**NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES**

**PROGRAM: SOFTWARE ENGINEERING**



***SOFTWARE CONSTRUCTION AND DEVELOPMENT LAB***

**LAB TASK 12**

**SUBMITTED BY:**

**Name: Ahmed Ali**

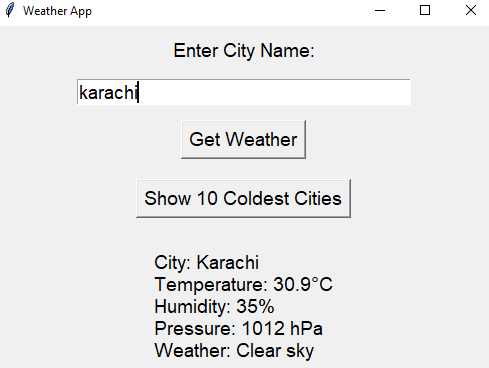
**Roll No: 22P-9318**

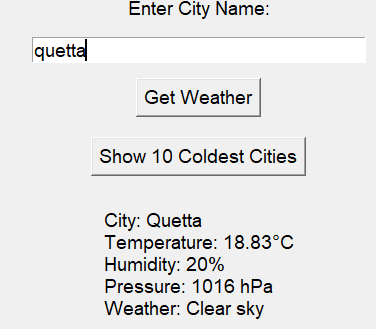
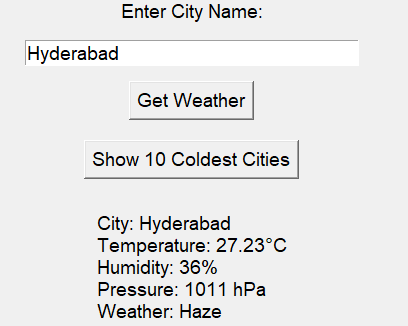
**SECTION: BS(SE)-5B**

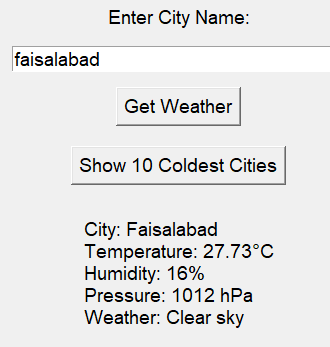
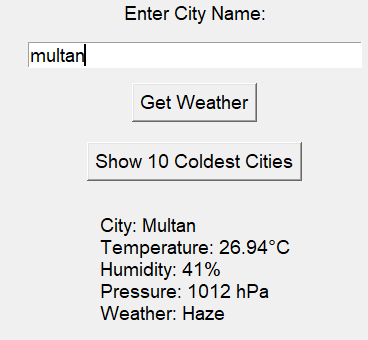
**INSTRUCTOR NAME: Sir Saood Sarwar**

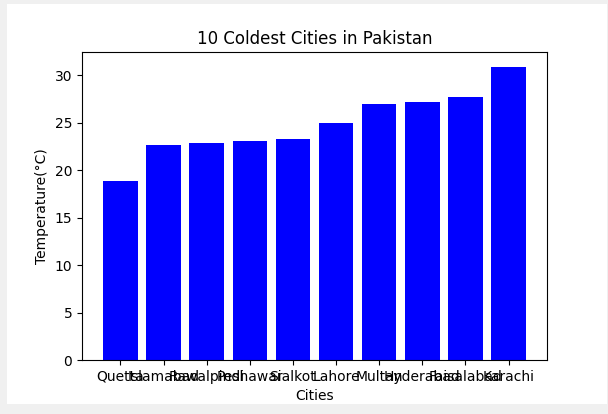
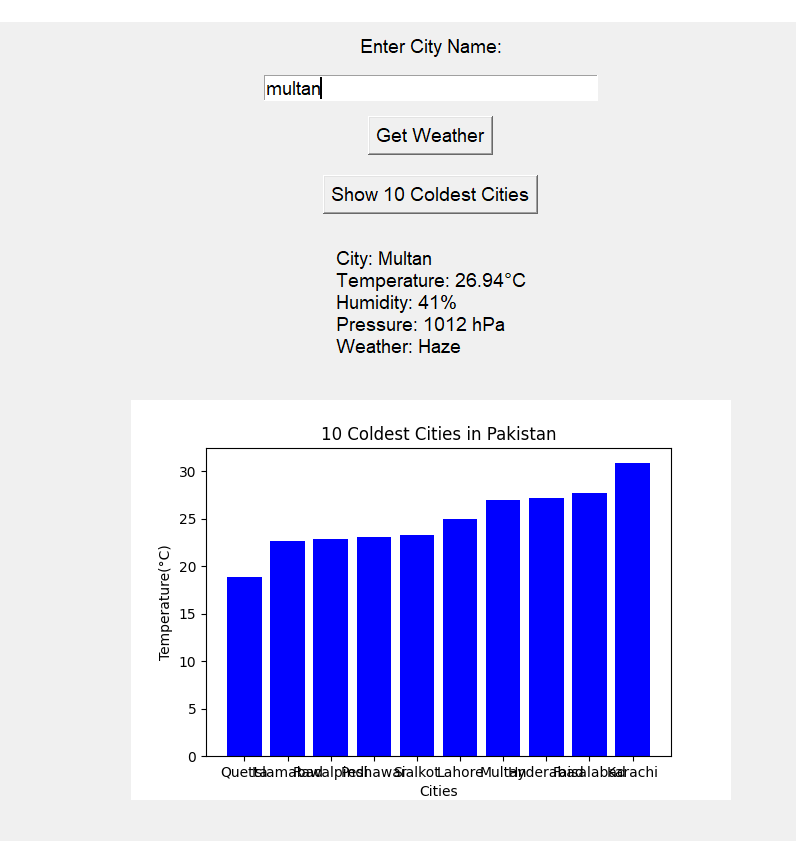
**A DEPARTMENT OF COMPUTER SCIENCE**

**Screenshots of GUI Showing Weather data for specific city, top 10 coldest cities In Pakistan, Baar chart for temperatures**

****







**A short description of implementation approach:**

I first analyze the OpenWeatherMap API's documentation to understand the endpoints needed for fetching current weather and forecast data Then, I develop the backend by using the requests library to fetch weather data for a given city and retrieve the temperatures of 10 coldest cities in Pakistan

For the frontend, I design a simple GUI using Tkinter, where I create input fields, buttons, and labels to display the weather information I also integrate a Matplotlib bar chart to visualize the temperatures of the coldest cities

Lastly, I handle errors, ensuring the application works smoothly by providing error messages for invalid input or API errors, while keeping the user interface clean and easy to use

I use the **FigureCanvasTkAgg module from matplotlib.backends.backend\_tkagg** to integrate Matplotlib charts into my Tkinter GUI first, I create a plot using Matplotlib, and then I use FigureCanvasTkAgg to convert that plot into a widget that can be displayed inside my Tkinter window

This allows me to add interactive visualizations, like bar charts or line graphs, directly into the GUI layout It's helpful for making the application more dynamic and visually appealing