



Факультет компьютерных наук

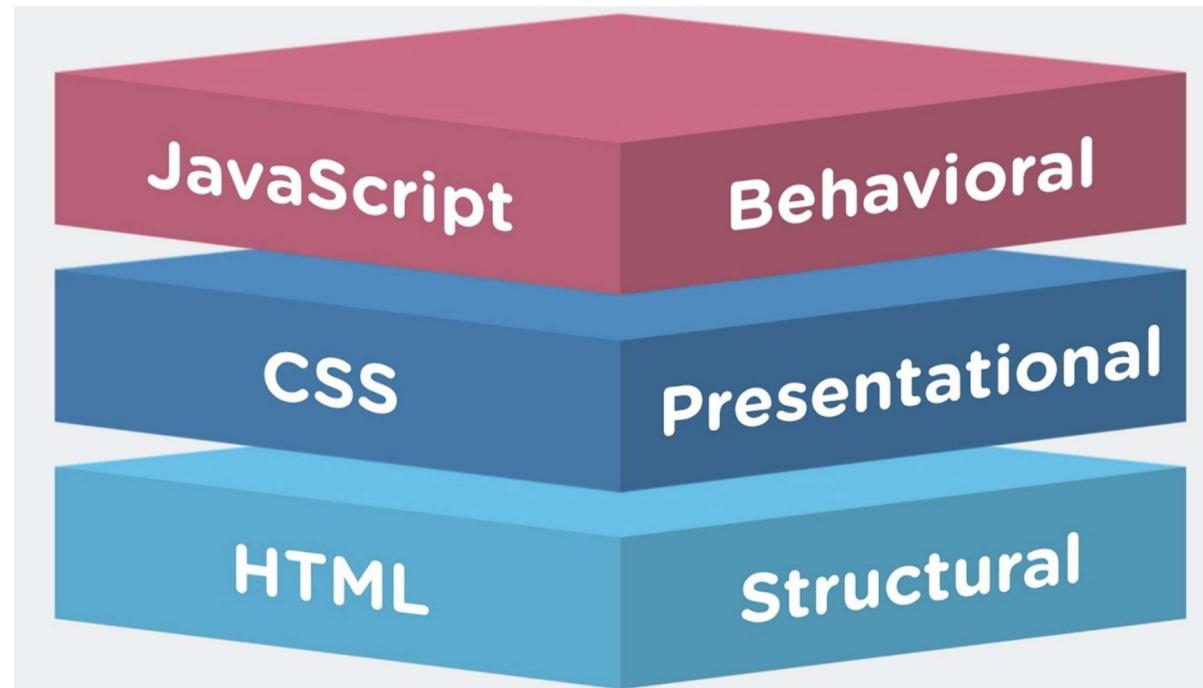
НИС «Введение в веб-разработку»

Москва
2024

Основы верстки 2

Преподаватель: Караваева Екатерина Андреевна
НИС «Введение в веб-разработку»

CSS



Немного истории



Термин «каскадные таблицы стилей» был предложен Хокон Виум Ли в 1994 году. Совместно с Бертом Босом он стал развивать CSS.

Версии:

- . Уровень 1 (CSS1) 1996 г
Параметры шрифтов: размер, стиль. Цвета текста, фона, рамок и других элементов. Выравнивание для текста, изображений, таблиц и других элементов. Высота, ширина, внутренние (padding) и внешние (margin) отступы и рамки.
- . Уровень 2 (CSS2) 1998 г
Блочная вёрстка. Относительное, абсолютное и фиксированное позиционирование. Расширенный механизм селекторов.
- . Уровень 2, ревизия 1 (CSS2.1) 2011 г
- . Модули уровня 3 (CSS3) с 2011 г. Возможность создавать анимированные элементы, поддержка линейных и радиальных градиентов, теней, сглаживания и многое другое.
- . Модули уровня 4 (CSS4) с 2011 г

Что может CSS?

Что может CSS?

1. Задавать внешний вид

Что может CSS?

1. Задавать внешний вид
2. Задавать размеры

Что может CSS?

1. Задавать внешний вид
2. Задавать размеры
3. Задавать расположение элементов

Как работает CSS?

Как работает CSS?

1. Браузер парсит HTML и строит DOM
2. Браузер парсит CSS и строит CSSOM

<https://web.dev/articles/critical-rendering-path/constructing-the-object-model?hl=ru>

HTML

```
<p>  
    Используем  
    <span>каскадные</span>  
    <span>таблицы</span>  
    <span>стилей</span>  
</p>
```

DOM

```
P  
└ "Используем"  
  └ SPAN  
    └ "каскадные"  
  └ SPAN  
    └ "таблицы"  
  └ SPAN  
    └ "стилей"
```

CSS

```
span {  
    border: 1px solid black;  
    background-color: lime;  
}
```

Используем **каскадные** **таблицы** **стилей**

Как подключить стили
к странице?

Внешняя таблица стилей

Это отдельный файл с расширением `.css`, в котором и хранится CSS. Вы ссылаетесь на него из HTML при помощи элемента `<link>`, который вы располагаете в секции `<head>`.

```
<html>
<head>
...
  <link rel="stylesheet" href="style.css">
</head>
...
</html>
```

Внутренняя таблица

Содержится внутри элемента **<style>** и не требует подключения внешних файлов.

```
<html>
<head>
...
<style>
  body {
    background-color: blue;
  }
</style>
</head>
...
</html>
```

Внутренняя таблица

Содержится внутри элемента **<style>** и не требует подключения внешних файлов.

```
<html>
<head>
...
<style>
  body {
    background-color: blue;
  }
</style>
</head>
...
</html>
```



Встроенные стили (inline css)

Это объявление CSS, которое применимо только к одному элементу и содержащееся в атрибуте **style**

```
<html>
...

```

Встроенные стили (inline css)

Это объявление CSS, которое применимо только к одному элементу и содержащееся в атрибуте **style**

```
<html>
...

```



Синтаксис CSS

Каждое правило CSS из таблицы стилей имеет две основные части — *селектор* и *блок объявлений*.

```
селектор {  
    свойство: значение;  
    свойство: значение;  
    свойство: значение;  
}
```

Синтаксис в CSS

Каждое правило CSS из таблицы стилей имеет две основные части — *селекторы* и *блок объявлений*.

```
/* Это комментарий CSS */
div {
    background: lightgreen;
    color: black;
}
/*
    Комментарии предназначены только для чтения людьми и компьютер
    их разбирать не будет
*/
```



Синтаксис в CSS

В CSS существует более 300 различных свойств и чуть ли не бесконечное количество значений.

Для каждого свойства определен некоторый список допустимых значений.

Обычно мы используем 50-100 свойств

<https://webref.ru/css>

<https://developer.mozilla.org/en-US/docs/Web/CSS/Reference>



Синтаксис css

Типы
селекторов

Синтаксис CSS

Универсальный селектор

```
* {  
    margin: 0;  
    padding: 0;  
}
```

```
<div>  
    <span>  
        <a href="/"></a>  
    </span>  
    <h1></h1>  
    <ul>  
        <li></li>  
    </ul>  
</div>
```



Синтаксис CSS

Селектор тега

```
p {  
    font-family: arial, Helvetica, sans-serif;  
}
```

<https://codepen.io/ngm/pen/qjGEeL>

```
<div>  
  <p>  
    <a href="/"></a>  
    <p></p>  
  </p>  
  <h1></h1>  
  <ul>  
    <li>  
      <p></p>  
    </li>  
  </ul>  
</div>
```

Синтаксис CSS

Селектор класса

```
.note {  
    font-weight: bold;  
    color: red;  
    background-color: yellow;  
}
```

<https://codepen.io/ngm/pen/yXWNyP>

```
<div>  
  <p class="note">  
    <a href="/"></a>  
    <p></p>  
  </p>  
  <h1></h1>  
  <ul>  
    <li>  
      <p></p>  
    </li>  
  </ul>  
</div>
```

Синтаксис CSS

Селектор по идентификатору (id)

```
#paragraph1 {  
    margin: 0;  
}
```

<https://codepen.io/ngm/pen/mwYJVJ>

```
<div>  
  <p>  
    <a href="/"></a>  
    <p></p>  
  </p>  
  <h1 id="paragraph1"></h1>  
  <ul>  
    <li>  
      <p></p>  
    </li>  
  </ul>  
</div>
```

Синтаксис CSS

Селектор атрибутов

```
a[href="https://www.somesite.com"] {  
    font-weight: bold;  
}
```

<https://codepen.io/ngm/pen/RgmPoP>

```
<div>  
  <p>  
    <a href="http://www.somesite.com"></a>  
    <p></p>  
  </p>  
  <h1></h1>  
  <ul>  
    <li>  
      <p></p>  
    </li>  
  </ul>  
</div>
```

Синтаксис CSS

Контекстные селекторы (селекторы потомков)

```
div#paragraph1 p.note {  
    color: red;  
}
```

<https://codepen.io/ngm/pen/pwmJrL>

```
<div>  
  <p>  
    <a href="/"></a>  
    <p></p>  
  </p>  
  <h1 id="paragraph1"></h1>  
  <ul>  
    <li>  
      <p></p>  
    </li>  
  </ul>  
  <div id="paragraph1">  
    <p class="note"></p>  
    <p></p>  
    <p class="note2"></p>  
  </div>  
</div>
```

Синтаксис CSS

Селектор дочерних элементов

```
div.note > b {  
    color: red;  
}
```

<https://codepen.io/ngm/pen/OgYVBg>

```
<div>  
  <p>  
    <a href="/"></a>  
    <p></p>  
  </p>  
  <h1></h1>  
  <ul>  
    <li>  
      <p></p>  
    </li>  
  </ul>  
  <div class="note">  
    <p class="note">  
      <b></b>  
    </p>  
    <b></b>  
  </div>  
</div>
```

Синтаксис CSS

Селектор сестринских элементов

```
h1 + p {  
  margin-top: 20px;  
}
```

<https://codepen.io/ngm/pen/qjGdeO>

```
<div>  
  <p>  
    <a href="/"></a>  
    <p></p>  
  </p>  
  <h1></h1>  
  <p></p>  
  <p></p>  
  <ul>  
    <li>  
      <p></p>  
    </li>  
  </ul>  
  <p></p>  
</div>
```

Синтаксис CSS

Селектор псевдоклассов

```
a:hover {  
    color: blue;  
}
```

<https://codepen.io/ngm/pen/ModmKr>

```
<div>  
    <p>  
        <a href="/"></a>  
        <p></p>  
    </p>  
    <h1></h1>  
    <p></p>  
    <p></p>  
    <ul>  
        <li>  
            <p></p>  
        </li>  
    </ul>  
    <p></p>  
</div>
```

Синтаксис CSS

Селектор псевдоэлементов

```
p::first-letter {  
    font-size: 32px;  
}
```

<https://codepen.io/ngm/pen/JJqNXw>

```
<div>  
  <p>  
    Текст  
    <p>Текст</p>  
  </p>  
  <h1></h1>  
  <p></p>  
  <p></p>  
  <ul>  
    <li>  
      <p></p>  
    </li>  
  </ul>  
  <p></p>  
</div>
```

Сброс CSS

На самом деле HTML без стилей не бывает.

Каждая веб-страница использует по крайней мере один CSS: **стиль клиентского приложения**.

Этот файл CSS включён в браузер и вызывается:

- **каждый раз**, когда веб-страница визуализируется;
- **до того**, как применяется любой из наших CSS.

Стили браузера по умолчанию могут **мешать** стилям, которые мы на самом деле хотим применить.

Единицы размера в CSS

Существует много свойств CSS, которые требуют размер в качестве единицы:

- **font-size** определяет размер текста
- **border-width** определяет толщину границ элементов
- **margin** определяет пространство между элементами
- **padding** определяет внутренние отступы элемента
- **top/left/right/bottom** позволяют позиционировать и перемещать элементы
- **width/height** определяет размеры элемента

Единицы размера в css

Наиболее часто используемые единицы:

- **px** для пикселей
- **%** для процентов
- **em/rem** для определения размера относительно родительского значения **font-size**
- **vw/vh** для определения размера относительно экрана

Единицы размера в css

Проценты

Относительные единицы: полагаются на родителя и/или предка элемента.

```
.container {  
    width: 80%;  
}  
  
.item {  
    margin-top: 10%;  
}
```

Единицы размера в CSS

em и rem

Относительные единицы: зависят от значения **font-size** родительского элемента (**em**) или корневого элемента (**rem**).

```
.container {  
    font-size: 20px;  
}  
  
h2 {  
    font-size: 2em; /* = 20px * 2 = 40px */  
}
```

```
html {  
    font-size: 20px;  
}  
  
h1 {  
    font-size: 3rem; /* = 20px * 3 = 60px */  
}  
  
p {  
    font-size: 0.8rem; /* = 20px * 0.8 = 16px */  
}
```

Единицы размера в css

vw и vh

Относительные единицы: зависят от размеров окна браузера (**viewport**).

- **1vw** – 1% ширины окна
- **1vh** – 1% высоты окна

```
h2 {  
    font-size: 3vw;  
}  
  
section {  
    height: 100vh;  
}
```

Цвета в css

Название цвета

На данный момент в css 147 названий цветов, от самых простых (black, white, orange, yellow, blue и др.) до более специфичных (lawngreen, orchid, crimson и др.).

```
body {  
    color: black;  
}  
  
a {  
    color: orange;  
}
```

Цвета в css

rgb и rgba

Является результатом сочетания красного, зеленого и синего, у каждого из этих трёх цветов есть 256 возможных значений, существует $256 * 256 * 256 = 16777216$ возможных цветов.

```
body {  
    background-color: rgba(0, 0, 0, 0.8);  
}  
  
a {  
    color: rgb(219, 78, 68);  
}
```

Цвета в css

Шестнадцатеричные значения (HEX)

Сочетание красного, зеленого и синего, причем каждый из них представляют в шестнадцатеричном значении, например DB для красного, 4E для зелёного и 44 для синего.

```
body {  
    background-color: #000000;  
}  
  
a {  
    color: #ADFF2F;  
}
```

Цвета в css

Что почитать?

<http://css.yoksel.ru/size-units/>

<https://learn.javascript.ru/css-units>

<https://habr.com/ru/company/ruvds/blog/505626/>

font-family

Свойство CSS определяет приоритетный список из одного или нескольких имен семейств шрифтов и/или общих имен семейств для выбранного элемента.**font-family**

<https://developer.mozilla.org/en-US/docs/Web/CSS/font-family>

font-style

Свойство **font-style** определяет, следует ли применять к шрифту обычное, курсивное или наклонное начертание в зависимости от его **.font-style** font-family

Свойство **font-weight** позволяет изменять толщину шрифта

font-style:italic;

font-weight:bold;

font-weight:800;

font-weight:10;

<https://developer.mozilla.org/en-US/docs/Web/CSS/font-style>

<https://developer.mozilla.org/ru/docs/Web/CSS/font-weight>

font-size

CSS Свойство **font-size** определяет размер шрифта. Это свойство также используется для вычисления размера `em`, `ex` и других относительных единиц. Подробнее: [<length>](#).

Выравнивание текста

С помощью CSS свойства `text-align` вы можете выровнять текст элемента по горизонтали. Текст может быть выровнен:

- по центру (значение `center`);
- по левому краю (`left`);
- по правому краю (`right`);
- по ширине (`justify`).

`text-align:center;`

`text-align:left;`

`text-align:right;`

`text-align:justify;`

Выравнивание текста

Добавляет оформление текста в виде его подчеркивания, перечеркивания, линии над текстом и мигания. Одновременно можно применить более одного стиля, перечисляя значения через пробел.

Значения

blink	Устанавливает мигающий текст. Такой текст периодически, примерно раз в секунду исчезает, потом вновь появляется на прежнем месте. Это значение в настоящее время не поддерживается браузерами и осуждается в CSS3, взамен рекомендуется использовать анимацию.
line-through	Создает перечеркнутый текст (пример).
overline	Линия проходит над текстом (пример).
underline	Устанавливает подчеркнутый текст (пример).
none	Отменяет все эффекты, в том числе и подчеркивания у ссылок, которое задано по умолчанию.
inherit	Значение наследуется у родителя.

Отступы и межстрочный интервал

- С помощью CSS свойства `letter-spacing` можно увеличивать или уменьшать отступ между буквами в тексте HTML элементов.
- С помощью свойства `word-spacing` можно увеличивать или уменьшать отступ между словами в тексте HTML элементов.
- С помощью `line-height` можно задавать расстояние между строками.

`letter-spacing: 10px;`

`word-spacing: 15px;`

`line-height : 150%;`

Оформление текста

Свойство	Описание	Значения
direction	Устанавливает направление текста.	ltr (слева направо) rtl (справа налево)
text-indent	Устанавливает величину отступа первого символа текста.	пиксели %
text-transform	Устанавливает регистр букв текста элемента.	none capitalize (все слова с больших букв) uppercase (все слова большими буквами) lowercase (все слова маленькими буквами)

Задание

Этот абзац с отступом между буквами 17 пикселей и
отступом между словами 5 пикселей. Цвет текста —
оранжевый.

Этот текст подчеркнут, отступ между буквами равен 15
пикселей, цвет текста — серый.

Этот текст выровнен по центру, отступ между словами — 10 пикселей. Цвет текста — #ff3366.

текст этого элемента выровнен по правому краю, отступ между буквами равен 6 пикселей. текст написан маленькими буквами красного цвета.

текст этого элемента выровнен по центру, подчёркнут, отступ между буквами 7
пикселей. текст написан большими буквами зелёного цвета.

Задание

Описание задания:

- Первый абзац имеет оранжевый цвет текста. Отступ между буквами составляет 17 пикселей, а между словами — 5 пикселей.
- Второй абзац серого цвета, подчеркнутый, с отступом между буквами 15 пикселей.
- Третий абзац выровнен по центру. Отступ между словами составляет 10 пикселей, а цвет текста — #ff3366.
- Четвёртый абзац красного цвета, текст выровнен по правому краю, написан маленькими буквами, с отступом между буквами 6 пикселей.
- Пятый абзац зеленого цвета, подчеркнутый, выровнен по центру, написан большими буквами, с отступом между буквами 7 пикселей

Э т о а б з а ц с о т с т у п о м м е ж д у б у к в а м и 1 7 п и к с е л е й и
о т с т у п о м м е ж д у с л о в а м и 5 п и к с е л е й . Ц в е т т е к с т а —
о р а н ж е в ы й .

Э т о т т е к с т п о д ч е р к н у т , о т с т у п м е ж д у б у к в а м и р а в е н 1 5
п и к с е л е й , ц в е т т е к с т а — с е р ы й .

Этот текст выровнен по центру, отступ между словами — 10 пикселей. Цвет текста — #ff3366.

т е к с т э т о г о э л е м е н т а в y р o в n e n p o p r a v o m u k r a y o , o t s t u p m e j d u b u k v a m i r a v e n 6 p i k s e l e y . t e k s t
n a p i s a n m a l e n y k i m i b u k v a m i k r a s n o g o c v e t a .

т е к с т э т о г о э л е м е н т а v y r o v n e n p o c e n t r u , p o d c h e r k n u t , o t s t u p m e j d u b u k v a m i 7
p i k s e l e y . t e k s t n a p i s a n b o l s h i m i b u k v a m i z e l e n o g o c v e t a .

Фон в CSS

background-attachment — определяет, будет ли фон фиксированным относительно окна браузера или будет прокручиваться вместе с содержимым страницы. Значения: scroll (по умолчанию), fixed, local.

background-color — задает цвет фона элемента. Значения: любые цветовые форматы (например, red, #ff0000, rgba(255, 0, 0, 0.5)).

background-image — указывает изображение в качестве фона элемента. Значения: путь к изображению (например, url('image.jpg')), либо none для отсутствия изображения.

background-position — задает положение фонового изображения относительно элемента. В качестве первого значения данного свойства должна задаваться величина смещения изображения по горизонтали, а в качестве второго величина смещения по вертикали.

background-repeat — определяет, будет ли фоновое изображение повторяться. Значения: repeat (по умолчанию), no-repeat, repeat-x, repeat-y.

<https://codepen.io/bdktvqzv-the-bold/pen/Vwopagy>

Фон в CSS

Свойство	Описание	Значение	
background-attachment	Указывает будет ли фоновая картинка привязана к одному месту, или будет прокручиваться вместе с текстом.	scroll	fixed
background-image	Устанавливает фоновую картинку для элемента	url (URL)	
background-size	Устанавливает размеры фоновой картинки	250px 200px – задание точных размеров 250px auto – второй размер изменится пропорционально cover – заполняет весь фон, сохраняя пропорции	
background-repeat	Указывает как фоновая картинка будет повторяться	repeat repeat-x	repeat-y no-repeat

Фон в CSS

Свойство	Описание	Значение	
background-position	Указывает координаты расположения фоновой картинки.	<code>left top left center left bottom center top center center center bottom</code>	<code>right top right center right bottom x% y% xpos ypos</code>
background-color	Устанавливает фоновый цвет элемента.	<code>rgb(r,g,b) #rrggbb название_цвета</code>	

Задание

Создать веб-страницу с фоновым изображением, которое занимает весь экран и не повторяется. Для этого необходимо использовать HTML и CSS.

1. **background-image**: задать фоновое изображение с указанием пути к файлу
2. **background-repeat: no-repeat;**: запретить повторение фонового изображения.
3. **background-size: cover;**: настроить изображение так, чтобы оно автоматически изменяло размер, заполняя весь экран, сохраняя свои пропорции.
4. **background-position: center;**: центрировать изображение.
5. **margin: 0;**: убрать стандартные отступы браузера для элемента body.
6. **height: 100vh;**: задать высоту страницы в 100% от высоты видимого окна браузера (viewport), чтобы фоновое изображение занимало всю видимую область.

Opacity

Opacity — это CSS-свойство, которое задает прозрачность элемента. Значение свойства может варьироваться от 0 (полностью прозрачный) до 1 (полностью непрозрачный).

- **opacity: 0** — элемент полностью прозрачный.
- **opacity: 1** — элемент полностью видимый.
- **Значения от 0.1 до 0.9** регулируют частичную прозрачность элемента.

```
<div style="opacity: 0.5; background-color: blue; color: white; padding: 20px;">
```

Это пример элемента с прозрачностью 0.5
</div>

Задание

Первый квадрат с **opacity: 1** (полностью видимый).

Второй квадрат с **opacity: 0.5** (полупрозрачный).

Третий квадрат с **opacity: 0.2** (более прозрачный).

CSS

Блочная модель (Box model)

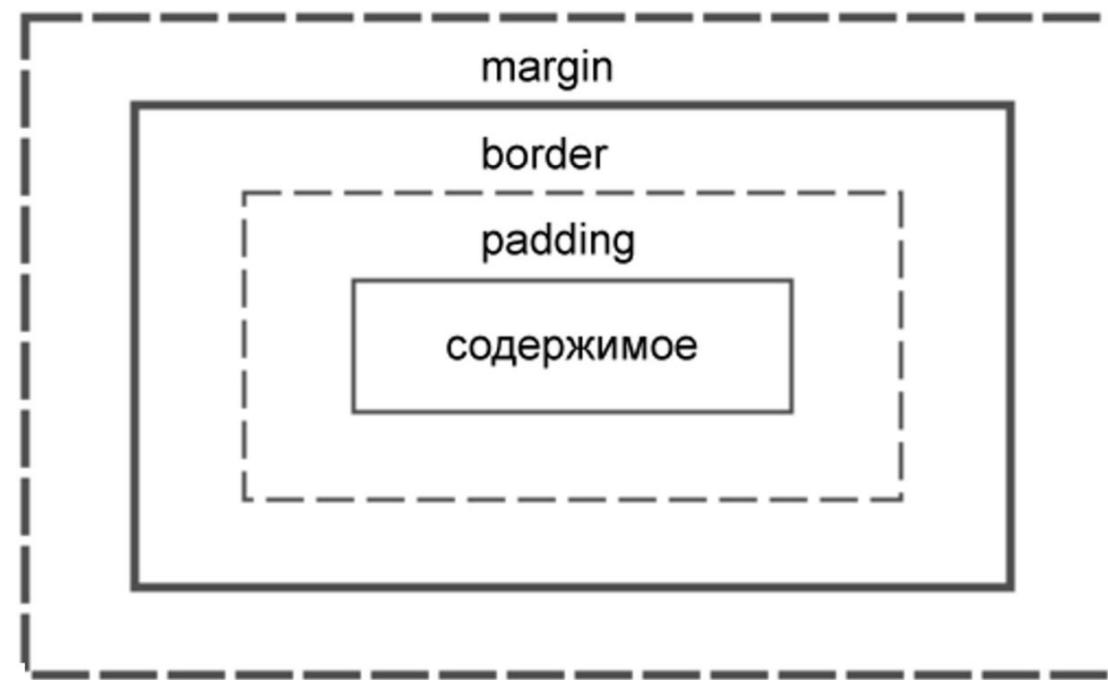
Блочная модель

Блочная модель (box model) — одно из фундаментальных понятий веб-верстки.

Элемент в HTML документе выводится внутри своего отдельного прямоугольного блока.

Блочная модель

Блок состоит из непосредственно контента (внутреннего содержимого), внутренних отступов (padding), границ (border) и, наконец, внешних отступов (margin).



Блочная модель

Размеры блока

Свойство **box-sizing** позволяет вам немного управлять моделью, применяемой для задания размеров блоков. Два возможных значения — это **content-box** и **border-box**.

content-box

По умолчанию. При вычислении размера блока добавляются **padding** и **border**.

border-box

При вычислении размера блока **padding** и **border** не учитываются.

Блочная модель

Размеры блока

Свойство **box-sizing** позволяет вам немного управлять моделью, применяемой для задания размеров блоков. Два возможных значения — это **content-box** и **border-box**.

content-box

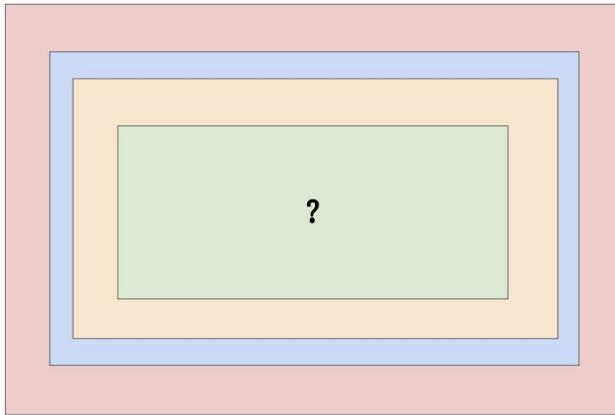
По умолчанию. При вычислении размера блока добавляются **padding** и **border**.

border-box

При вычислении размера блока **padding** и **border** не учитываются.

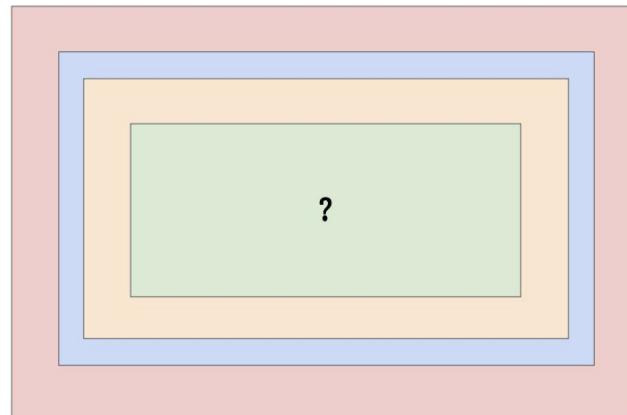
Блочная модель

content-box



```
{  
  width: 200px;  
  height: 100px;  
  padding: 20px;  
  border: 10px;  
  margin: 20px;  
}
```

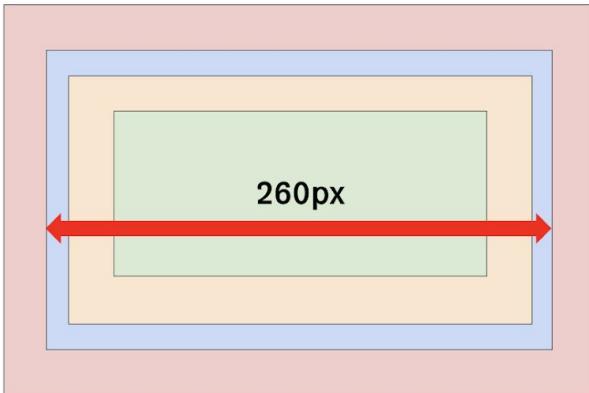
border-box



```
{  
  width: 200px;  
  height: 100px;  
  padding: 20px;  
  border: 10px;  
  margin: 20px;  
  box-sizing: border-box;  
}
```

Блочная модель

content-box



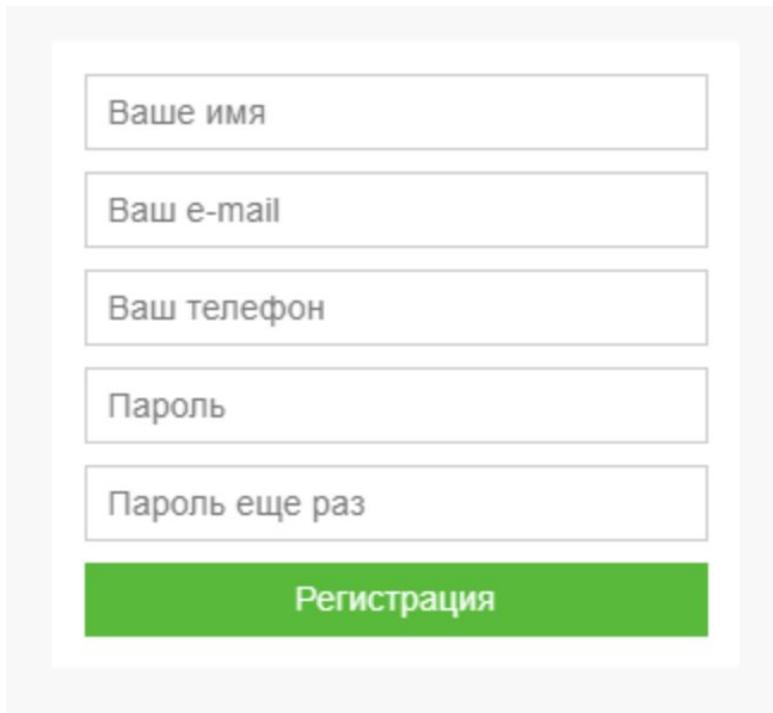
```
{  
  width: 200px;  
  height: 100px;  
  padding: 20px;  
  border: 10px;  
  margin: 20px;  
}
```

border-box



```
{  
  width: 200px;  
  height: 100px;  
  padding: 20px;  
  border: 10px;  
  margin: 20px;  
  box-sizing: border-box;  
}
```

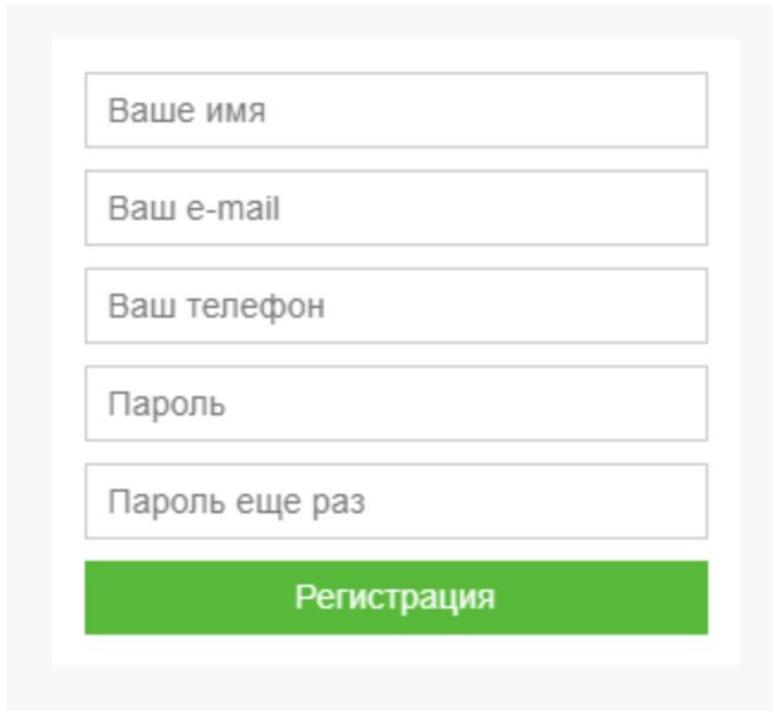
Задание



A registration form template consisting of five input fields and one button. The fields are labeled: 'Ваше имя' (Your name), 'Ваш e-mail' (Your email), 'Ваш телефон' (Your phone), 'Пароль' (Password), and 'Пароль еще раз' (Password again). Below these fields is a green rectangular button labeled 'Регистрация' (Registration).

- Цвет фона страницы **#f8f8f8**;
- Размер формы **320px**, внутренний отступ **15px**, фон **#fff**;
- Цвет границ input **#ccc**;
- Цвет фона кнопки **#1cbc11**;
- Цвет фона кнопки при наведении **#14a20a**;

Задание



A registration form template consisting of five input fields and one button. The fields are labeled: 'Ваше имя' (Your name), 'Ваш e-mail' (Your email), 'Ваш телефон' (Your phone), 'Пароль' (Password), and 'Пароль еще раз' (Password again). Below these fields is a green button labeled 'Регистрация' (Registration).

- Цвет фона страницы **#f8f8f8**;
- Размер формы **320px**, внутренний отступ **15px**, фон **#fff**;
- Цвет границ input **#ccc**;
- Цвет фона кнопки **#1cbc11**;
- Цвет фона кнопки при наведении **#14a20a**;

<https://codepen.io/bdktvqzv-the-bold/pen/QWepjOy>

Каскадность, специфичность, наследование

CSS - это сокращение от *Cascading Style Sheets* (**Каскадные таблицы стилей**)

Каскадность означает, что к одному и тому же элементу может применяться несколько CSS-правил.

- 1. Важность**
- 2. Специфичность**
- 3. Исходный порядок**

Важность

Запись свойства с **!important** в конце всегда имеет приоритет перед остальными

```
.better {  
    background-color: gray;  
    border: none !important;  
}
```

<http://codepen.io/ngm/pen/eRaWWK>

Специфичность

Побеждает тот селектор, который более специфичен.

4 составляющих специфичности:

1. Нахождение селектора внутри элемента **<style>** или внутри атрибута **style**
2. Наличие селектора ID
3. Наличие селектора класса, атрибута или псевдо-класса
4. Наличие селектора элемента или псевдо-элемента

Селектор	style=""	#id	.class	<tag>	Специфичность
h1	0	0	0	1	0001
#id	0	1	0	0	0100
h1 + p::first-letter	0	0	0	3	0003
li > a[href="site"] > .inline-warning	0	0	2	2	0022
#id div > div > a:hover, внутри элемента <style>	1	1	1	3	1113

Селектор	style=""	#id	.class	<tag>	Специфичность
span	0	0	0	0	0000

Какая специфичность у `span`?

Селектор	style=""	#id	.class	<tag>	Специфичность
span	0	0	0	1	0001

Селектор	style=""	#id	.class	<tag>	Специфичность
span	0	0	0	1	0001
div .class	0	0	0	0	0000

Селектор	style=""	#id	.class	<tag>	Специфичность
span	0	0	0	1	0001
div .class	0	0	1	1	0011

Селектор	style=""	#id	.class	<tag>	Специфичность
span	0	0	0	1	0001
div .class	0	0	1	1	0011
#id .class	0	0	0	0	0000

Селектор	style=""	#id	.class	<tag>	Специфичность
span	0	0	0	1	0001
div .class	0	0	1	1	0011
#id .class	0	1	1	0	0110
div span	0	0	0	0	0000

Селектор	style=""	#id	.class	<tag>	Специфичность
span	0	0	0	1	0001
div .class	0	0	1	1	0011
#id .class	0	1	1	0	0110
div span	0	0	0	2	0002

Селектор	style=""	#id	.class	<tag>	Специфичность
span	0	0	0	1	0001
div .class	0	0	1	1	0011
#id .class	0	1	1	0	0110
div span	0	0	0	2	0002
#id span	0	0	0	0	0000

Селектор	style=""	#id	.class	<tag>	Специфичность
span	0	0	0	1	0001
div .class	0	0	1	1	0011
#id .class	0	1	1	0	0110
div span	0	0	0	2	0002
#id span	0	1	0	1	0101

Display

Определяет, как будет показан (или не показан) элемент на странице

- **block**: элемент показывается как блочный, занимает всю доступную ширину и всегда начинается с новой строки
- **inline**: при возможности помещается в ту же строку, занимает столько места, сколько ему требуется

block

<p>
<div>
<h1> – <h6>
<section>
<form>
<header>
<footer>

inline

 <a> <button> <input> <label>

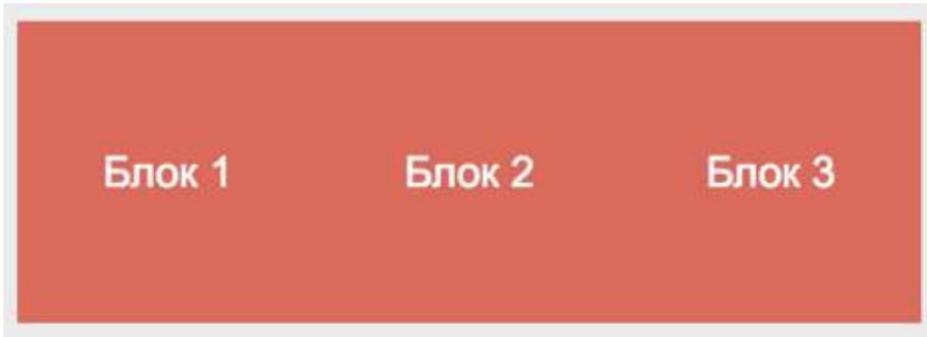
Display

Определяет, как будет показан (или не показан) элемент на странице

- **block**: элемент показывается как блочный, занимает всю доступную ширину и всегда начинается с новой строки
- **inline**: при возможности помещается в ту же строку, занимает столько места, сколько ему требуется
- **none**: элемент не отображается
- **inline-block**: ведут себя как инлайн-элементы, но им можно задать свойства блоков
- **flex, grid, table** и др.

Display

Есть особенность у `display: inline-block`



```
/* Способ 1 */
.box {
    font-size: 0;
}

<!-- Способ 2 --&gt;
&lt;div&gt;Блок 1&lt;/div&gt;&lt;div&gt;Блок 2&lt;/div&gt;</pre>
```

Задание

Создайте форму для выбора любимого цвета.
Используйте радио-кнопки для выбора цвета, а
также стилизуйте их с помощью CSS.

Шаги:

1. Создайте HTML-форму с несколькими радио-кнопками, каждая из которых будет представлять один из цветов: красный, синий и зеленый.
2. Используйте CSS для стилизации радио-кнопок, чтобы они выглядели аккуратно.

Выберите ваш любимый цвет:

- Красный
- Синий
- Зеленый

Отправить

Задание

Выберите ваш любимый цвет:

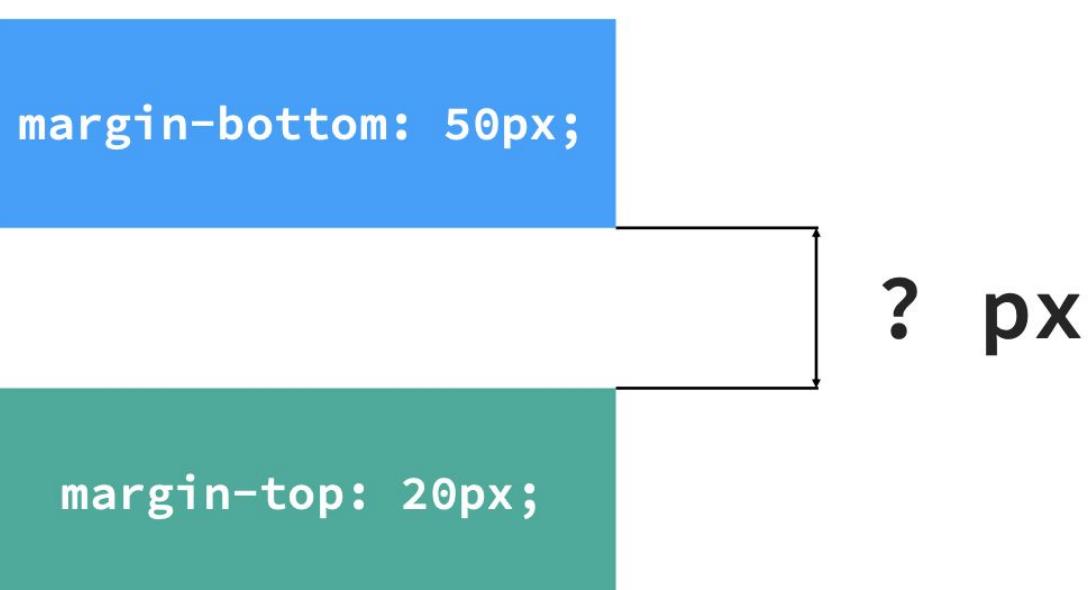
- Красный
- Синий
- Зеленый

Отправить

Вы выбрали: red

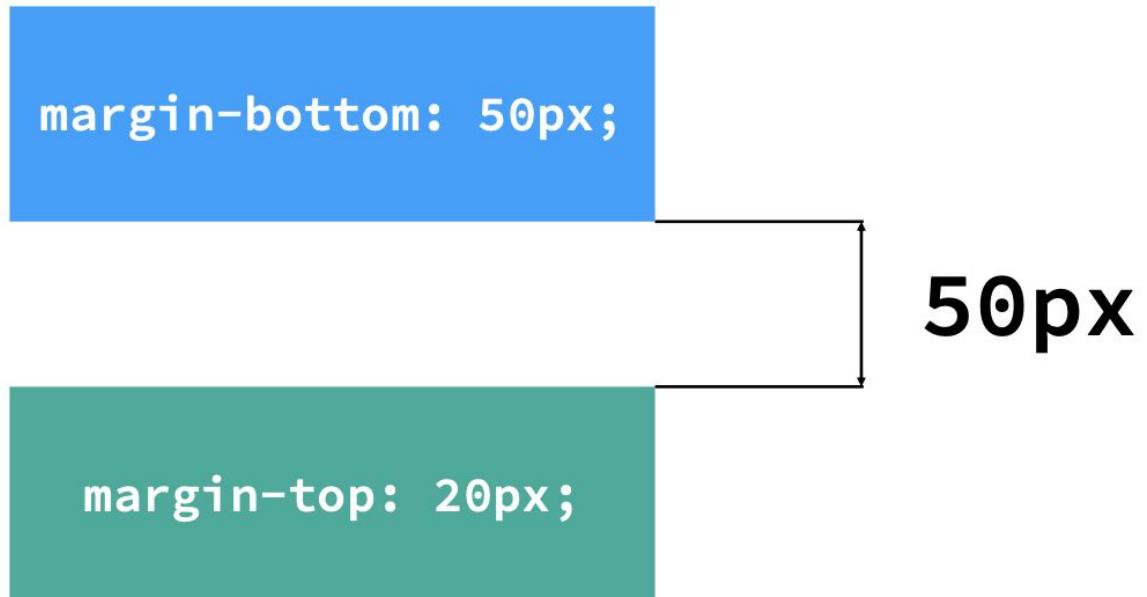
Особенности box-model

Какой будет отступ у блоков расположенныхных друг под другом?



Особенности box-model

Какой будет отступ у блоков расположенных друг под другом?



Особенности box-model

Какой будет отступ у блоков расположенных друг под другом?

`margin-bottom: 50px;`

50px

`margin-top: 20px;`

Отступы `margin-top` и `margin-bottom` иногда объединяются в один, с размером равным наибольшему из них (или размеру одного, если они равны).

Это поведение известно как схлопывание внешних отступов (margin collapsing).

max-width / max-height min-width / min-height

Используются для задания
максимальных/минимальных значений

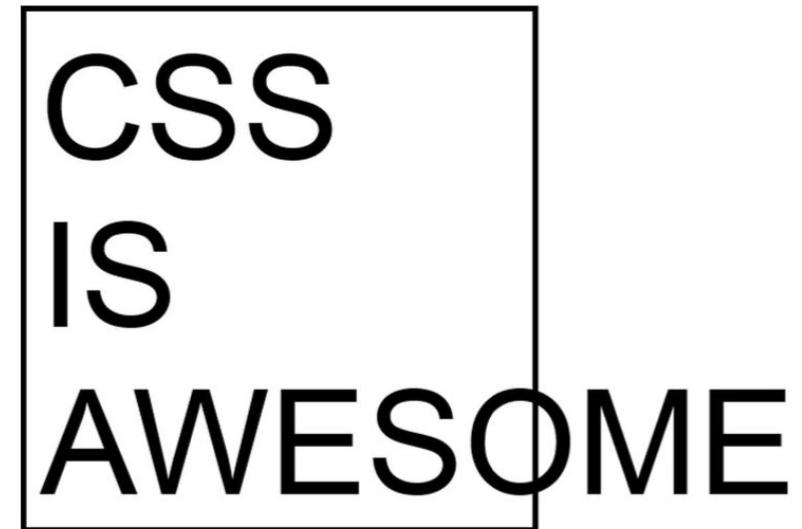
- Если элементу задано и свойство **width**,
и **max-width/min-width**, то последние
являются более приоритетными
- Разумеется, то же самое относится к свойствам
height/min-height/max-height

```
.box {  
    width: 300px;  
    max-width: 200px;  
}
```

overflow

Определяет, как отображать контент, если он не помещается в заданную область

- **visible**: контент отображается за пределами блока
- **hidden**: контент, который не помещается в заданную область, не отображается
- **scroll**: в блок добавляется скроллбар для прокрутки контента
- **auto**: если контент не помещается в заданную область, в блок добавляется скроллбар



`overflow: visible;`

Задание

< 1 2 3 >

Задание

< 1 2 3 >

<https://codepen.io/nosia/pen/QWVGmRj>

Задание

Инструкции:

1. Создайте HTML страницу с минимальной структурой: <html>, <head> и <body>.
2. Используйте следующие элементы:
 1. <div> для группировки элементов.
 2. для выделения части текста.
 3. для отображения изображения.
 4. <form> с текстовым полем и кнопкой.
 5. (неупорядоченный список) и (упорядоченный список) с не менее чем тремя элементами в каждом.
 6. <a> для создания ссылок с атрибутом href.
3. Добавьте к вашей странице CSS стилизацию для элементов.

Задание

Инструкции:

1. Создайте HTML страницу с минимальной структурой: <html>, <head> и <body>.
2. Используйте следующие элементы:
 1. <div> для группировки элементов.
 2. для выделения части текста.
 3. для отображения изображения.
 4. <form> с текстовым полем и кнопкой.
 5. (неупорядоченный список) и (упорядоченный список) с не менее чем тремя элементами в каждом.
 6. <a> для создания ссылок с атрибутом href.
3. Добавьте к вашей странице базовую CSS стилизацию для элементов.

<https://pastebin.com/rxYMeNzQ>

Тест

<https://forms.gle/3JXEFsznXQf85Fxw6>

