



# Frontender[1.0]. Вопросы по первому модулю HTML, CSS, SASS, SCSS, NPM, GIT, SVG, БЭМ.

🔗 YouTube	<a href="https://youtu.be/RCDnSqW4TLM">https://youtu.be/RCDnSqW4TLM</a>
🔗 Telegram	<a href="https://t.me/Dmitry_Kolotilshikov">https://t.me/Dmitry_Kolotilshikov</a>
# Номер урока	97



## 1. Что такое frontend и backend?

**Фронтенд (frontend)** - это клиентская часть продукта (интерфейс, с которым взаимодействует пользователь). В случае с сайтом ее формирует и выводит на экран браузер, который работает с **HTML, CSS и JavaScript**.

### Backend

— это внутренняя часть продукта, которая находится на сервере и скрыта от пользователей. Для ее разработки могут использоваться самые разные языки, например, Python, PHP, Java, JavaScript, C# и тп



## 2. Что такое HTML?

**HTML (Hypertext Markup Language)** - это язык разметки, который используется для создания веб-страниц. Он описывает структуру документа и ее содержимое, используя теги, такие как `<html>`, `<head>`, `<body>`, `<p>` и многие другие. Эти теги определяют, как документ будет отображаться в браузере.



### 3. Что такое CSS?

**CSS (Cascading Style Sheets - Каскадные таблицы стилей)** - это язык, который используется для определения внешнего вида веб-страниц, такого как цвет, шрифт и расположение элементов.

**CSS** позволяет разделить контент веб-страницы и его дизайн, что улучшает читаемость и поддержку кода.



### 4. Какие бывают теги (по структуре, все перечислять не нужно) ?

Теги бывают:

- **парные** и не **парные** (<></> и </>)
- **строчные** и **блочные**



### 5. Что такое атрибуты и как они используются в HTML ?

Например, атрибут

`src` у тега `img` определяет путь к изображению, атрибут `href` у тега `a` определяет ссылку, на которую будет переходить пользователь. Атрибуты всегда находятся внутри тега и имеют формат `имя="значение"`

Например: `<a href="https://example.com">ссылка</a>`



## 6. Что такое селекторы в HTML ?

**Селекторы в HTML** - это механизм, используемый для выбора элементов на странице для применения к ним стилей и скриптов. Существует несколько типов селекторов, каждый из которых используется для выбора элементов на странице по-разному.

- **Селекторы по типу тега:** выбирают элементы на странице по их типу тега. Например, `div` селектор выберет все элементы `<div>` на странице.
- **Селекторы по классу:** выбирают элементы на странице по их классу. Например, `.example` селектор выберет все элементы с классом "example".
- **Селекторы по ID:** выбирают элементы на странице по их ID. Например, `#example` селектор выберет элемент с ID "example".
- **Селекторы по атрибуту:** выбирают элементы на странице по их атрибутам и значениям. Например, `[type="button"]` селектор выберет все элементы с атрибутом "type" со значением "button".
- **Селекторы по состоянию:** выбирают элементы на странице в зависимости от их состояния, например, `.button:hover`.
- **Селекторы по соседним элементам:** выбирают элементы на странице в зависимости от их соседних элементов.

`.box + .text` или `.box ~ .text`

- **Составные селекторы:** используются для выбора элементов на основе нескольких различных критериев, например, выбор всех элементов с классом "example" внутри элемента с классом "container". `.container > example`



## 7. Что такое GIT ?

**GIT** - это система контроля версий, которая помогает командам разработчиков отслеживать изменения в исходном коде и работать вместе над проектом. Она позволяет хранить разные версии проекта и возвращаться к предыдущим версиям при необходимости.



## 8. В чем разница margin и padding ?

**Margin** и **padding** - это свойства CSS, которые используются для создания пространства вокруг элементов веб-страницы.

Основное различие между ними заключается в том, что

**margin** создает пространство вне границы элемента, а **padding** создает пространство внутри границы элемента.



## 9. Что такое Margin collapsing ?

**Margin collapsing** это явление в CSS, когда два вертикально смежных элемента имеют отступы, которые соприкасаются друг с другом. Результатом является то, что расстояние между ними равно большему из двух отступов, а не их сумме. Это может усложнить расчеты макета.



#### 10. Опишите свойство `box-sizing` ?

Свойство

**`box-sizing`** в CSS определяет, как вычисляются ширина и высота элемента: должны ли они включать **`padding`** и **`border`** или нет. Есть два основных значения для этого свойства: **`content-box`** и **`border-box`**



#### 11. Что такое `Desktop First` и `Mobile First` ?

**`Desktop first`** и **`Mobile first`** - это два подхода к разработке веб-сайтов, основанные на том, какой тип устройства должен иметь более высокий приоритет при проектировании.



## 12. Назовите несколько семантических тегов ?

**<header>** представляет собой вводный контент, обычно группу вводных или навигационных средств. Он может содержать другие элементы-заголовки, а также логотип, форму поиска, имя автора и другие элементы. Может быть несколько на странице, могут быть вложенными.

**<main>** предназначен для основного контента (содержимого) **<body>** документа (страницы).

**<section>** - Создание раздела сайта. Представляет собой автономный раздел – который не может быть представлен более точным по семантике элементом – внутри HTML-документа. Как правило, но не всегда, разделы имеют заголовок.

**<nav>** определяет отдельную секцию документа, назначение которой обозначение ссылок навигации. Не обязательно все ссылки должны быть обернуты в **<nav>**. **<nav>** следует использовать лишь для главных навигационных блоков. Например, **<footer>** часто содержит список ссылок, которые не стоит оборачивать в **<nav>**

**<footer>** представляет собой нижний колонтитул (футер, подвал) Футер обычно содержит информацию об авторе раздела, информацию об авторском праве или ссылки на связанные документы.

**<article>** представляет самостоятельную часть документа, страницы, приложения или сайта, предназначенную для независимого распространения или повторного использования.

**<address>** задаёт контактные данные для ближайшего родительского **<article>** или **<body>**; В последнем случае применяется ко всему документу, визуально выделяется курсивом.

**<aside>** представляет собой часть документа, чьё содержимое только косвенно связано с основным содержимым документа. Чаще всего представлен в виде боковой панели, сносок или меток.

**<figure>** (Иллюстрация с необязательной подписью) представляет самостоятельный контент, часто с подписью (заголовком), которая указывается с помощью элемента (**<figcaption>**).

**<details>** Нативный "аккордеон". Работает в связке с тегом **summary**. HTML-элемент **<details>** используется для раскрытия скрытой (дополнительной) информации.

**<form>** представляет (собой) раздел документа, содержащий интерактивные элементы управления, которые позволяют пользователю отправлять информацию на веб-сервер.

**<dialog>** определяет диалоговое окно или другой интерактивный элемент, например часто используется для модальных окон.

**<audio>** и **<video>**



### 13. Что такое Pixel perfect ?

**Pixel perfect** - метод верстки веб-страниц, при котором дизайн максимально точно соответствует готовой странице на уровне каждого пикселя.



### 14. Для чего служит БЭМ ?

**БЭМ (Блок, Элемент, Модификатор)** - это разработанная Яндексом методология именования классов в **HTML** и **CSS**, которая помогает **организовать код** и **обеспечить его масштабируемость** и **повторное использование**. БЭМ позволяет описывать компоненты страницы в терминах их составляющих частей, которые могут быть легко использованы и изменены в других частях проекта.



### 15. Что такое препроцессоры ?

**Препроцессор** - это **CSS с СУПЕРСИЛОЙ**

**SASS (Syntactically Awesome Style Sheets)** и **SCSS (Sassy CSS)** - это языки препроцессоров **CSS**, которые расширяют возможности обычного **CSS** и делают его более мощным и гибким.

На сегодняшний день SASS это уже как мини-язык программирование, который обладает широким набором возможностей.



### 16. Расскажите про HTML Accessibility ?

**HTML Accessibility**, также известный как "**веб-доступность**", это процесс создания веб-страниц, которые могут быть использованы и поняты людьми с различными формами инвалидности и ограничений, включая низкое зрение, слабослышащих, людей с ограниченными возможностями движения и т.д.



### 17. Расскажите про SEO ?

**SEO (Search Engine Optimization)** - это процесс оптимизации веб-сайта для улучшения его видимости и позиций в поисковых системах, таких как **Google, Bing, Yahoo** и других. Цель **SEO** заключается в привлечении органического (неоплачиваемого) трафика на сайт путем повышения его релевантности и авторитетности в глазах поисковых систем.



## 18. Псевдоклассы и псевдоэлементы в CSS ?

**Псевдоклассы** и **псевдоэлементы** в CSS позволяют изменять стили элементов на основе их состояния или позиции в документе. Это очень удобно для создания интерактивных элементов, таких как кнопки, ссылки и формы, которые меняют свой вид при наведении курсора мыши, при фокусировке или при заполнении.



## 19. Как создать адаптивный (респонсив) дизайн в CSS ?

Правило @Media

```
@media (max-width: 768px) {  
  body {  
    background-color: blue;  
  }  
}
```



## 20. Способы работы с анимацией в CSS ?

**Правило:** @Keyframes

**Свойства:**

Transform  
Transition  
Animation



## 21. SVG спрайты и их преимущества ?

**SVG спрайты** - это техника использования нескольких SVG-изображений, объединенных в один файл. Вместо того, чтобы загружать множество отдельных файлов, используется один файл, который содержит множество различных изображений. Каждое изображение в спрайте имеет свой уникальный идентификатор (ID), по которому оно может быть выбрано и использовано на веб-странице.

Преимущества использования **SVG спрайтов** включают:

1. Уменьшение числа запросов к серверу. Вместо загрузки множества отдельных файлов, используется один файл, что уменьшает количество запросов к серверу и снижает время загрузки страницы.
2. Уменьшение размера файла. SVG спрайты обычно имеют меньший размер, чем множество отдельных файлов, что уменьшает количество передаваемых данных и ускоряет загрузку страницы.
3. Гибкость и масштабируемость. Каждое изображение в спрайте может быть использовано многократно и масштабировано без потери качества.
4. Управление стилями и анимациями. Каждое изображение в спрайте может быть стилизовано и анимировано независимо от других изображений в спрайте.





## 22. Flexbox ?

**CSS Flexbox (Flexible Box)** - это технология верстки веб-страниц, которая позволяет создавать гибкие и адаптивные макеты без использования сложных таблиц и позиционирования.

**Flexbox** представляет собой контейнер, который может содержать один или несколько элементов, называемых "**флекс-элементами**". С помощью свойств CSS, определенных для контейнера и флекс-элементов, можно легко управлять расположением, порядком, размером и выравниванием элементов внутри контейнера.



## 23. Приоритеты селекторов ?

**Приоритетность селекторов** - это механизм, используемый для определения, какой стиль или скрипт должен быть применен к элементу на странице, когда существуют несколько стилей или скриптов, которые могут быть применены к одному и тому же элементу.

**Приоритетность селектора** определяется его специфичностью. Чем более специфичный селектор, тем выше его приоритет.

Приоритетность селекторов может быть также изменена с помощью **!important** правила, которое может быть использовано для установки максимального приоритета для стиля или скрипта.



## 24. Что такое <iframe> ?

Тег `<iframe>` означает «**встроенный** фрейм» и используется для встраивания **документов в текущий HTML-документ**. Например, с помощью тега можно добавить видео, карты, посты из социальных сетей, рекламу и другой тип внешнего контента.

Тег `<iframe>` не считается семантическим, так как не придаёт никакого конкретного значения содержимому.

### Ограничения

- Содержимое `<iframe>` должно быть из того же домена, что и родительский документ, или сервер должен специально разрешить отображение этого содержимого.
- Использование `<iframe>` может замедлить время загрузки страницы.

### Нюансы

- `<iframe>` может использоваться для атак `clickjacking`, чтобы обманом заставить пользователя нажать на кнопку или ссылку, выполняющую вредоносное действие.
- `<iframe>` может вызывать проблемы со скринридерами, потому что такие программы не всегда способны прочитать содержимое `<iframe>`.





## 25. Для чего используется Favicon и как его установить?

**Фавикон (favicon)** - это значок для избранного (**FAVoritesICON**), который обычно является уменьшенной версией фирменного логотипа или создан с целью напоминания о логотипе или товарном знаке.

**Фавикон** отображается **в адресной строке браузера** и на **закладках** веб-сайта. **Фавикон** помогает пользователям быстро идентифицировать сайт и отличить его от других вкладок в браузере.

```
<link rel="icon" type="image/
x-icon" href="/favicon.ico">
<link rel="icon" type="image/
png" href="/favicon.png">
<link rel="icon" type="image/
gif" href="/favicon.gif">
<link rel="icon" type="image/
svg" href="/favicon.svg">
<link rel="icon" type="image/
webp" href="/favicon.webp">
```



## 26. Функции в CSS ?

**CSS функции** - это встроенные функции, которые могут использоваться в **CSS** для выполнения различных операций и преобразований значений. Они предоставляют возможность изменять свойства элементов на основе динамических значений или применять определенные вычисления к свойствам CSS.

Некоторые из наиболее распространенных CSS функций включают:

1. `rgb()`, `rgba()`, `hsl()`, `hsla()` - функции для определения цветов в формате RGB или HSL. Например, `rgb(255, 0, 0)` задает красный цвет.
2. `calc()` - функция для выполнения арифметических вычислений непосредственно в CSS. Она позволяет комбинировать различные единицы измерения и математические операции. Например, `width: calc(100% - 20px)`.
3. `var()` - функция для определения пользовательских CSS переменных. Позволяет использовать именованные переменные для хранения значений и использовать их в различных частях CSS. Например, `color: var(--primary-color)`.
4. `url()` - функция для определения пути к изображению или другому ресурсу. Например, `background-image: url("image.jpg")`.
5. `translate()`, `rotate()`, `scale()` - функции для преобразования и трансформации элементов. Они могут использоваться совместно со свойством `transform`, чтобы изменить положение, поворот или масштаб элемента.
6. `linear-gradient()`, `radial-gradient()` - функции для создания градиентов. Они позволяют создавать плавные переходы между несколькими цветами.
7. `attr()` - функция для извлечения значения атрибута HTML и использования его в CSS. Например, `content: attr(data-text)`.



## 27. Что такое позиционирование (positioning) в CSS и какие основные значения свойства position вы знаете ?

Свойство `position` в CSS определяет, как элемент позиционируется на веб-странице. Существует несколько значений для этого свойства:

- `static` : значение по умолчанию для большинства элементов. Элемент располагается в потоке документа.
- `relative` : элемент также располагается в потоке документа, но его позиция может быть сдвинута относительно его исходной позиции.
- `absolute` : элемент вынимается из потока документа и его позиция определяется относительно ближайшего родительского элемента с позиционированием, который не имеет значение `static` . Если такой родительский элемент не найден, позиция элемента будет относительно самого окна браузера.
- `fixed` : элемент также вынимается из потока документа и его позиция фиксируется относительно окна браузера. Элемент остается на месте даже при прокручивании страницы.
- `sticky` : элемент ведет себя как `relative` до тех пор, пока он не пересекает определенную точку на экране, после чего он становится фиксированным и прилипает к месту.

Свойства

`top`, `bottom`, `left` и `right` используются для определения позиции элемента.



## 28. Как вставить изображение в HTML-страницу ?

1. Тег `<img>`
2. Тег `<picture>`
3. Свойство `background`



## 29. Что такое box model ?

Блочная модель в CSS определяет, как элементы HTML будут отображаться в браузере. Каждый элемент HTML является блоком, имеющим размер, границы, поля и заполнение. Все эти атрибуты вместе образуют блок-модель элемента.

Основные компоненты блочной модели в CSS включают:

- **Content (контент)**: Размер содержимого элемента, определяемый шириной и высотой.
- **Padding (поля)**: Пространство между содержимым и границей элемента.
- **Border (граница)**: Линия, окружающая содержимое и поля элемента.
- **Margin (заполнение)**: Пространство между границей элемента и соседними элементами.

Зная эти компоненты, можно точно определить размеры и расположение элементов на странице. Это основа для создания адаптивных и отзывчивых веб-сайтов.



### 30. Как с помощью тега `img` можно предусмотреть загрузку нескольких изображений ?

```

```



### 31. Что такое DOCTYPE в HTML и зачем он нужен ?

**!DOCTYPE html** - Информация для браузера о типе (версии) HTML документа.

Сообщает браузеру, какую версию HTML или XHTML вы используете. Это помогает браузеру правильно отображать страницу.



### 32. Как обеспечить кроссбраузерную совместимость в HTML ?

Если вкратце, то **Вендорные префиксы** нужны для кроссбраузерности.

**Вендорные префиксы** — это приставки перед свойствами, селекторами, функциями или другими сущностями в CSS, позволяющие браузерам внедрять экспериментальные фичи до того, как они будут полностью стандартизированы и готовы для использования.

Основные браузеры используют следующие префиксы:

- **-webkit-** — Safari, Chrome, Opera 15+ и другие браузеры на основе движка WebKit или Blink.
- **-moz-** — Firefox и браузеры на движке Gecko.
- **-o-** — Opera 12 и раньше, на движке Presto.
- **-ms-** — Internet Explorer и старый Microsoft Edge 12–18.



### 33. Что такое метатеги и как они используются в HTML ?

**Мета-теги** являются частью HTML-кода и предоставляют информацию о странице для поисковых систем. Они помогают определить заголовок страницы, описание, ключевые слова и другие важные атрибуты.



#### 34. Как создать выпадающий список (выпадающее меню) в HTML ?

```
<label for="select">Choose an option:</label>
<select id="select">
  <option value="">Please choose an option</option>
  <option value="one">One</option>
  <option value="two">Two</option>
  <option value="three">Three</option>
</select>
```



#### 35. Как создать нативный аккордион в HTML ?

```
<details>
  <summary>Details</summary>
  Something small enough to escape casual notice.
</details>
```



#### 36. Как скрыть элемент на странице с помощью CSS ?

1. display: none;
2. visibility: hidden;
3. opacity: 0;



#### 37. В чем отличие display: none и visibility: hidden ?

**display: none** - элемент полностью исчезает с веб-страницы, не оставляя и следа.

**visibility: hidden** убирает элемент из вида, при этом не затрагивая поток html-документа, оставляя пустое пространство в том месте, где данный элемент должен был находиться.

Для абсолютно позиционированных элементов visibility: hidden и display: none ведут себя одинаково.



#### 38. Как центрировать элементы по горизонтали и вертикали в CSS ?

1. text-align: center;
2. margin: 0 auto;
3. line-height: высота блока;
4. position: absolute, relative, fixed, sticky;
5. float: left; (на сегодняшний день не использовать)
6. display: flex; (предпочтительно)
7. display: grid;



### 39. Какие глобальные атрибуты есть в HTML ?

Популярные глобальные атрибуты

1. class
2. contenteditable
3. data-\*
4. hidden
5. id
6. lang
7. spellcheck
8. style
9. tabindex
10. title



### 40. Есть ли у HTML элементов свои дефолтные стили ?

Да, у многих элементов есть свои дефолтные стили, также стили зависят от браузера. Например одна кнопка в разных браузерах может выглядеть по-разному.



### 41. Для чего нужна вкладка `computed` в dev tools браузера ?

Вкладка «**Computed**» – полезный инструмент, который поможет понять, какое именно свойство CSS применяется к элементу. Обычно это хорошее место для начала понимания структуры элемента на странице. Также в этой вкладке можно посмотреть абсолютно любой стиль на любом элементе.



### 42. В чем отличия DOM и Accessibility tree в браузере ?

**DOM** и **Accessibility tree** используют одни и те же данные, но они немного отличаются. DOM предоставляет все данные, весь объект, вся модель.

**Accessibility tree** показывает только данные доступности. Простейший способ описать это – сказать, что дерево доступности – это, по сути, фильтр, который показывает нам только информацию, связанную с доступностью, в DOM.



### 43. Accessibility (доступность) vs Usability (удобство использования) ?

**Accessibility** (Доступность) - сайт спроектирован в соответствии со стандартами по accessibility, доступ к контенту возможен каждому.

**Usability** (Удобство использования) - Пользователь может осуществлять навигацию без проблем, дизайн помогает а не мешает людям выполнять необходимые действия, интуитивность использования.



#### 44. Что такое NPM ?

**NPM** (аббр. **node package manager**) — это стандартный менеджер пакетов, автоматически устанавливающийся вместе с **Node.js**. Он используется для скачивания пакетов из облачного сервера **npm**, либо для загрузки пакетов на эти сервера.