Programming Project 8

This assignment is worth 50 points (5.0% of the course grade) and must be **completed and turned in before 11:59 on Monday, November 12, 2018.**

Assignment Overview

- Dictionaries
- Lists and Tuples

Assignment Background

Diabetes is a disease that affects the production of insulin, which helps regulate the glucose levels in the blood. Many health related complications like heart disease and obesity have been linked to diabetes. The International Diabetes Federation (IDF) is comprised of over 220 countries and over 230 national diabetes associations in an effort to raise awareness on the problems this disease causes around the world. The IDF has published a diabetes atlas since 2000 that provides data about the prevalence of the disease in seven regions: North America and Caribbean, South and Central America, Middle East and Northern Africa, Africa, Europe, South East Asia, and the Western Pacific (http://diabetesatlas.org/resources/2017-atlas.html).

For this project, you will create a program that will store the information of diabetes prevalence, both type-1 and type-2, in over 220 countries within a dictionary. From this dictionary, you need to select a region to visualize the number of people with diabetes, their age group distribution, and gender distribution for all the countries covered by the selected region. You will use tables, bar charts and pie charts for data visualization.

Project Specifications

1. You must implement the following functions:

    a) **open_file()** prompts the user to enter a filename containing the diabetes prevalence data. The program will try to open a file. An error message should be shown if the file cannot be opened. This function will loop until it receives proper input and successfully opens the file. It returns a file pointer. On most computers you will need to include the following argument in your open statement so the Unicode characters in these files can be read: `encoding ="windows-1252"` e.g.
       `fp = open("diabetes_data_small.csv",encoding ="windows-1252")`

    (Likely the original csv file came from an Excel spreadsheet on a MS Windows computer.)

    b) **create_dictionary(file)** This function receives the opened file object, reads the file, and creates the dictionary containing the diabetes prevalence information. The dictionary will have three keys (region, country, and age group).   That is, it is a dictionary of

dictionaries of dictionaries of lists of tuples. The function returns the dictionary. For each line in the file, you need to read the following:

```
country = line_list[1]
region = line_list[2]
age_group = line_list[3]
gender = line_list[4]
geographic_area = line_list[5]
diabetes = int(float(line_list[6])*1000)
population = int(float(line_list[7])*1000)

tup = (gender, geographic_area, diabetes, population)
# there is more needed here to properly set up the dictionary D
D[region][country][age_group].append(tup)
```

c) **get_country_total(data)** This function receives a dictionary from a specific region, and returns a new dictionary of tuples with the number of people with diabetes and the total population for each country—the country name is the key, the tuple has total number of people with diabetes followed by the total population for the country.

d) **display_table(data, region)** This function receives a dictionary with the diabetes data for a specific region (the data returned from the `get_country_total` function), and the full name of the region. This function returns nothing. It displays (**In alphabetical order!**) the country name, number of people with diabetes, and the total population of that country. There are two header lines and at the bottom is one line with totals for diabetes and population. Truncate country names to 24 characters. The formatting string for entries in the table is: `"{:<25s}{:>20,d}{:>16,d}"`

e) **prepare_plot(data)** This function receives a dictionary for a specific region. We want to plot the region by age group and gender, that is, combine all the countries in the region's data so it can be displayed by age group and gender. It returns a new dictionary with the age group and gender as its keys—the value totals all the countries in the region's data for that age group and gender. This data will be used to plot the bar chart and pie chart that visualizes the age group distribution per gender, and the gender distribution of people with diabetes respectively for all the countries in a specific region.

f) **plot_data(plot_type,data,title)** This function receives a string indicating which plot will be used (`"BAR"` or `"PIE"`), a dictionary with the diabetes data for a specific region (the dictionary returned by the `prepare_plot` function), and the title of the plot. This function returns nothing. **This function is already provided for this project.**

g) **main()** The main function of the program. This function begins by prompting for and opening a file. Next, this function will call the `create_dictionary()` function to

build the data dictionary. Then it will prompt the user for a region to visualize the data. If the region is a valid region, the program will display a table of the diabetes prevalence for each country represented by the region (the program will continue prompting until a valid region is entered). Then, the program will prompt the user whether they want to display the age group and gender distributions of the data and plot the data. **For this prompt, the program only accepts yes/no answers!** The program will stop once the user enters **"quit"** for the region prompt. Keep in mind that the program evaluates "Quit" and "qUIt" and "quiT" as the same word (**Hint: Use the upper() function**)

2. Hints and Suggestions

    a) Dictionaries can contain other elements like float, list, and even other dictionaries. How to access a dictionary with multiple keys? Just like any other dictionary:

        Dictionary = {'KeyA' :{ 'Key1': value_list1, 'Key2': value_list2}}

        Dictionary[KeyA][Key2] would give value_list2

    b) Dictionaries, just like lists, are not always sorted. One way to sort a dictionary is by having a sorted list of keys. To get the list of keys on a dictionary, use the keys() function.

    c) Entering a non-existing or incorrect region can be a very troublesome **error**! Python has a way to **try** these conditions and avoid such errors.

Deliverables

The deliverable for this assignment is the following file:

    proj08.py – the source code for your Python program

Be sure to use the specified file name and to submit it for grading via the **Mimir system** before the project deadline.

Sample Output:

Function Test create_dictionary

Reads `diabetes_data_tiny.csv`

Returns `{'MENA': {'Afghanistan': {'20-24': [('Female', 'Urban', 6786, 442695), ('Male', 'Urban', 2699, 474429)], '35-39': [('Female', 'Urban', 17834, 237228), ('Male', 'Urban', 14852, 262910)], '50-54': [('Female', 'Urban', 21715, 117219), ('Male', 'Urban', 23055, 126786)], '65-69': [('Female', 'Urban', 14604, 57459), ('Male',`

'Urban', 11716, 51738)]}, 'Egypt': {'20-24': [('Female', 'Urban', 60631, 1649047), ('Male', 'Rural', 53521, 2241365)], '35-39': [('Female', 'Rural', 180200, 1841436), ('Male', 'Rural', 153985, 1907127)], '50-54': [('Female', 'Rural', 256800, 1127517), ('Male', 'Rural', 185168, 1130588)], '65-69': [('Female', 'Urban', 251500, 477673), ('Male', 'Urban', 159297, 429078)]}}, 'WP': {'Australia': {'20-24': [('Female', 'Rural', 192, 82406), ('Male', 'Urban', 1550, 764040)], '35-39': [('Female', 'Urban', 11780, 733000), ('Male', 'Rural', 1424, 84157)], '50-54': [('Female', 'Rural', 5219, 81662), ('Male', 'Rural', 6533, 81384)], '65-69': [('Female', 'Urban', 83815, 554127), ('Male', 'Urban', 92794, 542297)]}, 'French Polynesia': {'20-24': [('Female', 'Rural', 643, 5226), ('Male', 'Urban', 504, 7007)], '35-39': [('Female', 'Rural', 1016, 4416), ('Male', 'Rural', 807, 4500)], '50-54': [('Female', 'Urban', 1610, 5044), ('Male', 'Rural', 1287, 4363)], '65-69': [('Female', 'Urban', 825, 2391), ('Male', 'Urban', 792, 2374)]}}}

Function Test get_country_total

input data: {'Afghanistan': {'20-24': [('Female', 'Urban', 6786, 442695),
    ('Male', 'Urban', 2699, 474429)],
   '35-39': [('Female', 'Urban', 17834, 237228),
    ('Male', 'Urban', 14852, 262910)],
   '50-54': [('Female', 'Urban', 21715, 117219),
    ('Male', 'Urban', 23055, 126786)],
   '65-69': [('Female', 'Urban', 14604, 57459),
    ('Male', 'Urban', 11716, 51738)]},
 'Egypt': {'20-24': [('Female', 'Urban', 60631, 1649047),
    ('Male', 'Rural', 53521, 2241365)],
   '35-39': [('Female', 'Rural', 180200, 1841436),
    ('Male', 'Rural', 153985, 1907127)],
   '50-54': [('Female', 'Rural', 256800, 1127517),
    ('Male', 'Rural', 185168, 1130588)],
   '65-69': [('Female', 'Urban', 251500, 477673),
    ('Male', 'Urban', 159297, 429078)]}}

Returns:
{'Afghanistan': (113261, 1770464), 'Egypt': (1301102, 10803831)}

Function Test prepare_plot

Input:
{'Australia': {'20-24': [('Female', 'Rural', 192, 82406),
    ('Male', 'Urban', 1550, 764040)],
   '35-39': [('Female', 'Urban', 11780, 733000),
    ('Male', 'Rural', 1424, 84157)],
   '50-54': [('Female', 'Rural', 5219, 81662), ('Male', 'Rural', 6533, 81384)],
   '65-69': [('Female', 'Urban', 83815, 554127),

```
    ('Male', 'Urban', 92794, 542297)]},
 'French Polynesia': {'20-24': [('Female', 'Rural', 643, 5226),
    ('Male', 'Urban', 504, 7007)],
  '35-39': [('Female', 'Rural', 1016, 4416), ('Male', 'Rural', 807,
4500)],
   '50-54': [('Female', 'Urban', 1610, 5044), ('Male', 'Rural', 1287,
4363)],
   '65-69': [('Female', 'Urban', 825, 2391), ('Male', 'Urban', 792,
2374)]}}
```

Returns:
{'50-54': {'MALE': 7820, 'FEMALE': 6829}, '35-39': {'MALE': 2231,
'FEMALE': 12796}, '65-69': {'MALE': 93586, 'FEMALE': 84640}, '20-24':
{'MALE': 2054, 'FEMALE': 835}}

Test Case 1:

```
Please enter a file name: diabetes_data.csv

                Region Codes
    MENA: Middle East and North Africa
    EUR: Europe
    AFR: Africa
    NAC: North America and Caribbean
    SACA: South and Central America
    WP: Western Pacific
    SEA: South East Asia

Enter region code ('quit' to terminate): MENA
     Diabetes Prevalence in Middle East and North Africa
Country Name            Diabetes Prevalence        Population

Afghanistan                    1,032,605        15,428,666
Algeria                        1,782,257        25,704,996
Bahrain                          165,265         1,022,356
Egypt                          8,222,618        54,443,686
Iraq                           1,411,455        18,738,616
Islamic Republic of Iran       4,985,503        55,742,446
Jordan                           408,122         4,302,026
Kuwait                           440,926         2,922,506
Lebanon                          585,350         4,003,266
Libya                            442,474         3,951,516
Morocco                        1,641,878        22,403,366
Oman                             367,675         3,450,516
Pakistan                       7,474,005       107,734,376
Qatar                            259,184         1,844,106
Saudi Arabia                   3,851,963        20,770,146
State of Palestine               168,754         2,415,876
Sudan                          2,247,026        20,642,716
Syrian Arab Republic             705,670         9,947,656
```

```
Tunisia                                        762,172        7,795,546
United Arab Emirates                         1,185,490        7,607,816
Yemen                                          530,469       13,838,856

TOTAL                                       38,670,861      404,711,056
```

Do you want to visualize diabetes prevalence by age group and gender
(yes/no)?: no

```
                Region Codes
    MENA: Middle East and North Africa
    EUR: Europe
    AFR: Africa
    NAC: North America and Caribbean
    SACA: South and Central America
    WP: Western Pacific
    SEA: South East Asia
```

Enter region code ('quit' to terminate): saca
      Diabetes Prevalence in South and Central America
Country Name   Diabetes Prevalence   Population

Country Name	Diabetes Prevalence	Population
Argentina	1,757,462	28,530,256
Bolivia	391,014	6,283,026
Brazil	12,465,808	143,348,476
Chile	1,199,785	12,885,856
Colombia	2,671,421	32,796,946
Costa Rica	319,109	3,361,356
Cuba	897,578	8,481,476
Dominican Republic	520,740	6,447,876
Ecuador	554,451	10,144,616
El Salvador	332,644	3,835,546
French Guiana	13,034	160,356
Guatemala	752,634	8,909,396
Honduras	285,767	4,789,336
Nicaragua	373,387	3,738,996
Panama	215,906	2,550,368
Paraguay	297,942	4,052,596
Peru	1,130,754	20,094,526
Puerto Rico	400,542	2,596,546
Uruguay	152,745	2,323,166
Venezuela	1,311,414	19,942,026
TOTAL	26,044,137	325,272,742

Do you want to visualize diabetes prevalence by age group and gender
(yes/no)?: no

```
                Region Codes
    MENA: Middle East and North Africa
    EUR: Europe
    AFR: Africa
```

```
     NAC: North America and Caribbean
     SACA: South and Central America
     WP: Western Pacific
     SEA: South East Asia

Enter region code ('quit' to terminate): quit
```

Test Case 2:

```
Please enter a file name: xxx
File not found. Please enter a valid file name: Diabetes
File not found. Please enter a valid file name: diabetes_data.csv
Please enter a file name: xxx
File not found. Please enter a valid file name: Diabetes
File not found. Please enter a valid file name: diabetes_data.csv

                 Region Codes
     MENA: Middle East and North Africa
     EUR: Europe
     AFR: Africa
     NAC: North America and Caribbean
     SACA: South and Central America
     WP: Western Pacific
     SEA: South East Asia

Enter region code ('quit' to terminate): This is America!
Error with the region key! Try another region

                 Region Codes
     MENA: Middle East and North Africa
     EUR: Europe
     AFR: Africa
     NAC: North America and Caribbean
     SACA: South and Central America
     WP: Western Pacific
     SEA: South East Asia

Enter region code ('quit' to terminate): NAC
     Diabetes Prevalence in North America and Caribbean
Country Name               Diabetes Prevalence        Population

Anguilla                                1,273            9,670
Antigua and Barbuda                     8,443           62,166
Aruba                                  11,234           76,626
Bahamas                                37,838          279,456
Barbados                               35,593          202,866
Belize                                 31,497          215,116
Bermuda                                 6,636           43,770
British Virgin Islands                  2,816           20,236
```

```
Canada                                 2,603,183        27,141,126
Cayman Islands                             5,437            39,969
Curaçao                                   18,720           114,246
Dominica                                   6,001            47,586
Grenada                                    6,465            68,316
Guadeloupe                                53,521           319,696
Guyana                                    52,359           463,076
Haiti                                    351,384         6,150,276
Jamaica                                  209,319         1,842,476
Martinique                                51,229           281,746
Mexico                                12,030,035        81,413,316
Montserrat                                   446             3,346
Sint Maarten                               3,541            26,050
St Kitts and Nevis                         4,983            36,856
St Lucia                                  14,137           125,626
St Vincent and the Grena                   8,410            72,406
Suriname                                  45,635           351,556
Trinidad and Tobago                      117,400           976,996
US Virgin Islands                         12,281            74,616
United States of America              30,187,454       231,954,966

TOTAL                                 45,917,270       352,414,153
```

Do you want to visualize diabetes prevalence by age group and gender
(yes/no)?: no

```
              Region Codes
    MENA: Middle East and North Africa
    EUR: Europe
    AFR: Africa
    NAC: North America and Caribbean
    SACA: South and Central America
    WP: Western Pacific
    SEA: South East Asia
```

Enter region code ('quit' to terminate): QuIT

Test Case 3: Plot Test

Please enter a file name: diabetes_data.csv

```
              Region Codes
    MENA: Middle East and North Africa
    EUR: Europe
    AFR: Africa
    NAC: North America and Caribbean
    SACA: South and Central America
    WP: Western Pacific
```

    SEA: South East Asia
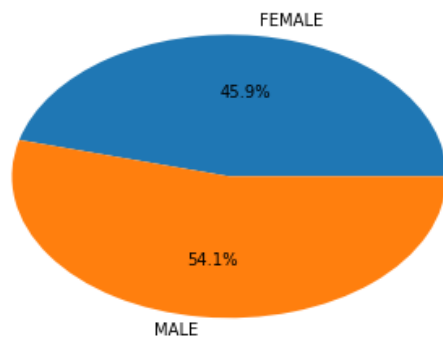
Enter region code ('quit' to terminate): wp
          Diabetes Prevalence in Western Pacific
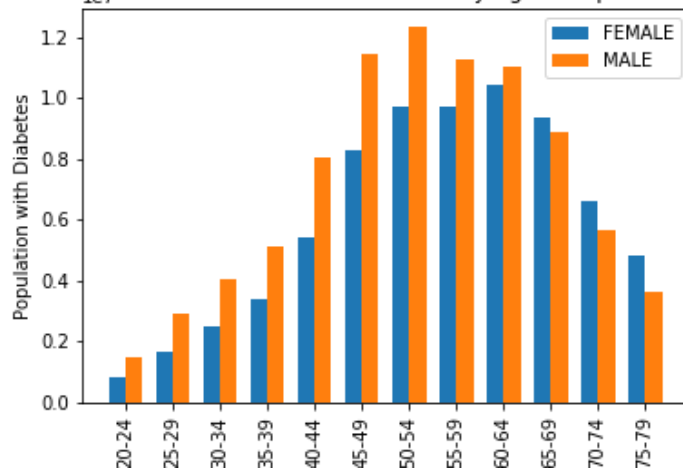Country Name   Diabetes Prevalence   Population

Australia   1,132,949   17,519,716
Brunei Darussalam   41,055   298,116
Cambodia   246,183   9,445,886
China   114,394,800   1,048,632,566
Cook Islands   1,466   12,616
Democratic People's Repu   842,004   17,858,246
Federated States of Micr   6,047   57,256
Fiji   81,703   562,706
French Polynesia   45,396   197,996
Guam   26,040   113,036
Hong Kong China   635,950   5,809,340
Indonesia   10,276,094   166,531,646
Japan   7,234,202   93,683,276
Kiribati   13,028   63,806
Lao People's Democratic   115,199   3,859,976
Macau China   44,947   485,030
Malaysia   3,492,628   20,722,786
Marshall Islands   10,553   32,126
Mongolia   97,735   1,930,826
Myanmar   1,398,932   34,752,706
Nauru   1,488   6,240
New Caledonia   46,155   185,116
New Zealand   326,037   3,213,346
Niue   232   946
Palau   2,331   13,146
Papua New Guinea   639,789   4,172,576
Philippines   3,721,882   60,327,436
Republic of Korea   3,465,422   39,214,336
Samoa   7,389   101,156
Singapore   605,977   4,411,450
Solomon Islands   42,979   302,916
Taiwan   1,957,928   18,017,406
Thailand   4,208,574   50,727,396
Timor L'Este   32,880   568,936
Tokelau   197   790
Tonga   7,234   55,756
Tuvalu   1,746   5,966
Vanuatu   16,142   148,956
Vietnam   3,535,669   64,843,596

TOTAL   158,756,962   1,668,887,124

Do you want to visualize diabetes prevalence by age group and gender
(yes/no)?: yes

```
                Region Codes
   MENA: Middle East and North Africa
   EUR: Europe
   AFR: Africa
   NAC: North America and Caribbean
   SACA: South and Central America
   WP: Western Pacific
   SEA: South East Asia

Enter region code ('quit' to terminate): quit
```

Test Case 4: Blind Test

Grading Rubrics

```
Computer Project #08                                       Scoring Summary

General Requirements:
```

__0__   (7 pts) Coding Standard 1-9

          (descriptive comments, function headers, etc...)

Implementation:

__0__ (3 pts) open_file function (No Mimir test)

       -1 Did not use try/except

       -2 Did not while loop

__0__ (6 pts) create_dictionary function

__0__ (6 pts) get_country_total function

__0__ (6 pts) prepare_plot function

__0__ (6 pts) Pass Test1

__0__ (6 pts) Pass Test2

__0__ (5 pts) Pass Test3: Plot Test (no Mimir test)

__0__ (5 pts) Pass Test4: Blind Test