ПРАКТИЧЕСКАЯ РАБОТА №8

«Основы CSS3. Свойства. Цвет в CSS. Стилизация шрифтов»

**Цвет в CSS**

В CSS широкое распространение находит использование цветов. Чтобы установить цвет текста, фона или границы, нам надо указать цвет.

Например, определим красный цвет для фона элемента div:

|  |  |
| --- | --- |
| 1  2  3 | div{      background-color: red;  } |

В CSS есть несколько различных свойств, которые в качестве значения требует определенный цвет. Например, за установку цвета текста отвечает свойство **color**, за установку фона элемента - свойство **background-color**, а за установку цвета границы - **border-color**.

Существует несколько различных способов определения цвета текста.

* **Шестнадцатеричного значение**. Оно состоит из отдельных частей, которые кодируют в шестнадцатеричной системе значения для красного, зеленого и синего цветов.

Например, **#1C4463**. Здесь первые два символа 1C представляю значение красной компоненты цвета, далее 44 - значение зеленой компоненты цвета и 63 - значение уровня синего цвета. Финальный цвет, который мы видим на веб-странице, образуется с помощью смешивания этих значений.

Если каждое из трех двухзначных чисел содержит по два одинаковых символа, то их можно сократить до одного. Например,#5522AA можно сократить до #52A, или, к примеру, #eeeeee можно сократить до #eee. При этом не столь важно, в каком регистре будут символы.

* **Значение RGB**. Значение RGB также представляет последовательно набор значений для красного, зеленого и синего цветов (Red — красный, Green — зеленый, Blue — синий). Значение каждого цвета кодируется тремя числами, которые могут представлять либо процентные соотношения (0–100%), либо число от 0 до 255.

Например

|  |  |
| --- | --- |
| 1 | background-color: rgb(100%,100%,100%); |

Здесь каждый цвет имеет значение 100%. И в итоге при смешивании этих значений будет создаваться белый цвет. А при значениях в 0% будет генерироваться черный цвет:

|  |  |
| --- | --- |
| 1 | background-color: rgb(0%, 0%, 0%); |

Между 0 и 100% будут находиться все остальные оттенки.

Но, как правило, чаще применяются значения из диапазона от 0 до 255. Например,

|  |  |
| --- | --- |
| 1 | background-color: rgb(28, 68, 99); |

* **Значение RGBA**. Это тоже самое значение RGB плюс компонент прозрачности (Alpha). Компонент прозрачности имеет значение от 0 (полностью прозрачный) до 1 (не прозрачный). Например:

|  |  |
| --- | --- |
| 1 | background-color: rgba(28, 68, 99, .6); |

* **Значение HSL**. HSL представляет аббривиатуру: Hue — тон, Saturation — насыщенность и Lightness — осветленность. HSL задает три значения. Первое значение Hue угол в круге оттенков - значение в градусах от 0 до 360. Например, красный — 0 (или 360 при полном обороте круга). Каждый цвет занимает примерно 51°.

Второе значение - Saturation - представляет насыщенность, то указывает, насколько чистым является цвет. Насыщенность определяется в процентах от 0 (полное отсутствие насыщенности) до 100% (яркий, насыщенный цвет).

Третье значение - Lightness - определяет осветленность и указывается в процентах от 0 (полностью черный) до 100 (полностью белый). Для получения чистого цвет применяется значение 50 %.

Например:

|  |  |
| --- | --- |
| 1 | background-color: hsl(206, 56%, 25%); |

Данный цвет является эквивалентом значений #1C4463 и rgb(28, 68, 99)

* **Значение HSLA**. Аналогично RGBA здесь к HSL добавляется компонента прозрачности в виде значения от 0 (полностью прозрачный) до 1 (не прозрачный). Например:

|  |  |
| --- | --- |
| 1 | background-color: hsl(206, 56%, 25%, .6); |

* **Строковые значения**. Существует ряд константных строковых значений, например, red (для красного цвета) или green (для зеленого цвета). К примеру,

|  |  |
| --- | --- |
| 1 | color: red; |

* является эквивалентом

|  |  |
| --- | --- |
| 1 | color: #ff0000; |

* Полный перечень цветов можно найти на странице <https://developer.mozilla.org/en-US/docs/Web/CSS/color_value>

В заключение следует отметить, что существует множество бесплатных онлайн-генераторов цвета, где можно настроить и посмотреть цвет в нужном формате. Например, [генератор цвета на mdn](https://developer.mozilla.org/ru/docs/Web/CSS/CSS_Colors/Color_picker_tool).

**Прозрачность**

Ряд настроек цвета позволяют установить значение для альфа-компоненты, которая отвечает за прозрачность. Но также в CSS есть специальное свойство, которое позволяет установить прозрачность элементов - свойство **opacity**. В качестве значения оно принимает число от 0 (полностью прозрачный) до 1 (не прозрачный):

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | div{      width: 100px;      height: 100px;        background-color: red;      opacity: 0.4;  } |

**Стилизация шрифтов**

**Семейство шрифтов**

Свойство **font-family** устанавливает семейство шрифтов, которое будет использоваться. Например:

|  |  |
| --- | --- |
| 1  2  3 | body{      font-family: Arial;  } |

В данном случае устанавливается шрифт Arial.

Шрифт свойства font-family будет работать, только если у пользователя на локальном компьютере имеется такой же шрифт. По этой причине нередко выбираются стандартные шрифты, которые широко распространены, как Arial, Verdana и т.д.

Также нередко применяется практика нескольких шрифтов:

|  |  |
| --- | --- |
| 1  2  3 | body{      font-family: Arial, Verdana, Helvetica;  } |

В данном случае основным шрифтом является первый - Arial. Если он на компьютере пользователя не поддерживается, то выбирается второй и т.д.

Если название шрифта состоит из нескольких слов, например, Times New Roman, то все название заключается в кавычки:

|  |  |
| --- | --- |
| 1  2  3 | body{      font-family: "Times New Roman";  } |

Кроме конкретных стилей также могут использоваться общие универсальные шрифты, задаваемые с помощью значений **sans-serif** и**serif**:

|  |  |
| --- | --- |
| 1  2  3 | body{      font-family: Arial, Verdana, sans-serif;  } |

Так, если ни Arial, ни Verdana не поддерживаются на компьютере пользователя, то используется sans-serif - универсальный шрифт без засечек.

**Типы шрифтов**

**Шрифты с засечками**

Шрифты с засечками названы так, потому что на на концах основных штрихов имеют небольшие засечки. Считается, что они подходят для больших кусков текста, так как визуально связывают одну букву с другой, делая текст более читабельным.

Распространенные шрифты с засечками: Times, Times New Roman, Georgia, Garamond. Универсальный обобщенный шрифт с засечками представляет значение **serif**.

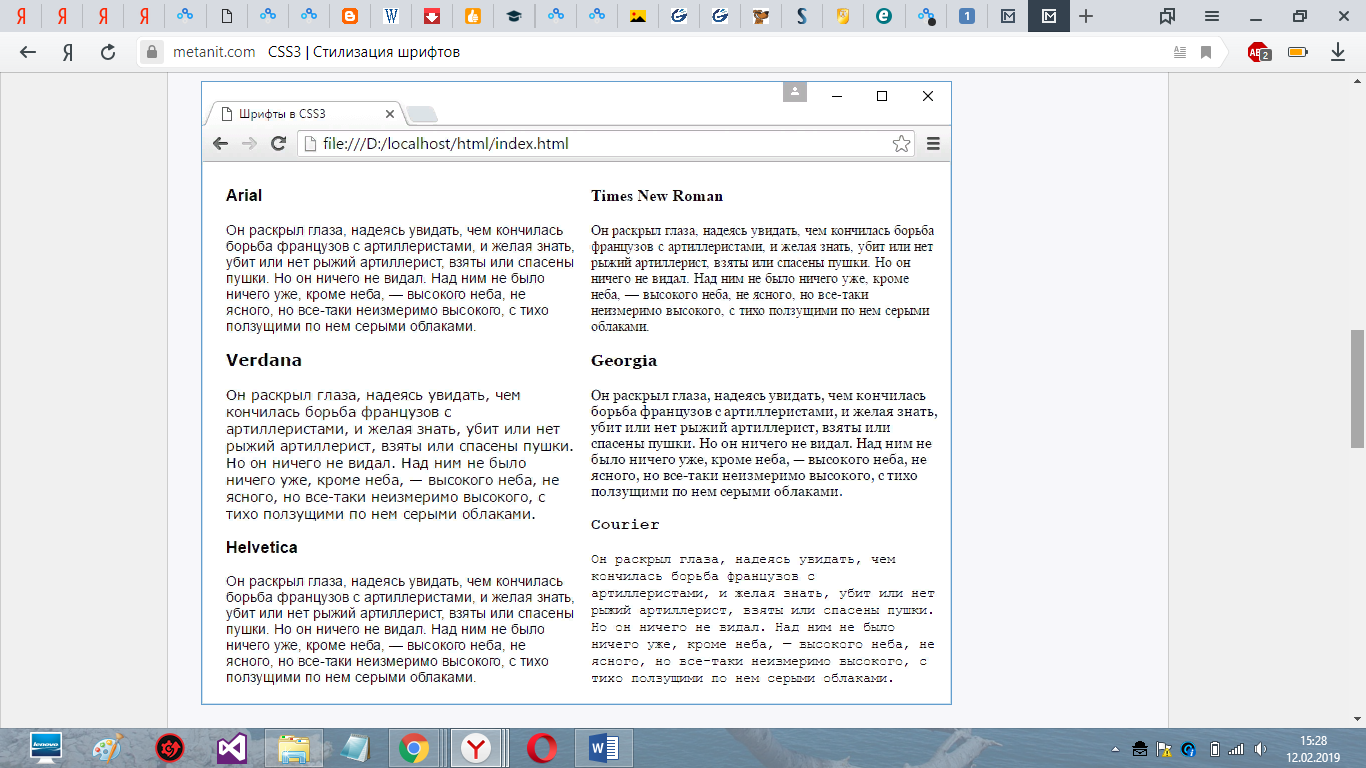
**Шрифты без засечек**

В отличие от шрифтов с засечками шрифты из этой группы не имеют засечек. Наиболее распространенные шрифты этой группы: Arial, Helvetica, Verdana.

**Моноширинные шрифты**

Моноширинный шрифт преимущественно применяется для отображения программного кода и не предназначен для вывода стандартного текста статей. Свое название эти шрифты получили от того, что каждая буква в таком шрифте имеет одинаковую ширину. Примеры подобных шрифтов: Courier, Courier New, Consolas, Lucida Console.

Примеры шрифтов:



**Толщина шрифта**

Свойство **font-weight** задает толщину шрифта. Оно может принимать 9 числовых значений: 100, 200, 300, 400,...900. 100 - очень тонкий шрифт, 900 - очень плотный шрифт.

В реальности чаще для этого свойства используют два значения: normal (нежирный обычный текст) и bold (полужирный шрифт):

|  |  |
| --- | --- |
| 1  2 | font-weight: normal;  font-weight: bold; |

**Курсив**

Свойство **font-style** позволяет выделить текст курсивом. Для этого используется значение italic:

|  |  |
| --- | --- |
| 1 | p {font-style: italic;} |

Если надо отменить курсив, то применяется значение normal:

|  |  |
| --- | --- |
| 1 | p {font-style: normal;} |

**Цвет шрифта**

Свойство **color** устанавливает цвет шрифта:

|  |  |
| --- | --- |
| 1  2  3 | p {      color: red;  } |