**Weather prediction by using python**

**A Project Report submitted in the fulfill of the requirement**

**for the degree of**

**INFORMATIONAL TECHNOLOGY AND MANAGEMENT**

**UNDER**

**UTKAL UNIVERSITY,**

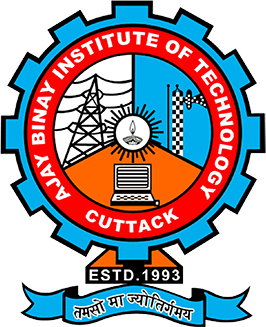
**BHUBANESWAR, ODISHA**

**Submitted by: Guided by:**

Bijoy Laxmi Behera

Dr. Amaresh Sahu

Evita Bhowmik



**AJAY BINAYA DEGREE COLLEGE**

**CUTTACK, ODISHA**

**ABSRACT**

1. **Objective:**

The primary objective of weather prediction is to provide accurate and timely information about future weather conditions. This information is used to help individuals and organizations make informed decisions regarding a wide range of activities, such as agricultural planning, transportation, disaster preparedness, and energy management.

Weather prediction involves using scientific models and data analysis techniques to forecast the conditions of the atmosphere at a specific time and location. This process requires gathering data from a variety of sources, including satellites, radar systems, and weather stations, and using this data to create computer simulations of the atmosphere.

By providing accurate weather forecasts, weather prediction can help individuals and organizations make informed decisions that can improve safety, increase efficiency, and reduce costs. For example, farmers can use weather forecasts to plan their planting and harvesting schedules, while airlines can use them to make decisions about flight routes and schedules. In addition, accurate weather forecasts can help emergency responders and disaster management teams prepare for severe weather events, such as hurricanes and tornadoes.

1. **Introduction:**

Weather prediction, also known as weather forecasting, is the process of using scientific and mathematical models to forecast the state of the atmosphere for a specific location and time in the future. This involves collecting and analyzing various types of data, such as temperature, humidity, wind speed and direction, and atmospheric pressure, to create a picture of the current weather conditions.

Here we are using python by using pip, pip request, and a URL <https://wttr.in>.The main purpose of this program is to predict the weather of any state of any country. This python program shows the predicted weather report for 3 days and also forecasts the weather in morning, afternoon, evening, and night of that upcoming predicted days.

1. **Prerequisites:**

The necessary things requirements to run our program are as follows:

* Install the latest version of Python.
* Install PyCharm for better results.
* Install the necessary libraries required.

1. **Installation:**

* **Install Python :**

Go to <https://www.python.org/downloads/> and download the latest version of python that suits your Laptop/PC.

* **Install PyCharm:**

Go to <https://www.jetbrains.com/pycharm/download/#section=windows> and download the community version as its free to use.

* **Install the required python libraries:**

open the terminal and run the following commands to install the required python libraries.

* + - * Python get-pip.py
      * Pip install requests
* **“wttr.in”:**

You can have a look at this website from GitHub where a short detail is mentioned regarding that website: <https://github.com/chubin/wttr.in>.

* **Run the project:**

To run the project, open PyCharm and navigate the project directory . then run the program.

1. **Databases:**

For data base we have taken a url <https://wttr.in> . This website is the way to curl the weather

wttr.in is a console-oriented weather forecast service that supports various information representation methods like terminal-oriented ANSI-sequences for console HTTP clients (curl, httpie, or wget), HTML for web browsers, or PNG for graphical viewers.

Originally started as a small project, a wrapper for wego, intended to demonstrate the power of the console-oriented services, wttr.in became a popular weather reporting service, handling tens of millions of queries daily.

You can see it running here: wttr.in.

1. **Project Structure:**

import requests

print("Welcome to the Weather Forecaster!\n\n")

print("Enter the City you want the weather report!\n\n")

city\_name = input("Enter the name of the City : ")

print("\n\n")

def Gen\_report(C):

    url = 'https://wttr.in/{}'.format(C)

    try:

        data = requests.get(url)

        T = data.text

    except:

        T = "Error Occurred"

    print(T)

Gen\_report(city\_name)

1. **Code Explanation:**
2. import requests

*To import requests library file in python 1st we need to install Pip package. The full form of Pip is Python Package Index. It is a package manager for Python, which is used to install and manage Python packages (libraries or modules).*

*The ‘requests’ is a very popular library used for making HTTP (URL) requests to web servers. It provides a simple and intuitive API for sending HTTP requests and handling responses, making it easier to work with web APIs and web services.*

print("Welcome to the Weather Forecaster!\n\n")

print("Enter the City you want the weather report!\n\n")

*These two lines of code are used to display a welcome message and prompt the user to enter the name of a city for which they would like to see the weather report.*

* *The first line uses the* ***print()*** *function to display the string "Welcome to the Weather Forecaster!\n\n" on the console. The* ***"\n"*** *characters are used to add two new lines after the message, which creates some space before the next line of text.*
* *The second line also uses the* ***print()*** *function to display the string "Enter the City you want the weather report!\n\n" on the console. This line prompts the user to enter the name of the city for which they want to see the weather report.*

*Overall, these two lines of code are used to provide a simple and user-friendly interface for the weather forecaster program, allowing the user to easily input the desired city name and receive the corresponding weather report.*

city\_name = input("Enter the name of the City : ")

print("\n\n")

*These two lines of code prompt the user to enter the name of a city and store the input in the city\_name variable. The input() function is used to read a line of text from the user, and the string "Enter the name of the City : " is passed as an argument to provide a prompt to the user.*

*Once the user enters a city name and presses the "Enter" key, the input text is assigned to the city\_name variable, which can then be used in the Gen\_report() function to generate a weather report for the specified city.*

*The second line of code simply prints two new lines on the console, which creates some space between the input prompt and the next line of text. This is done to make the program output easier to read and more visually appealing.*

*Overall, these two lines of code allow the user to input the desired city name and store it in a variable, which can then be used in the Gen\_report() function to generate a weather report for the specified city.*

*And again* ***"\n"*** *characters are used to add two new lines after the message, which creates some space before the next line of text.*

def Gen\_report(C):

    url = 'https://wttr.in/{}'.format(C)

    try:

        data = requests.get(url)

        T = data.text

    except:

        T = "Error Occurred"

    print(T)

*This is a function definition for* ***Gen\_report()*** *that takes a single argument* ***C****, which represents the name of a city for which the weather report is to be generated.*

*Within the function, the* ***url*** *variable is defined as a string that contains the URL for the weather report API endpoint for the specified city. The* ***{}*** *placeholder in the URL string is replaced with the value of the* ***C*** *argument using the* ***str.format()*** *method, which generates the appropriate URL for the specified city.*

*The* ***requests.get()*** *method is then used to send an HTTP GET request to the API endpoint using the generated URL, and the response data is stored in the* ***data*** *variable.*

*The* ***data.text*** *attribute is used to extract the text content of the response data, which contains the weather report for the specified city in a plain text format.*

*If the* ***requests.get()*** *method encounters an error while trying to retrieve the weather report data, the* ***except*** *block is executed and the* ***T*** *variable is set to the string "Error Occurred".*

*Finally, the weather report for the specified city is printed to the console using the* ***print()*** *function, which displays the value of the* ***T*** *variable on the console.*

*Overall, this function takes a city name as input, retrieves the corresponding weather report using an external API, and prints the weather report to the console.*

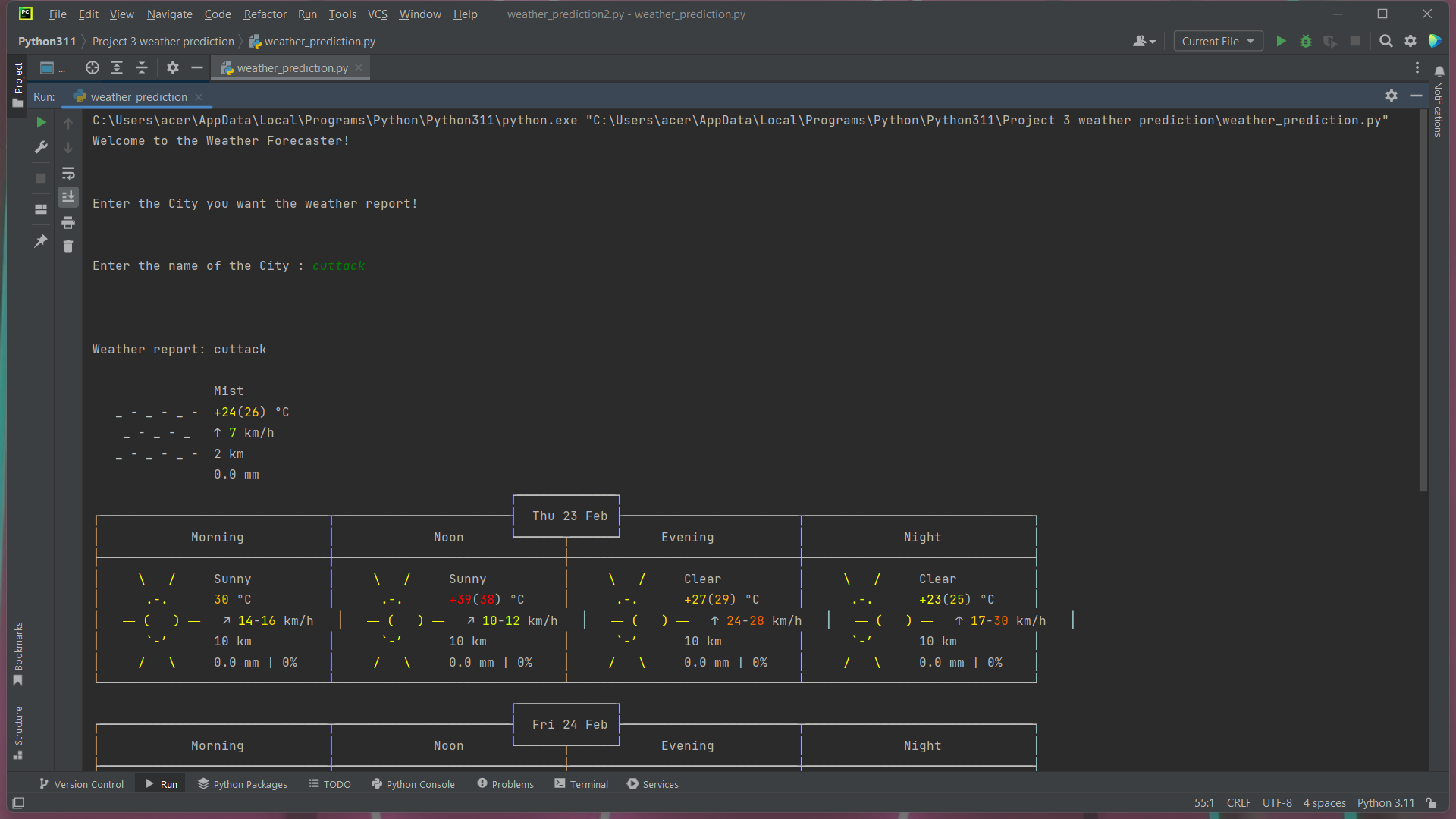
Gen\_report(city\_name)

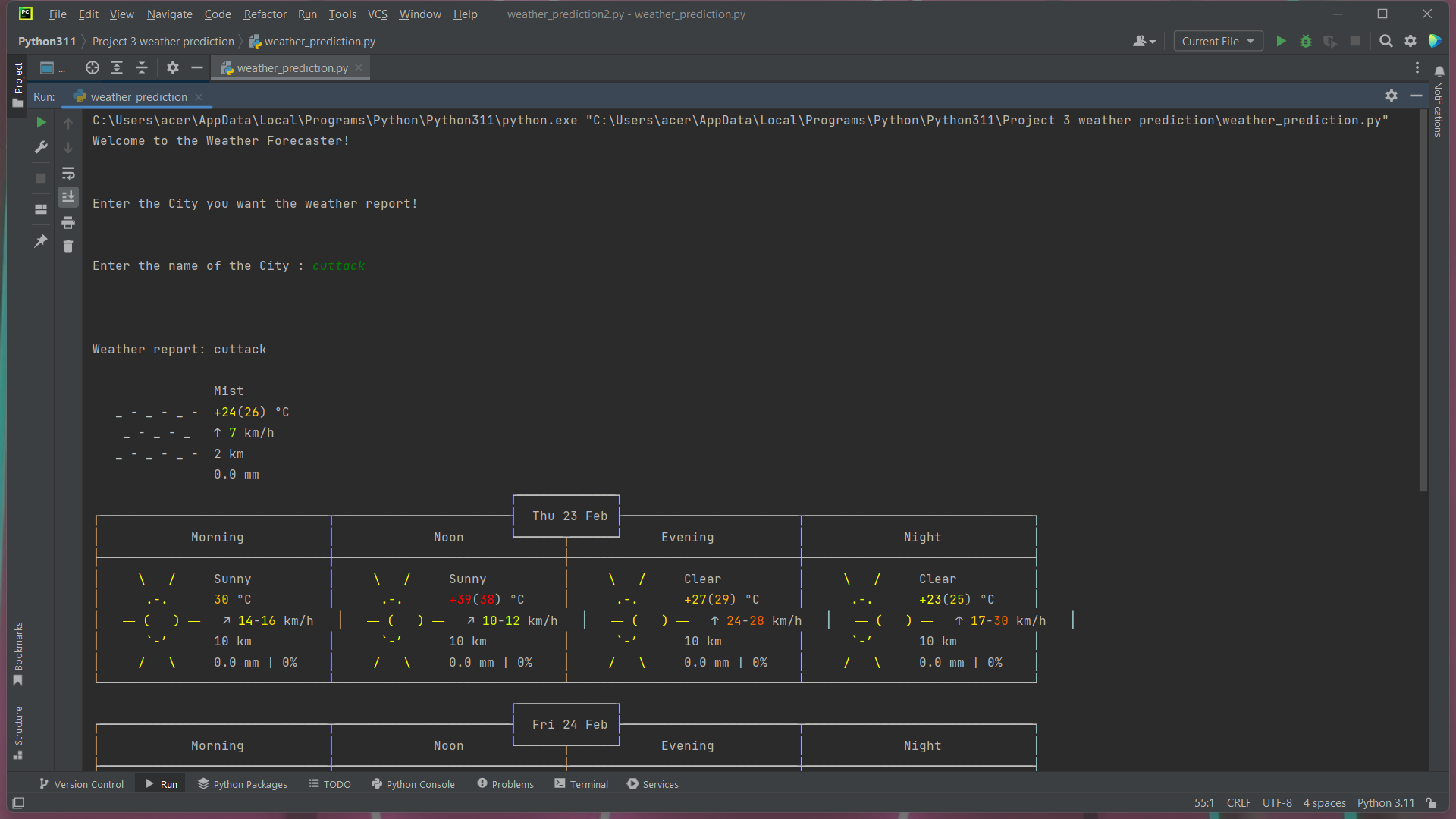
*This line of code calls the Gen\_report() function and passes the value of the city\_name variable as an argument.*

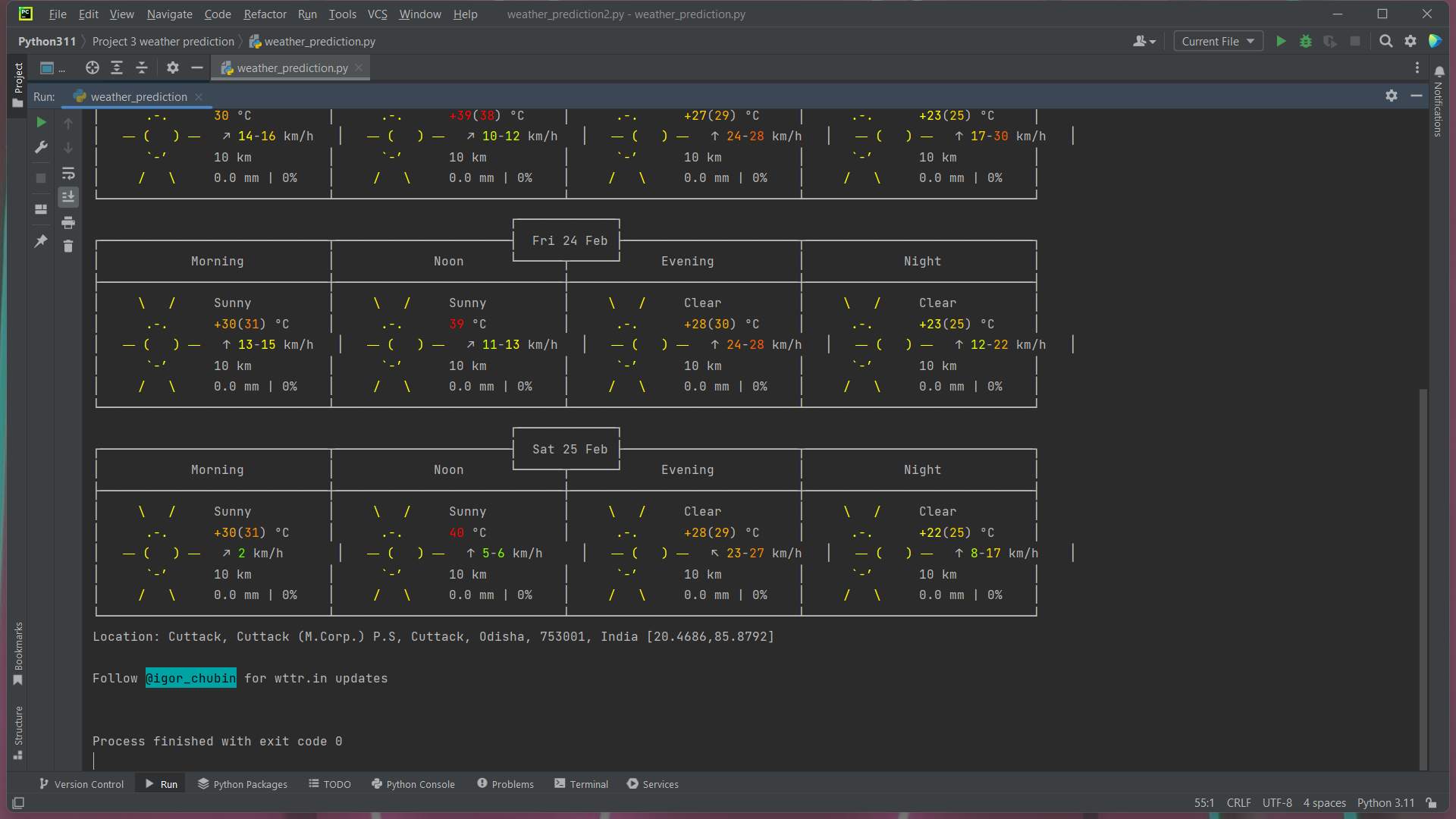
*When executed, the Gen\_report() function uses the input city name to generate a URL for the weather report API endpoint and sends an HTTP GET request to retrieve the weather report data for the specified city.*

*The function then extracts the text content of the response data and prints the weather report to the console.*

*Overall, this line of code generates a weather report for the specified city using the Gen\_report() function and displays the weather report on the console.*

****

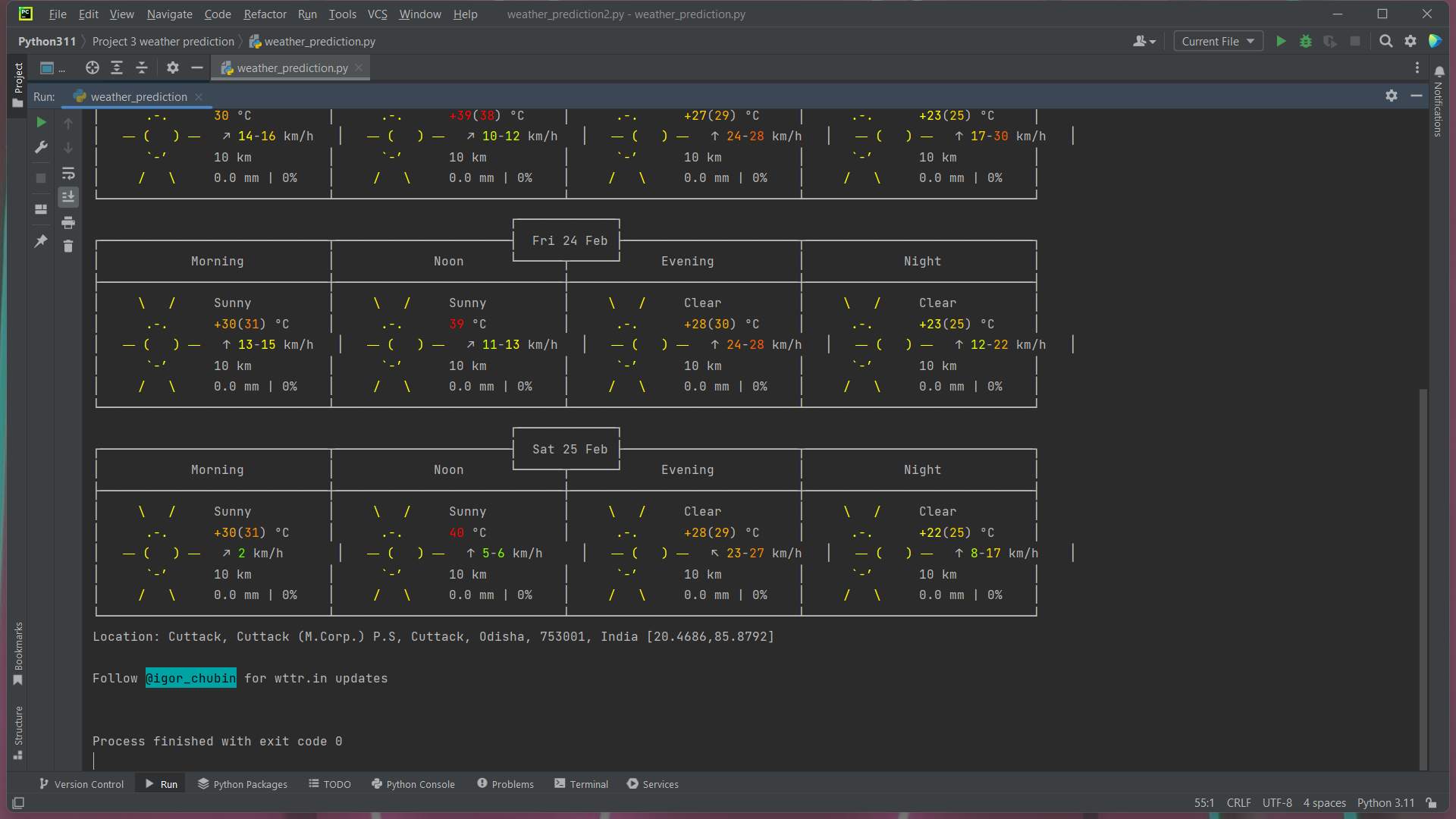
1. **Output:**

****

1. **Conclusion:**

This program is a simple Python script that uses the requests library to retrieve weather information from the wttr.in API based on user input. The program prompts the user to enter the name of a city, which is then used to generate a weather report for that city.

In terms of future use, this program could be expanded to include additional functionality, such as the ability to retrieve weather data for multiple cities, or the ability to display the weather data in a more user-friendly format. It could also be integrated into a larger application, such as a weather dashboard or mobile app, to provide users with up-to-date weather information for a variety of locations.

****