**Словарь frontend-разработчицы**

* **Первая неделя обучения**

**Тег** — это специальное зарезервированное слово, заключенное в угловые скобки (например, <body>). Кратко говоря: «Тег – элемент HTML», все, что внутри тега, называют содержанием элемента.

**HTML-атрибуты** это специальные слова, которые управляют поведением HTML-элемента. Они добавляют дополнительную функциональность, либо меняют поведение элемента по умолчанию.

**директория** = папка = folder

**родитель** (родительский элемент) тот тег, внутрь которого вложен другой. На картинке выше родительский тег — <a>

**ребенок** — тег, который вложен внутрь другого тега. На картинке выше тег <img> ребенок по отношению к тегу <a>. Важно понимать, что ребенком считается только тот тег, который располагается на первом уровне вложенности. На изображении ниже тег <ul> является родителем только для трех тегов <li>, для тега <a> тег <ul> не является родителем. Это немного похоже на матрешку 😄

**Фавикон** — иконка (обычно логотип, или какая-то узнаваемая картинка) которая стоит перед названием сайта во вкладке браузера или перед названием сайта в выдаче поисковика

**Легаси код** — Устаревший код. код, написанный на старой версии языка или с использованием устаревшего фреймворка; код, который не документировался, о нём недостаточно информации; код, в котором нет тестов. Его трудно поддерживать и развивать, потому что программист не знает, сломает ли он что-то в процессе или нет. Бррррр, жуткая история 😬

# ****Стандарты оформления HTML и CSS кода****

### ****Общие правила оформления кода****

* **Используй два пробела для отступа**. Применяй для отступов пробелы, не используй табуляцию или сочетания пробелов и табуляции.

<body> <div> <h1>Привет, мир!</h1> <p>Это пример отступов с использованием двух пробелов.</p> <p>Вместо табуляции или сочетания пробелов и табуляции</p> </div> </body>

* **Применяй кодировку UTF-8.** Применение кодировки UTF-8 является рекомендацией для работы с текстом в HTML и других языках программирования. UTF-8 (Unicode Transformation Format-8) является универсальной кодировкой, которая поддерживает широкий спектр символов и позволяет представлять текст на разных языках и символьных наборах.

Чтобы указать кодировку UTF-8 в HTML, ты можешь использовать следующий тег в разделе \*\*<\*\*head\*\*>\*\* вашего HTML-документа:

<meta charset="UTF-8">

Этот тег сообщает браузеру, что текст на странице должен быть интерпретирован с использованием кодировки UTF-8. Указывая кодировку в вашем HTML-коде, ты гарантируешь правильное отображение и обработку текста на разных языках и символьных наборах.

Вот пример использования тега <meta charset="UTF-8"> в HTML-документе:

<!DOCTYPE html> <html> <head> <meta charset="UTF-8"> <title>Моя веб-страница</title> </head> <body> <h1>Привет, мир!</h1> <p>Это моя веб-страница с кодировкой UTF-8.</p> </body> </html>

* **Используй только строчные символы при написании кода**. Это включает именование элементов HTML, атрибутов, значений атрибутов и других текстовых частей кода.

Использование только строчных символов имеет несколько преимуществ:

* 1. **Согласованность**. При использовании только строчных символов обеспечивается единообразие в коде. Весь код выглядит одинаково и более легко читается.
  2. **Совместимость**. Строчные символы считаются более надежными при обработке различными системами, включая разные операционные системы и веб-серверы.
  3. **Удобство ввода**. Написание кода с использованием только строчных символов может быть более удобным и быстрым.

Примеры правильного использования строчных символов в HTML коде:

<!-- Не рекомендуется --> <H1>Заголовок</H1> <!-- Рекомендуется --> <h1>Заголовок</h1> <div class="container"> <p>это абзац текста</p> <img src="изображение.jpg" alt="описание изображения"> </div>

* **Удаляй пробелы в конце строк**

<!-- Не рекомендуется --> <p>Пробел в конце строки? </p> <!-- Рекомендуется --> <p>Пробелы не нужны</p>

* **Комментируй свой код**

Комментарии - это важный аспект при написании кода. Вот пример использования комментариев в HTML коде:

<!-- Это комментарий, описывающий назначение данного блока кода --> <div class="container"> <!-- Этот div содержит основное содержимое страницы --> <h1>Заголовок страницы</h1> <p>Это абзац текста.</p> <!-- Конец блока основного содержимого --> </div>

Комментарии помогают разъяснить, что делает определенная часть кода, указать на начало и конец логических блоков, или просто оставить заметку для других разработчиков. При написании комментариев можно использовать как латиницу, так и кириллицу.

### ****Правила оформления HTML****

* **Валидируй свой HTML код.** Проверяй свой код на валидность, для этого есть специальные инструменты, например – [W3C HTML validator](https://validator.w3.org/).

Валидация поможет вам понять технические требования и ограничения при написании кода, обнаружить и исправить структурные и синтаксические ошибки.

* **Используй семантические элементы HTML**, такие как <header>, <nav>, <main>, <section>, <article>, <footer>, чтобы ясно обозначить структуру страницы и улучшить её доступность для поисковых систем и пользователей.

### ****Какие плагины установить****

* **Live Server** - нужен для автоматического обновления страницы в браузере после редактирования. Для запуска нажимаем Ctrl + Shift + P, вводим в поиске команду “Live Server: Open with Live Server” и выбираем её.
* **Vscode-icons** — заменяет стандартные иконки файлов
* **Material Icon Theme** - добавляет красивые иконки для файлов и папок, что упрощает навигацию по проекту
* **Auto Close Tag** - автоматически закрывает HTML-теги при вводе закрывающего символа **>**
* **Indent Rainbow** - делает разноцветные отступы в коде и подсвечивает некорректные. Вместо радужных цветов можно установить оттенки серого.
* **Prettier - Code formatter** - автоматически форматирует CSS-код, делая его более читабельным и стандартизированным. При установке расширения прочитайте вот эту [**статью**](https://www.digitalocean.com/community/tutorials/how-to-format-code-with-prettier-in-visual-studio-code-ru)**.** Обязательно выполни шаги, указанные в этой статье, чтобы твой код был всегда красивым и аккуратным.
* **Image Preview -** с помощью этого плагина будет легко понять, если набирая адрес изображения, ты где-то ошиблась. Он выводит это изображение слева маленьким размером, если же что то не так — изображение не появится.
* **eCSStractor -** облегчает работу с классами в CSS. С его помощью ты можешь выделить хоть весь HTML-документ, а eCSStractor автоматически «вытащит» оттуда классы, которые установила к блокам и элементам на странице. Тебе остается только нажать Ctrl + V в CSS-файле.
* **IntelliSense for CSS class names** - предоставляет подсказки и автодополнение для классов CSS в HTML-файлах
* **CSS Peek** - позволяет быстро переходить от CSS-стилей в HTML-файле к соответствующим стилям в CSS-файле
* **CSS IntelliSense** - предоставляет автозаполнение и подсказки для CSS-свойств и значений
* **ESLint** - это утилита, проверяющая стандарты кодирования на JavaScript, который помогает обнаруживать ошибки и потенциальные проблемы в коде

### Основные действия:

### Cmd + S: Сохранить файл

### Cmd + N: Создать новый файл

### Cmd + O: Открыть файл

### Cmd + P: Быстрое открытие файла

### Cmd + F: Поиск в файле

### Cmd + H: Замена в файле

### Cmd + G: Перейти к строке

### Cmd + Z: Отменить последнее действие

### Cmd + Y: Повторить последнее действие

### Cmd + X: Вырезать выделенный текст

### Cmd + C: Копировать выделенный текст

### Cmd + V: Вставить скопированный или вырезанный текст

### Cmd + A: Выделить весь текст в файле

### Работа с редактором:

### Cmd + B: Показать/скрыть панель бокового меню

### Cmd + `: Открыть/закрыть терминал

### Cmd + K, Cmd + C: Закомментировать выделенный блок кода

### Cmd + K, Cmd + U: Раскомментировать выделенный блок кода

### Cmd + /: Закомментировать/раскомментировать текущую строку

### Навигация и манипуляции с текстом:

### Cmd + D: Выбрать следующее вхождение текущего выделенного текста

### Cmd + F2: Выбрать все вхождения текущего выделенного текста

### Option + ↑ / ↓: Переместить строку вверх или вниз

### Shift + Option + ↑ / ↓: Дублировать текущую строку вверх или вниз

### Cmd + ] / [: Перейти вперед или назад по блоку кода

### Работа с закладками:

### Cmd + F2: Установить или удалить закладку на текущей строке

### F2: Перейти к следующей закладке

### Shift + F2: Перейти к предыдущей закладке

### Работа с файлами и окнами:

### Cmd + Shift + N: Открыть новое окно VS Code

### Cmd + Tab: Переключиться между открытыми вкладками

### Cmd + Shift + T: Переоткрыть последнюю закрытую вкладку