Белгородский Государственный Технологический Университет им. В. Г. Шухова

Кафедра программного обеспечения вычислительной техники  
и автоматизированных систем

## Лабораторная работа №1 по теме: «Использование HTML и CSS для создания веб-сайтов»

**Выполнил:**  
студент группы ПВ-41  
Адаменко И. И.

**Проверил:**  
старший преподаватель  
Гарибов А. И.

Белгород  
2015

**Цель работы:** изучить основы HTML и CSS.

# Задание

Для выбранной предметной области создать веб-сайт, который должен содержать 3-4 HTML-страницы в едином стиле. На каждой странице должно быть меню, которое ведёт на другие страницы сайта.

# Теоретическая часть

HTML-страница имеет определённую структуру, которая позволяет интернет-браузеру интерпретировать содержимое страницы и отображать его требуемым образом. Базовым элементом HTML-страницы является «тег». Общий синтаксис написания тегов следующий:

<тег атрибут1="значение" атрибут2="значение">  
<тег атрибут1="значение" атрибут2="значение">данные</тег>

Как видно из этого примера, теги бывают двух типов — одиночные и парные (контейнеры). Одиночный тег используется самостоятельно, а парный может включать внутри себя другие теги или текст. У тегов допустимы различные атрибуты, которые разделяются между собой пробелом.

Если открыть любую веб-страницу, то она будет содержать в себе типичные элементы, которые не меняются от вида и направленности сайта. В качестве примера код простого документа, содержащего основные теги:

<!DOCTYPE html>  
<html lang="ru">  
<head>  
 <meta charset="UTF-8">  
 <title>Документ</title>  
</head>  
<body>  
 <h1>Заголовок</h1>  
 <!-- Комментарий -->  
 <p>Абзац</p>  
</body>  
</html>

Тег <html> определяет начало HTML-файла. Тег DOCTYPE до него описывает версию стандарта HTML для этого файла для лучшей интерпретации браузером. Тег <html> же хранит внутри себя заголовок — <head> и тело документа — <body>. На странице пользователю видно только содержимое тела, а заголовок же используется для передачи определённой информации браузеру.

В языке разметки HTML используется достаточно большое число различных тегов. В этом примере можно увидеть тег заголовка <h1> и тег абзаца <p>.

HTML задаёт основную структуру страницы, а также указывает, какие элементы на ней присутствуют. Само оформление страницы, положение и вид элементов регулируется с помощью каскадных таблиц стилей. Стилевые свойства в них записываются в своём, отличном от HTML формате. Основным понятием здесь выступает селектор — это некоторое имя стиля, для которого и добавляются параметры форматирования. В качестве селектора выступают теги и описывающие их атрибуты (классы, идентификаторы и пр.). Общий способ записи имеет следующий вид:

селектор { свойство: значение; }

# Практическая часть

Для этой лабораторной работы в качестве предметной области я выбрал сайт, посвящённый моей CSS-библиотеке.

Сайт (igoradamenko.github.io/awsm.css) состоит из следующих страниц:

* Overview — страница с кратким описанием того, для чего нужна библиотека и как ей пользоваться (рис. 1);
* Elements — страница с перечислением всех элементов, которые используются в библиотеке (рис. 2);
* Download — страница, на которой можно скачать библиотеку (рис. 3);
* Plugins — страница с описанием плагинов, которые я советую использовать вместе с библиотекой (рис. 4).

# Скриншоты

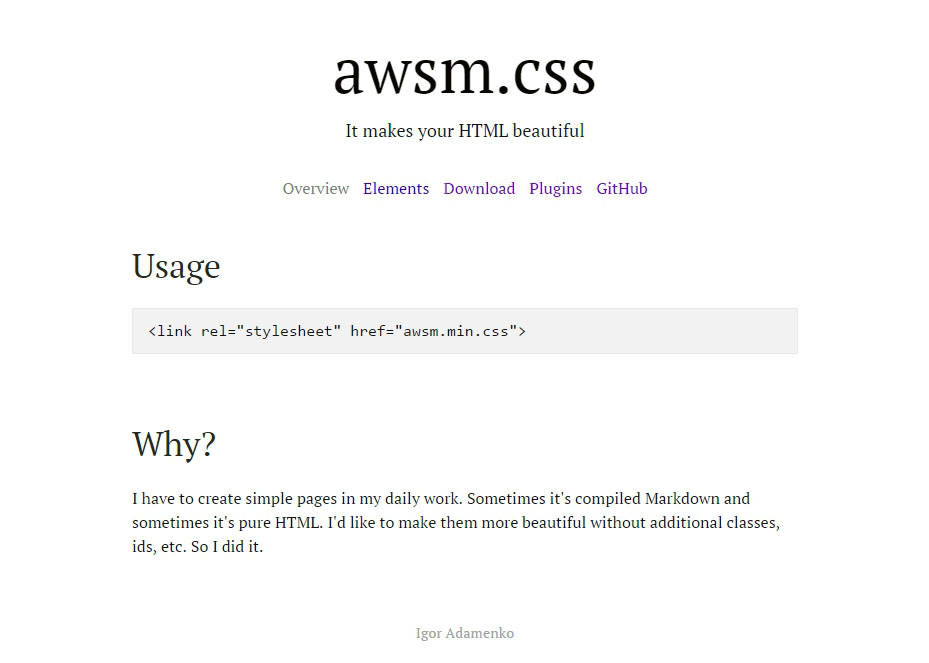


Рисунок 1. Overview

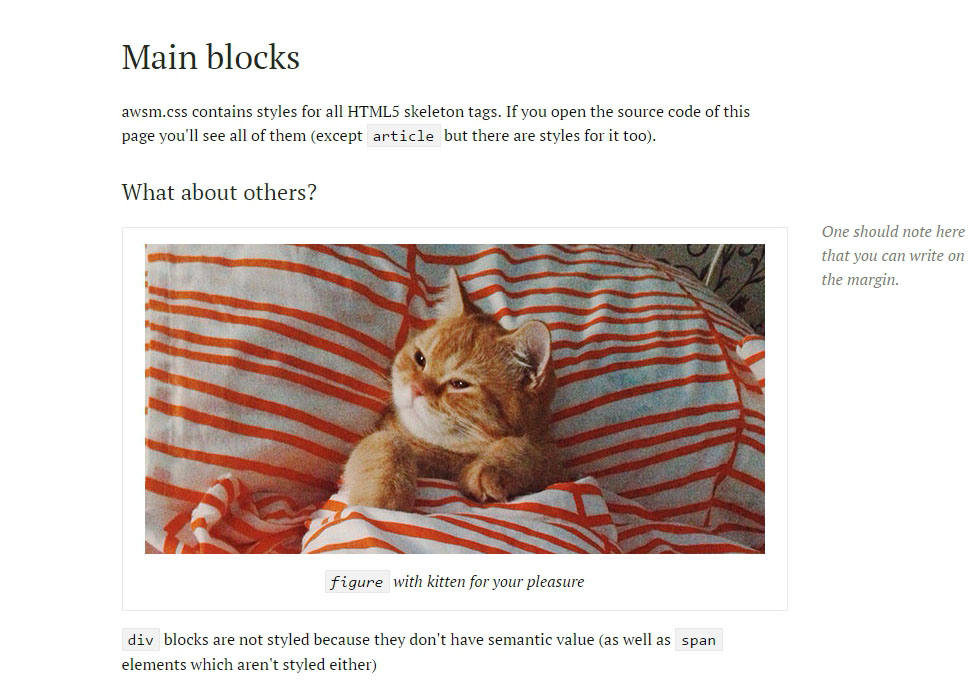


Рисунок 2. Elements (частично)

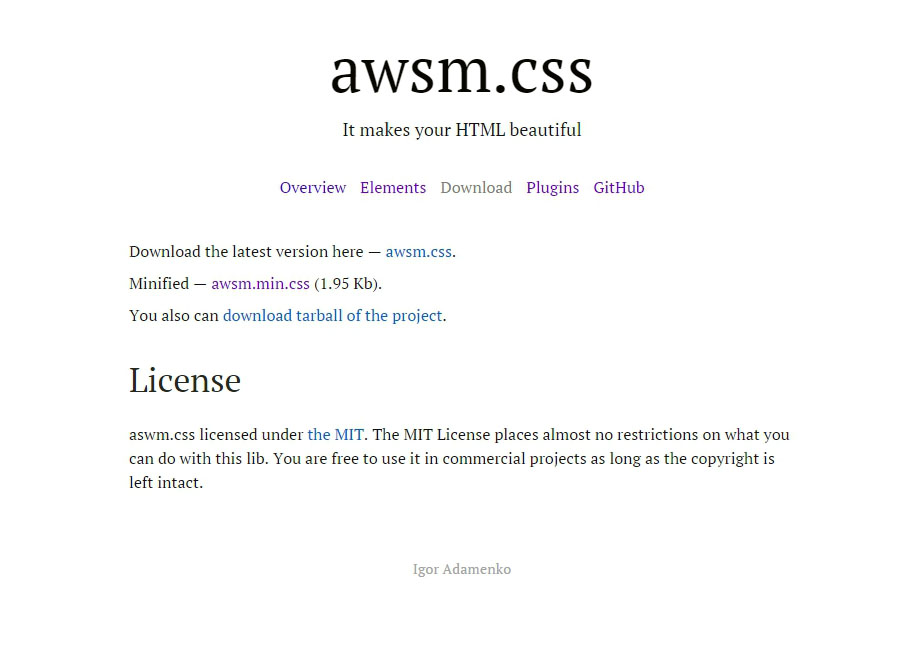


Рисунок 3. Download



Рисунок 4. Plugins

# Исходный код

Поскольку исходного кода много, то прикладываю только некоторые его части.

index.html:

1. <!DOCTYPE html**>**
2. **<html** lang="en"**>**
3. **<head>**
4. **<meta** charset="utf-8"**>**
5. **<meta** name="viewport" content="width=device-width, initial-scale=1.0"**>**
6. **<title>**awsm.css**</title>**
7. **<link** rel="stylesheet" href="css/awsm.min.css"**>**
8. **</head>**
9. **<body>**
11. **<header>**
13. **<h1>**awsm.css**</h1>**
14. **<p>**It makes your HTML beautiful**</p>**
16. **<nav>**
17. **<ul>**
18. **<li>**Overview**</li>**
19. **<li><a** href="elements.html"**>**Elements**</a></li>**
20. **<li><a** href="download.html"**>**Download**</a></li>**
21. **<li><a** href="plugins.html"**>**Plugins**</a></li>**
22. **<li><a** href="https://github.com/igoradamenko/awsm.css"**>**GitHub**</a></li>**
23. **</ul>**
24. **</nav>**
26. **</header>**
27. **<main>**
29. **<section>**
30. **<h2>**Usage**</h2>**
31. **<pre><code><link** rel="stylesheet" href="awsm.min.css"**></code></pre>**
32. **</section>**
34. **<section>**
35. **<h2>**Why?**</h2>**
36. **<p>**I have to create simple pages in my daily work. Sometimes it's compiled Markdown and sometimes it's pure HTML. I'd like to make them more beautiful without additional classes, ids, etc. So I did it.**</p>**
37. **</section>**
39. **</main>**
40. **<footer>**
42. **<address>**Igor Adamenko**</address>**
44. **</footer>**
46. **</body>**
47. **</html>**

css/awsm.css (частично):

1. html {
2. **color**: #111;
3. **font-family**: "PT Serif", Georgia, sans-serif;
4. **font-size**: 16px;
5. **line-height**: 1.5; }
7. body {
8. **margin**: 1em; }
10. body > header,
11. body > main,
12. body > footer,
13. body > article {
14. **display**: block;
15. **position**: relative;
16. **max-width**: 666px;
17. **margin**: 0 auto; }
19. body > header {
20. **margin-bottom**: 2.5em;
21. **text-align**: center; }
23. body > footer {
24. **margin-top**: 4em;
25. **color**: #A3A39B;
26. **font-size**: 0.9em;
27. **text-align**: center; }
29. a:hover,
30. a:focus, a:visited:hover {
31. **color**: #3000c1;
32. -webkit-transition: color 0ms;
33. transition: color 0ms; }
35. header p {
36. **font-size**: 18px; }
38. section, article {
39. **margin-bottom**: 4em; }
41. address {
42. **font-style**: normal; }
44. nav {
45. **margin**: 2em 0; }
47. nav ul {
48. **padding**: 0;
49. **margin**: 0;
50. **list-style**: none; }
52. nav ul li {
53. **display**: inline-block;
54. **margin**: 0 5px;
55. **color**: #7D7D77; }
57. aside {
58. **position**: absolute;
59. right: -200px;
60. **max-width**: 150px;
61. **padding-right**: 1em;
62. **color**: #7D7D77;
63. **font-style**: italic; }
65. @media (**max-width**: 1066px) {
66. aside {
67. **position**: relative;
68. right: auto;
69. **max-width**: none;
70. **padding**: 0; } }
72. a {
73. **color**: #0064C1;
74. **text-decoration**: none;
75. -webkit-transition: color 300ms;
76. transition: color 300ms; }
78. a:visited {
79. **color**: #6d00c1; }
81. input {
82. box-sizing: border-box;
83. **display**: block;
84. **width**: 100%;
85. **margin-bottom**: 20px;
86. **padding**: 0.3em 0.5em;
87. **border**: 1px solid #b5b5b5;
88. **outline**: none;
89. **font-size**: 1em;
90. **font-family**: inherit;
91. -webkit-transition: border 200ms;
92. transition: border 200ms; }
94. input:focus {
95. **border**: 1px solid dimgray; }
97. input[type=submit] {
98. **width**: auto;
99. **margin-bottom**: 0;
100. border-radius: 2px;
101. **background**: #E8E8E8;
102. **color**: #25291C;
103. **cursor**: pointer; }
105. input[type=submit]:focus, input[type=submit]:hover {
106. **border**: 1px solid dimgray;
107. **background**: #c2c2c2;
108. **color**: #070600; }