



Преподаватель:

Коляда

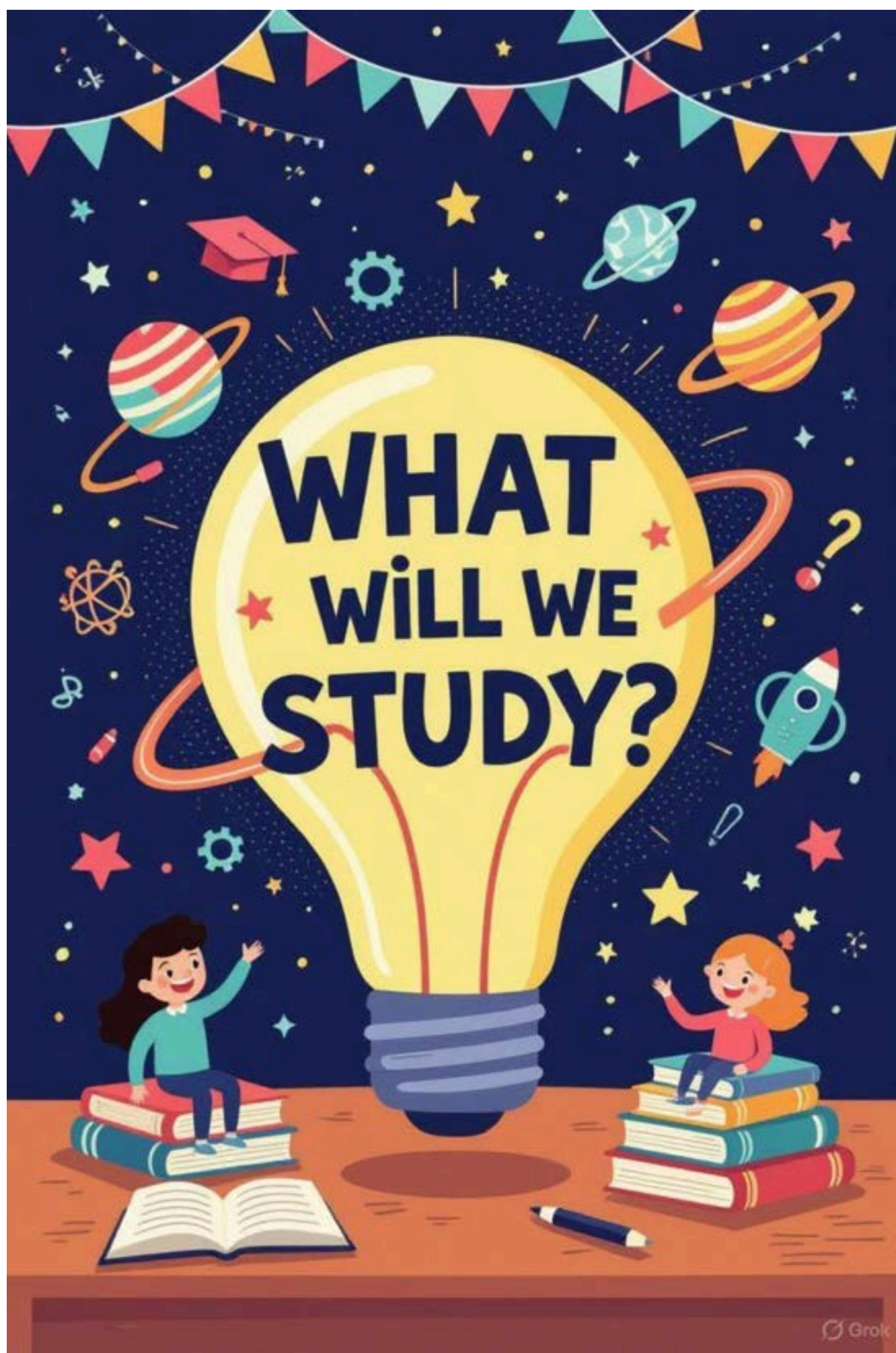
Никита Владимирович

Обратная связь:

- сообщения на inStudy

БАЗОВЫЕ CSS-СВОЙСТВА БЛОЧНАЯ МОДЕЛЬ

ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА
ИНТЕРФЕЙСОВ ПОЛЬЗОВАТЕЛЯ



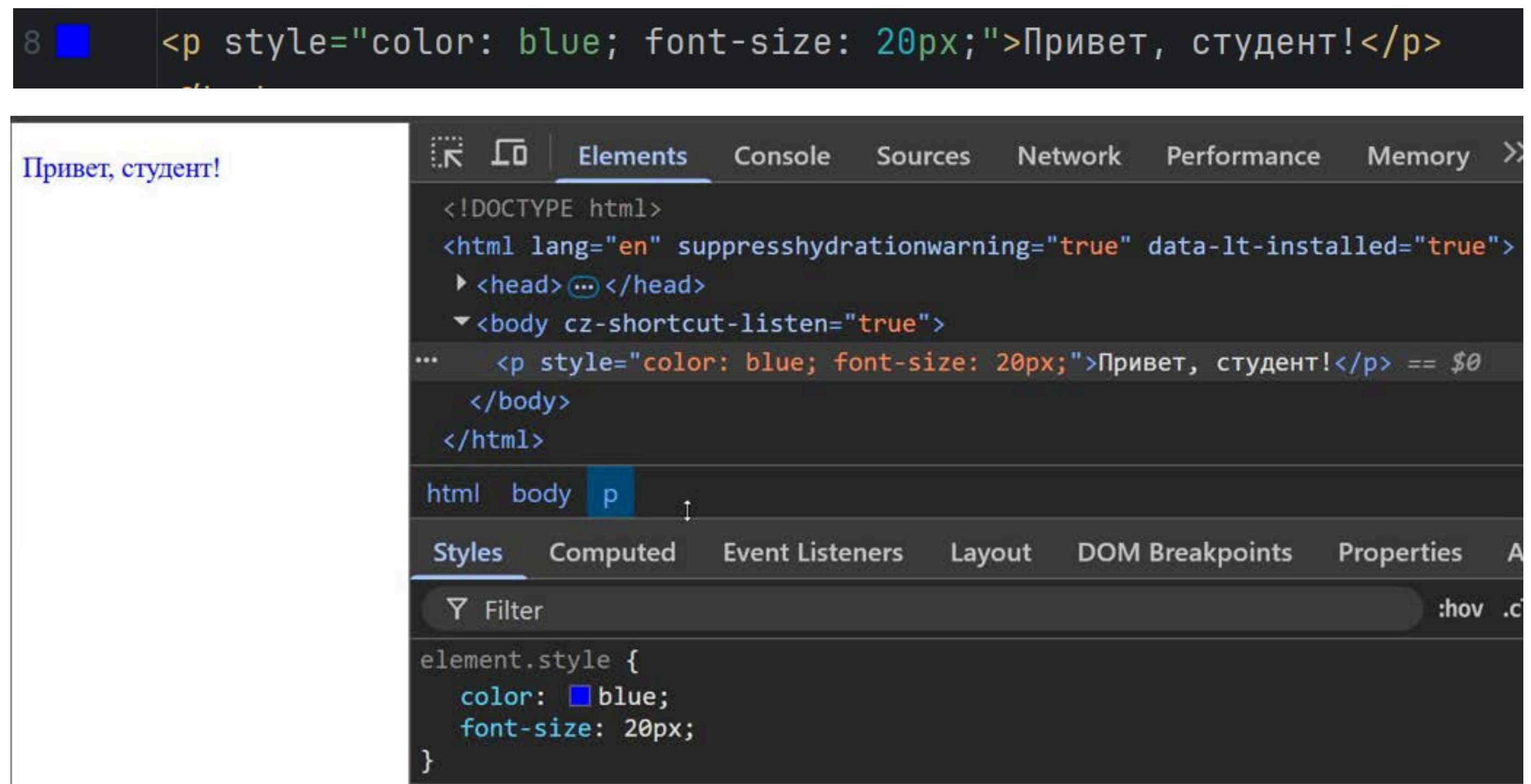
ЧТО ИЗУЧИМ СЕГОДНЯ?

1. Что такое каскадные стили (CSS)?
2. Варианты подключения CSS стилей к странице
3. Синтаксис CSS
4. Подключение шрифтов
5. Блочная модель

ЧТО ТАКОЕ **CSS**?

CSS (Cascading Style Sheets) — это язык, который отвечает за внешний вид веб-страниц.

Если HTML описывает структуру документа (какие есть заголовки, абзацы, изображения), то **CSS управляет тем, как всё это выглядит**: цвета, шрифты, размеры, расположение элементов.



ПОДКЛЮЧЕНИЕ CSS СТИЛЕЙ К СТРАНИЦЕ

Три основных способа:

а) Встроенные стили (`inline`)

Стили задаются прямо в атрибуте `style` у тега.

+ Быстро и удобно для теста.

- Непрактично для больших проектов.

```
8 <p style="color: blue; font-size: 20px;">Привет, студент!</p>
```

ПОДКЛЮЧЕНИЕ CSS СТИЛЕЙ К СТРАНИЦЕ

в) Внутренние стили (внутри `<style>`)

CSS пишется в секции `<head>` внутри тега `<style>`.

+ Удобно для небольших

- Стили работают только в этом файле.

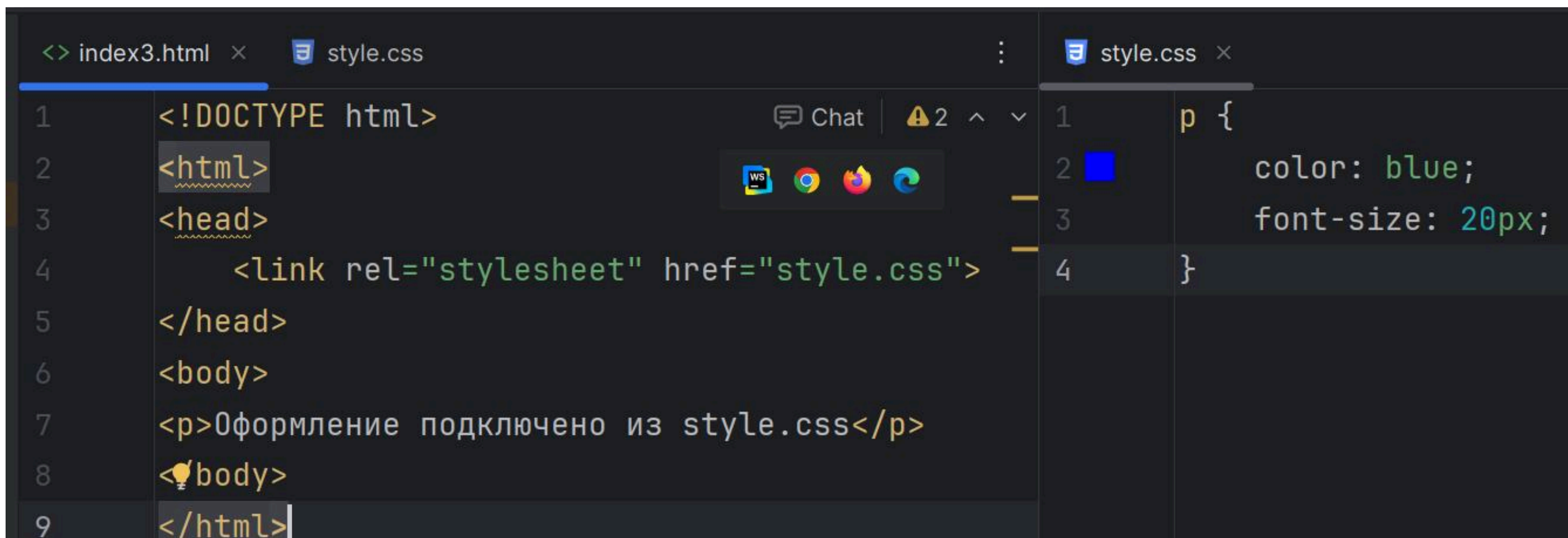
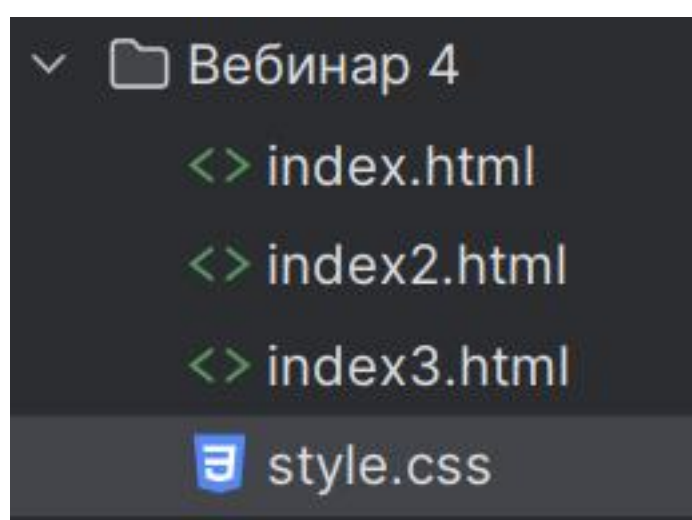
```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <style>
5          p {
6              color: green;
7              font-size: 18px;
8          }
9      </style>
10 </head>
11 <body>
12 <p>Зеленый текст</p>
13 </body>
14 </html>
```

ПОДКЛЮЧЕНИЕ CSS СТИЛЕЙ К СТРАНИЦЕ

с) Внешний файл CSS

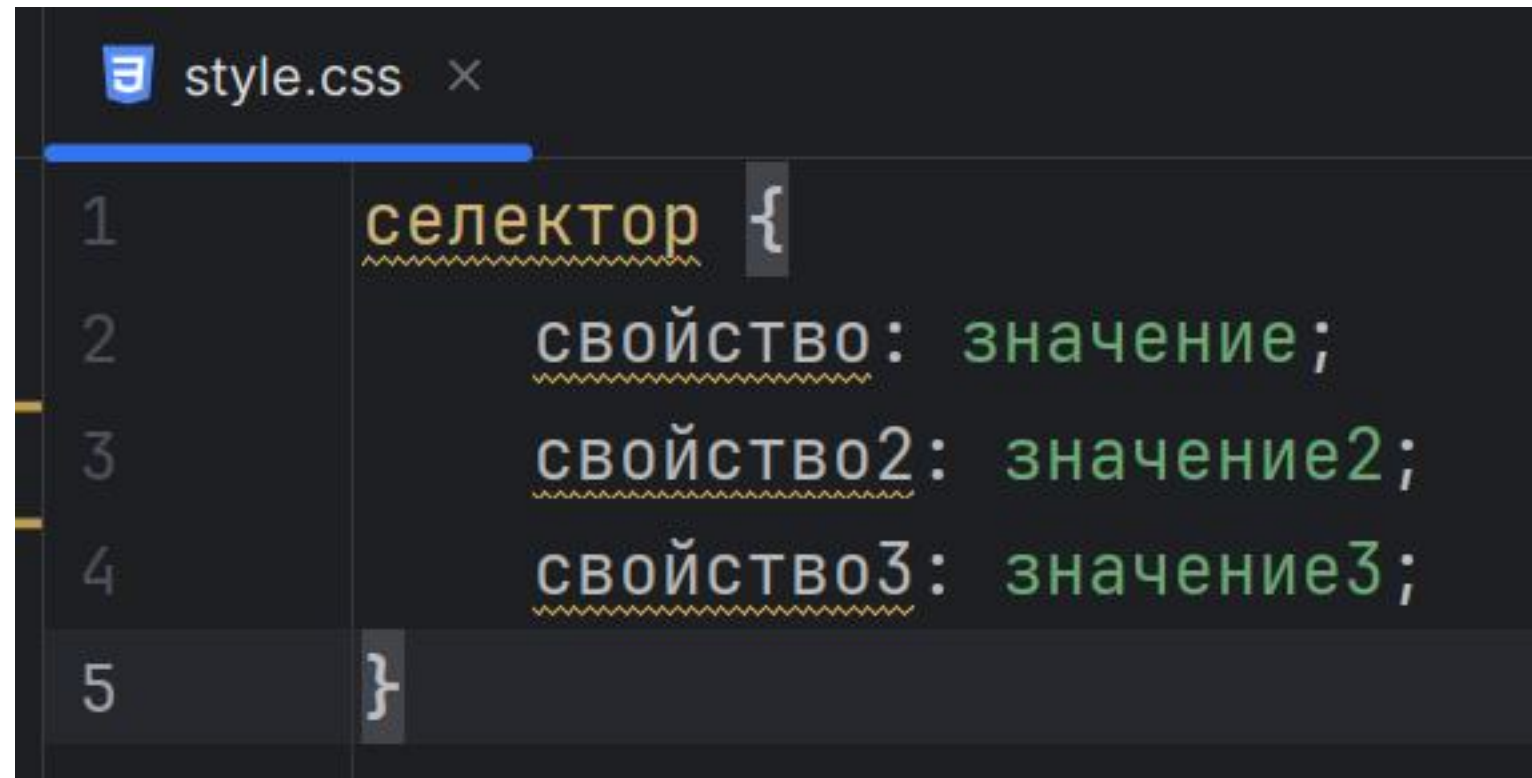
Стили выносятся в отдельный файл .css

- + Правильный способ для реальных проектов.
- + CSS можно подключить к нескольким страницам.
- Нужно больше файлов, чем при *inline*-стилях.



ОСНОВНЫЕ СВОЙСТВА CSS

CSS состоит из селектора и свойств



```
style.css x
1  селектор {
2      свойство: значение;
3      свойство2: значение2;
4      свойство3: значение3;
5  }
```

Примеры популярных свойств:

- **color** — цвет текста
- **font-size** — размер шрифта
- **font-family** — шрифт
- **background-color** — цвет фона
- **width / height** — ширина и высота
- **margin** — внешние отступы
- **padding** — внутренние отступы
- **border** — рамка

ОСНОВНЫЕ СВОЙСТВА CSS

Пример

⌕ ⌂ Elements Console

```
<!DOCTYPE html>
<html lang="en">
  > <head> ... </head>
  > <body>
...    <p>Пример </p> == $0
    </body>
  </html>
```

html body p

Styles Computed Event Listen

▼ Filter

```
element.style {
}

p {
  color: ■ white;
  background-color: ■ black;
  font-size: 22px;
  padding: ▶ 10px;
  border: ▶ 2px solid ■ red;
}
```


СИНТАКСИС CSS

1. СЕЛЕКТОРЫ (SELECTORS)

Селекторы — это «адреса» элементов HTML, к которым мы хотим применить стили.

Главные типы:

Селектор	Пример	Что делает
По тегу	<code>p { ... }</code>	Все <code><p></code> на странице
По классу	<code>.className { ... }</code>	Все элементы с классом <code>className</code>
По идентификатору	<code>#idName { ... }</code>	Один элемент с <code>id="idName"</code>
Группировка	<code>h1, h2, h3 { ... }</code>	Стили для нескольких тегов
Вложенность	<code>nav ul li a { ... }</code>	Стили только для ссылок в меню

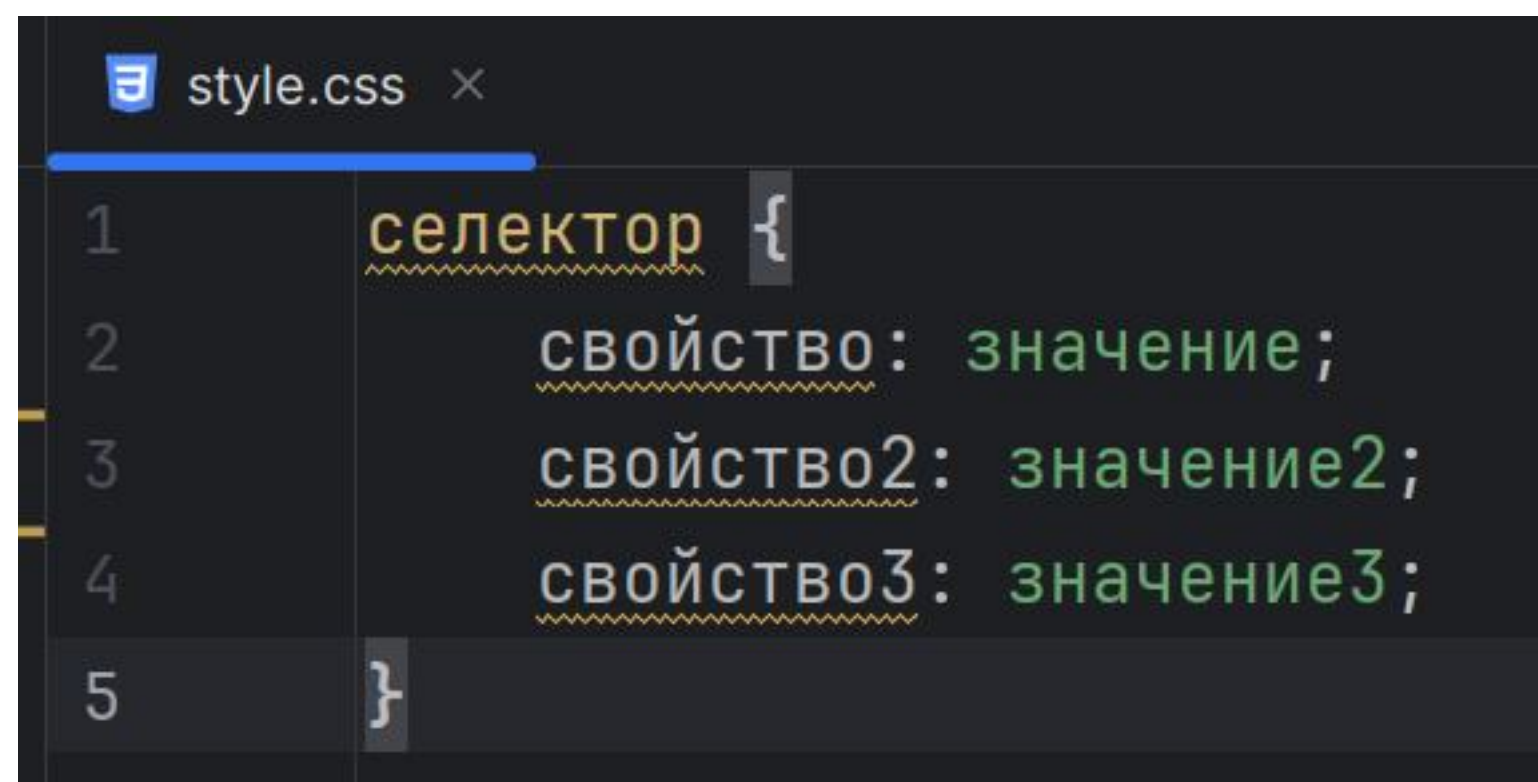
```
1  p {
2    color: blue; /* Все параграфы синие */
3  }
4
5  .note {
6    background-color: yellow; /* Элементы с классом note имеют
7                                жёлтый фон */
8  }
9
10 #mainTitle {
11   font-size: 32px; /* Элемент с id=mainTitle крупный */
12 }
```

СИНТАКСИС CSS

2. СВОЙСТВА (PROPERTIES)

Свойства определяют что именно мы меняем: цвет, размер, отступы, рамки.

Каждое свойство имеет значение:



```
1  селектор {
2      свойство: значение;
3      свойство2: значение2;
4      свойство3: значение3;
5  }
```

Примеры популярных свойств:

- **color** — цвет текста
- **font-size** — размер шрифта
- **font-family** — шрифт
- **background-color** — цвет фона
- **width / height** — ширина и высота
- **margin** — внешние отступы
- **padding** — внутренние отступы
- **border** — рамка

СИНТАКСИС CSS

3. ЗНАЧЕНИЯ (VALUES)

Значения свойств бывают разными:

- Цвета — red, #ff0000, rgb(255,0,0)
- Размеры — px, em, %
- Стили рамок — solid, dashed, dotted

```
p {  
    color: #333;           /* Цвет в HEX */  
    font-size: 18px;       /* Размер шрифта в пикселях */  
    line-height: 1.5;      /* Межстрочный интервал */  
    border-bottom: 2px dotted red; /* Нижняя пунктирная рамка */  
}
```


ПОДКЛЮЧЕНИЕ ШРИФТОВ В CSS

Шрифты — важная часть дизайна сайта. С их помощью текст становится читаемым и красивым.

В CSS есть **несколько способов** работать со шрифтами:

1. Системные (стандартные) шрифты

Это шрифты, которые уже установлены у пользователя на компьютере. Мы просто указываем их название в CSS.

```
p {  
    font-family: Arial, sans-serif; /* Используем Arial, если  
    нет — любой sans-serif */  
}
```

- Arial — конкретный шрифт
- sans-serif — семейство шрифтов без засечек (подстраховка, если Arial не найден)

Популярные стандартные семейства:

- serif (с засечками: Times New Roman)
- sans-serif (без засечек: Arial, Helvetica)
- monospace (моноширинные: Courier New)

ПОДКЛЮЧЕНИЕ ШРИФТОВ В CSS

2. Подключение через Google Fonts (или другой CDN)

Если хотим необычный шрифт, которого может не быть у пользователя, мы можем подключить его с сервера (например, Google Fonts).

Шаги:

1. Заходим на <https://fonts.google.com/>
2. Выбираем шрифт (например, Roboto)
3. Копируем ссылку `<link>` из Google Fonts в `<head>` нашего HTML

```
1  <head>
2      <link href="https://fonts.googleapis.com/css2?family=Roboto&display=swap" rel="stylesheet">
3      <style>
4          body {
5              font-family: 'Roboto', sans-serif; /* Подключенный
6              шрифт */
7          }
8      </style>
9  </head>
```

ПОДКЛЮЧЕНИЕ ШРИФТОВ В CSS

3. Подключение собственного (локального) шрифта через @font-face

Если у нас есть файл шрифта (например, MyFont.woff2), мы можем подключить его напрямую из проекта. важная часть дизайна сайта. С их помощью текст становится читаемым и красивым.

```
24  @font-face {
25      font-family: 'MyFont'; /* Имя шрифта, которым мы будем
                               пользоваться */
26      src: url('fonts/MyFont.woff2') format('woff2'),
27           url('fonts/MyFont.woff') format('woff'); /* Форматы шрифтов
                               */
28      font-weight: normal;
29      font-style: normal;
30  }
31
32  p {
33      font-family: 'MyFont', sans-serif; /* Применяем к параграфу
                               */
34  }
```

Форматы шрифтов:

- .woff2 — современный, лёгкий формат (лучше всего)
- .woff — старее, поддерживается почти везде
- .ttf, .otf — старые форматы, тоже можно подключать

Шрифты нужно положить в папку проекта (например, **fonts/**).

ПОДКЛЮЧЕНИЕ ШРИФТОВ В CSS

4. Импорт шрифта через @import в CSS

Это почти как <link> в HTML, только подключаем шрифт прямо из CSS.

Пример:

```
@import url('https://fonts.googleapis.com/css2?family=Montserrat  
&display=swap');  
  
body {  
    font-family: 'Montserrat', sans-serif;  
}
```

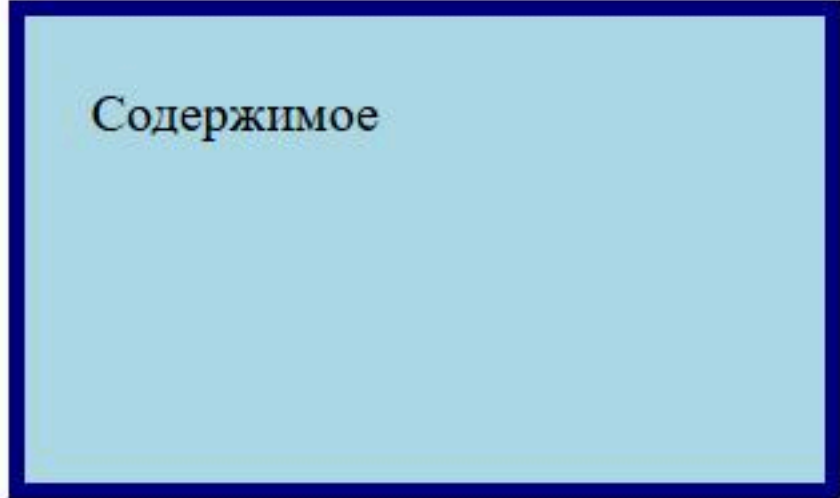
ПОДКЛЮЧЕНИЕ ШРИФТОВ В CSS

5. Практический пример

```
1 <!DOCTYPE html>
2 <html lang="ru">
3 <head>
4   <meta charset="UTF-8">
5   <title>Пример шрифтов</title>
6   <link href="https://fonts.googleapis
7     .com/css2?family=Roboto&display=swap" rel="stylesheet">
8     <!-- Подключение Google Fonts -->
9   <link rel="stylesheet" href="style5.css"> <!-- Подключение
10     своего CSS -->
11 </head>
12 <body>
13 <h1 class="google-font">Заголовок с Google Fonts</h1>
14 <p class="system-font">Параграф с системным шрифтом</p>
15 <p class="local-font">Параграф с локальным шрифтом</p>
16 </body>
17 </html>
```

```
1 /* Системный шрифт */
2 .system-font {
3   font-family: Arial, sans-serif; /* Стандартный */
4 }
5
6 /* Google Fonts */
7 .google-font {
8   font-family: 'Roboto', sans-serif; /* Google Fonts */
9 }
10
11 /* Локальный шрифт */
12 @font-face {
13   font-family: 'MyFont';
14   src: url('fonts/MyFont.woff2') format('woff2');
15 }
16 .local-font {
17   font-family: 'MyFont', sans-serif; /* Подключённый локально
18   */
19 }
```


БЛОЧНАЯ МОДЕЛЬ CSS



⌕ ↺ Elements Console Sources

<!DOCTYPE html>
<html lang="en">
 <head> ⋮ </head>
 <body>
 ⋮ <div class="box">Содержимое</div>
 </body>
</html>

html body div.box

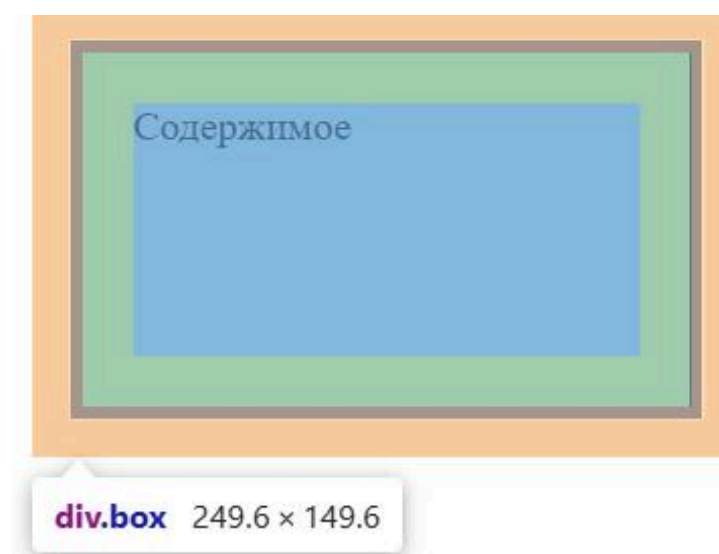
Styles Computed Event Listeners Lay

⌵ Filter

element.style {
}

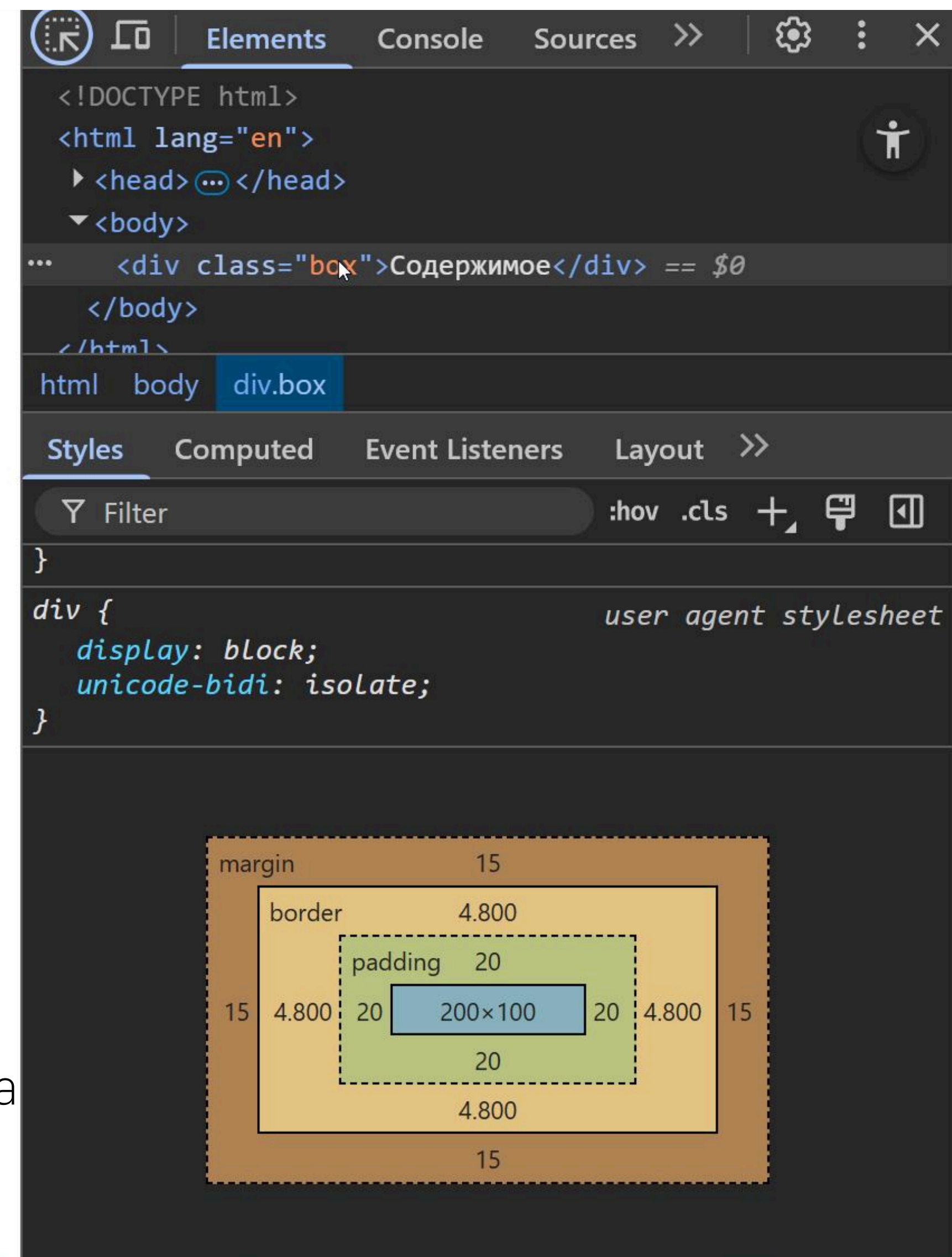
.box {
 width: 200px;
 height: 100px;
 background-color: ■ lightblue;
 border: ▶ 5px solid ■ navy;
 padding: ▶ 20px;
 margin: ▶ 15px;
}

БЛОЧНАЯ МОДЕЛЬ CSS



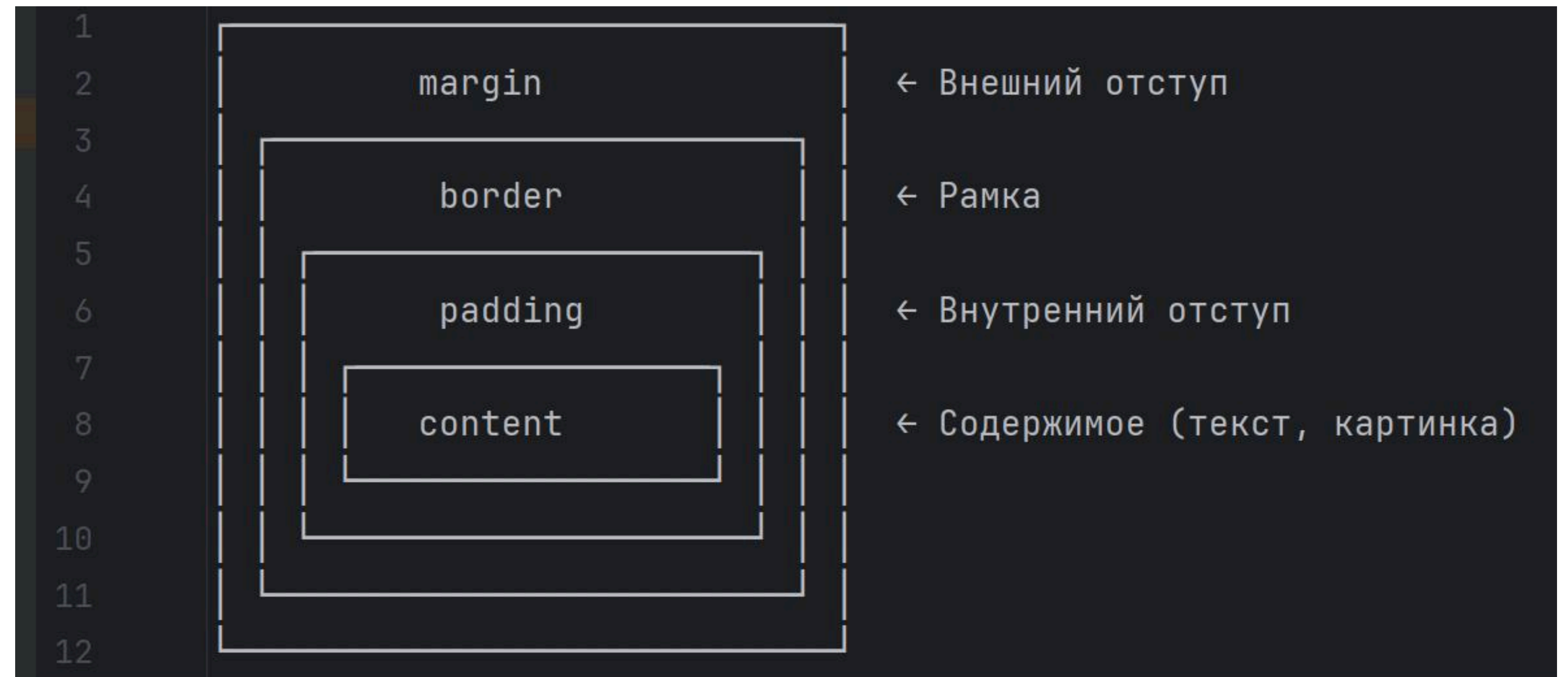
Разберём по слоям:

- **content** → текст "Содержимое!"
- **padding: 20px** → отодвигает текст от краев внутри блока
- **border: 5px solid navy** → рамка вокруг блока
- **margin: 15px** → расстояние от других элементов



БЛОЧНАЯ МОДЕЛЬ CSS

Каждый HTML-элемент можно представить как прямоугольную коробку (box). Она состоит из 4 частей:



МЕДИА ЗАПРОСЫ

МЕДИА ЗАПРОСЫ

Медиа-запросы (media queries) — это инструмент CSS, который позволяет адаптировать внешний вид сайта под разные устройства (телефон, планшет, компьютер). Благодаря им сайт становится отзывчивым (responsive).

```
@media (условие) {  
    /* Стили, которые применяются при выполнении условия */  
}
```

Пример

```
48      /* По умолчанию (для больших экранов) */  
49      body {  
50          font-size: 18px;  
51      }  
52  
53      /* Для экранов шириной меньше 600px */  
54      @media (max-width: 600px) {  
55          body {  
56              font-size: 14px; /* Уменьшаем текст на маленьких экранах  
57              */  
58          }  
59      }
```

МЕДИА ЗАПРОСЫ

Основные параметры для медиа-запросов

- width / max-width / min-width — ширина окна
- height / max-height / min-height — высота окна
- orientation: portrait — вертикальная ориентация
- orientation: landscape — горизонтальная ориентация

Примеры:

```
64  /* Применяем стиль, если ширина окна меньше 768px */
65  @media (max-width: 768px) {
66      .container {
67          flex-direction: column; /* Ставим элементы в колонку */
68      }
69  }
70
71  /* Применяем стиль, если ширина окна больше 1200px */
72  @media (min-width: 1200px) {
73      .container {
74          max-width: 1100px; /* Ограничиваем ширину контента */
75      }
76  }
77
78  /* Применяем стиль только для вертикального режима */
79  @media (orientation: portrait) {
80      body {
81          background-color: lightblue;
82      }
83  }
```

МЕДИА ЗАПРОСЫ

Как работают медиа-запросы

- Сначала CSS читается сверху вниз.
- Браузер применяет базовые стили, а потом проверяет условия медиа-запросов.
- Если условие верно — применяются стили внутри @media.

Это позволяет строить «**гибкий**» дизайн:

1. Пишем базовые стили (для больших экранов).
2. Добавляем медиа-запросы для меньших экранов.

ПРИМЕР

Иван Иванов

Обо мнеКонтактыПортфолио

Обо мне

Я создаю *современные сайты* и обучаю студентов веб-разработке. Люблю чистый код, понятные интерфейсы и адаптивный дизайн.

Контакты

Email: ivan@example.com
Телефон: +7 (900) 123-45-67

Портфолио

Сайт-визитка

Простой сайт-визитка для начинающих с использованием HTML, CSS и семантических тегов.

Интернет-магазин

Учебный проект с каталогом, корзиной и адаптивной версткой.

Интересный факт

Я веду блог про веб-разработку и делюсь учебными материалами.

© 2025 Иван Иванов


```
<!DOCTYPE html> <!-- Объявляем HTML5 -->
<html lang="ru"> <!-- Указываем язык документа -->

<head>
  <meta charset="UTF-8"> <!-- Кодировка -->
  <meta name="viewport" content="width=device-width, initial-scale=1.0"> <!-- Чтобы
сайт был адаптивным -->
  <meta name="description" content="Учебный пример HTML и CSS: семантика, шрифты,
селекторы, медиа-запросы">
  <title>Учебная страница</title>
  <link rel="stylesheet" href="style.css">
</head>

<body>

<!-- HEADER: Шапка страницы -->
<header>
  <div class="container">
    <h1>Иван Иванов</h1>
    <p class="highlight">Веб-разработчик</p>
    <nav>
      <ul>
        <li><a href="#about">Обо мне</a></li>
        <li><a href="#contacts">Контакты</a></li>
        <li><a href="#portfolio">Портфолио</a></li>
      </ul>
    </nav>
  </div>
</header>

<!-- MAIN: Основной контент -->
<main class="container">

  <!-- ОБО МНЕ -->
  <section id="about">
    <h2>Обо мне</h2>
    <article>
      <p>Я создаю <span class="highlight">современные сайты</span> и обучаю
студентов веб-разработке.
      Люблю чистый код, понятные интерфейсы и адаптивный дизайн.</p>
    </article>
  </section>

  <!-- КОНТАКТЫ -->
  <section id="contacts">
    <h2>Контакты</h2>
    <ul>
      <li>Email: <a href="mailto:ivan@example.com">ivan@example.com</a></li>
      <li>Телефон: +7 (900) 123-45-67</li>
    </ul>
  </section>

  <!-- ПОРТФОЛИО -->
  <section id="portfolio">
    <h2>Портфолио</h2>

    <article>
      <h3>Сайт-визитка</h3>
      <p>Простой сайт-визитка для начинающих с использованием HTML, CSS и
семантических тегов.</p>
    </article>

    <article>
      <h3>Интернет-магазин</h3>
      <p>Учебный проект с каталогом, корзиной и адаптивной версткой.</p>
    </article>
  </section>

  <!-- ASIDE: Боковая панель -->
  <aside>
    <h2>Интересный факт</h2>
    <p>Я веду блог про веб-разработку и делюсь учебными материалами.</p>
  </aside>

</main>

<!-- FOOTER: Подвал -->
<footer>
  <p>© 2025 Иван Иванов</p>
</footer>

</body>
</html>
```

```
/* Подключение шрифта */
@import url('https://fonts.googleapis.com/css2?family=Roboto:wght@400;
700&display=swap');

/* Сброс отступов */
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

/* Базовые стили для всего документа */
body {
  font-family: 'Roboto', sans-serif; /* Используем подключенный шрифт */
  line-height: 1.6;
  color: #333;
}

/* Класс .container для основных блоков: центрирует и ограничивает ширину */
.container {
  max-width: 1100px;
  margin: 0 auto;
  padding: 20px;
}

/* HEADER: Шапка сайта */
header {
  background-color: #3498db;
  color: white;
  padding: 20px 0;
  text-align: center;
}

/* Стили навигации */
nav ul {
  list-style: none;
  display: flex;
  justify-content: center;
  gap: 20px;
}

nav a {
  color: white;
  text-decoration: none;
  font-weight: bold;
}

nav a:hover {
  text-decoration: underline;
}

/* Основной контент */
main {
  padding: 20px 0;
}

section {
  margin-bottom: 40px;
}

h2 {
  margin-bottom: 10px;
  font-size: 24px;
  color: #2c3e50;
}

article {
  background-color: #f9f9f9;
  border-left: 4px solid #3498db;
  padding: 15px;
  margin-bottom: 15px;
}

/* ASIDE: Боковая панель */
aside {
  background-color: #f1f1f1;
  padding: 20px;
  margin-top: 20px;
  border-radius: 5px;
}

/* FOOTER: Подвал */
footer {
  background-color: #2c3e50;
  color: white;
  text-align: center;
  padding: 15px 0;
}

/* Класс .highlight — выделение текста курсивом */
.highlight {
  font-style: italic;
  color: #3498db;
}

/* Пример CSS-селектора по id */
#contacts ul {
  list-style: none;
}

#contacts li {
  padding: 5px 0;
}

/* Медиа-запрос: перестраиваем навигацию и aside */
@media (max-width: 600px) {
  nav ul {
    flex-direction: column;
    gap: 10px;
  }

  aside {
    background-color: #dbeafe; /* Меняем цвет боковой панели на маленьких экранах */
  }
}
```


УПРАЖНЕНИЕ 3: СЕМАНТИЧЕСКАЯ ВИЗИТКА

Цель: Доработать HTML-страницу (визитку), используя классы и блочную модель CSS вместо встроенных `style`. Научиться структурировать стили через классы и применять CSS-свойства для управления внешним видом страницы.

Требования:

- Использовать только классы для задания стилей.
- Вынести все стили в тег `<style>` внутри `<head>` или в подключаемый файл `style.css`.
- Применить базовые CSS-свойства:
 - `margin`, `padding`, `border`, `background`, `width`, `max-width`, `box-sizing`, `text-align`, `display`, `flex`, `border-radius`, `box-shadow`, `font-family`.
- Все блоки должны быть оформлены согласно блочной модели — с отступами, границами и фоном.
- Форма и расписание должны быть аккуратно оформлены и читабельны.