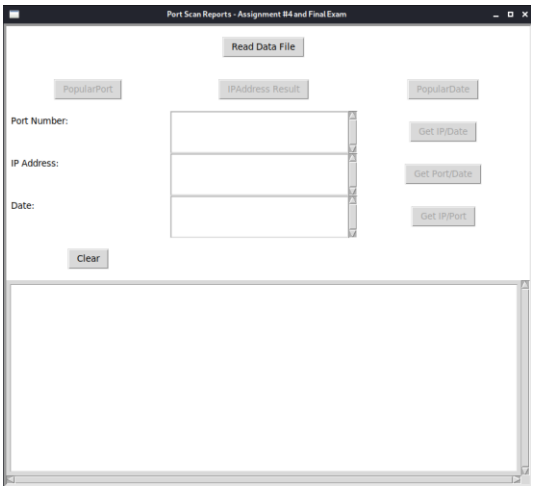
Problem: You are required to use the classes, dictionaries and functions that you developed for Assignment # 3 to develop a Graphical User Interface (GUI) for the work done for Assignment #3 as explained below. You have been sent a port scan history file to your email (***LNamePortScanHistory.csv***). Each student has different data so you have to use the file sent to you. Over the last twenty years or so, as an ethical hacker, you were called to scan ports. The comma separated (CSV) file contains a log of your findings of open ports. (OpenPortNumber, IPAddress,DateofScan).

This Assignment is required to have all the functionality of Assignment # 3, but to use a graphical interface. You have to write a python program that uses breezypythongui (optional) and / or tkinter modules to create a GUI. Specific components are: (with grading rubric):

1. (***10 points***) Design the GUI for the assignment. The GUI shown below is representational, and you are encouraged to design your own GUI. Submit the design as a rough handdrawn sketch picture saved as pdf file ***LastnameA04GUIDesign.pdf***



2. (***15 points***) The GUI has a button for Reading the data file. On clicking, a file dialog is opened and the user is allowed to select the appropriate data file. The choice should be restricted to .py, .txt, .log, and .csv files only. Note that all actions (all other buttons) are disabled unless the data file is read successfully.
3. The program uses three classes (***TrackByCount, TrackByIPAddress, TrackByDate***) of assignment # 3. Each of the classes contains methods and properties to create dictionaries within themselves. The GUI interacts with those classes for the following actions. The result of the actions is displayed in the Text Area of the screen.:
    a. ***(20 points)*** The GUI has a button called ***Popular Port***. On clicking this button, the program uses the ***TrackByCount*** class methods to create a dictionary {OPN:Count,…} where the port number is the key and the number of times that port was found to be open. Items in the dictionary are sorted by port numbers. Save into file **LNamePortCount.csv** where each line contains the port number and count separated by a comma. The most popular open port and the number of times it was found to be open is displayed in the TextArea.

```
The most popular port that was found to be open was:
PortNumber : 49
Count: 484
```

    b. ***(20 points)*** The GUI has a button called ***IPAddressResult***. On clicking this button, the program uses the ***TrackByIPAddress*** class methods to create a dictionary {IPAddress:[(Portnum, Date), …]} where the ipAddress is the key and the value is the list of tuples PortNum, and Date. Items in the dictionary are sorted by IP numbers (string). Save into file **LNameIPAddress.csv** where each line contains the IPAddress, Port number and Date separated by commas. The number of IP Addresses that have only one {Portnum, Date} combination is displayed in the TextArea.

```
The number of IP Addresses that have only one (Portnum, Date) combination is : 104836
```

    c. ***(20 points)*** The GUI has a button called ***PopularDate***. On clicking this button, the program uses the ***TrackByDate*** class methods to create a dictionary {Date:[(IpNum, PortNum),…]} where the Date is the key and the value is the list of tuples IpNum, and PortNum. Items in the dictionary are sorted by Date. Save into file **LNameDate.csv** where each line contains the Date, IPNum, and Port number separated by commas. The date where most PortNumbers were found to be open and how many of them, and the list of them, sorted by Port number, is shown in the TextArea

```
The most Port numbers were found to be open : 31. on Date: 2021-09-27
The ports on that day are:
1
7
19
80
161
161
220
369
389
427
444
520
563
587
```

4. ***(5 points):*** Correct design and test written in docstring. The GUI has a Clear button that clears the TextArea.
5. ***(10 points):*** Submit the output of screen shots of your run as a pdf file (***LastnameA04Output.pdf***). Submit the python file named ***LastnameA04.py*** where *Lastname* is your last name. Submit the three csv output files and the three class python files (rename them as ***LastnameA04TrackbyCount.py***, ***LastnameA04TrackbyIPAddress.py*** ***LastnameA04TrackbyDate.py*** into BB with names as given above.

The student has to design the GUI individually, as per their own design considerations. ***Note:*** (i) Since data file is different for each student, the answers will be different. (ii) Code should be well commented, and have appropriate error checks to prevent crashing. (iii) must use relative path names (iv) Code that **does not use classes and GUI as mentioned above** or that does not meet the intent of the assignment will get 0 points.

Note: Only working programs will be graded using the above rubric. Non-working programs, or those that do not run due to syntactic errors, will get a grade of zero.

Hint: The program uses GUI concepts discussed on pages 276-279, among other concepts of chapter 9. Students are strongly urged to complete the code snippets given in chapter 9 to ensure you have proper practice of using GUI elements, before attempting this assignment.