px;  
}
```

Здесь задали фиксированную ширину 250 пикселей для сайдбара, прибили его к левому краю и отодвинули колонку с контентом вправо на 280 пикселей от левого края. Также добавили отступ у контента снизу.

Подменю

Подменю создаем аналогично главному меню. Для этого в теге `aside` прописываем следующее:

```
<aside>
<nav>
  <ul class="aside-menu">
    <li class="active">LOREM IPSUM</li>
    <li><a href="/donec/">DONEC          TINCIDUNT
LAOREET</a></li>
    <li><a href="/vestibulum/">VESTIBULUM ELIT</a></li>
    <li><a href="/etiam/">ETIAM PHARETRA</a></li>
    <li><a href="/phasellus/">PHASELLUS PLACERAT</a></li>
    <li><a href="/cras/">CRAS ET NISI VITAE ODIO</a></li>
  </ul>
</nav>
</aside>
```

И применяем к подменю следующие стили:

```
.aside-menu li {
  font-weight: 300;
  list-style-type: square;
  border-top: 1px solid #e7e7e7;
}

.aside-menu li:first-child {
  border: none;
}

.aside-menu li.active {
  color: #29c5e6;
}

.aside-menu a {
  color: #8f8f8f;
}
```

Для подменю применяется более тонкий шрифт и квадратные маркеры. Для отображения разделителей рисуем верхнюю границу у каждого элемента списка, кроме первого.

Контент сайдбара

В контенте сайдбара помимо подменю в макете располагается также изображение с расположением офисов.

В html это выглядит так:

```
<h2>OUR OFFICES</h2>
<p>

</p>
```

В стилях укажем шрифты, цвета и отступы:

```
aside > h2 {
  background: #29c5e6;
  font: 14px 'Oswald', sans-serif;
  color: #fff;
  padding: 10px;
  margin: 30px 0 0 0;
}
```

```
aside > p {
  background: #f3f3f3;
  border: 1px solid #e7e7e7;
  padding: 10px;
  margin: 0;
}
```

Данные стили применяются только к заголовкам и параграфам, лежащим непосредственно внутри сайдбара, но не глубже.

Цитата

Верстку контента начнём с добавления цитаты. Добавим код цитаты в раздел section

```
<section>
<blockquote>
<p>
  “QUISQUE IN ENIM VELIT, AT DIGNISSIM EST. NULLA UL
  CORPER, DOLOR AC PELLENTESQUE PLACERAT, JUSTO TELLUS
  GRAVIDA ERAT, VEL PORTTITOR LIBERO ERAT.”
</p>
<cite>John Doe, Lorem Ipsum</cite>
</blockquote>
</section>
```

И применим для него стили:

```
blockquote {  
  margin: 0;  
  background: #29c5e6;  
  padding: 10px 20px;  
  font-family: 'Oswald', sans-serif;  
  font-weight: 300;  
}
```

```
blockquote p {  
  color: #fff;  
  font-style: italic;  
  font-size: 33px;  
  margin: 0;  
}
```

```
blockquote cite {  
  display: block;  
  font-size: 20px;  
  font-style: normal;  
  color: #1d8ea6;  
  margin: 0;  
  text-align: right;  
}
```

Контент

Все стили для текста контента уже добавил, остается добавить только три параграфа с самим текстом после:

```
<p>Lorem ipsum dolor sit amet...</p>
```

```
<p>Donec vel nisl nibh...</p>
```

```
<p>Donec vel nisl nibh...</p>
```

Следующим шагом нужно добавить два изображения, которые находятся в конце текста контента. Делается это с помощью тега:

```
<figure>
```

```
  
```

```
</figure>
```

```
<figure>
```

```
  
```

```
</figure>
```

```
figure {  
  display: inline-block;  
  margin: 0;
```

```

    font-family: 'Oswald', sans-serif;
    font-weight: 300;
}

figure img {
    display: block;
    border: 1px solid #fff;
    outline: 1px solid #c9c9c9;
}

figure figcaption {
    font-size: 16px;
    font-weight: 300;
    margin-top: 5px;
}

figure figcaption span {
    display: block;
    font-size: 14px;
    color: #29c5e6;
}

section > figure + figure {
    margin-left: 28px;
}

```

Здесь убираем стандартные отступы у figure, отобраем его как инлайновый блок и применили нужный шрифт. Изображение отображаем как блочный элемент с белой рамкой. Вторую серую рамку можно сделать через css-свойство outline. Самое интересное находится в последнем правиле, которое задает левый отступ у всех figure кроме первого внутри тега section.

Блок «Our team»

При верстке этого блока добавим сначала заголовок:

```

<h2>OUR TEAM</h2>
section > h2 {
    background: #29c5e6;
    font: 30px 'Oswald', sans-serif;
    font-weight: 300;
    color: #fff;
    padding: 0 10px;
    margin: 30px 0 0 0;
}

```

А затем два блока-строки с карточками сотрудников:

```

<div class="team-row">
  <figure>
    
    <figcaption>John Doe<span>ceo</span></figcaption>
  </figure>
  <figure>
    
    <figcaption>Saundra                                Pittsley<span>team
leader</span></figcaption>
  </figure>
  ...
</div>
<div class="team-row">
  <figure>
    
    <figcaption>Ericka                                Nobriga<span>art
director</span></figcaption>
  </figure>
  <figure>
    
    <figcaption>Cody                                Rousselle<span>senior          ui
designer</span></figcaption>
  </figure>
  ...
</div>

```

Таким образом, карточка (figure) состоит из фотографии (img), подписи (figcaption) с именем сотрудника и его должностью (div). Карточки будут иметь следующие стили:

```

figure figcaption {
  font-size: 16px;
  font-weight: 300;
  margin-top: 5px;
}

```

```

figure div {
  font-size: 14px;
  color: #29c5e6;
}

```

```

.team-row figure {
  margin-top: 20px;
}

```

```
.team-row figure + figure {  
    margin-left: 43px;  
}
```

Здесь мы задали шрифт и отступ для подписи, размер и цвет должности, добавили верхний отступ для карточек и указали, что для всех карточек в строке, кроме первой, должен быть отступ слева.

Футер

Футер состоит из четырёх больших блоков: ленты Твиттера, карты сайта, социальных ссылок и логотипа с копирайтом.

Для начала создадим контейнер футера с этим блоками:

```
<footer>  
  <div id="footer">  
    <div id="twitter"></div>  
    <div id="sitemap"></div>  
    <div id="social"></div>  
    <div id="footer-logo"></div>  
  </div>  
</footer>
```

И применим к нему оформление:

```
footer {  
    background: #7e7e7e;  
    color: #dbdbdb;  
    font-size: 11px;  
}
```

```
#footer {  
    max-width: 960px;  
    margin: auto;  
    padding: 10px 0;  
    height: 90px;  
}
```

Контейнер с id="footer" находится внутри тега footer, это даёт нам возможность через тег footer задать серую область по всей ширине экрана, а через внутренний div отцентрировать с максимальной шириной 960 пикселей. Также этот div задает обоим блокам высоту 90 пикселей.

Лента Твиттера

Верстаем содержимое ленты Твиттера:

```
<div id="twitter">  
  <h3>TWITTER FEED</h3>
```

```
<time datetime="2012-10-23"><a href="#">23 oct</a></time>
```

<p>

In ultricies pellentesque massa a porta. Aliquam ipsum enim, hendrerit ut porta nec, ullamcorper et nulla. In eget mi dui, sit amet scelerisque nunc. Aenean aug

</p>

</div>

Стили:

```
footer h3 {
  font: 14px 'Oswald', sans-serif;
  color: #fff;
  border-bottom: 1px solid #919191;
  margin: 0 0 10px 0;
}
```

```
#twitter time a {
  color: #b4aeae;
}
```

```
footer p {
  margin: 5px 0;
}
```

```
#twitter {
  float: left;
  width: 300px;
}
```

```
#twitter p {
  padding-right: 15px;
}
```

Из интересных моментов здесь следующее: подчеркивание у заголовка сделали через нижнюю границу, а сам блок твиттера, как и последующие блоки выровняем по левому краю и задаём ширину.

Карта сайта

Карта сайта представляет собой два блока со ссылками:

```
<div id="sitemap">
  <h3>SITEMAP</h3>
  <div>
    <a href="/home/">Home</a>
    <a href="/about/">About</a>
    <a href="/services/">Services</a>
  </div>
```

```

<div>
    <a href="/partners/">Partners</a>
    <a href="/customers/">Support</a>
    <a href="/contact/">Contact</a>
</div>
</div>

```

Ссылкам задаем цвет и оставляем подчеркивание только для наведённых.

Колонки со ссылками делаем через инлайновые блоки и затем свойством #sitemap div + div отодвигаем вторую колонку от первой:

```

footer a {
    color: #dbdbdb;
}

#sitemap {
    width: 150px;
    float: left;
    margin-left: 20px;
    padding-right: 15px;
}

#sitemap div {
    display: inline-block;
}

#sitemap div + div {
    margin-left: 40px;
}

#sitemap a {
    display: block;
    text-decoration: none;
    font-size: 12px;
    margin-bottom: 5px;
}

#sitemap a:hover {
    text-decoration: underline;
}

```

Социальные ссылки

Вставляем набор ссылок в контейнер;

```

<div id="social">
  <h3>SOCIAL NETWORKS</h3>
  <a href="http://twitter.com/" class="social-icon twitter"></a>
  <a href="http://facebook.com/" class="social-icon facebook"></a>
  <a href="http://plus.google.com/" class="social-icon google-plus"></a>
  <a href="http://vimeo.com/" class="social-icon-small vimeo"></a>
  <a href="http://youtube.com/" class="social-icon-small youtube"></a>
  <a href="http://flickr.com/" class="social-icon-small flickr"></a>
  <a href="http://instagram.com/" class="social-icon-small instagram"></a>
  <a href="/rss/" class="social-icon-small rss"></a>
</div>

```

И стилизуем их:

```

#social {
  float: left;
  margin-left: 20px;
  width: 130px;
}
.social-icon {
  width: 30px;
  height: 30px;
  background: url(../images/social.png) no-repeat;
  display: inline-block;
  margin-right: 10px;
}
.social-icon-small {
  width: 16px;
  height: 16px;
  background: url(../images/social-small.png) no-repeat;
  display: inline-block;
  margin: 5px 6px 0 0;
}
.twitter {
  background-position: 0 0;
}
.facebook {
  background-position: -30px 0;
}
.google-plus {
  background-position: -60px 0;
}
.vimeo {
  background-position: 0 0;
}

```



```
.youtube {
  background-position: -16px 0;
}
.flickr {
  background-position: -32px 0;
}
.instagram {
  background-position: -48px 0;
}
.rss {
  background-position: -64px 0;
}
```

Здесь применили технику спрайтов – когда один файл с изображением применяется для разных картинок. Все ссылки разделились на большие иконки (.social-icon) и маленькие (.social-icon-small). Мы задали этим классом отображение в виде инлайнового блока с фиксированными размерами и одинаковым фоном. А затем с помощью CSS сдвинули этот фон так, чтобы на каждой ссылке отобразилось соответствующее изображение.

Копирайт

Блок с копирайтом и логотипом – это картинка со ссылкой и параграф с текстом под ним.

```
<div id="footer-logo">
  <a href="/"><img src="" alt="Whitesquare logo"></a>
  <p>Copyright      ©      2012      Whitesquare.      А      <a
href="http://pcklab.com">pcklab</a> creation</p>
</div>
```

Стили делают аналогично предыдущим блокам с той лишь разницей, что блок прибавляется к правому краю и выравнивание внутри него так же по правому краю:

```
#footer-logo {
  float: right;
  margin-top: 20px;
  font-size: 10px;
  text-align: right;
}
```

5. Порядок выполнения работы

1. Получить у преподавателя вариант задания psd-макета.
2. Сверстать HTML-страницу, представленную в psd-макете.
3. В файле style.css описать все свойства для оформления страницы.
4. Подключите файл style.css к документу *.html.

6. Форма отчета о работе

Лабораторная работа № _____

Номер учебной группы _____

Фамилия, инициалы учащегося _____

Дата выполнения работы _____

Тема работы: _____

Цель работы: _____

Оснащение работы: _____

Результат выполнения работы: _____

7. Контрольные вопросы

1. Перечислите способы использования каскадных таблиц стилей. Укажите преимущества и недостатки каждого из них.
2. Укажите формат правила CSS.
3. Что такое селектор?
4. Что такое псевдокласс?
5. Какие основные блоки используются для макетирования?
6. Какие виды селекторов вы знаете?
7. Чем класс отличается от идентификатора?

8. Рекомендуемая литература

1. **Макфарланд, Д.** Новая большая книга CSS / Дэвид Макфарланд. – СПб.: Питер, 2016. – 720с.
2. **Никсон, Р.** Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5 /Р. Никсон. – 4-е изд. – СПб.: ООО «ПИТЕР М», 2017. – 768 с.
3. **Прохоренок, Н.А.** HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера / Н.А.Прохоренок. – СПб.: БХВ-Петербург, 2010. – 912с.
4. **Робсон, Э.** Изучаем HTML, XHTML и CSS / Э. Робсон. – 2-е изд. – СПб.: ООО «ПИТЕР М», 2017. – 720 с.
5. **Фрейн, Б.** HTML5 и CSS3. Разработка сайтов для любых браузеров и устройств / Бен Фрейн. – СПб.: Питер Пресс, 2017. — 272с.