

Task 1:- Command Line Weather Checking Application Documentation

1. Introduction

The Command Line Weather Checking Application is a simple, yet powerful tool that allows users to retrieve weather information from various locations directly from the command line interface (CLI). This application uses a third-party weather API to provide up-to-date weather data, including current conditions, forecasts, and more.

Key features of this application include:

- ✓ **Real-time weather data retrieval**
- ✓ **Support for multiple locations**
- ✓ **Configurable units (e.g., Fahrenheit or Celsius)**
- ✓ **Simple and user-friendly interface**

2. Installation

Before installing the application make sure you have installed all the necessary libraries required using the '**Pip**' command.

Now, to install the Command Line Weather Checking Application, follow these steps:

1. Open a terminal window.

2. Clone the repository from GitHub:

```
``shell  
  
git clone https://github.com/yourusername/weather-cli.git  
``
```

3. Change to the application directory:

```
``shell  
  
cd weather-cli  
``
```

4. Install any required dependencies (if applicable).

3. Usage

Define the URL you want to send an HTTP GET request to

Base_URL = "<http://api.openweathermap.org/data/2.5/weather?>"

Send an HTTP GET request to the URL

response = requests.get(url).json()

OR

response = rq.get(URL).json()

#Url to access the weather output

url = {BASE_URL + "appid=" + API_KEY + "&q=" + CITYNAME}

OR

url = base_url + "appid=" + api_key + "&q=" + cityname

To retrieve weather information for a specific location, use the following basic command:

```
``shell
```

```
./weather.py <location>
```

```
``
```

Replace `**<location>**` with the name of the city or location you want to check the weather for.

4. Configuration

The application can be configured by modifying the `'json'` command codes, which is typically located in the application's directory. You can customize settings such as the default temperature units, API key (if required by the weather service provider), and other options.

Make sure to follow the format specified in the `'json'` file when making changes.

5. Examples

Here are some example commands to demonstrate how to use the Command Line Weather Checking Application:

- Retrieve the current weather for Delhi:

```
```shell
./python weather.py Delhi
```
```

- Retrieve the current weather for Mumbai:

```
```shell
./python weather.py Mumbai
```
```

- Retrieve the current weather for Hyderabad:

```
```shell
./python weather.py Hyderabad
```
```

6. Output of Examples

Below are the examples of real time data that we get after executing the above commands in our environment or command prompt.

1) Output 1

```
C:\Users\ASUS\Weather>python weather.py
Enter your city name : delhi
Temperature in delhi is : 28.05°C or 147.20°F
Temperature in delhi feels like : 32.71°C or 165.84°F
Humidity in delhi is : 83 %
Wind speed in delhi : 3.6 Kmph
General weather in delhi : mist
Sun rises in delhi at 2023-09-09 06:02:58 local time.
Sun sets in delhi at 2023-09-09 18:34:48 local time.
```

2) Output 2

```
C:\Users\ASUS\Weather>python weather.py
Enter your city name : mumbai
Temperature in mumbai is : 29.99°C or 154.96°F
Temperature in mumbai feels like : 36.99°C or 182.96°F
Humidity in mumbai is : 84 %
Wind speed in mumbai : 4.63 Kmph
General weather in mumbai : haze
Sun rises in mumbai at 2023-09-09 06:25:12 local time.
Sun sets in mumbai at 2023-09-09 18:47:30 local time.
```

3) Output 3

```
C:\Users\ASUS\Weather>python weather.py
Enter your city name : hyderabad
Temperature in hyderabad is : 26.23°C or 139.92°F
Temperature in hyderabad feels like : 26.23°C or 139.92°F
Humidity in hyderabad is : 83 %
Wind speed in hyderabad : 5.66 Kmph
General weather in hyderabad : haze
Sun rises in hyderabad at 2023-09-09 06:03:26 local time.
Sun sets in hyderabad at 2023-09-09 18:24:17 local time.
```

7. Troubleshooting

If you encounter any issues or errors while using the application, consider the following troubleshooting steps:

1. Check your internet connection.
2. Ensure that you have provided the correct location name. ['Check for any typos']
3. Review the configuration in the ``.json`` file.

If the issue persists, please consult the API provider documentation or contact the support team for assistance.

8. Frequently Asked Questions

Visit the project's README or openweatherapi website for a list of frequently asked questions and answers.

9. Acknowledgments

I would like to thank the openweatherapi community for their contributions to this project by providing the real time weather data services and APIs codes.

For more information and updates, please visit the project's GitHub repository:

<https://github.com/Sanket3332/INFOTRIXS/tree/main>