

Objective:

The objective of this program is to parse and display a graph from a csv file. This is accomplished by using numpy, pandas, matplotlib and sys packages. Upon starting the program the user is given a menu to choose to go into the population menu or housing menu. From here the user can choose to display data from Apr 1, Jul 1 or the changes in population. (NOTE: the data sets from the csv files are unedited in their original format. In the current file the data does display some errors due to outlying data.)

Test cases:

Below are some test cases that were used to test the program.

Test Cases:	User Input:	Predicted Output:	Actual Output:	Pass/Fail:
1	random key input	Invalid input	Invalid input	Pass, fig 1
2	3	Invalid input	Invalid input	Pass, fig 1
3	1	You're in population, select column	You're in population, select column	Pass, fig 1
4	1	Invalid input	Invalid input	Pass, fig 2
5	random key input	Invalid input	Invalid input	Pass, fig 2
6	A, a	Display graph, count, mean, min and max	Display graph, count, mean, min and max	Pass, fig 2 Outlier in data
7	B, b	Display graph, count, mean, min and max	Display graph, count, mean, min and max	Pass, fig 3 Outlier in data
8	C, c	Display graph, count, mean, min and max	Display graph, count, mean, min and max	Pass, fig 4 Outlier in data
9	0	Exit to main menu	Exit to main menu	Pass, fig 5
10	2	In housing data, select from menu	In housing data, select from menu	Pass, fig 6
11	A,a	Display graph, count, mean, min and max	Display graph, count, mean, min and max	Pass, fig 6 Outlier in data
12	B, b	Display graph, count, mean, min and max	Display graph, count, mean, min and max	Pass, fig 7

13	C, c	Display graph, count, mean, min and max	Display graph, count, mean, min and max	Pass, fig 8
14	D, d	Display graph, count, mean, min and max	Display graph, count, mean, min and max	Pass, fig 9
15	E, e	Display graph, count, mean, min and max	Display graph, count, mean, min and max	Pass, fig 10
16	0	Exit to main menu	Exit to main menu	Pass, fig 11
17	0	Exit program, thank user	Exit program, thank user	Pass, fig 11

Below are the images associated with the above test cases.

```

##### -MAIN MENU- #####
Select the file you want to analyze:

    1: Population Data
    2: Housing Data
    0: Exit the Program

Please select an item: c
##### That is not a valid input! #####
##### Welcome to the Python Data Analysis App #####
##### -MAIN MENU- #####
Select the file you want to analyze:

    1: Population Data
    2: Housing Data
    0: Exit the Program

Please select an item: c
##### That is not a valid input! #####
##### Welcome to the Python Data Analysis App #####
##### -MAIN MENU- #####
Select the file you want to analyze:

    1: Population Data
    2: Housing Data
    0: Exit the Program

Please select an item: c
##### -YOU'RE IN POPULATION DATA- #####
Select the column you want to analyze:

    A: Population Apr 1
    B: Population Jul 1
    C: Change Population
    D: Exit Population Menu

Please select an item: c
  
```

Fig 1

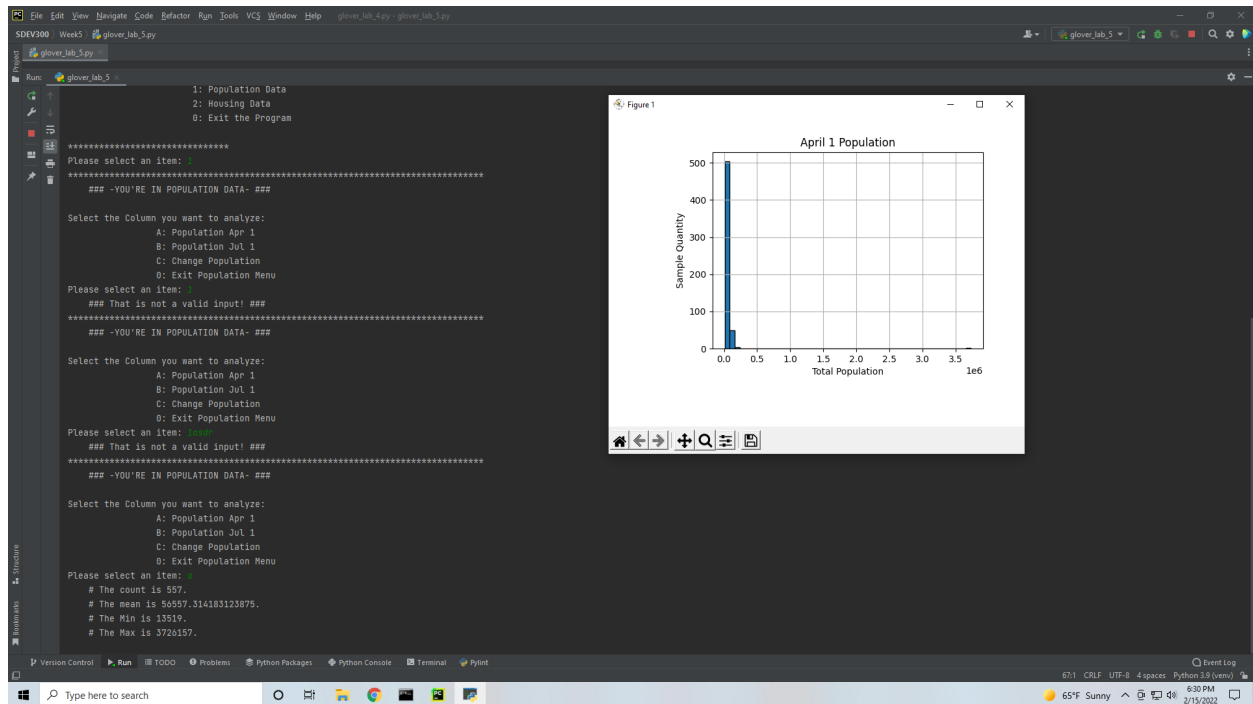


Fig 2, Outlier in data

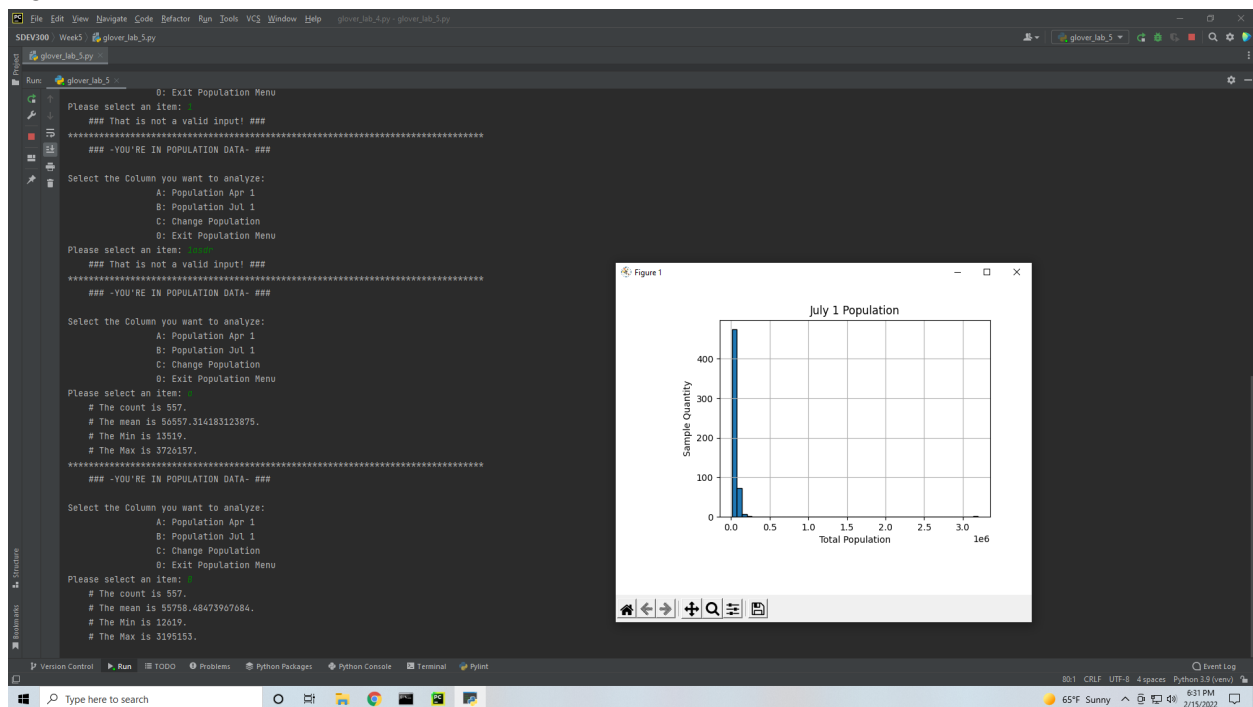


Fig 3, Outlier in data

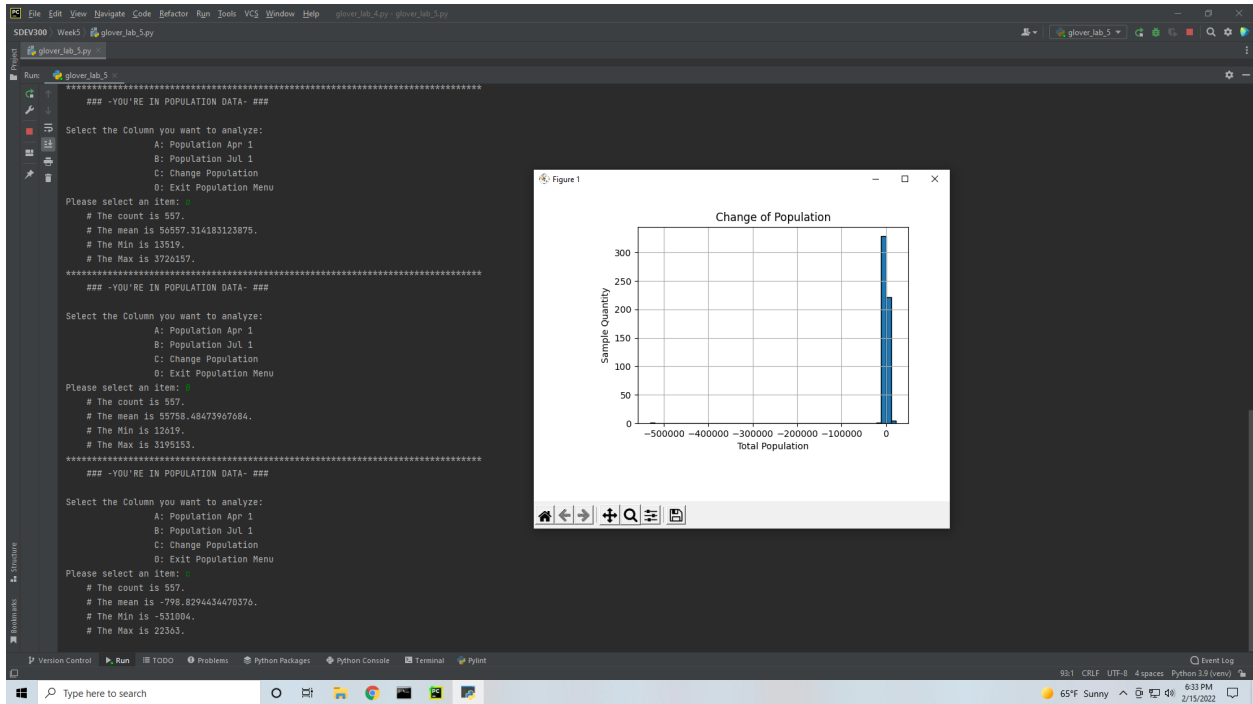


Fig 4, Outlier in data

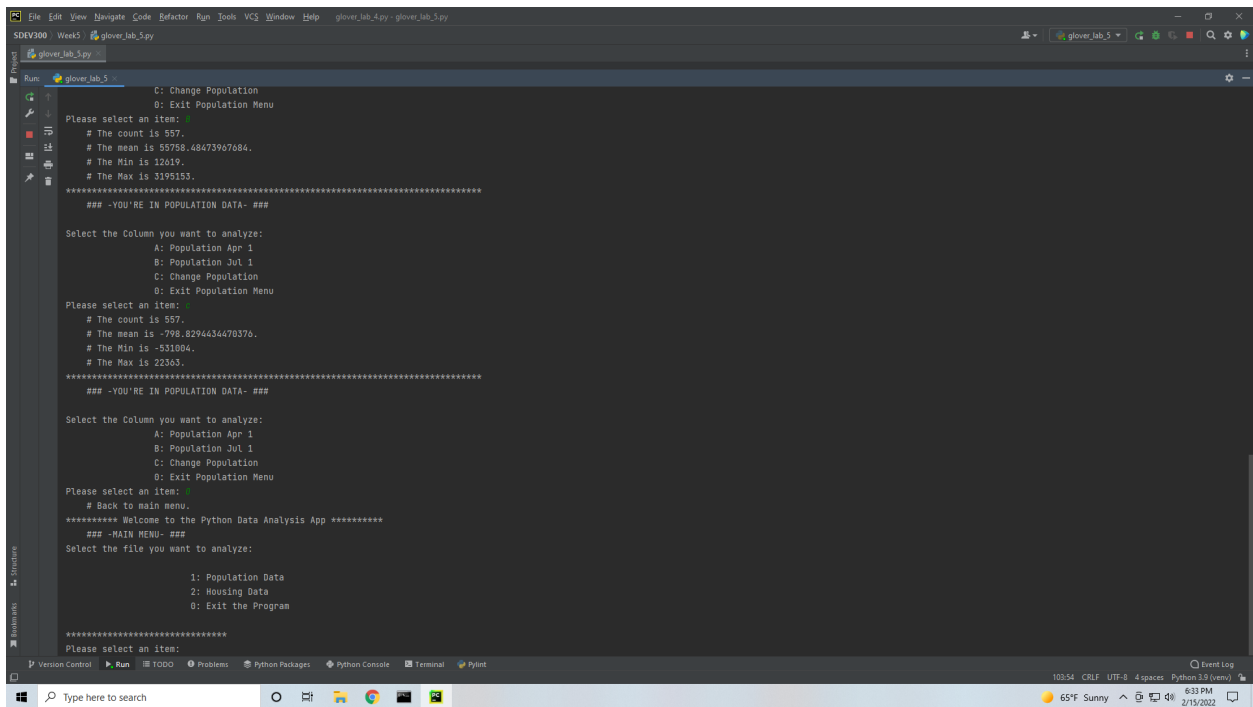


Fig 5

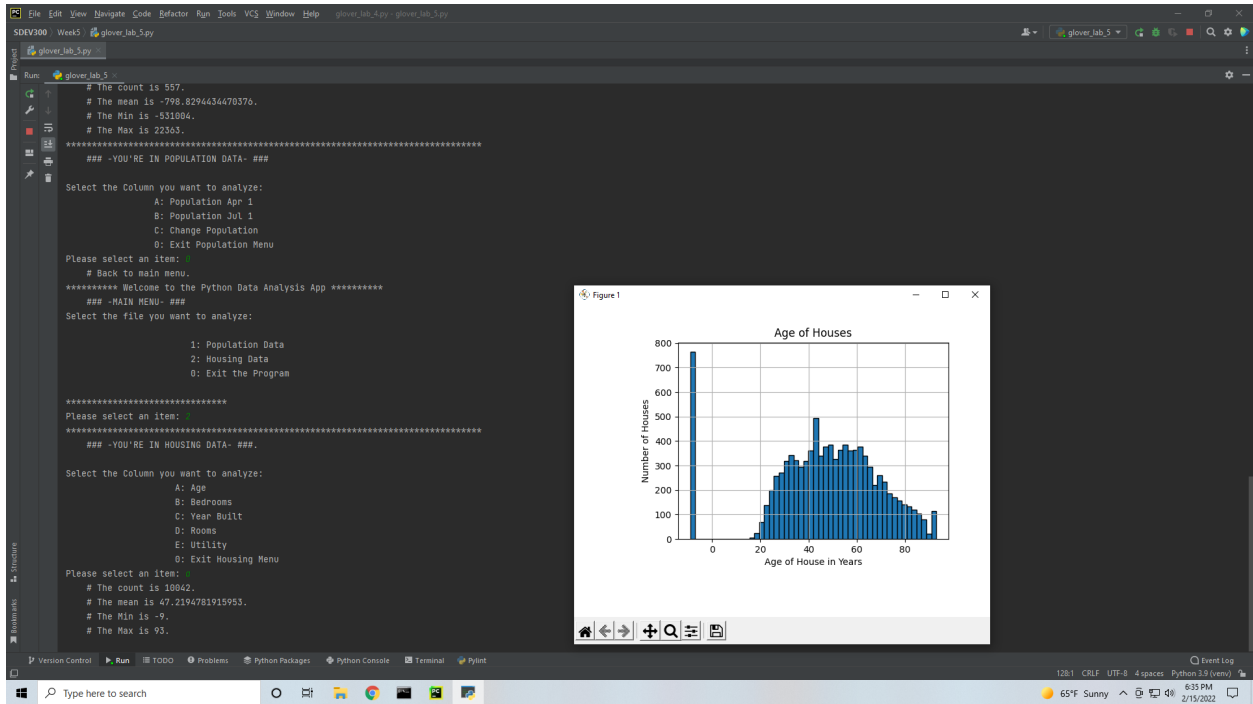


Fig 6, Outlier in data

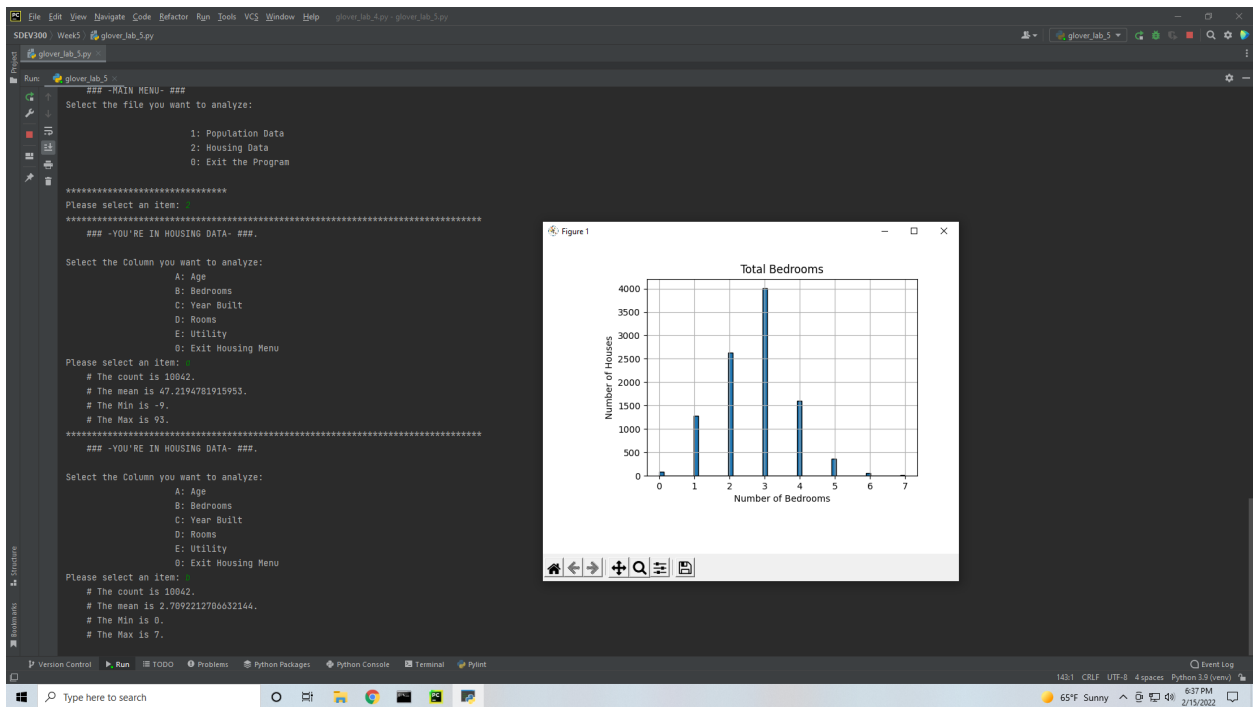


Fig 7

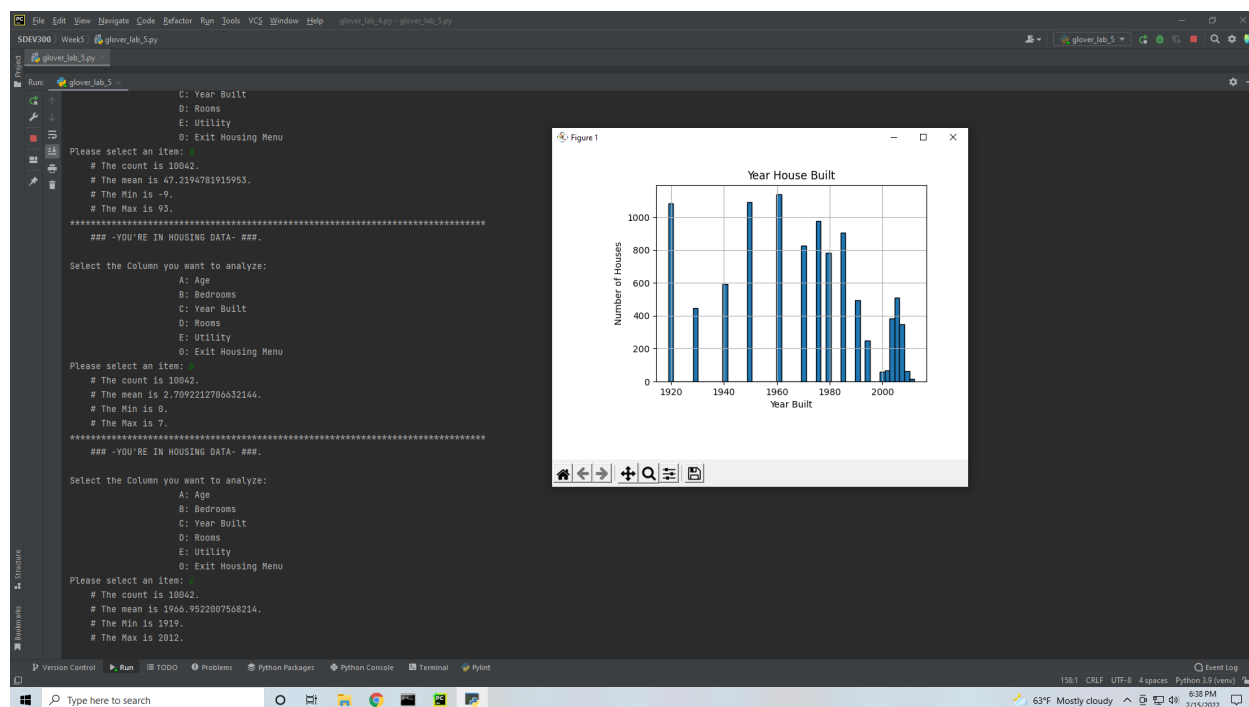


Fig 8

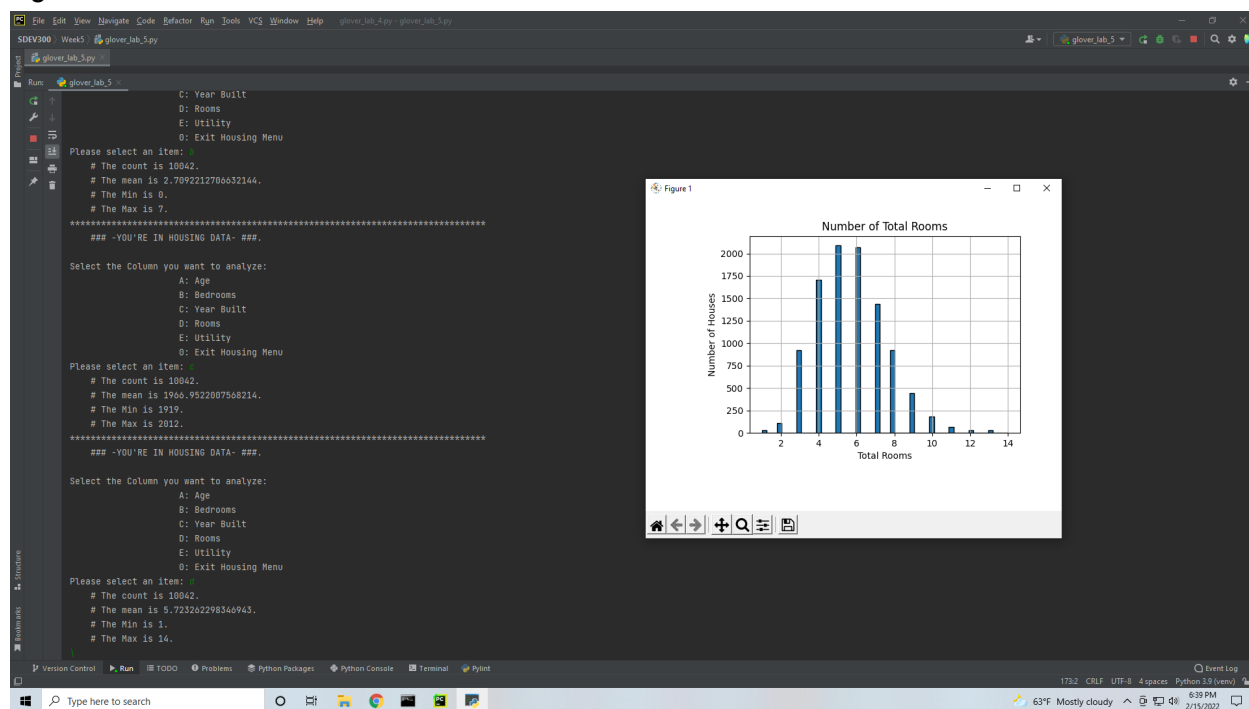


Fig 9

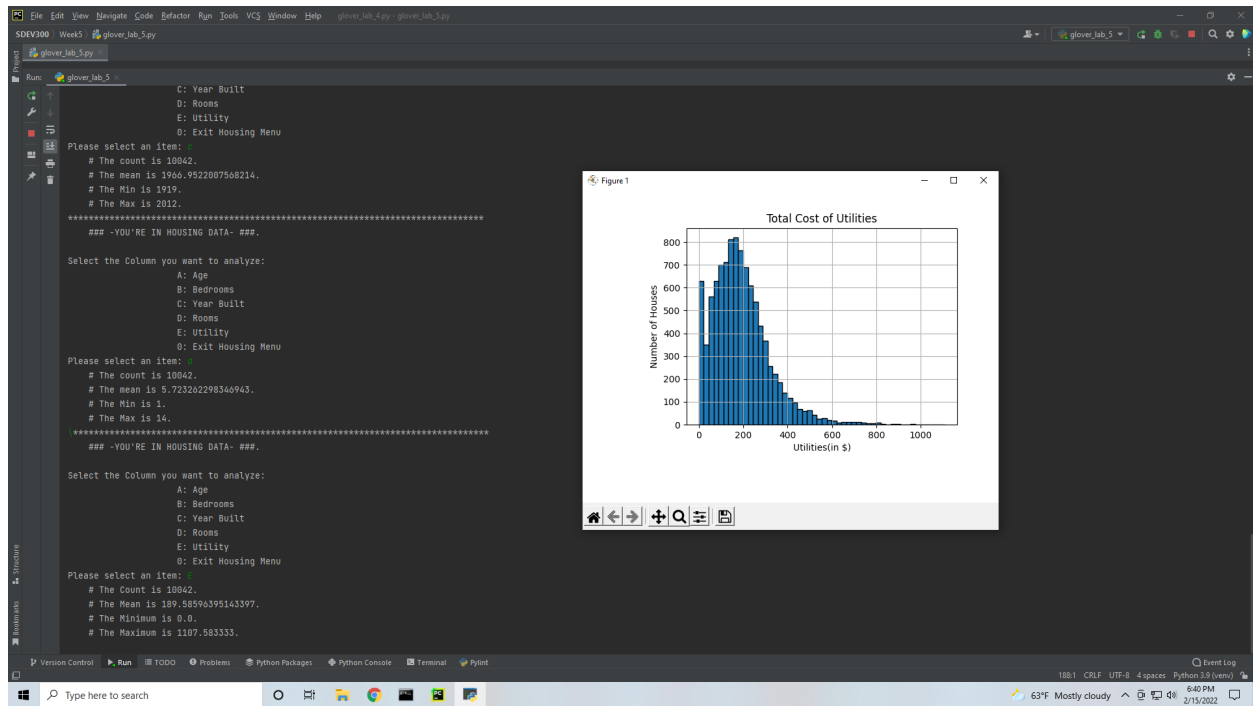


Fig 10

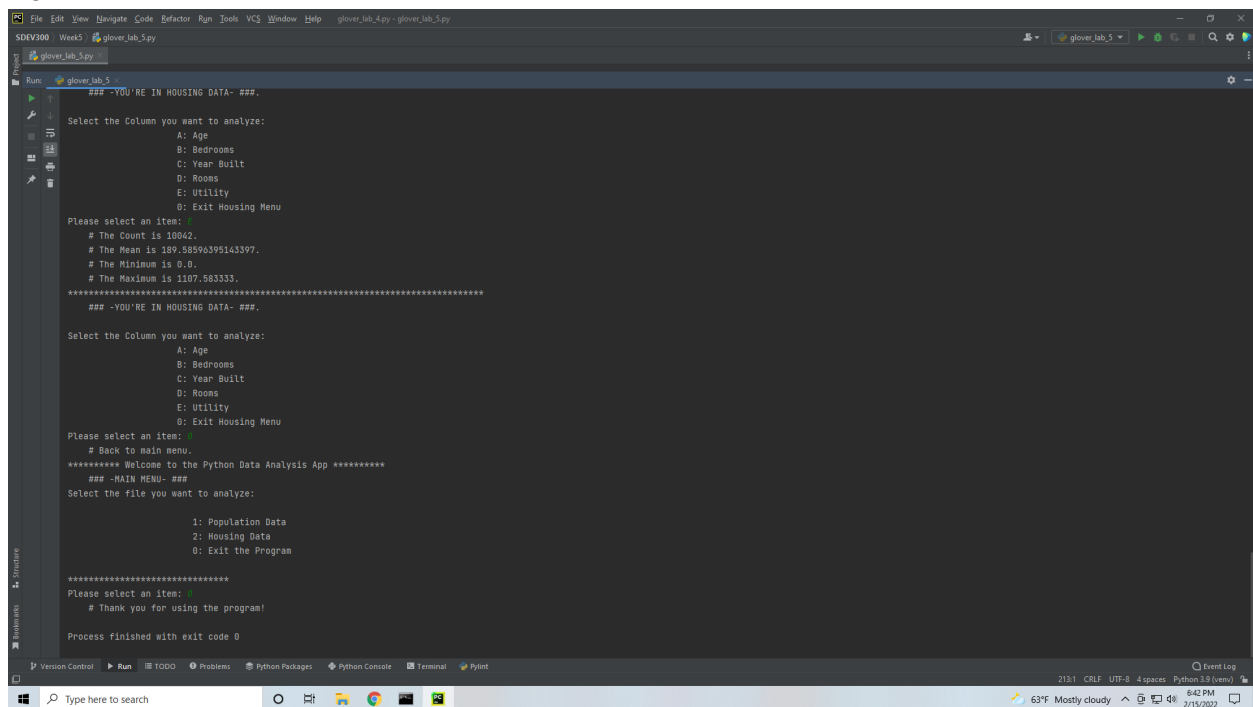
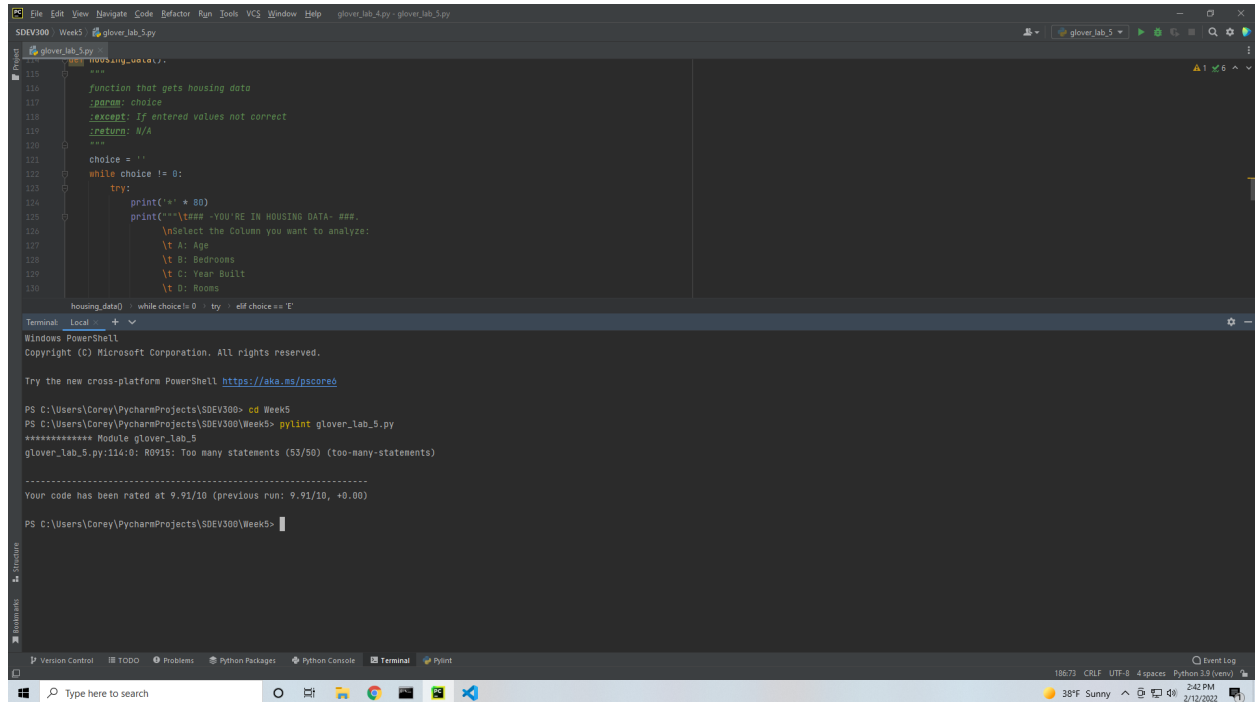


Fig 11

Pylint Testing

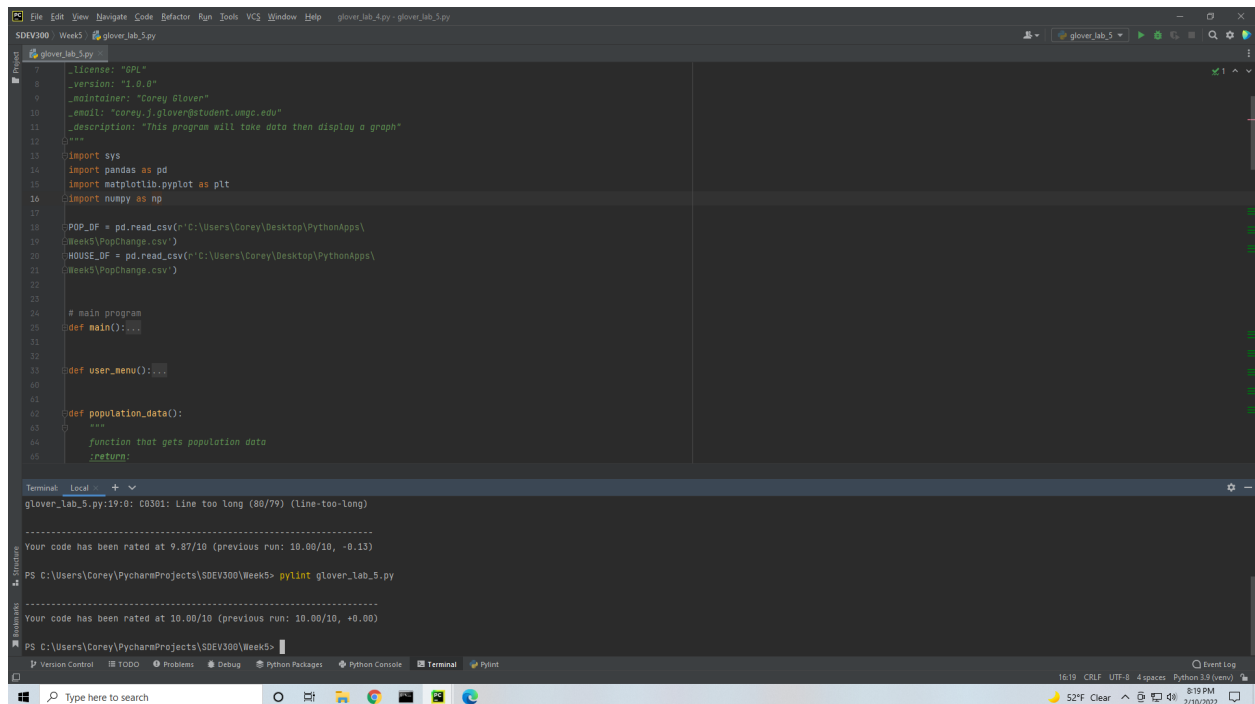
Below are some screenshots while writing, testing and running the program



The screenshot shows the PyCharm IDE with a Python file named `glover_lab_5.py`. The code defines a function `housing_data` that prompts the user to select a column to analyze. The Pylint output in the terminal shows a warning: `***** Module Glover_lab_5 *****` and `glover_lab_5.py:114:8: R0915: Too many statements (53/50) (too-many-statements)`. The code is as follows:

```
115 def housing_data():
116     """
117     function that gets housing data
118     """
119     choice = None
120     while choice != 0:
121         try:
122             print('\n * 80)')
123             print("""\n\n YOU'RE IN HOUSING DATA- $$$.\n\n
124             \nSelect the Column you want to analyze:
125             \n\t A: Age
126             \n\t B: Bedrooms
127             \n\t C: Year Built
128             \n\t D: Rooms
129             \n\n""")
130             choice = input('Enter your choice: ')
131             while choice != 'A' and choice != 'B' and choice != 'C' and choice != 'D':
132                 print('\nInvalid choice. Please enter A, B, C, or D.\n')
133                 choice = input('Enter your choice: ')
134             if choice == 'A':
135                 # Code for Age
136             elif choice == 'B':
137                 # Code for Bedrooms
138             elif choice == 'C':
139                 # Code for Year Built
140             elif choice == 'D':
141                 # Code for Rooms
142             else:
143                 # Code for exit
144                 choice = 0
145         except ValueError:
146             print('\nInvalid input. Please enter a number.\n')
147             choice = input('Enter your choice: ')
148     return choice
```

Pylint states too many statements in section. The was corrected by separating the program in to its multiple functions.



The screenshot shows the PyCharm IDE with a Python file named `glover_lab_5.py`. The code is organized into functions: `main`, `user_menu`, and `population_data`. The Pylint output in the terminal shows a warning: `glover_lab_5.py:19:8: C0301: Line too long (80/79) (line-too-long)`. The code is as follows:

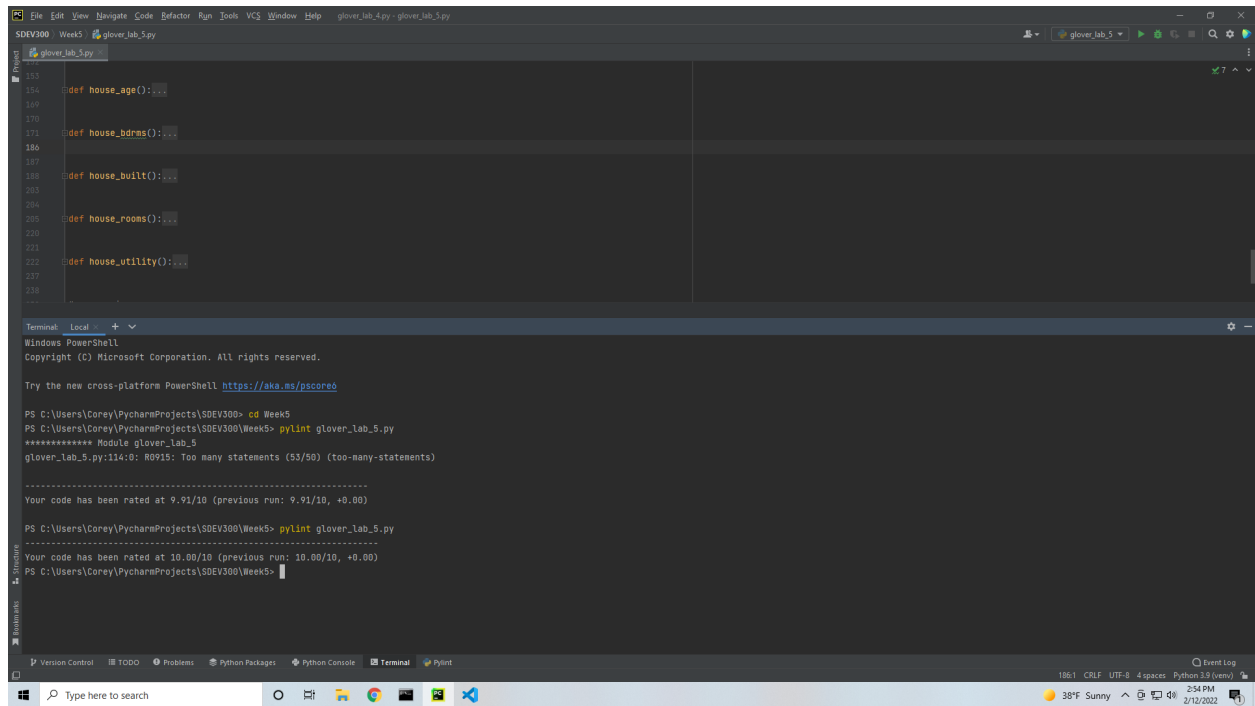
```
1 # License: "GPL"
2 _version: "1.0.0"
3 _maintainer: "Corey Glover"
4 _email: "corey.j.glover@student.umcg.edu"
5 _description: "This program will take data then display a graph"
6
7 import sys
8 import pandas as pd
9 import matplotlib.pyplot as plt
10 import numpy as np
11
12 POP_DF = pd.read_csv("C:\Users\Corey\Desktop\PythonApps\Week5\PopChange.csv")
13 HOUSE_DF = pd.read_csv("C:\Users\Corey\Desktop\PythonApps\Week5\PopChange.csv")
14
15 # main program
16 def main():
17     user_menu()
18
19 def user_menu():
20     """
21     function that gets population data
22     """
23     return
```

The issue here was that the line was too long in comparison to .pylintrc. Returned line to allow pylint to pass.

Name: Corey Glover

SDEV300-6382

Date: Feb 15, 2022



The screenshot shows the PyCharm IDE interface. The top pane displays the code for `glover_lab_5.py`, which contains several function definitions: `def house_age():`, `def house_bdrms():`, `def house_built():`, `def house_rooms():`, and `def house_utility():`. The bottom pane shows a terminal window with the following output:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/powershell

PS C:\Users\Corey\PycharmProjects\SDEV300> cd Week5
PS C:\Users\Corey\PycharmProjects\SDEV300\Week5> pylint glover_lab_5.py
***** Module glover_lab_5
glover_lab_5.py:114:0: R0915: Too many statements (53/50) (too-many-statements)

-----
Your code has been rated at 9.91/10 (previous run: 9.91/10, +0.00)

PS C:\Users\Corey\PycharmProjects\SDEV300\Week5> pylint glover_lab_5.py
-----
Your code has been rated at 10.00/10 (previous run: 10.00/10, +0.00)
PS C:\Users\Corey\PycharmProjects\SDEV300\Week5>
```

Final run of pylint, 10/10