**How to run the MyWeatherApp code**

Welcome to the **MyWeatherApp**! This Spring Boot application allows you to compare daylight hours between two cities and check which city is currently raining. The application uses the **Visual Crossing Weather API** to fetch real-time weather data.

**Prerequisites**

Before running the application, ensure you have the following installed:

* **Java SDK 17** (Download from [OpenJDK](https://openjdk.java.net/projects/jdk/17/) or [Oracle JDK](https://www.oracle.com/java/technologies/javase/jdk17-archive-downloads.html))
* **Maven 3.6.3+** (Download from [Apache Maven](https://maven.apache.org/install.html))
* **Visual Studio Code** (Download from [VS Code](https://code.visualstudio.com/))

**Setting Up the Project**

1. Clone the Repository using the URL.
2. **Add the API Key**:
   1. Sign up for a free account on [Visual Crossing](https://www.visualcrossing.com/) and get an API key.
   2. Open the application.properties file in src/main/resources/.
   3. Add your API key:



1. **Install Dependencies**:
   1. Open the project in **VS Code**.
   2. Open the integrated terminal and run: **mvnw.cmd clean install**

**Running the Application**

1. **Start the Application**:
   * In the VS Code terminal, run: **mvnw.cmd spring-boot:run**
   * You should see logs indicating that the application has started.
2. **Access the Application**:
   * Open your browser and navigate to: <http://localhost:8080>
   * If the application runs correctly, you should see a Whitelabel Error Page. This is expected because the root endpoint (/) is not mapped. You can proceed to test the specific endpoints.

**Testing the Endpoints**

The application exposes the following endpoints:

**1. Forecast by City**

* **Endpoint**: GET /forecast/{city}
* **Example**: <http://localhost:8080/forecast/London>
* This returns weather data for the specified city in JSON format.

A screen shot of a computer

Description automatically generated

**2. Compare Daylight Hours**

* **Endpoint**: GET /compare-daylight/{city1}/{city2}
* **Example**: <http://localhost:8080/compare-daylight/London/Mumbai>
* This compares the daylight hours between two cities and returns the city with the longest day.

A screenshot of a computer

Description automatically generated

**3. Check Rain**

* **Endpoint**: GET /check-rain/{city1}/{city2}
* **Example**: <http://localhost:8080/check-rain/London/Mumbai>
* This checks which of the two cities is currently raining.

A screenshot of a computer

Description automatically generated

**Unit Tests**

To run the unit tests:

1. Open the VS Code terminal.
2. Run the following command: **mvnw.cmd test**
3. You should see an output indicating that all tests have passed.

A screenshot of a computer error

Description automatically generated