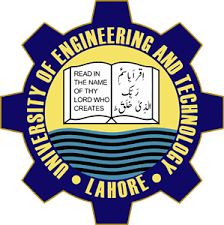
**University Of Engineering And Technology**

****

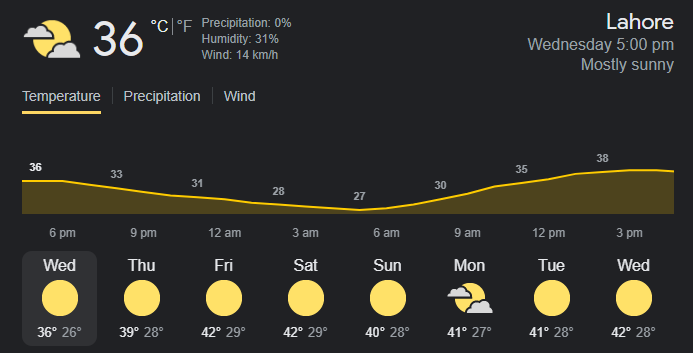
**Weather Forecast Project in Python**

**Group Members:**

1. Abdul Basit 2021-CE-50
2. Mohsin Qais 2021-CE-96
3. Talal Muzammal 2021-CE-47
4. Altamash Zaheer 2021-CE-49
5. Safiullah 2021-CE-100

**Submitted To*:***

Raja Muzammal Muneer



**Contents:**

**Flow Chart……………………………………………………………3**

**Introduction:…………………………………………………………4**

**Features: …………………………………………………………....4**

**Objectives:… ………………………………………………………...4**

**Socio-economic benefits:…………………………………………….4**

**Libraries………………………………………………………….......5**

**Code snippets used:………………………………………………… 6**

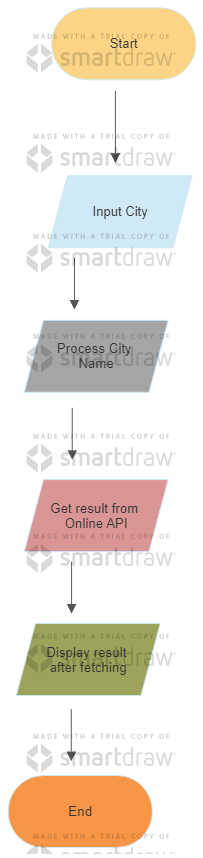
**Code…………………………………………………………………10**

**Working:……………………………………………………………11**

**Limitations and future enhancement……………………….…….12**

**Ending Note:…………………….……………………………….....12**

**References:………………………………………………………….13**

**Flow Chart**

# **Introduction:**

Applying science and technology to forecast the weather is known as weather forecasting. It involves determining the atmosphere's state at a certain location and moment in time. For Millennia, people have tried to forecast the weather informally, and professionally since the 19th Century. Using meteorology to predict how the atmosphere will change at a certain location, Weather predictions are created by gathering quantitative data on the status of the atmosphere, Land, and ocean at the moment.

As weather’s Data are stored and predicted on different API’s, Here we are going to link that data of API in our program (python based weather program) in order to show latest and updated forecast for the public in a better way.

# **Features:**

This Program Tells us about:

* Wind (it gives the blowing rate of air in KPH)
* Temperature (it will show temperature in Centigrade)
* Humidity (a quantity representing the amount of water vapors in the atmosphere or in a gas)
* Pressure
* Description (overcast clouds)

# **Objectives:**

1. This project will provide us with the critical knowledge of python programing.
2. This is a fully functional GUI base project developed in python that covers all of the features that an IT student needs to make a weather forecasting application
3. We will become familiar to interact with Graphical interface within python
4. It will provide practical experience and understanding by working on different projects.

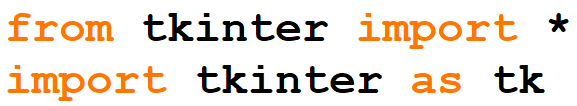
**Tool Software:**

Visual Studio Code (Vs Code)

# **Socio-economic benefits:**

1. It provides information to the people and organization to reduce weather related loses and to ensure health, safety and quality of life.
2. Aircraft and shipping rely heavily on accurate weather forecasting.
3. People can make smart choices about when and where to go on vacation.
4. Sensors will be placed in two different places in river so that pollution comparison can be made.
5. Farmers can be known when to plant or harvest crops.
6. Regions can be evacuated if typhons and floods are excepted.

# **Libraries:**

The default Python interface for the Tcl/Tk GUI toolkit is the tkinter package ("Tk interface"). The majority of Unix systems, including macOS, as well as Windows machines support Tk and tkinter.

A Python client for numerous well-known geocoding web services is called geopy.

Using third-party geocoders and other data sources, geopy enables Python programmers to quickly find the coordinates of addresses, cities, nations, and landmarks all around the world.

For the OpenStreetMap Nominatim, Google Geocoding API (V3), and several more geocoding services, geopy offers geocoder classes. The Geocoders doc section has the whole list accessible. Geopy.geocoders contains geocoder classes.



Your apps can show message boxes by using the tkMessageBox module. You can utilise a variety of the capabilities offered by this module to show the proper message.

Showinfo, ShowWarning, ShowError, AskQuestion, AskOKCancel, AskYesNo, and AskRetryIgnore are a few of these features.



This is a quick and lightweight python tool for seeking up the matching timezone for specified coordinates on earth fully offline.

TimezoneFinder is a lite version of the TimezoneFinder class. Without consuming as much computing resources, it is helpful for swiftly recommending likely timezones (cf. speed tests). This class rapidly retrieves the timezone based only on precomputed "shortcuts" rather than timezone polygon data.



Python does not have a data type for dates, but we may import the datetime module to interact with dates as date objects. Year, month, day, hour, minute, second, and microsecond are all included in the date.

There are several ways to get data about a date object from the datetime module.



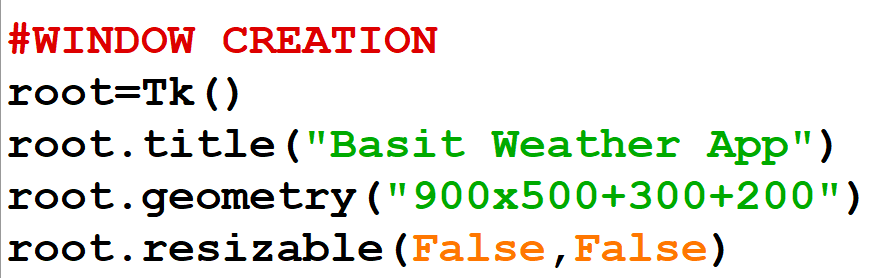
The requests module allows you to send HTTP requests using Python.

The whole response data is returned in a Response Object by the HTTP request (content, encoding, status, etc).

Python's pytz module integrates the Olson tz database. Python versions 2.4 and up may be used to calculate timezones accurately and across platforms. It also resolves the problem of hazy moments after the conclusion of daylight saving time.



# **Code Snippets Used:**

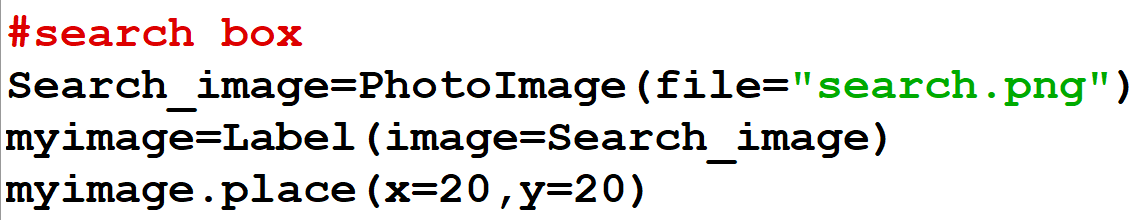
****

Tk(): It controls all the other elements of the tkinter programme and aids in displaying the root window.

Root is the window you create, and root.title("My Title") sets the title of that window.

Geometry(): The size, location, and other characteristics of the screen layout we're going to design are determined by the geometry funvtion, which is a fundamental technique.

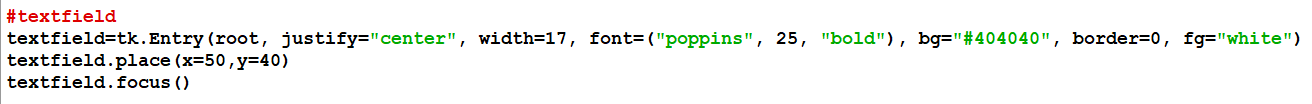
Resizable(): This method is used to allow Tkinter root window to change it's size according to the users need as well we can prohibit resizing of the Tkinter window.

****

PhotoImage(): It returns the image object. Output. That is all there is to displaying a picture in Python. The issue is that the PhotoImage class only supports the GIF and PGM/PPM file formats.

Label(): Picture labelling is the process of locating and indicating different aspects of an image. When creating meta data automatically or giving consumers recommendations based on specifics in their photographs, image tagging is helpful.

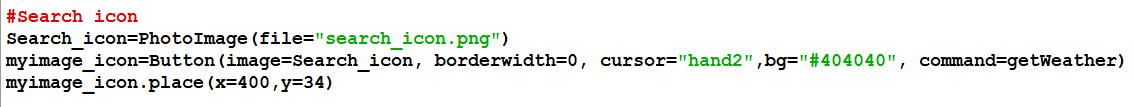
Place(): It enables you to specifically define the location and size of a window, either in absolute terms or in relation to another window.

****

Entry(): In order to receive a value from the user, the Entry widget provides a single line text box. To receive text strings from the user, we may utilise the Entry widget.

Place():  It enables you to specifically define the location and size of a window, either in absolute terms or in relation to another window.

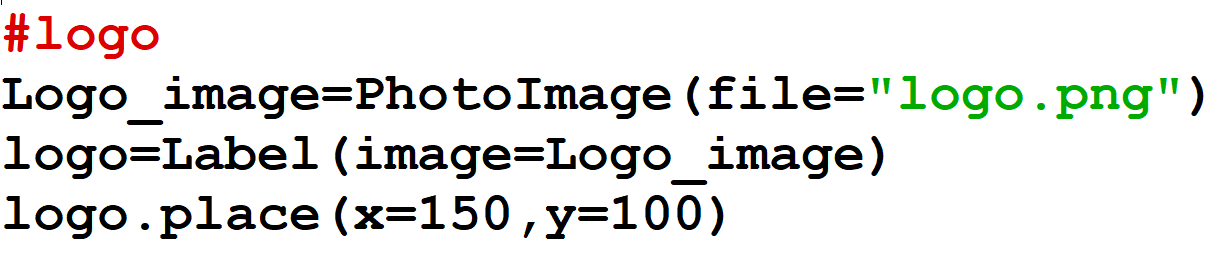
Focus(): Python is a fantastic programming language. Calling the focus() function on a tkinter widget instance we previously generated will allow us to easily set the focus to a particular control when the GUI displays. We used a variable we called nameEntered to hold the instance of the ttk.Entry class in our current GUI example. It can now receive our attention.

****

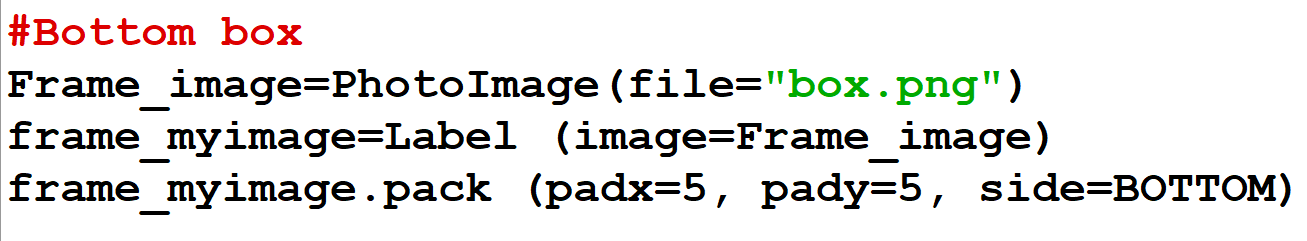
PhotoImage(): It returns the image object. Output. That is all there is to displaying a picture in Python. The issue is that the PhotoImage class only supports the GIF and PGM/PPM file formats.

Button():In a Python programme, buttons are added using the Button widget. These buttons can show text or graphics that explain what they are for. When a button is clicked, you may connect a function or method that is called automatically.

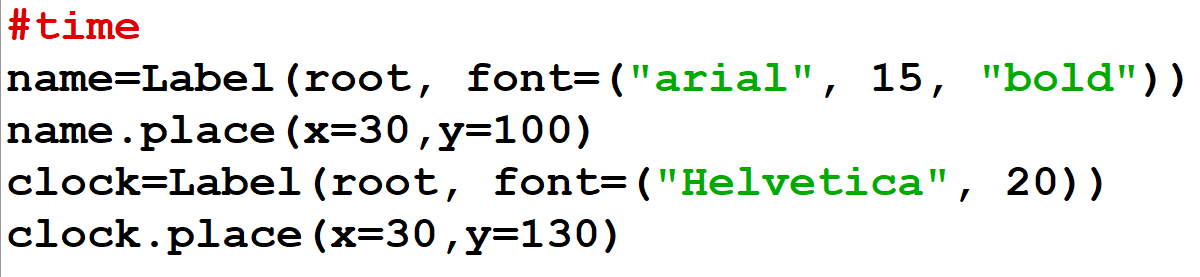
Place():  It enables you to specifically define the location and size of a window, either in absolute terms or in relation to another window.

****

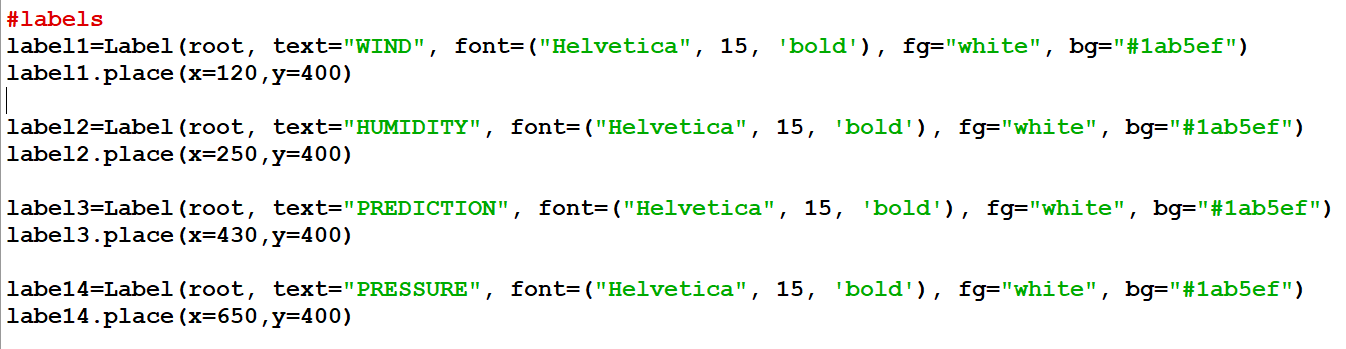
These functions are as the same as upper explained.

****

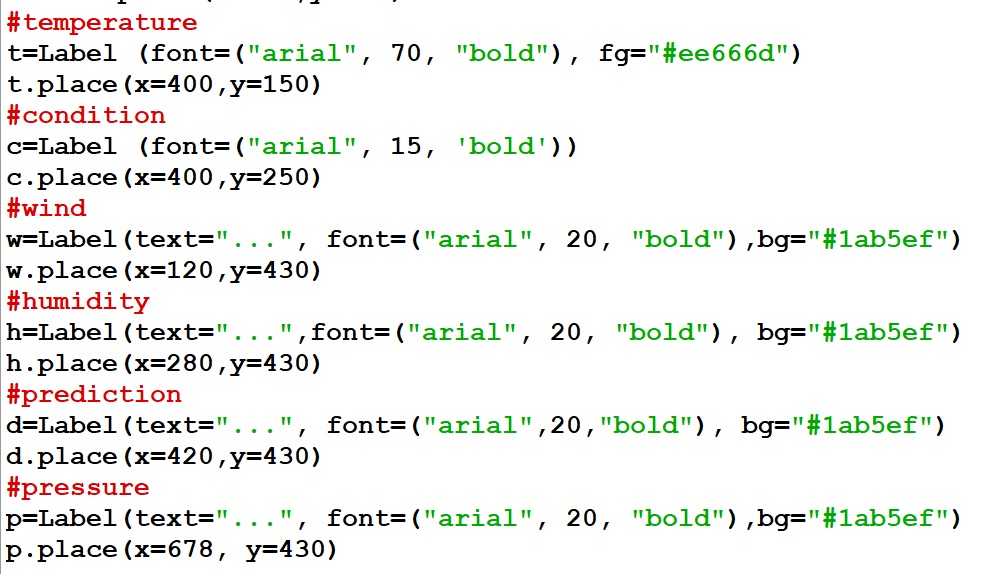
Pack(): Pack, Grid, and Place managers are just a few of the three main geometry managers that Tkinter offers. The pack manager, or widget organization technique in Tkinter, will be covered in this article. The pack manager arranges the widgets in blocks and boxes in the parent widget or window. This manager is used to manage the widgets when they must fit inside the frames, or when they need to be stacked on top of one another or placed side by side.

****

The Label and the place function has already been explained.

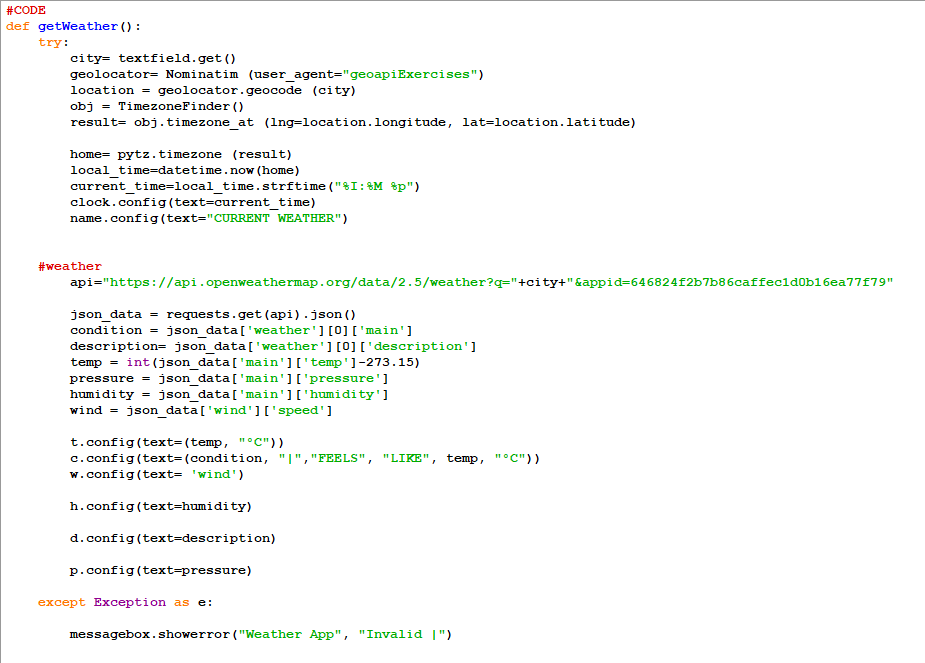
****

Label(): Picture labelling is the process of locating and indicating different aspects of an image. When creating meta data automatically or giving consumers recommendations based on specifics in their photographs, image tagging is helpful.

****

Place(): It enables you to specifically define the location and size of a window, either in absolute terms or in relation to another window.

**Code:**

****

With a start position argument and an optional end argument to determine the end position of the text to be fetched, the get() method of the Tkinter Text widget allows users to retrieve input from text boxes. Python.

The Latin word "nominatim" means "by name." Additionally, it provides a tool for searching OpenStreetMap data by location or address (geocoding). Nominatim is a part of the Python library for GeoPy.

Geolocation is another word for location. It recognizes and tracks the location of linked electronic devices using a variety of location technologies, including IP addresses and GPS. We'll make use of the Python Geopy package.

This is a quick and lightweight Python package for completely offline timezone lookup for specified coordinates on earth.

Timezonefinder module is able to find the timezone of any point on earth (coordinates) offline.

This is a quick and lightweight Python package for completely offline timezone lookup for specified coordinates on earth.

current\_time=local\_time.strftime("%I:%M %p")

The strftime() function in R is a built-in function that transforms time into character objects. Use the strftime() function to alter the class of a time object to character.

To retrieve material from a specific resource URI, Python requests are typically utilised.

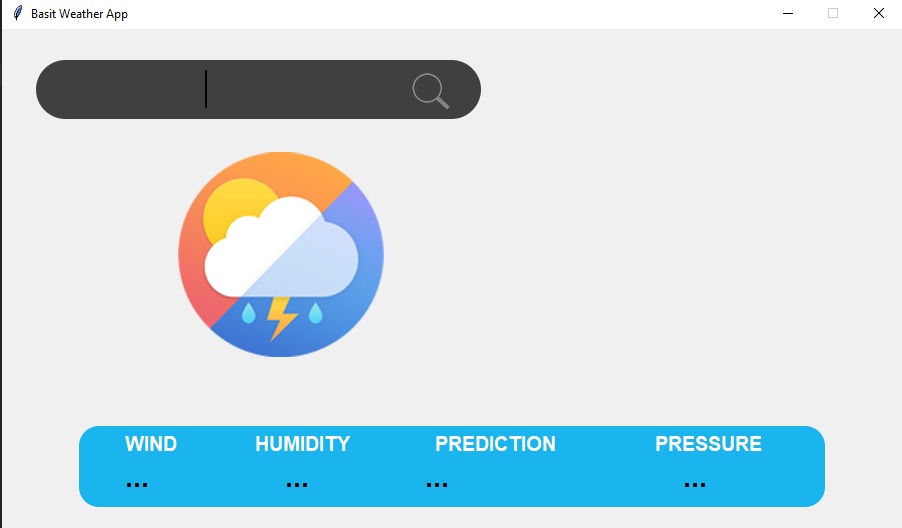
It is obvious that the response returns a json array in which the dictionary at index 0 does not have a key named max . requests.get

Tkinter Config() function, which can be used on any widget to change settings that you may have applied earlier, or haven't applied yet.

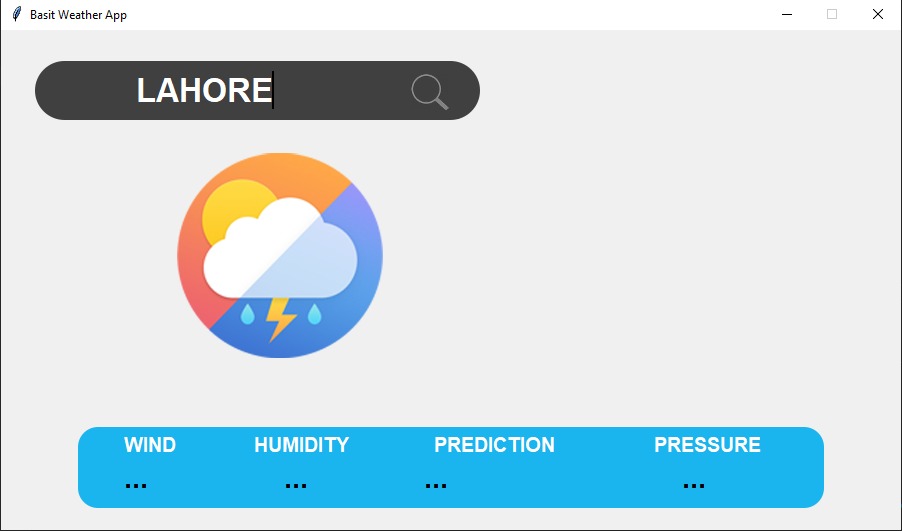
then string slicing is taken place. A slice object is the result of the method slice(). The method for slicing a sequence is specified by a slice object. Both the beginning and ending points of the slicing can be specified. You may optionally define a step that, for instance, lets you slice only the remain.

**Working:**

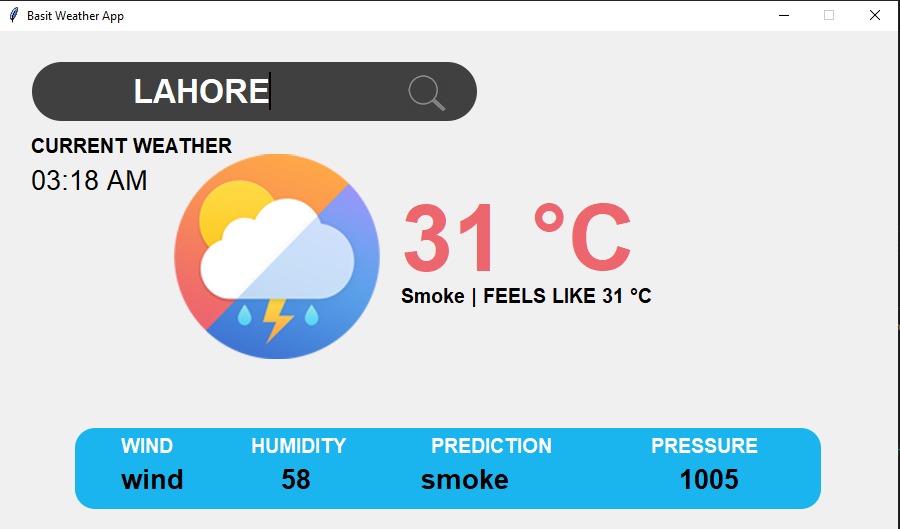
First of all, this interface is shown to us when we open the program which includes images, text fields, labels etc.

****

Then we typed any well-known city name and clicked on searched icon.

****

At the end, we get all the current and predicted info of the weather about that region.

****

**Limitations and future enhancement:**

Although this is a well-equipped Graphical user interface program but it doesn’t provide the facility of 7 days prediction. The reason is the API we have used is free which provides limited services. In future we have plan to enhance this project to be more efficient in order to predict long time weather forecast. This will help mankind for their welfare so that they can be aware of natural disasters and they may take the necessary measures for their safety.

**Ending Note:**

This paper aids in comprehending how the Python weather forecasting system was created. It is a straightforward console-based application that uses machine learning to assist assess if the current weather conditions are suitable for outside activity or not. The system provides information on the wind pressure, temperature, and humidity. It also has other socioeconomic advantages, including Accurate weather predictions are crucial for commerce and aviation, and if tornadoes and floods are excluded, regions can be evacuated. On August 1, 1861, The Times published the first ever daily weather predictions. It comprises a review of the work completed through team cooperation and an explanation of how the weather forecasting system operates with the use of code.

# **References:**

* DataQuest. (2022). 55 Fun (and Unique) Python Project Ideas for Beginners in 2022.
* *GitHub*. (n.d.). Retrieved from https://www.github.com
* Harry, C. W. (2020). Python Project For Absolute Beginners.