**Пояснения:**

1. **HTML и CSS:**
   * Элемент <video> предназначен для отображения видеопотока.
   * CSS задаёт, чтобы видео занимало 80% ширины экрана с сохранением пропорций.
2. **JavaScript:**
   * Функция navigator.mediaDevices.getUserMedia() запрашивает доступ к USB-камере и микрофону.
   * В случае успешного получения потока он присваивается свойству srcObject элемента <video>.
3. **Автоматический запуск:**
   * Свойства autoplay и playsinline обеспечивают автоматический запуск видео.
   * controls позволяет пользователю управлять видео (например, ставить на паузу или отключать звук).



**Что добавлено:**

1. **Выбор камеры:**
   * Используется <select> для перечисления доступных камер, найденных через navigator.mediaDevices.enumerateDevices().
2. **Настройки параметров:**
   * Добавлены элементы <input type="range"> для регулировки яркости и контраста.
   * Кнопка Применить настройки применяет параметры через API applyConstraints.
3. **Максимальное разрешение:**
   * Используются свойства width: { ideal: 1920 } и height: { ideal: 1080 } для запроса максимального разрешения.
4. **Управление текущим потоком:**
   * При переключении камеры текущий поток останавливается, чтобы освободить ресурсы.
5. **Универсальность:**
   * Код автоматически обнаруживает устройства камеры и позволяет пользователю их переключать.

**Как это работает:**

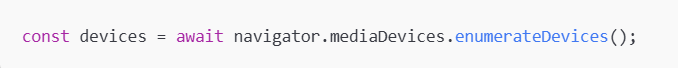
1. Загружается список доступных камер.
2. По умолчанию используется первая камера в списке с максимальным разрешением.
3. Пользователь может переключить камеру и настроить параметры яркости/контраста, которые применяются динамически.

**Что такое navigator.mediaDevices?**

navigator.mediaDevices — это интерфейс для работы с мультимедийными устройствами (веб-камерами, микрофонами, динамиками и т. д.) через WebRTC. Он предоставляет доступ к:

1. **getUserMedia()** — позволяет запрашивать доступ к видеопотоку, аудиопотоку или и тому, и другому.
2. **enumerateDevices()** — возвращает список всех доступных мультимедийных устройств (видеокамер, микрофонов, динамиков).
3. **ondevicechange** — событие, которое срабатывает при подключении или отключении устройства.

**Разбор строки кода:**

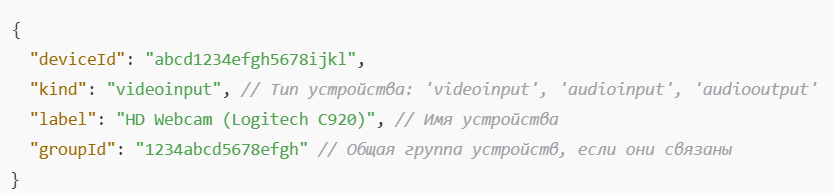


#### Расшифровка:

1. **navigator.mediaDevices.enumerateDevices()**:
   * Возвращает промис, который резолвится в массив объектов, представляющих мультимедийные устройства, подключённые к компьютеру.
2. **await**:
   * Используется для ожидания завершения выполнения промиса. Это позволяет получить результат вызова enumerateDevices() без использования явного .then().
3. **const devices**:
   * Переменная, в которую сохраняется результат — массив объектов с информацией о доступных устройствах.

### Пример результата вызова enumerateDevices():

Результатом будет массив объектов, содержащих информацию о каждом мультимедийном устройстве. Вот пример структуры одного объекта:



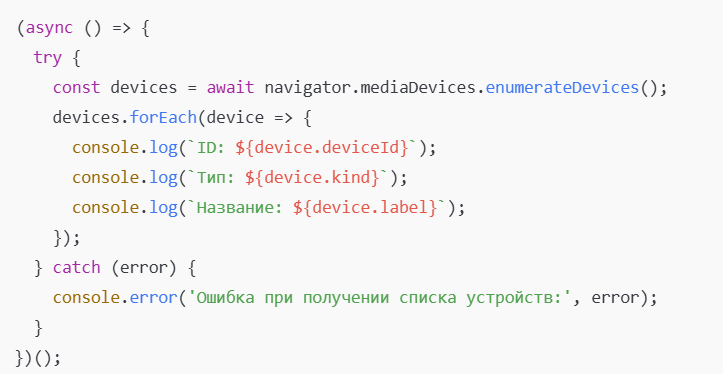
**Полезные свойства объектов устройства:**

1. **deviceId**:
   * Уникальный идентификатор устройства.
   * Используется для явного указания, к какому устройству подключиться.
2. **kind**:
   * Тип устройства:
     + videoinput — видеокамера.
     + audioinput — микрофон.
     + audiooutput — динамик.
3. **label**:
   * Название устройства. Например, "HD Webcam (Logitech C920)".
   * Обычно доступно только после того, как пользователь предоставил доступ к устройству (например, через getUserMedia()).
4. **groupId**:
   * Идентификатор группы, указывающий, что устройства (например, камера и микрофон) принадлежат одному физическому устройству.

**Как это применяется?**

Вызов enumerateDevices() используется, чтобы:

* Узнать, какие камеры или микрофоны доступны.
* Дать пользователю возможность выбрать устройство.
* Реализовать поддержку нескольких устройств (например, переключение между камерами).



Этот код выведет в консоль список всех доступных устройств с их параметрами.