

# CSS

## iNTRO

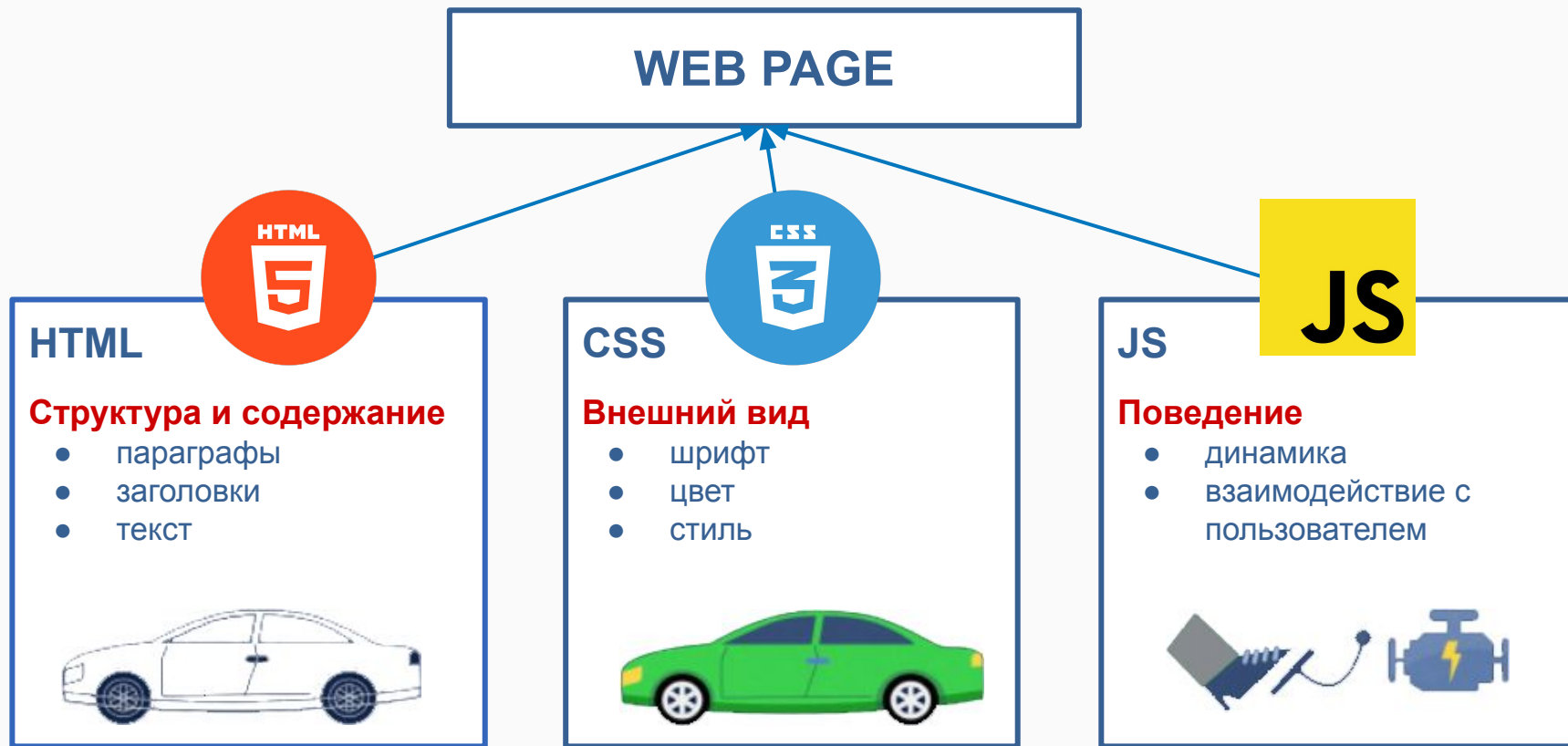


HTML



HTML + CSS

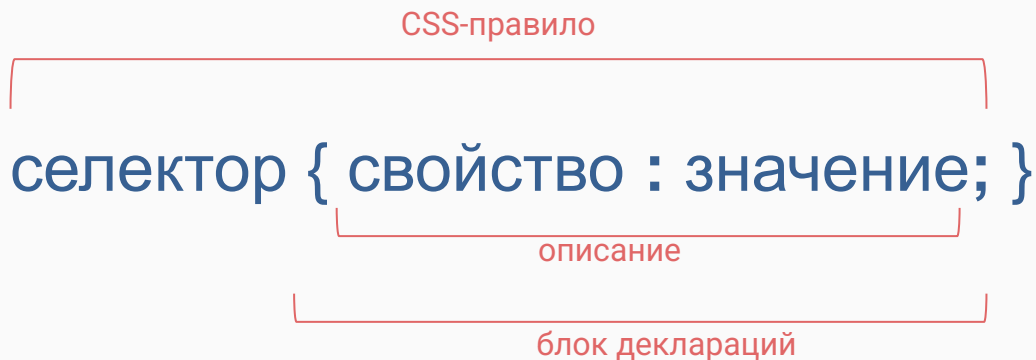
**Cascading Style Sheets (CSS)** —  
формальный язык описания  
внешнего вида документа,  
написанного с использованием  
языка разметки.



CSS отделяет код оформления страницы от ее структуры и содержания.

## CSS позволяет:

- определяет внешний вид элементов страницы (размер, цвет, оформление и т.д.)
- адаптировать внешний вид страницы к устройству отображения
- изменить оформление страницы не затрагивая ее структуру и содержание
- определить/изменить стиль централизованно, сразу для нескольких страниц



**Селектор (Selector)** - определяет набор элементов HTML на который распространяется данное правило

**Блок деклараций (Declarations block)** - содержит одно или несколько, разделенных точкой с запятой, *описаний*.

**Описание (Declaration)** - представляет собой пару `свойство : значение` разделенную двоеточием. В конце описания ставится точка с запятой.

**Свойство (Property)** - ключевое слово, которое определяет способ стилизации.

## CSS

```
p { /* свойство : значение */  
  font      : 18px sans-serif;  
  font-style : italic;  
  color     : #026afd;  
  text-align : center;  
}
```

селектор

блок  
деклараций

## HTML

```
<h1>hello world</h1>
```

```
<p>
```

Это параграф.<br>

Текст внутри параграфа будет отображен  
синим шрифтом, выровнен по центру и  
напечатан курсивом

```
</p>
```

# hello world

Это параграф.

Текст внутри параграфа будет отображен синим  
шрифтом, выровнен по центру и напечатан курсивом

- ВСТРОЕННЫЙ (INLINE) CSS
- ВНУТРЕННИЙ CSS
- ВНЕШНИЙ CSS



# Встроенный (inline) CSS

HTML

```
<body style="background-color:gray">  
  
  <h1 style="font-weight: bold; color: red;">hello world</h1>  
  
</body>
```

**hello world**

**Встроенные стили**, мы пишем CSS-код в HTML-файл, непосредственно внутри элемента с помощью атрибута style

# Внутренний CSS

HTML

```
<head>
```

```
  <style>
```

```
    body{
```

```
      background-color:gray;
```

```
    }
```

```
    h1{
```

```
      font-weight: bold;
```

```
      color: red;
```

```
    }
```

```
  </style>
```

```
</head>
```

```
<body>
```

```
  <h1>hello world</h1>
```

```
</body>
```

hello world

**Внутренние стили** встраиваются в раздел `<head></head>` HTML-документа и определяются внутри элемента `<style></style>`

# Синтаксис CSS

.\css\style.css

```
body{
  background-color:gray;
}
h1{
  font-weight: bold;
  color: red;
}
```

HTML

```
<head>
  <meta charset="UTF-8">
  <link rel="stylesheet" href=".\css\style03.css">
  <title>Document</title>
</head>
<body>
  <h1>hello world</h1>
</body>
</html>
```

Внешняя таблица стилей представляет собой текстовый файл с расширением .css

Внешняя таблица стилей подключается к веб-странице с помощью элемента `<link>`, расположенного внутри раздела `<head></head>`

`rel="stylesheet"` указывает тип ссылки (ссылка на таблицу стилей)

## ПРИОРИТЕТ CSS:

1. ВСТРОЕННЫЙ (INLINE) CSS
2. ВНУТРЕННИЙ CSS
3. ВНЕШНИЙ CSS
4. ВСТРОЕННЫЕ СТИЛИ БРАУЗЕРА

Если конфликтуют несколько правил, применяется то, у которого специфичность выше



*Чем точнее правило  
определяет элемент, тем  
выше его приоритет*

# CSS селекторы

Селектор CSS ( от англ. select - выбирать) — это специальное выражения, которые указывает браузеру (“выбирает”) элементы страницы, к которым должно быть применено правило CSS.

## Селектор по элементу:

CSS

```
p{  
  background-color:gray;  
}
```

HTML

```
<p>  
  hello  
</p>
```

Данный стиль будет применен ко всем элементам `<p>`

## Селектор по id:

CSS

```
#msg{  
    background-color:gray;  
}
```

HTML

```
<div id="msg">  
    hello  
</div>
```

Данный стиль будет применен к элементу страницы с id = "msg"



## Селектор по class:

CSS

```
.big{  
  background-color:gray;  
}
```

Данный стиль будет применен ко всем элементам страницы с атрибутом class = "big"

HTML

```
<p>нормальный текст</p>  
<p class = "big">большой текст</p>
```

нормальный текст

БОЛЬШОЙ ТЕКСТ

## Селектор “по нескольким class-ам”:

CSS

```
.big{ font-size: 2em;}  
.green{ color: green;}
```

Данный стиль будет применен ко всем элементам страницы с атрибутом class = “big”

HTML

```
<p>нормальный текст</p>  
<p class = "big">большой текст</p>  
<p class = "big green">большой зеленый текст</p>
```

нормальный текст

большой текст

большой зеленый текст

## Универсальный селектор \*:

```
* {  
    font: 1em sans-serif;  
}
```

Данный стиль будет применен ко всем элементам страницы

## Селекторы потомков:

Селекторы потомков применяют стили к элементам, расположенным внутри элемента-контейнера.

- **ul li**{ — выберет все элементы li, являющиеся потомками всех элементов ul
- **ul .time li**{ — выберет все элементы li, являющиеся потомками всех элементов ul с классом “time”
- **p.box a**{ — данный стиль применится ко всем ссылкам <a>, потомкам <p class=“box”>;
- **p .box a**{ — будут стилизованы ссылки, расположенные внутри любого элемента класса .box который является потомком элемента <p>
- **.box b**{ — к любому элементу <b>, расположенному внутри элемента, обозначенного классом box.

CSS: шрифт и текст

## font-family

тип шрифта

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

- основной шрифт (Tahoma), если он установлен у пользователя.
- запасной шрифт (Verdana), если основной недоступен
- если нет — используется любой системный семейства sans-serif (шрифт без засечек).

### СЕМЕЙСТВО ШРИФТОВ

- **Serif** Шрифты с засечками имеют небольшой штрих по краям каждой буквы. Они создают ощущение формальности и элегантности.
- **Sans-serif** Шрифты без засечек имеют чистые линии (без небольших штрихов). Они создают современный и минималистичный вид.
- **Monospace** Моноширинные шрифты — здесь все буквы имеют одинаковую фиксированную ширину. Они создают механический вид.
- **Cursive** Курсивные шрифты имитируют почерк человека.
- **Fantasy** Фантастические шрифты — это декоративные/игривые шрифты.

## font-size

размер шрифта

```
p{  
  font-size: 30px;  
}
```

### Единицы измерения

<b>px</b>	пиксели (фиксированная величина)	font-size: 16px;
<b>em</b>	относительная единица от родительского элемента	font-size: 1.2em;
<b>rem</b>	относительная единица от корня документа (html)	font-size: 1rem;
<b>%</b>	процент от размера родительского элемента	font-size: 120%;

## text-align

выравнивание текста

```
✓ p.left {  
  text-align: left;  
}  
  
✓ p.right {  
  text-align: right;  
}  
  
✓ p.center {  
  text-align: center;  
}  
  
✓ p.justify {  
  text-align: justify;  
}
```

<b>center</b>	Центрирует текст по горизонтали
<b>left</b>	Выравнивает текст по левому краю (по умолчанию)
<b>right</b>	Выравнивает текст по правому краю
<b>justify</b>	Выравнивает текст по ширине блока. Полезно для больших абзацев текста, например в статьях.
<b>start</b> <b>end</b>	<p>Выравнивание учитывает направление текста в блоке, заданное через свойство <code>direction</code>.</p> <p><b>direction: ltr</b> (left to right) — слева направо (по умолчанию, для английского, русского и т.д.) то:</p> <ul style="list-style-type: none"><li>• <code>text-align: start</code> — выравнивание по левому краю</li><li>• <code>text-align: end</code> — выравнивание по правому краю</li></ul> <p><b>direction: rtl</b> (right to left) — справа налево (например, для арабского или иврита), то наоборот</p>



## text-decoration оформление текста

```
p {  
  text-decoration: underline;  
  text-decoration-style: wavy;  
  text-decoration-color: crimson;  
}
```



hello html & css

```
p { text-decoration: underline wavy crimson;  
}
```

<code>text-decoration-line</code>	Задаёт, какие линии будут  <code>underline</code>   <code>overline</code>   <code>line-through</code>   <code>none</code> )
<code>text-decoration-style</code>	Стиль линии  <code>solid</code>   <code>dotted</code>   <code>dashed</code>   <code>double</code>   <code>wavy</code>
<code>text-decoration-color</code>	Цвет линии
<code>text-decoration-thickness</code>	Толщина линии (поддерживается не во всех браузерах)

Свойство `text-decoration` позволяет задать все четыре свойства оформления текста, при этом, обязательно нужно указать только тип оформления (`text-decoration-line`)

<b>font-weight</b>	Толщина шрифта (жирность)	<code>font-weight: bold;</code> <code>font-weight: 400;</code>
<b>font-style</b>	Наклон шрифта	<code>font-style: italic;</code>  Возможные значения: <code>oblique</code>   <code>normal</code>   <code>italic</code>
<b>font-variant</b>	Использование капитель (малых прописных букв)	<code>font-variant: small-caps;</code> <code>font-variant: normal;</code>
<b>line-height</b>	Межстрочный интервал	<code>line-height: 1.5;</code>
<b>letter-spacing</b>	Расстояние между буквами	<code>letter-spacing: 2px;</code>
<b>text-transform</b>	Преобразование регистра текста	<code>text-transform: none;</code>  Возможные значения: <code>uppercase</code>   <code>lowercase</code>   <code>capitalize</code>   <code>none</code>

<b>color</b>	Цвет текста	<code>color: #333;</code>
<b>text-align</b>	Выравнивание текста	<code>text-align: center;</code>  Возможные значения: <code>left   right   justify   start   end   center</code>
<b>text-decoration</b>	Подчеркивание, перечёркивание, линия над текстом. Позволяет задать не только тип оформления, но и цвет линии, тип линии, толщину	<code>text-decoration: underline;</code>  Возможные значения: <code>underline   overline   line-through   none</code>
<b>word-spacing</b>	Расстояние между словами	<code>word-spacing: 5px;</code>
<b>text-indent</b>	Отступ первой строки абзаца	<code>text-indent: 30px;</code>
<b>direction</b>	Направление текста (слева направо или справа налево)	<code>direction: rtl;</code> <code>direction: ltr;</code>

```
p {  
  color: #b8107a;  
  font-family: "Georgia", serif;  
  font-size: 18px;  
  font-weight: normal;  
  font-style: italic;  
  text-align: justify;  
  text-decoration: underline;  
  text-transform: capitalize;  
  letter-spacing: 1px;  
  word-spacing: 3px;  
  line-height: 1.6;  
  text-indent: 30px;  
  white-space: normal;  
  direction: ltr;  
}
```

```
p {  
  color: #b8107a;           /* цвет */  
  font-family: "Georgia", serif; /* название шрифта */  
  font-size: 18px;          /* размер */  
  font-weight: normal;      /* жирность нормальная */  
  font-style: italic;       /* курсив */  
  text-align: justify;      /* выравнивание по ширине */  
  text-decoration: underline; /* подчеркнутый */  
  text-transform: capitalize; /* заглавными буквами */  
  letter-spacing: 1px;      /* расстояние между буквами */  
  word-spacing: 3px;        /* расстояние между словами */  
  line-height: 1.6;         /* высота строки */  
  text-indent: 30px;        /* абзац 30px */  
  white-space: normal;      /* переносы */  
  direction: ltr;           /* направление слева направо */  
}
```

CSS: ссылки

```
/* тег <a> */
a {
    color: #009578;
    text-decoration: none;
}
/* Нормальная ссылка (ещё не посещенная) */
a:link {
    color: blue;
    text-decoration: none;
}
/* Посещённая ссылка */
a:visited { color: purple; }

/* стиль при наведение на ссылку */
a:hover {
    color: rgb(20, 220, 53);
    text-decoration: underline;
    font-size: 1.1em;
}
/* Активная ссылка (в момент нажатия) */
a:active { color: orange; }
```

- **a:link** — нормальная ссылка (ещё не посещенная)
- **a:visited** — посещенная ссылка
- **a:hover** — ссылка при наведении мыши
- **a:active** — активная ссылка (в момент нажатия)

Необходимо соблюдать порядок указания правил (LoVe HA):

1. link
2. visited
3. hover
4. active

# CSS: списки



<b>list-style-type</b>	Управляет видом маркеров или номеров в списке.	<code>font-style: italic;</code>  Возможные значения для <code>&lt;ul&gt;</code> : <code>disk</code>   <code>circle</code>   <code>square</code>   <code>none</code>  Возможные значения для <code>&lt;ol&gt;</code> : <code>decimal</code>   <code>lower-alpha</code>   <code>upper-roman</code>
<b>list-style-position</b>	Определяет, где находится маркер	<code>list-style-position: inside;</code>  Возможные значения: <code>outside</code>   <code>inside</code>
<b>list-style-image</b>	Позволяет использовать картинку вместо стандартного маркера	<pre>ul {   list-style-image:     url("bullet.png"); }</pre>