

## Лабораторная работа № 5

### Каскадные таблицы стилей. Форматирование текста с помощью CSS

**Цель работы** – приобрести начальные навыки создания простейших Internet-документов и форматирования созданных Web-страниц с помощью каскадных таблиц стилей.

#### 1. Основные понятия

*Каскадные таблицы стилей* применяются для оформления Web-страниц. Таблица стилей содержит набор правил (*стилей*), описывающих оформление самой Web-страницы и отдельных ее фрагментов.

Назначить определенный стиль для какой-либо части Web-документа можно тремя способами.

*1 Встроенный стиль.* Встроенные стили применяются к конкретному элементу.

Синтаксис: `<тег style="свойство: значение"> текст </тег>`.

*2 Внедренный стиль.* Внедренные стили позволяют управлять отображением той страницы, где они определены. Они используют тег `<style>`, который размещается в заголовке HTML-документа.

Синтаксис:

```
<head>
  <style type="text/css">
    селектор1 {свойство: значение}
    селектор2 {свойство: значение}
    ...
  </style>
</head>
```

Данный стиль будет применяться для фрагментов документа, выделенных с помощью отмеченных селекторов.

*Селектором* называется имя стилей, в котором указаны параметры форматирования. Селекторы делятся на несколько типов: селекторы тегов, классы и идентификаторы.

*Свойство* сообщает Web-обозревателю о том, что изменить, а *значение* позволяет ему узнать, каким должно быть это изменение.

В качестве селектора может использоваться любой из тегов языка HTML, если к нему возможно употребить какое-нибудь свойство.

Селектор может состоять из нескольких тегов (контекстный селектор). В этом случае значение указанного свойства будет применено только к тому фрагменту документа, который помечен сразу всеми участвующими в декларировании тегами с соблюдением указанной последовательности вложения тегов.

*Классы* используются для создания стилей, которые можно применять к любому тегу HTML, для выделений или изменения стиля блока текста; они обозначаются точкой перед именем. Обращение к селектору класса происходит через параметр `class`, значением которого выступает имя класса без точки.

Синтаксис: `селектор.класс {свойство: значение}`.

Можно также описать класс без явного указания селектора.

*Синтаксис:* .класс {свойство: значение}.

*Синтаксис* связывания какого-либо тега с объявленным классом:

<тег class="класс"> текст </тег>.

*Идентификаторы* похожи на классы, за исключением того, что они появляются в документе только один раз. Для определения того, что это идентификатор, к имени добавляется символ #. Стиль любого тега можно связать с идентификатором, если к тегу добавить параметр id с указанием имени идентификатора без символа решетки.

Селектор id может появляться много раз в документе CSS, но элемент, на который ссылается селектор id, встречается в HTML-документе единожды.

*3 Присоединенный стиль.* В этом случае описание стилей выносится во внешний файл.

*Синтаксис:*

```
<head>
  <link rel="stylesheet" type="text/css"
    href="file.css">
</head>
```

Для элемента link используется несколько атрибутов:

- rel - атрибут связи, который описывает связь ссылки. В данном случае, связь осуществляется с основной таблицей стилей, поэтому используется значение stylesheet;
- type - как и для элемента style внедренного стиля, необходимо определить тип языка и используемый формат - в данном случае, text/css;
- href - атрибут ссылки. В данном случае, было указано только имя файла но файл может располагаться на другом компьютере в сети. В этом случае атрибут href должен содержать ссылку на URL нужного компьютера;
- media - позволяет определять различные стили для различных устройств вывода. Если создать отдельную таблицу стилей для данного документа, предназначенную для вывода информации на портативных устройствах, в элементе link использовалась бы следующая информация: media="handheld". Подобным образом, атрибут media= "print" предполагает использование таблицы стилей только для печати. По умолчанию, значением атрибута media является all.

Подключить внешний CSS-файл можно также с помощью правила @import. Синтаксис:

```
@import url(<URL-адрес>)[ <Тип устройства>];
@import <URL-адрес>[ <Тип устройства>];
```

Правило @import должно быть расположено внутри тега <style>:

```
<style type="text/css">
  @import url("style.css");
</style>
```

Представленные методы назначения стилей описывают иерархию стилевых решений. Существует следующий приоритет стилей:

- стиль, заданный таблицей стилей, будет отменен, если в HTML-коде явно описано форматирование блока;

- стиль, заданный в теге <style>, будет отменен, если в параметре style тега указан другой стиль;
- стиль, заданный в отдельном файле, будет отменен, если в теге <style> указано другое определение стиля.

Именно из-за такой структуры приоритетов таблицы стилей называют *каскадными*.

Кроме того, следует учитывать, что стиль, заданный через идентификатор, будет иметь более высокий приоритет, чем стиль, заданный через класс.

С помощью свойства !important можно изменить приоритет. Синтаксис: {свойство: значение !important}.

## 2. Основные понятия форматирования

Каскадные таблицы стилей позволяют задать название, цвет и размер шрифта, его стиль и "жирность". Атрибуты форматирования можно задавать как для всего текста целиком, так и для отдельных фрагментов текста.

Изменение начертания шрифта и его размера происходит через свойства CSS, которые описаны в табл. 2.1.

Таблица 2.1 – Свойства шрифта

Свойство	Значение	Описание	Пример
font-family	Имя шрифта	Задаёт список шрифтов	p {font-family: arial, sans-serif}
font-style	normal italic oblique	Нормальный шрифт Курсив Наклонный шрифт	p {font-style: italic}
font-variant	normal small-caps	Трансформация букв шрифта в уменьшенные заглавные буквы	p {font-variant: small-caps}
font-weight	normal lighter bold bolder 100-900	Нормальная жирность Светлое начертание Полужирный Жирный 100-светлый шрифт, 900-самый жирный	p {font-weight: bold}
font-size	normal px mm cm in= 2.54 cm pt=1/72 in	нормальный размер пиксели миллиметры сантиметры дюймы пункты	font-size: normal font-size: 14pt font-size: 14px

Размеры в CSS можно также задавать в относительных единицах; процент (%); высота текущего шрифта (em); высота буквы "x" текущего шрифта (ex).

Кроме изменения параметров шрифтов, можно управлять и свойствами всего текста. Значения свойств приведены в таблице 2.2.

Таблица 2.2 – Свойства текста

Свойство	Значение	Описание	Пример
line-height	normal множитель точно %	Межстрочный интервал	line-height: normal line-height: 1.5 line-height: 12px line-height: 120%
text-decoration	none underline overline line-through blink	Убрать все оформление Подчеркивание Линия над текстом Перечеркивание Мигание текста	text-decoration: none
text-transform	none capitalize uppercase lowercase	Убрать все эффекты Начинать С Прописных ВСЕ ПРОПИСНЫЕ все строчные	text-transform: capitalize
text-align	left right center justify	Выравнивание текста	text-align: justify выравнивание по ширине
text-indent	точно %	Отступ первой строки	text-indent: 15px; text-indent: 10%

Гарнитура определяет художественное изображение шрифта, позволяющее различать его среди других шрифтов. *Синтаксис:*

`font-family: <список имен шрифтов, разделенных запятыми>.`

Если имя шрифта содержит пробелы, его необходимо взять в кавычки. Если данный атрибут стиля присутствует во встроенном стиле, кавычки заменяют апострофами: `<p style="font-family: 'Times New Roman'">.`

В ряде случаев шрифт может отсутствовать на компьютере пользователя. Поэтому лучше указывать несколько альтернативных шрифтов. Имена шрифтов указываются через запятую, заканчивая семейством шрифтов:

`p { font-family: "Verdana", "Tahoma", sans-serif}.`

Семейства шрифтов представляют собой наборы аналогичных шрифтов. Таких семейств пять: `serif` (шрифты с засечками); `sans-serif` (шрифты без засечек); `cursive` (шрифты, имитирующие рукописный текст); `fantasy` (декоративные шрифты); `monospace` (моноширинные шрифты).

На практике отдельно изменяют ширину символов и их наклон. Синтаксис: `<span style="font-weight: ширина_символов"> текст </span>` или `<span style="font-style: наклон"> текст </span>.`

Предусмотрено три метода указания размера шрифта:

- абсолютный размер: задается один из семи базовых размеров (`xx-large`, `x-large`, `large`, `medium`, `small`, `x-small`, `xx-small`);
- относительный размер: задается либо один из двух параметров – `larger` (на шаг больше текущего базового размера) и `smaller` (на шаг меньше), либо относительная величина – `em`, `ex`, `%`;

- явное указание размера: задается в дюймах, сантиметрах, миллиметрах, пунктах, пиках.

*Синтаксис:* `<span style="font-size: размер"> текст </span>`.

На практике отдельно рассматривают написание текста прописными буквами, но меньшего, чем предусмотрено для данного шрифта в нормальном значении размера, оформление текста, смещение текста (надстрочный и подстрочный). *Синтаксис:*

`<span style="font-variant: значение"> текст </span>`

или

`<span style="text-decoration: значение"> текст </span>`

или

смещение текста `<sup>` вверх `</sup>` и `<sub>` вниз `</sub>`

*Синтаксис* изменения расстояние между буквами и словами следующий:

`<span style="letter-spacing: значение"> текст </span>`

или

`<span style="word-spacing: значение"> текст </span>`.

Значение атрибута может быть задано параметром `normal` (в этом случае расстояние задается браузером, исходя из размера шрифта и выравнивания текста) или выражено конкретной величиной (в сантиметрах, дюймах, пунктах и др.).

*Синтаксис* изменения цвета текста следующий:

`<span style="color: цвет"> текст </span>`.

CSS имеет несколько опций для определения цвета текста и фоновых областей на Web-странице. В таблице 2.3 перечислены свойства элементов, предназначенных для задания цвета.

Таблица 2.3 – Опции для определения цвета текста и фоновых областей

Свойство	Значение	Описание	Пример
color	цвет	Устанавливает цвет текста	<code>p {color:#330000}</code>
background-color	цвет transparent	Цвет фона	<code>body {background-color:#6699ff}</code>
background-image	url none	Фоновый рисунок	<code>body {background-image:url(bg.gif)}</code>
background-repeat (Повторяемость фонов рисунка)	repeat repeat-x repeat-y no-repeat	Заполняется все доступное пространство Повтор по горизонтали Повтор по вертикали Выводится один раз	<code>body {background-image: url (bg.gif) background-repeat: repeat-y}</code>
background-attachment	scroll fixed	Прокручиваемость фона вместе с документом	<code>body {background-image: url(bg.gif) background-attachment: fixed}</code>
background-position	top, bottom left, right	Начальное положение фонов рисунка	<code>body {background-position: left top}</code>

Фон документа, можно задавать двумя способами:

- цвет фона. *Синтаксис*: `<body style="background: цвет">`;
- фоновое изображение. *Синтаксис*:

`<body style="background: url('файл') повтор фиксация х-позиция у-позиция">`.

Атрибут url представляет собой сетевой адрес графического файла.

Значением по умолчанию для цвета фона является transparent, который устанавливает прозрачный фон. Для установки фонового рисунка используется абсолютный или относительный адрес к файлу.

Как правило, фоновое изображение повторяется на странице столько раз, сколько необходимо для покрытия всего свободного пространства. Существует возможность управлять направлением повтора, используя свойство background-repeat или вообще запрещать его посредством команды повтор.

Режим фиксации (команда fixed) запрещает перемещение фонового изображения при прокрутке содержимого экрана браузера.

Если изображение не повторяется, то представляется возможным указать его положение на странице с указанием координат или методов позиционирования. Положение фона определяется параметром background-position.

Горизонтальные и вертикальные отступы от края браузера до содержимого Web-страницы встроены в браузер по умолчанию. Отступы задаются параметрами margin и padding. *Синтаксис*:

`<p style="text-indent: значение"> текст </p>`.

Значение атрибута может быть задано абсолютной или относительной величиной.

Управлять отдельными отступами от разных краев экрана можно с помощью параметров margin-top, margin-bottom, margin-right и margin-left, которые соответственно изменяют расстояние от верхнего, нижнего, правого и левого края окна браузера (рис. 2.1). Прежде чем их использовать, следует задать нулевое значение параметрам margin и padding.

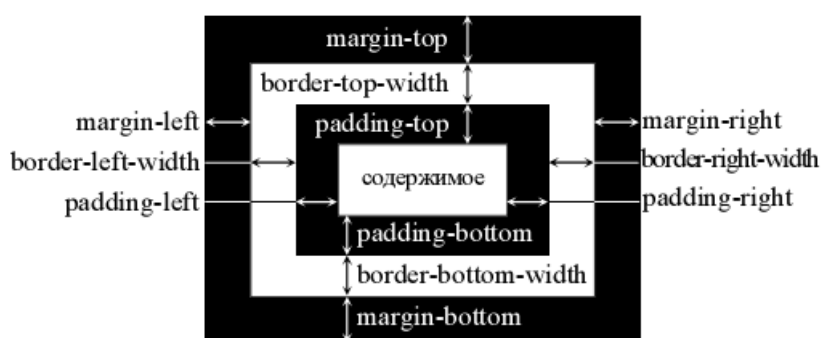


Рисунок 2.1 – Обозначение отступов от разных краев экрана

Селектор body, к которому применяется стиль, задает отступы на всей Web-странице целиком. То же можно сделать и применительно к отдельным элементам, например заголовкам.

Для сокращения кода можно использовать универсальный параметр margin.

*Обрамление* (border) – это формирование границы вокруг текстового блока. Граница формируется с помощью трех параметров: толщина (width), стиль (style) и цвет (color). Объединяет эти три параметра свойство border:

border-color:<цвет>; border-width:<значение>; border-style:<стиль>.

Допустимые значения для border-width: thin – тонкая граница; medium – граница средней толщины; thick – толстая граница.

Допустимые значения для border-style: none – нет границы (принято по умолчанию); solid – обычная граница; double – делает границу двойной; inset – создает эффект вдавленного текста; outset – эффект выпуклости; groove – это как будто сначала сделали outset, а потом – inset; ridge – то же, что и groove, только все цвета в границе наоборот.

Можно для каждой стороны границы задать разную толщину. Необходимо вставить вместо свойства border-width соответствующие конструкции из схемы выше. Каждое из свойств задает толщину границы для соответствующей стороны. Если для какой-то из сторон толщина не задана, то она будет по умолчанию равна нулю.

Для создания линий используется свойство CSS border, которое устанавливает рамку вокруг блока. В частных случаях, для создания линии лишь с одной стороны элемента, используются параметры border-bottom, border-top, border-left и border-right, которые соответственно задают линию внизу, сверху, слева и справа.

Значение этих атрибутов перечисляются через пробел и сразу устанавливают тип линии, ее толщину и цвет. Стиль линии может принимать одно из восьми значений, как показано на рис. 2.2, а толщина задается, как правило, в пикселях.

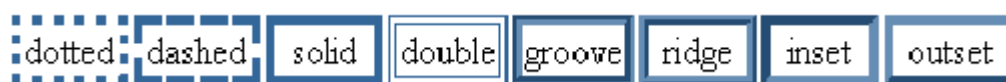


Рисунок 2.2 – Типы линий

Точно также линии можно создавать для любых блочных элементов (теги <table>, <p>, <div>), а также для встроенных элементов (тег <span>).

### 3. Типовое задание

Загрузите в Вашу папку файлы FotoHtml.jpg.

Реализуйте и проанализируйте решение практического задания:

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
<title> Форматирование с использованием CSS</title>
```

```
<meta charset="windows - 1251">
```

```
<style type="text/css">
```

```
body {
```

```
    background-color: purple; /* Цвет фона */
```

```
    background-attachment: fixed; /* фиксируем фон */
```

```
    background-image: url('FotoHtml.jpg'); /* Путь к фоновому рисунку */
```

```
    background-repeat: no-repeat; /* Отменяем повторение фона */
```

```
    background-position: right bottom /* Положение фона */
```

```
}
```

```
h1 {
```

```
    text-align: center;
```

```
    margin-top: 0px; /* Отступ сверху */
```

```
    margin-bottom: 1em; /* Отступ снизу */
```

```
    color: lime;
```

```
    font-family: Arial, Helvetica, Verdana, Sans-Serif;
```

```
    font-size: 150%;
```

```
    font-weight: lighter }
```

```
h3 {font-family: 'courier new'; font-size: 100%; font-weight: lighter }
```

```
p {
```

```
    color: yellow;
```

```
    margin: 0.5em 0px 1em /* Отступ сверху, справа-слева и снизу */
```

```
}
```

```
p.line {
```

```
    border-left: solid 5px black; /* Линия слева от текста */
```

```
    margin-left: 20px; /* Отступ слева от края до текста */
```

```
    padding-left: 7px /* Расстояние от линии до текста */
```

```
}
```

```
hr {
```

```
    color: red; /* Цвет линии для */
```

```
    height: 3px; /* Высота линии */
```

```
    width: 800px; /* Ширина линии */
```

```
    text-align: center /* Выравнивание по центру */
```

```
}
```

```
.line_b { border-bottom: 3px groove blue }
```

```
.underline {
```

```
    border-bottom: 1px dashed blue /* Добавляем пунктирную линию под текстом */
```



```

}
#cursive {font-style: italic} /* Курсивный текст */
#oblique {font-style: oblique} /* Наклонный текст */
</style>
</head>
<body>
<h1> Это заголовок первого уровня </h1>
<hr>
<h3> Это заголовок третьего уровня, далее идут элементы
форматирования </h3>
<hr>
<p> <span class="line_b"> Горизонтальная линия </span> внизу текста.
</p>
<p class="line"> Вертикальная линия слева от текста. </p>
<p style="text-indent: 20mm"> Отступ первой строки в 20 мм. </p>
<p><span style="letter-spacing: 10mm"> Установка расстояния в 10
миллиметров между символами текста. </span></p>
<p><span style="word-spacing: 20mm"> Установка расстояния в 20
миллиметров между словами. </span></p>
<p><span style="text-transform: capitalize"> Каждое слово начинается с
прописной буквы </span></p>
<p><span style="font-variant: small-caps"> написание текста малыми
прописными буквами </span></p>
<p><span style="text-decoration: underline"> Подчеркнутый текст
</span></p>
<p><span style="font-size: 20pt"> Задаем размер текста в 20 pt</span></p>
<p>Выделение слов <span style="font-weight: bold; font-style: italic">
наклонным полужирным </span> шрифтом</p>
<p><span style="font-family: 'Lucida Console', Monospace"> Гарнитура
шрифта </span></p>
<p>Часть текста <span style="color: red"> красного </span> цвета</p>
<span class="underline">Динамический HTML</span> - способ управления
свойствами объектов на Web-странице, такие как, графика, текст, элементы
форм и другое, без необходимости обновления всей страницы.
<hr>
<p id=cursive><span style="font-style: normal"> Курсивный текст </span>
хорошо использовать для цитат и прямой речи </p>
<p id=oblique>Наклонный шрифт пригодится для создания ненавязчивого
выделения в тексте </p>
<hr>
</body>
</html>

```

#### 4. Контрольные задания

На основе изученного материала, самостоятельно создайте свою Web-страницу, отображающую следующие пункты:

1. Выберите любую тему для своей Web-страницы, которая будет отображаться в заголовке. Оформите фон сайта, используя фоновый рисунок, который прокручивается вместе с документом.

2. Выделите в окне браузера произвольный текст по теме без форматирования. Задайте цвет шрифта, используя встроенный стиль в соответствии с вариантом:

№ варианта	Цвет
1	Lime
2	Purple
3	Navy
4	Fuchsia
5	Maroon
6	Teal
7	Olive
8	Aqua
9	Silver
10	Yellow
11	Red

3. Используя идентификатор, задайте курсивное начертание для следующего абзаца текста.

4. Выведите в окне браузера заголовок h1 без форматирования. Используя селекторы, задайте для заголовка следующие параметры форматирования.

№ варианта	Цвет	Размер шрифта
1	Green	20 pt
2	Red	12 mm
3	Teal	14 cm
4	Yellow	10 in
5	Olive	22 em
6	Blue	18 pt
7	Maroon	22 px
8	Fuchsia	24 mm
9	Aqua	11 em
10	Purple	10 cm
11	Yellow	12 px

5. Используя классы селекторов, задайте цвет текста для следующего абзаца в соответствии с вариантом.

№ варианта	Цвет
1	Maroon
2	Navy
3	Fuchsia
4	Pink
5	Silver
6	Olive
7	Purple
8	Teal
9	Lime
10	Blue
11	Green

6. В виде отдельного файла mystyle.css создайте таблицу стилей, в которой задайте контекстный селектор. Примените его к произвольному фрагменту документа, который помечен сразу всеми участвующими в декларировании тегами с соблюдением указанной последовательности вложения тегов.

7. Отформатируйте произвольный фрагмент текста по Вашей теме, состоящий из трех абзацев, в соответствии с вариантом:

№ варианта	1 абзац	2 абзац	3 абзац
1	Отступ первой строки в 15 мм, выравнивание текста по ширине, цвет - желтый	Мигание текста, шрифт размером 14 pt, цвет – синий, подчеркнутый	Выровнять по центру, расстояние между буквами 10 мм
2	Отступ первой строки на расстояние, равное 10% открытого окна браузера, шрифтом типа Courier	Перечеркнутый полужирный шрифт, размер - 12 pt, цвет - красный	Выровнять по правому краю, расстояние между словами 10 мм
3	Вертикальную линию справа от текста, задав ее тип, толщину и цвет	Все слова начинать с прописных букв, шрифт текста типа Arial, курсив, размер 18 pt	Выровнять по левому краю, расстояние между буквами 15 мм
4	Отступ первой строки на расстояние на 10% превышающее размер шрифта абзаца	Все буквы прописные, наклонный шрифт зеленого цвета размером 20 pt	Выровнять по ширине, расстояние между словами 20 мм
5	Рамка вокруг текстового блока, оформленная стилем, создающим эффект выпуклости	Шрифт размером 22 pt, красный цвет, полужирный	Выровнять по правому краю, расстояние между буквами 5 мм
6	Отступ сверху и снизу абзаца, выравнивание текста по центру, цвет - розовый	Шрифт текста типа Times New Roman, размер шрифта большого базового размера, нормальной жирности	Выровнять по центру, расстояние между словами 30 мм
7	Вертикальную линию слева от текста, задав ее тип, толщину и цвет	Мигание текста, Шрифт размером 22 pt, зеленый цвет, полужирный	Выровнять по ширине, расстояние между буквами 10 мм
8	Рамка вокруг текстового блока, оформленная стилем, создающим эффект вогнутости	Шрифт текста типа Garamond, размер шрифта меньшего базового размера, курсив	Выровнять по левому краю, расстояние между словами 15 мм
9	Перечеркнутый полужирный шрифт, размер - 10 pt, цвет - зеленый	Выровнять по правому краю, расстояние между словами 8 мм	Отступ первой строки на расстояние, равное 15% открытого окна браузера, шрифтом типа Courier
10	Выровнять по ширине, расстояние между буквами 6 мм	Вертикальную линию слева от текста, задав ее тип, толщину и цвет	Шрифт текста типа Garamond, размер шрифта меньшего базового размера, курсив, цвет - фиолетовый
11	Все буквы прописные, наклонный шрифт синего цвета размером 15 px	Рамка вокруг текстового блока, оформленная стилем, создающим эффект вогнутости	Вертикальную линию слева от текста, задав ее тип, толщину и цвет

