**Weather Prediction System** that integrates real-time weather data with historical data to forecast weather conditions, such as temperature, humidity, and the probability of rain. This system provides valuable insights by combining **live data from OpenWeatherMap** with **past weather records** to make accurate predictions using machine learning models.

**1. Objective of the Project**

The main purpose of this project is to **analyze weather conditions** and provide **predictions for future weather** using a combination of:  
✅ **Current weather data** (fetched using OpenWeatherMap API)  
✅ **Historical weather data** (loaded from weather.csv)  
✅ **Machine Learning** (Random Forest models for classification & regression)

It aims to answer two key questions:  
**Will it rain tomorrow?** (Classification Problem)  
W**hat will be the temperature & humidity in the next few hours?** (Regression Problem)

This helps in weather forecasting and planning for different activities like agriculture, travel, and daily life.

**2. How the Project Works?**

The project consists of **several interconnected steps**, each responsible for handling different parts of weather prediction.

**Step 1: Fetching Live Weather Data**

* The project starts by **asking the user to enter a city name**.
* Once the city name is provided, the system **fetches real-time weather data** from the **OpenWeatherMap API**.
* This data includes:
  + **Temperature (Current, Min, Max)** – Helps understand current weather conditions.
  + **Humidity Levels** – Important for predicting rainfall.
  + **Wind Speed & Direction** – Wind data helps in forecasting weather changes.
  + **Atmospheric Pressure** – Affects weather patterns like storms.
  + **General Weather Conditions** (Sunny, Cloudy, Rainy, etc.).

📌 **Why is this important?**  
This real-time weather data serves as an input for the prediction model, ensuring that forecasts are up to date and accurate.

**Step 2: Loading Historical Weather Data**

* The project **loads a dataset** containing past weather records stored in weather.csv.
* This dataset contains **several months or years of weather observations** collected from different locations.
* The data is **cleaned and pre-processed** by:
  + Removing **duplicate records**
  + Handling **missing values**
  + Converting **textual information** into machine-readable formats

📌 **Why is this important?**  
Machine learning models require **a lot of past data** to learn patterns in weather conditions. The historical data helps train models to **understand trends and predict future weather accurately**.

**Step 3: Processing and Preparing Data for Prediction**

* Once the data is loaded, it is **prepared for machine learning models**.
* This involves selecting **key weather parameters** like:
  + Minimum & Maximum Temperature
  + Wind Speed and Direction
  + Humidity & Pressure
  + Previous Rainfall Data
* Some categorical variables like **wind direction** are converted into **numerical values** so they can be used for training models.
* The dataset is then **split into features (inputs) and labels (outputs)**:
  + Inputs: Weather conditions from previous days.
  + Outputs: Rain occurrence (Yes/No) and future temperature/humidity values.

📌 **Why is this important?**  
This step ensures that the **machine learning model receives structured and meaningful data** for training.

**Step 4: Training the Rain Prediction Model (Classification)**

* A **Random Forest Classifier** is used to train a model that predicts **whether it will rain tomorrow**.
* The model learns patterns based on **past weather conditions** and the **occurrence of rain**.
* Once trained, the model can analyze **current weather conditions** and **predict the likelihood of rain for the next day**.

📌 **Why is this important?**  
This model is useful for **weather forecasting applications**, helping users prepare for rain in advance.

**Step 5: Training the Temperature & Humidity Prediction Model (Regression)**

* A **Random Forest Regressor** is used to **predict future temperature and humidity**.
* The model is trained on past **temperature, humidity, and time-based trends**.
* Once trained, it can **forecast temperature and humidity for the next few hours or days** based on current conditions.

📌 **Why is this important?**  
Accurate temperature and humidity predictions help in **planning agricultural activities, energy consumption, and outdoor events**.

**Step 6: Making Predictions for the Future**

* After training, the models can make **real-time predictions** based on the most recent weather data.
* The system uses the trained models to predict:
  + **Whether it will rain tomorrow** (Yes/No)
  + **Temperature for the next 5 hours**
  + **Humidity levels for the next 5 hours**

📌 **Why is this important?**  
This provides users with a **realistic forecast** of upcoming weather conditions, improving decision-making for travel, farming, and city planning.

**Step 7: Displaying Results**

* The final step is to **display all relevant weather details**, including:
  + **Current Weather** (fetched from OpenWeatherMap)
  + **Past Weather Data Insights**
  + **Rain Prediction for Tomorrow**
  + **Temperature & Humidity Forecast for the Next 5 Hours**

📌 **Why is this important?**  
This combines **real-time data, historical trends, and AI-powered predictions** into one system, providing an easy-to-understand weather forecast for users.

**3. Why Does This Project Use Both Real-Time & Historical Data?**

The combination of **live weather data** and **historical records** makes the predictions more **accurate and reliable**.

**🔹 Real-time Data (from OpenWeatherMap API)**

* Ensures that the system gets **current weather conditions**.
* Helps in making **instant predictions** about upcoming weather.

**🔹 Historical Data (from weather.csv)**

* Provides a **large dataset for training machine learning models**.
* Helps the models **understand long-term weather patterns**.

By combining **real-time observations** with **past trends**, the system **adapts to changing weather conditions** and makes better forecasts.

**4. Real-World Applications of Your Project**

✅ **Weather Forecasting Apps** – Improve weather prediction accuracy.  
✅ **Agriculture Planning** – Farmers can plan irrigation and crops based on rainfall predictions.  
✅ **Travel & Logistics** – Helps airlines, shipping, and transport services adjust schedules.  
✅ **Smart City Planning** – Helps cities manage resources based on weather patterns.  
✅ **Disaster Preparedness** – Early rain predictions can help prevent flooding & other disasters.

This project is a **great blend of real-time weather monitoring, data analysis, and AI-driven forecasting**, making it useful for various industries. 🚀

**Technologies Used**

* **Python** – Main programming language
* **OpenWeatherMap API** – Fetches real-time weather data
* **Pandas** – Used for handling historical weather data
* **NumPy** – For numerical computations
* **Scikit-learn (sklearn)** – For machine learning models
* **Datetime & Pytz** – To handle time zones
* **CSV File (weather.csv)** – Stores past weather data