**Взаимодействие с DOM: динамическое управление веб-страницей**

DOM (Document Object Model) — это интерфейс, позволяющий взаимодействовать с HTML-документом как с объектной структурой. Каждый элемент страницы (тег, текст, атрибут) представлен в DOM в виде объекта, с которым можно работать с помощью JavaScript. Это дает возможность динамически изменять содержимое страницы, реагировать на действия пользователя и добавлять интерактивность.

**Что такое DOM**

Когда браузер загружает HTML-документ, он создает DOM-дерево. Это структура, где каждая часть HTML представлена в виде узла. Существуют три основных типа узлов:

1. **Элементные узлы**: представляют HTML-теги (<div>, <p>, <img> и т.д.).
2. **Текстовые узлы**: содержат текст внутри элементов.
3. **Атрибуты узлов**: описывают свойства элементов (например, class, id, src).

С помощью JavaScript можно находить элементы, изменять их содержимое, стили и атрибуты, а также добавлять или удалять их.

**Поиск элементов**

Для работы с DOM сначала нужно найти нужный элемент. JavaScript предоставляет методы для поиска:

1. **getElementById** — возвращает элемент по его уникальному идентификатору (id).
2. **querySelector** — находит первый элемент, соответствующий CSS-селектору.
3. **querySelectorAll** — возвращает все элементы, соответствующие CSS-селектору.

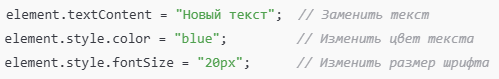


**Изменение содержимого и стилей**

После получения элемента можно изменять его содержимое:

* **textContent** изменяет текст внутри элемента.
* **innerHTML** позволяет вставлять HTML-код.

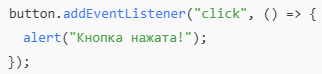
Кроме того, с помощью свойства style можно изменять стили элементов:



**Обработка событий**

События позволяют реагировать на действия пользователя, такие как клики, ввод текста или прокрутка. Для обработки событий используется метод addEventListener, который добавляет функцию-обработчик. Эта функция выполняется при наступлении события.

Пример события:



<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Взаимодействие с DOM</title>

  <style>

    #title {

      color: black;

    }

    .btn {

      margin-top: 20px;

      padding: 10px 20px;

      background-color: lightblue;

      border: none;

      cursor: pointer;

    }

  </style>

</head>

<body>

  <h1 id="title">Привет, мир!</h1>

  <button class="btn">Изменить текст</button>

  <script>

    // Найти элементы

    const title = document.getElementById("title");

    const button = document.querySelector(".btn");

    // Добавить обработчик события

    button.addEventListener("click", () => {

      title.textContent = "Текст изменен!"; // Изменить текст заголовка

      title.style.color = "blue";          // Изменить цвет текста

    });

  </script>

</body>

</html>

DOM предоставляет разработчику мощные инструменты для изменения веб-страницы в реальном времени. С помощью JavaScript можно находить элементы, управлять их содержимым и стилями, а также реагировать на действия пользователя через события. В этой лекции мы изучили основные способы взаимодействия с DOM и реализовали пример динамического изменения текста. В дальнейшем эти навыки позволят создавать интерактивные и функциональные веб-приложения.