



Брестский государственный
технический университет



web технологии

A computer monitor with a silver frame and stand is centered in the image. The screen is black and displays the text 'web технологии' in a light gray, sans-serif font. The monitor is flanked by decorative horizontal bands of semi-transparent circles in blue and gray on the left, and pink, orange, and yellow on the right.

CSS3

Весна 2022



ФАКУЛЬТЕТ
ЭЛЕКТРОННО-
ИНФОРМАЦИОННЫХ
СИСТЕМ

План лекции

- CSS



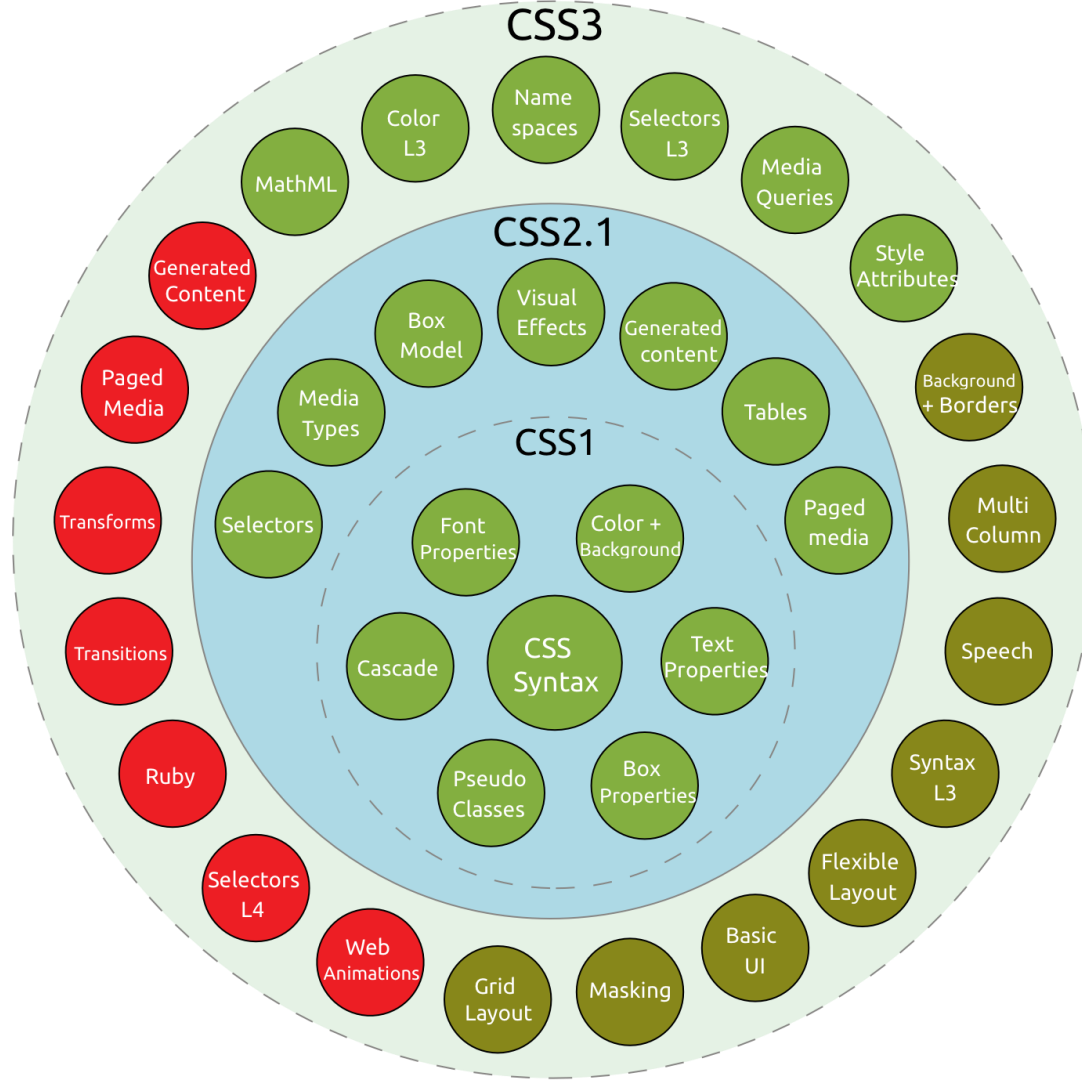
1991 год

В начале веб дизайны были основаны на таблицах, а стилизация переплеталась с содержимым.

Каскадные таблицы стилей (**Cascading Style Sheets** —

CSS) были созданы в качестве способа разделения дизайна и содержимого веб документа.

1991	HTML
1994	HTML 2
1996	CSS 1 + JavaScript
1997	HTML 4
1998	CSS 2
2000	XHTML 1
2002	Tableless Web Design
2005	AJAX
2009	HTML 5





Основные свойства

На 1 июня 2018 года CSS содержал **415** уникальных свойств, относящихся к объекту style в любом элементе браузера Chrome.

По состоянию на декабрь 2018 года уже насчитывалось **522** уникальных свойства.

Всего за семь месяцев в Chrome была добавлена поддержка более **100** новых свойств.

Этот процесс будет происходить постоянно, так как спецификация CSS продолжает развиваться.



Основные свойства

На 1 июня 2018 года CSS содержал **415** уникальных свойств, относящихся к объекту style в любом элементе браузера Chrome.

По состоянию на декабрь 2018 года уже насчитывалось **522** уникальных свойства.

Всего за семь месяцев в Chrome была добавлена поддержка более 100 новых свойств.

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, pink, and green, creating a vibrant, abstract pattern.

Основные свойства

На 14 апреля 2022 года количество селекторов составляло **588**.

Этот процесс будет происходить постоянно, так как спецификация CSS продолжает развиваться.

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, arranged in a gradient-like fashion from top to bottom.

Внешнее размещение

Код CSS можно сохранить в отдельном внешнем файле и включить с помощью HTML-элемента `link`

Исходный код файла `style.css`:

```
body p
```

```
{
```

```
background: white;
```

```
color: black;
```

```
font-family: Arial, sans-serif;
```

```
font-size: 16px;
```

```
line-height: 1.58;
```

```
text-rendering: optimizeLegibility;
```

```
-webkit-font-smoothing: antialiased;
```

```
}
```


Внешнее размещение

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6
7   <link rel="stylesheet" href="путь/до/файла.css">
8   _____
9 </head>
10 <body>
11
12 </body>
13 </html>
```

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, arranged in a gradient-like fashion from top to bottom.

Внешнее размещение

Пример ссылки на внешний CSS-файл:

```
<html>
```

```
  <head>
```

```
    <title>Добро пожаловать на сайт.</title>
```

```
    <link rel = "stylesheet"
```

```
      type = "text/css"
```

```
      href = "style.css" />
```

```
  </head>
```

```
  <body>
```

```
    <p>CSS-стили записаны в файле style.css
```

```
    и применяются к содержимому этой страницы.</p>
```

```
  </body>
```

```
</html>
```

Внутреннее размещение

```
<html>
<head>
  <style type = "text/css">
    body p
    {
      background: white;
      color: black;
      font-family: Arial, sans-serif;
      font-size: 16px;
      line-height: 1.58;
      text-rendering: optimizeLegibility;
      -webkit-font-smoothing: antialiased;
    }
  </style>
</head>
<body>
  <p>CSS-стили внутри элемента style
    и применяются к этому абзацу
    в HTML-коде веб-страницы</p>
</body>
</html>
```

Возможно ввести CSS-код
непосредственно в HTML-документ
между двумя тегам элемента style:

Строковое размещение

Строковое размещение CSS-кода с использованием атрибута `style` в элементе HTML:

```
<html>
```

```
  <head></head>
```

```
  <body style = "font-family: Arial;">
```

```
    <p>При выводе в браузере этот абзац наследует  
      форматирование шрифтом Arial из строкового  
      стиля в родительском элементе.</p>
```

```
  </body>
```

```
</html>
```

Строковое размещение

Приоритеты стилей

- Стили



- !important



- Порядок каскадирования:

- По источнику ("ближе" к тегу)
- Специфичность, более узкое, точное определение
- Порядок следования

- Как правило стили приоритетнее атрибутов



Селекторы

Грамматика языка CSS это правила синтаксиса для определения свойств и значений.

Наиболее распространенный селектор. — само имя HTML-элемента.

Использование имени тега приведет к выделению всех элементов данного типа.

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, arranged in a gradient from top to bottom.

Selector

p

{

color:

red;

}

Property

Property value

Declaration



selector

declaration

p

{

font-size:

1.2em;

}

property

value



Селекторы

Использование имени тега приведет к выделению всех элементов данного типа.

Возможно выбрать элемент `body` по его имени:

```
body { /* Сюда помещаются свойства CSS */ }
```

На первый взгляд, поскольку в HTML-документе есть только один элемент `body`, это единственное, что будет выбрано.

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, arranged in a gradient-like fashion from top to bottom.

Селекторы

Однако из-за каскадной специфики CSS любое свойство, которое мы берем в скобки, также будет применяться ко всем его потомкам (дочерним элементам, содержащимся в элементе `body`, даже если мы не станем явно указывать их стиль).

Это пустой селектор.

Он выбирает элемент `body`, но пока не назначает ему никаких свойств.

A vertical decorative bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, creating a bokeh-like effect.

Селекторы

Примеры выбора объектов по имени их HTML-элемента.
Это самые распространенные приемы.

/* Выбираем все элементы абзаца, p */

p { }

/* Выбираем все элементы div */

div { }

**/* Выбираем все элементы p, только если они находятся
в элементах div */**

div p { }

Ваши стили CSS будут заключены в {....эти....} скобки.



Селекторы

Инструкция CSS состоит из селектора и пары

свойство: значение;

Несколько свойств должны быть разделены точкой с запятой.

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, arranged in a gradient-like fashion from top to bottom.

Селекторы

Рассмотрим одно свойство, просто чтобы посмотреть, как выглядит синтаксис CSS-свойства:

`<div id = "box">контент<div>`

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, creating a vibrant, abstract pattern.

Селекторы

В CSS идентификатор указывается в виде символа хештега #:

#box { свойство: значение; }

Необходимо использовать идентификатор (id) для маркировки элементов всякий раз при наличии уникального контейнера.

```
<!doctype html>
```

```
<html lang="ru">
```

```
  ▶ <head>...</head>
```

```
  ▼ <body>
```

ПРЕДКИ

РОДИТЕЛЬ

```
  ▶ <header class="container">
```

СОСЕД

```
    <h1>Заголовок страницы</h1>
```

```
    ... ▼ <div class="section"> == $0
```

```
      ▼ <nav class="navbar">
```

```
        <li>Пункт списка 1</li>
```

```
        <li>Пункт списка 1</li>
```

```
        <li>Пункт списка 1</li>
```

```
      </nav>
```

```
      
```

```
    </div>
```

```
  </header>
```

```
  ▶ <main class="content">...</main>
```

```
    <footer>...</footer>
```

```
  </body>
```

```
</html>
```

ДЕТИ

ПОТОМКИ



Селекторы

Не нужно форматировать каждый отдельный HTML-элемент с помощью идентификаторов, необходимо использовать их для именования глобальных родительских элементов или для более значимых элементов (например, тех, которые необходимо часто обновлять из-за изменения содержимого).

Селекторы

Как поступить если необходимо выбрать несколько элементов одновременно?

``

`<li class = "элемент">1`

`<li class = "элемент">2`

`<li class = "элемент">3`

``

Аналогично атрибут класса (class) обозначается селектором точки(.):

`.item { line-height: 1.80; }`

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, creating a vibrant, abstract pattern.

Селекторы

В примере описанном выше точка используется для выбора нескольких элементов, имеющих одно и то же имя класса.

Так свойству `line-height` (высота строки) присваивается значение `1.50` (что примерно соответствует `150%` высоты шрифта).

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, creating a vibrant, abstract pattern.

Селекторы

Специальные правила CSS: селектор `:root` применяет их ко всем HTML-элементам.

Возможно использовать `:root`, чтобы установить значения CSS по умолчанию для всего документа.

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, arranged in a gradient-like fashion from top to bottom.

Селекторы

В качестве ещё одного примера установим Arial в качестве шрифта по умолчанию для всего документа или sans-serif, если шрифт Arial недоступен;

возможно указать столько шрифтов, сколько необходимо:

```
:root { font-family: Arial, sans-serif; }
```

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, arranged in a gradient-like fashion from top to bottom.

Селекторы

Селектор `:root` также часто используется для глобального хранения CSS-переменных.

Создадим CSS-переменную с именем **`--red-color`** и присвоим ей значение цвета `red`:

```
:root { --red-color: red; }
```

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, arranged in a gradient-like fashion from top to bottom.

Селекторы

Необходимо запомнить, что все имена переменных CSS должны начинаться с двойного дефиса —

Теперь возможно использовать CSS-переменную `--red-color` в качестве значения в стандартных селекторах CSS:

```
div { color: var(--red-color) ; }
```

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, creating a vibrant, abstract pattern.

Селекторы

Селектор звездочка (*) выполняет те же функции.

Можно использовать селектор * для достижения эффекта, вызываемого применением **:root**.

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, creating a vibrant, abstract pattern.

Селекторы

Единственное отличие состоит в том, что селектор `*` нацелен абсолютно на все элементы в документе, а `:.root` — только на контейнер документа без его дочерних элементов:

`* { font-family: Arial, sans-serif; }`

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, creating a vibrant, abstract pattern.

Селекторы

Несмотря на то что добавление селектора * дает тот же эффект, менее целесообразно использовать его для применения стилей ко всему документу

(вместо этого задействуйте **:root**).

Лучше всего селектор * подходит для пакетного выбора «всех элементов» в пределах определенного родительского элемента.

Селекторы

Селектор `#parent *` может использоваться для выбора всех потомков родительского элемента независимо от их типа:

```
<div id = "parent">  
  <div>A</div>  
  <div>B</div>  
  <ul>  
    <li>1</li>  
    <li>2</li>  
  </ul>  
  <p>Текст.</p>  
</div>
```

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, creating a vibrant, abstract pattern.

Селекторы

Если Вы продолжите экспериментировать с селекторами, вы заметите, что можно выбирать одни и те же HTML-элементы, используя разные комбинации селекторов.

Например, все следующие комбинации выбирают один и тот же набор элементов (все потомки родительского элемента, без учета самого предка).

Селекторы

Селектор `#parent *` может использоваться для выбора всех потомков родительского элемента независимо от их типа:

```
/* Выбираем все дочерние элементы #parent */
```

```
#parent * { color: blue; }
```

```
/* Объединяем несколько селекторов, используя запятую */
```

```
#parent div,
```

```
#parent ul,
```

```
#parent p { color: blue; }
```

```
/* Применяем псевдоселекторы :nth-child */
```

```
#parent nth-child(1),
```

```
#parent nth-child(2),
```

```
#parent nth-child(3),
```

```
#parent nth-child(4) { color: blue; }
```



Селекторы

При работе над реальным проектом и ввиду постоянного усложнения макета приложений вы не сможете даже предположить, как часто конкретное CSS свойство не будет работать нужным образом.

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, creating a vibrant, abstract pattern.

Селекторы

Когда пропущен определенный сценарий и допущена ошибка, разработчики часто используют ключевое слово **!Important**, чтобы быстро исправить проблему.

Вы можете переопределить любой стиль CSS, добавив ключевое слово **!Important** в конец кода CSS

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, pink, and green, creating a vibrant, abstract pattern.

Селекторы

```
/* Выбираем все дочерние элементы #parent */  
#parent * { color: blue; }
```

```
/* Выбираем только div в #parent и меняем цвет  
на красный */  
#parent div { color: red; }
```

```
/* Проверяем, что все div во всем документе  
зеленого цвета */  
div { color: green !important; }
```


Селекторы

Весьма заманчиво использовать ключевое слово **!important** для принудительной установки стиля.

Такая практика считается порочной, поскольку игнорируется каскадная логика таблиц стилей!





Не все CSS свойства одинаковы.

В зависимости от типа свойства значение может быть мерой пространства, заданного в пикселах, единицах pt, em или fr, цветом, указанным в виде имени (red, blue, black и т.д.), шестнадцатеричного значения (#0F0 или #00FF00) или rgb (r, g, b).



Взаимосвязь между свойствами и значениями

Значение уникально для конкретного свойства, и его нельзя использовать с любым другим свойством.

Например, CSS-свойство `transform` может принимать значение, указанное с помощью ключевого слова `rotate`.


A vertical decorative bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, pink, and green, creating a vibrant, abstract pattern.

Взаимосвязь между свойствами и значениями

В данном случае принимается угол в градусах — CSS требует добавления букв deg к числовому значению градуса.

```
/* поворот этого элемента на 45 градусов по  
часовой стрелке */
```


```
#box {  
transform: rotate(45deg);  
}
```

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, arranged in a gradient-like fashion from top to bottom.

Взаимосвязь между свойствами и значениями

Однако это не единственный способ указать угол.

CSS предлагает еще **три типа** единиц, специально предназначенных для указания угла поворота: **grad**, **rad** и **turn**.

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, arranged in a gradient from top to bottom.

Взаимосвязь между свойствами и значениями

```
/* 200 градиан (град)*/
```


```
transform: rotate(200grad);
```

```
/* 1,4 радиан */
```

```
transform: rotate(1.4rad);
```


```
/* 0,5 оборота или 180 градусов (1 оборот =  
360 градусов) */
```

```
transform: rotate(0.5turn);
```



Взаимосвязь между свойствами и значениями

В данном случае были использованы **grad** (градусы), **rad** (радианы) и **turn** (повороты) в качестве альтернативного способа задания угла поворота HTML-элемента.



Взаимосвязь между свойствами и значениями

Альтернативные способы указания значений не редкость для многих других CSS-свойств.

Например, #F00, #FF0000, red, rgb(255, 255, 255) и rgba(255, 255, 255, 1.0) задают один и тот же цвет.

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, pink, and green, creating a vibrant, abstract pattern.

Комментарии в CSS-коде

Для создания комментариев в коде CSS поддерживается только синтаксис блочных комментариев.

Это делается путем вставки блока текста с использованием символов.

`/*комментарий*/`

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, arranged in a gradient from top to bottom.

Комментарии в CSS-коде

/* Установить белый цвет шрифта, используя **название цвета** */

```
p { color: white; }
```

/* Установить белый цвет шрифта, используя **шестнадцатеричное значение** */

```
p { color: #FFFFFF; }
```

/* Установить белый цвет шрифта, используя **сокращенное шестнадцатеричное значение** */

```
p { color: #FFF; }
```



Комментарии в CSS-коде

/* Установить белый цвет шрифта, используя значение RGB */

```
p { color: rgb(255,255,255); }
```

/* Создать переменную CSS --white-color (обратите внимание на двойной дефис) */

```
:root { --white-color: rgba(255, 255, 255, 1.0); }
```

/* Установить белый цвет шрифта, используя CSS-переменную */

```
p { color: var(--white-color); }
```

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, arranged in a gradient-like fashion from top to bottom.

Комментарии в CSS-коде

Также возможно закомментировать раздел CSS-кода целиком.

Далее показано временное отключение блока кода CSS для тестирования нового кода или будущей ссылки и.т.д.:

```
/* Временно отключить данный блок CSS
content: "hello";
border: 1px solid gray;
color: transparent;
line-height: 48px;
padding: 32px;
*/
```



Комментарии в CSS-коде

CSS не поддерживает `//` встроенные комментарии, или, скорее, они не оказывают влияния на CSS-интерпретатор браузера.



Оформление стилей

В CSS имеется множество свойств, связанных с габаритами и размерами (left, top, width, height и др.).

Было бы излишним перечислять их все. Поэтому далее в примерах будет использовано слово **свойство**.



Оформление стилей

Для указания значения служит шаблон

свойство: значение.

Такая комбинация позволит установить фоновые изображения, цвет и другие основные свойства HTML-элементов.



Оформление стилей

В качестве альтернативы можете использовать шаблон **свойство : значение.**

Значение для установки нескольких значений одному свойству.



Оформление стилей

Без сокращенной формы записи вы бы указали каждую часть свойства в отдельной строке.

```
/* Фон */
```

```
background-color: black;
```

```
background-image: url("image.jpg");
```

```
background-position: center;
```

```
background-repeat: no-repeat;
```



Оформление стилей

Это и есть сокращенная форма записи.
Значения разделяются пробелом.

/* Сокращенная форма записи, только одна
строка кода! */

```
background: black url("image.jpg") center no-  
repeat;
```

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, creating a vibrant, abstract pattern.

Оформление стилей

За все эти годы в CSS произошли значительные изменения.

Прежде чем изучать визуальные схемы, описывающие каждое свойство, необходимо понять, как CSS интерпретирует шаблоны свойств и значений.

Большинство свойств используют следующие шаблоны:



Оформление стилей

/* Самая распространенная форма */
свойство: значение;

/* Значения, разделенные пробелом */
свойство: значение значение;

/* Значения, разделенные запятой */
свойство: значение, значение, значение;

Оформление стилей

Свойства, связанные с размером, могут быть рассчитаны с помощью ключевого слова calc:

/* Вычисление */

свойство: calc(значение px);

/* Вычисление значений в % и px – ок. */

свойство: calc(значение% - значение px);



Оформление стилей

/* Вычисление значений в % и % – ок. */
свойство: calc(значение% - значение%);

/* Сложение px и px – ок. */
свойство: calc(значение px + значение px);

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, creating a vibrant, abstract pattern.

CSS-переменные

Вы можете задействовать CSS переменные, чтобы избежать многократного определения одних и тех же значений в различных CSS-селекторах.

Имена переменных CSS всегда начинаются с двух дефисов.

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, creating a vibrant, abstract pattern.

CSS-переменные

Определить CSS-переменную в глобальной области видимости можно с помощью селектора `:root`.

Здесь элемент используется только в качестве заполнителя, в реальной же работе он будет заменен допустимым именем HTML-элемента:

The new way

```
:root {  
  --red: #ff6f69;  
}
```

Declaring a variable



```
#title {  
  color: var(--red);  
}
```

Referencing it

```
.quote {  
  color: var(--red);  
}
```

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, arranged in a gradient-like fashion from top to bottom.

CSS-переменные

/* Определяем переменную --default-color */

```
:root { --default-color: yellow; }
```

/* Определяем переменную --variable-name */

```
:root { --variable-name: 100px; }
```

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, creating a vibrant, abstract pattern.

CSS-переменные

```
/* Устанавливаем значение переменной цвета  
фона --default-color */
```

```
элемент { background-color: var(--default-color);  
}
```

```
/* Устанавливаем ширину 100px */
```

```
элемент { width: var(--variable-name); }
```





Локальные переменные

Вы можете создавать локальные переменные, содержащиеся только в определенном родительском элементе.

Таким образом, они не проникают.

В глобальную область видимости и не получают другие определения переменных, потенциально объявленных с тем же именем.

Локальные переменные

```
:root {  
  --color:  black;  
}  
  
.block {  
  color:  #8cacea;  
  --color: blue  
}
```

Globally scoped variable, `--color`

Locally scoped variable. The `--color` variable here is scoped to the `.block` selector

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, creating a vibrant, abstract pattern.

Локальные переменные

// Определение локальной переменной

```
.notifications { --notification-color: blue; }
```

// Локализация переменной для дочерних элементов

```
.notifications div {
```

```
color: var(--notification-color);
```

```
border: 1px solid var(--notification-color);
```

```
}
```

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, arranged in a gradient-like fashion from top to bottom.

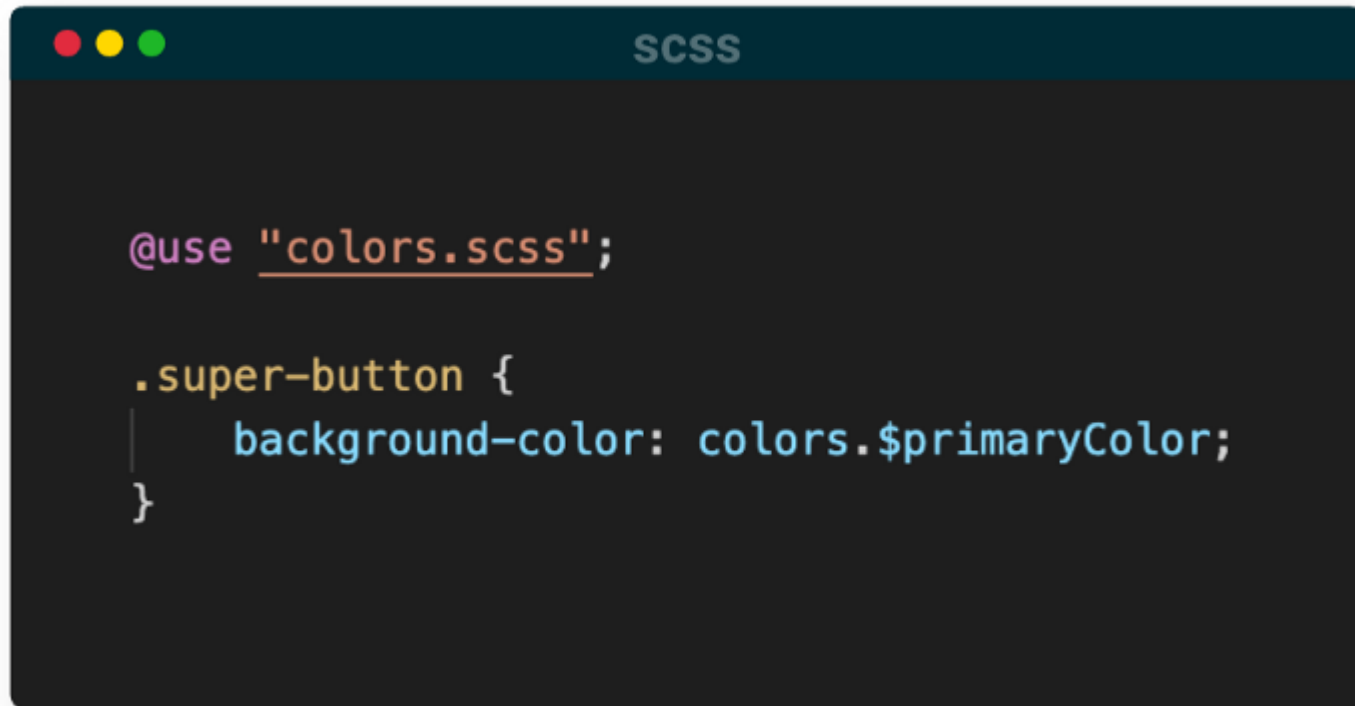
Метаязык SASS

Syntactically Awesome Stylesheet, или SASS, — это препроцессор CSS, добавляющий новые функции, которые в настоящее время недоступны в стандартной версии CSS.

SASS — расширенный набор стандартных CSS.

То есть все, что работает в CSS, будет работать в SASS.

Метаязык SASS



```
@use "colors.scss";

.super-button {
  background-color: colors.$primaryColor;
}
```




Метаязык SASS

Прежний синтаксис SASS больше не используется с файлами в формате SASS. Вместо этого применяется формат scss — новая (и улучшенная) версия SASS.

SCSS рекомендуется для опытных специалистов CSS.

С 10 декабря 2018 года SASS/SCSS не встраивается в браузеры. Для включения такой функции на своем веб-сервере необходимо установить компилятор SASS из командной строки.



Метаязык SASS

Если хотите начать экспериментировать с SASS, то зайдите на сайт www.codepen.io, где сможете легко начать использовать интерпретацию.

SASS без предварительной настройки.

CodePen.— это среда разработки для дизайнеров и программистов.

Имена переменных SASS определяются начальным символом \$, так же как в языке PHP!

Editor Preview

Choose your theme and fonts below and see how they look here.

```
html, body {  
  height: 100%; /* Full Screen */  
}  
body::after {
```

Syntax Highlighting

Dark Themes

- ☐ Twilight
- ☐ Solarized Dark
- ☐ Tomorrow Night

Light Themes

- ☐ Classic
- ☐ Solarized Light
- ☐ VS Light

Fonts

Code Font Family

MonoLisa

Made in Vienna by Component-Driven

Code Indentation

- ☒ Spaces
- ☐ Tabs

Indent Width

2

Editor Options

- ☒ Format On Save
- ☒ Line Numbers
- ☒ Line Wrapping

Emmet

CodePen supports Emmet.

Active ☒ Inactive

Key Bindings

I've been using Vim for about 2 years now, mostly because I can't figure out how to exit it.

- ☐ Normal
- ☒ Sublime Text
- ☐ Vim

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, arranged in a gradient-like fashion from top to bottom.

Метаязык SASS

```
$font: Helvetica, sans-serif;
```

```
$dark-gray: #333;
```

```
body {
```

```
  font: 16px $font;
```

```
  color: $dark-gray;
```

```
}
```



Метаязык SASS

Что можно сделать с помощью SASS?

```
$a: #E50C5E;
```

```
$b: #E16A2E;
```

```
.mixing-colors {  
  background-color: mix($a, $b, 30%);  
}
```

В данном случае SASS использовался для смешивания двух цветов, определенных в переменных SASS \$a и \$b.



Суть каскадных таблиц стилей

Содержит правило на основании, которого выбираются элементы HTML страницы, к которым применяются указанные в **фигурных скобках** стили

селектор {

свойство_CSS: значение;

свойство_CSS: значение;

...

}





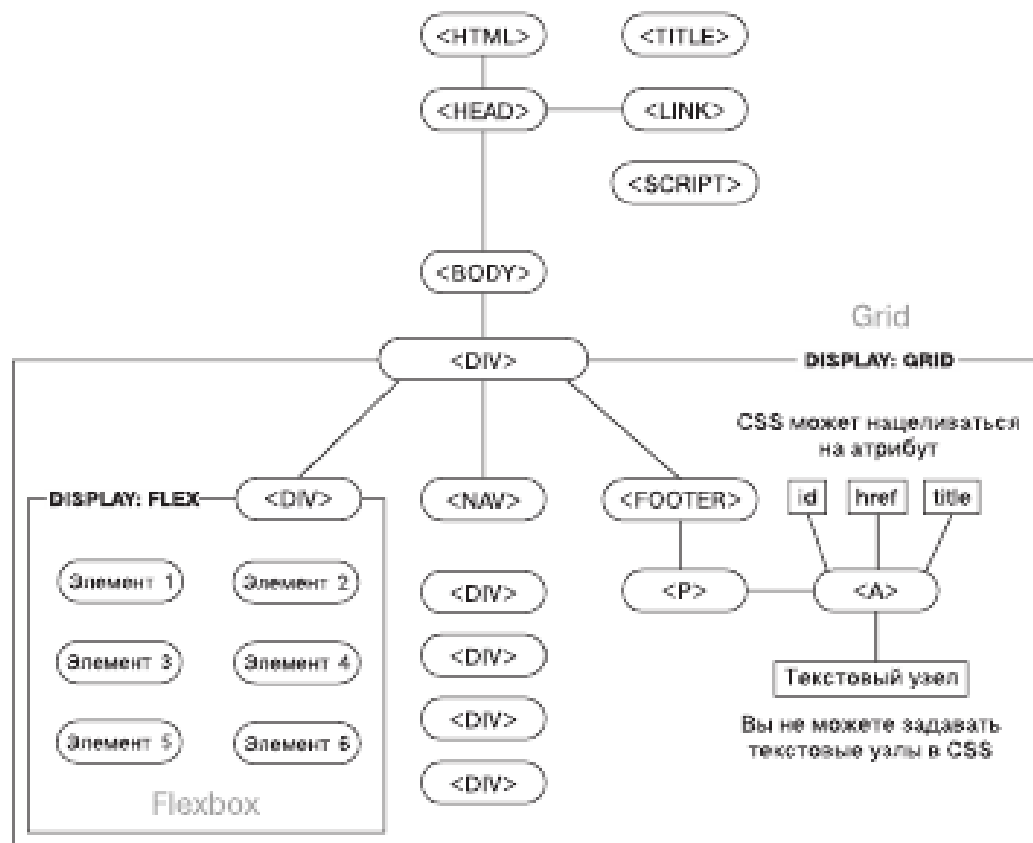
Суть каскадных таблиц стилей

Посмотрите на следующую схему.

Стили CSS буквально «сползают» по иерархии DOM, представляющей древовидную структуру вашего сайта.

Язык CSS (с помощью некоторых селекторов) позволяет контролировать этот обычно странный процесс.

Суть каскадных таблиц стилей



Суть каскадных таблиц стилей

Чтобы представить основную концепцию CSS, рассмотрим простую структуру сайта.





Суть каскадных таблиц стилей

На схеме представлено несколько элементов, вложенных в основной контейнер сайта.

CSS, подобно пинцету, помогает выбирать элементы, к которым мы хотим применить определенный стиль.

Если вы примените черный фон к элементу `body`, то все вложенные элементы в нем автоматически унаследуют черный фон:

```
body { background: black color: white; };
```

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, pink, and green, creating a vibrant, abstract pattern.

Суть каскадных таблиц стилей

Данный стиль будет «каскадно» перемещаться по родительской иерархии, заставляя все следующие HTML-элементы наследовать белый текст на черном фоне.

A vertical decorative bar on the left side of the slide, composed of numerous overlapping circles in shades of blue, yellow, orange, and pink, creating a vibrant, abstract pattern.

Суть каскадных таблиц стилей

Базовая.структура.HTML:

```
<body>
```

```
    <header>
```

```
        <p>Шапка сайта</p>
```

```
    </header>
```

```
    <article>
```

```
        <p>Основной контент/p>
```

```
    </article>
```

```
    <footer>
```

```
        <p>Privacy Policy. <span>&copy;
```

```
2019 Copyright</span></p>
```

```
    </footer>
```

```
</body>
```

Суть каскадных таблиц стилей

В том случае, если вы хотите оформить подвал (footer) и выделить слова Privacy Policy красным цветом, а 2018 Copyright — зеленым, можете расширить каскадный принцип, добавив следующие CSS- команды.

```
body      { background: black; color: white; }
```

```
footer    { color: red; }
```

```
footer span { color: green; }
```



Суть каскадных таблиц стилей

Обратите внимание: между словами footer и span есть пробел.

В CSS он выступает актуальным символом селектора CSS.

Это означает следующее: «найти в ранее указанном теге» (в данном примере footer).

Селекторы CSS

Основные CSS-селекторы:

/* Выбираем один элемент с идентификатором id */

```
#id { }
```

/* Выбираем все элементы с классом class */

```
.class { }
```

/* Выбираем все элементы с классом class1, создающие иерархию с другим родительским элементом с идентификатором parent */

```
#parent .class1 { }
```



Лояльность CSS

Поскольку разработка CSS велась для сред, в которых загрузка полной копии сайта не всегда гарантирована, он является одним из самых лояльных языков, похожих на HTML.

Если вы допустите ошибки или по какой-то причине страница полностью не загрузится, то CSS-код будет постепенно утрачивать свойства настолько, насколько возможно



Лояльность CSS

Это значит, что вы все еще можете использовать //
встроенные комментарии, но, вероятно, не стоит
этого делать.



Распространенные комбинации

Рассмотрим некоторые из наиболее распространенных комбинаций CSS-свойств и значений:

```
/* Устанавливаем белый цвет шрифта */  
color: #FFFFFF;
```

```
/* Устанавливаем черный цвет фона */  
background-color: #000000;
```

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, arranged in a gradient-like fashion from top to bottom.

Распространенные комбинации

`/* Создаем вокруг элемента синюю границу
толщиной 1 пиксел */`

`border: 1px solid blue;`

`/* Устанавливаем белый цвет шрифта */`

`font-family: Arial, sans-serif;`

`/* Устанавливаем размер шрифта 16px */`

`font-size: 16px;`

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles in various colors including blue, yellow, orange, pink, and green, of different sizes, creating a bokeh-like effect.

Распространенные комбинации

/* Добавляем отступы размером 32px */
padding: 32px;

/* Добавляем вокруг области контента отступ
размером 16px */
margin: 16px;

Сокращенные нотации

Назначим три различных свойства, способствующих появлению фонового изображения HTML-элемента:

```
background-color: #000000;
```

```
background-image: url("image.jpg");
```

```
background-repeat: no-repeat;
```

```
background-position: left top;
```

```
background-size: cover;
```

```
background-attachment: fixed;
```

Сокращенные нотации

То же самое можно сделать, используя одно сокращенное свойство.

`background`, разделяя значения пробелом:

```
background: background-color background-image background-repeat;
```

```
background: #000000 url("image.jpg") left top no-repeat  
fixed;
```

Сокращения также применимы к различным свойствам `grid`- и `flex`- верстки.