### **Документация: Проект Weather App**

Этот проект представляет собой приложение на основе HTML, CSS и JavaScript, которое отображает текущую погоду и прогноз погоды для указанного города или текущего местоположения пользователя. Приложение использует OpenWeatherMap API для получения данных о погоде.

### **1. HTML (Структура приложения)**

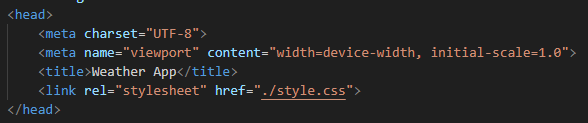
Файл index.html определяет структуру интерфейса пользователя.



#### **Основные элементы:**

* **<div id="weather-container">** Контейнер приложения. Включает элементы для ввода города, отображения температуры, погодной информации и почасового прогноза.
* **<input type="text" id="city">** Поле для ввода названия города.
* **<button onclick="getWeather()">** Кнопка для вызова функции поиска погоды.
* **<img id="weather-icon">** Изображение, которое отображает иконку текущей погоды.
* **<div id="temp-div">** Элемент для отображения текущей температуры.
* **<div id="weather-info">** Элемент для отображения текстовой информации о погоде, включая город и описание.
* **<div id="hourly-forecast">** Блок для отображения почасового прогноза погоды.

#### **Подключение:**



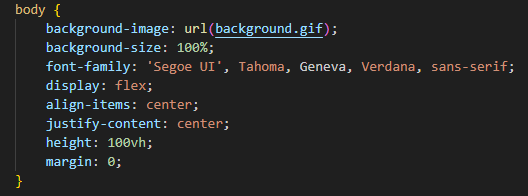


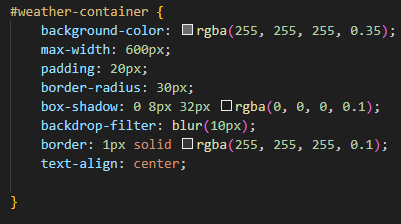
* **<link rel="stylesheet" href="./style.css">** Подключение файла стилей.
* **<script src="./script.js"></script>** Подключение JavaScript-кода.

### **2. CSS (Стилизация приложения)**

Файл style.css определяет внешний вид приложения.

#### **Основные стили:**

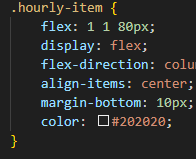
* **Фон приложения:** Используется анимация с эффектом полного заполнения (background-size: 100%).  
  
* **Контейнер погоды (#weather-container):**
  + Полупрозрачный фон с эффектом размытия (backdrop-filter).
  + Закругленные углы (border-radius: 30px).
  + Тень для создания объема (box-shadow).



* **Кнопки (button):**
  + Цвет текста и фона изменяются при наведении (:hover).
  + Скругленные углы и мягкие отступы для удобства.



* **Почасовой прогноз (.hourly-item):**
  + Элементы выровнены по центру, включают иконки и текст.
  + Используется адаптивная сетка с гибкими отступами (flex-wrap).



### **3. JavaScript (Функциональность приложения)**

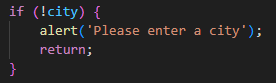
#### **Подключение к API**

Приложение использует [OpenWeatherMap API](https://openweathermap.org/api). Для работы необходим API-ключ, который используется в запросах.



#### **Функция: getWeather()**

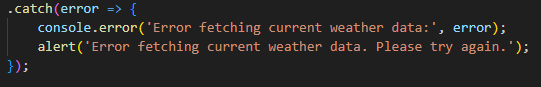
Эта функция вызывается при нажатии на кнопку поиска.

1. **Получение ввода города:** 

* Если пользователь не ввел город, отображается предупреждение.

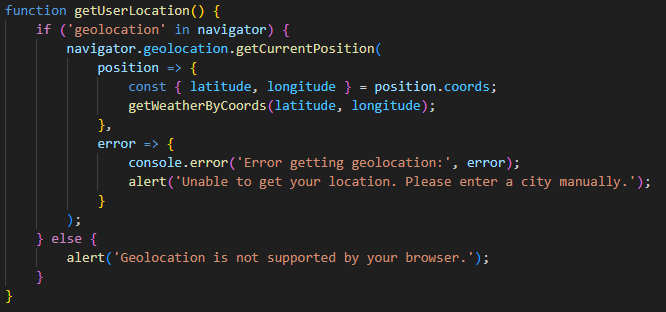
1. **Запрос текущей погоды и прогноза:**

Формируются два URL для получения текущей погоды и прогноза.

1. **Обработка ответа:**
   * displayWeather(data) для отображения текущей погоды.
   * displayHourlyForecast(data.list) для почасового прогноза.
2. **Обработка ошибок:  
   **

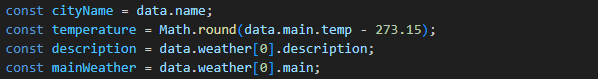
#### **Функция: getUserLocation()**

Используется для получения текущего местоположения пользователя.

1. **Геолокация:  
   **
2. **Запрос к API с координатами:** Используется getWeatherByCoords(lat, lon).

#### **Функция: displayWeather(data)**

Отображает текущую погоду.

1. **Данные API:  
   **
   * Город: data.name.
   * Температура (в °C): Math.round(data.main.temp - 273.15).
   * Описание: data.weather[0].description.
2. **Фоновые изображения:** Зависит от погодных условий (data.weather[0].main).
3. **Иконка погоды:**

#### **Функция: displayHourlyForecast(hourlyData)**

Отображает прогноз погоды на следующие 24 часа.

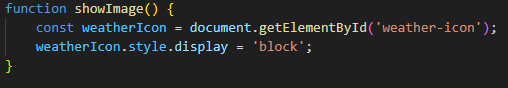
1. **Обработка данных:  
   **
2. **Создание HTML:** Каждое почасовое условие добавляется в #hourly-forecast.

#### **Функция: getWeatherByCoords(lat, lon)**



Получает погоду на основе широты и долготы.

#### **Функция: showImage()**



Отображает иконку погоды, скрытую по умолчанию.

### **4. Использование проекта**

1. Введите название города или дайте разрешение на доступ к геолокации.
2. Получите данные о текущей погоде и прогнозе.
3. Фон меняется в зависимости от погоды.