DATA ANALYSIS REPORT

Diabetes Prevalence and Glycemic Control among Adults of Age 20 Years and over, by Sex, Age, Race and Federal Poverty Level in the United States of America in the Selected Years of 2015 – 2018

MAY 10, 2023

YATISHA RAJANALA

G01419023 – HAP-618-003 – FINAL PROJECT REPORT

SPRING 2023

BACKGROUND

Diabetes Mellitus (DM) is an irreversible chronic metabolic disorder, involving inappropriately elevated glucose levels, due to insufficient production of insulin or inadequate receptor sensitivity to the action of insulin. It is one of the top 10 causes of death and one of the fastest growing health problems of the 21st Century, apart from being the prime comorbidity in the contemporary COVID-19 scenario.

In the United States of America, Diabetes Mellitus is a significant public health issue, affecting millions of people and contributing to significant morbidity and mortality. In 2015, according to the Centers for Disease Control and Prevention (CDC), an estimated 9.4% of the population had the disorder, and by 2017, it was the seventh leading cause of death in the U.S.A, with over 83,000 deaths attributed to the disease.

Diabetes Mellitus can lead to a range of complications, including cardiovascular disease, neuropathy (nerve damage), nephropathy (kidney damage), and retinopathy (vision loss). In addition to the human toll, it also has a significant economic impact in 2018, with an estimated $327 billion loss in direct medical costs.

Overall, the high prevalence of Diabetes Mellitus in the United States highlights the ongoing need for effective prevention and management strategies to reduce the burden of this disease on individuals and society. Also, healthcare professionals can adapt certain individualized drug regimens in patients with a high potential of developing side effects or medical emergencies. For this purpose, there have been inclusions of combinational therapies and personalized medicine, using Evidence-Based Medicine (EBM) to maintain homeostasis.

PURPOSE OF REPORT

Diabetes Mellitus is highly prevalent and significant in the United States, posing the following multifaceted problems:

1. The high prevalence of Diabetes Mellitus in the U.S.A. indicates a substantial public health problem, which can lead to serious complications, such as heart disease, kidney failure, blindness, and lower-limb amputations, finally resulting in reduce quality of life and increase healthcare costs.
2. Diabetes Mellitus is associated with a range of common risk factors, including obesity, physical inactivity, and poor diet, making it crucial for healthcare professionals to address them.
3. The disproportionate burden of Diabetes Mellitus among certain demographic groups, including racial and ethnic minorities and individuals living in poverty, highlights the need for targeted interventions to address health disparities and promote health equity.
4. The economic costs of diabetes are significant, and thus addressing the issue is vital for reducing healthcare costs and promoting economic productivity.

The main aim of the Data Analysis Report for "Diabetes Prevalence and Glycemic Control among Adults of Age 20 Years and over, by Sex, Age, Race, and Federal Poverty Level in the United States of America in the Selected Years of 2015 – 2018" is to provide a comprehensive picture of the prevalence and management of diabetes in the United States during the time.

The report is to present detailed information and analysis using Python library functions and HTML interface to identify disparities and highlight areas where targeted interventions may be done to improve outcomes for future reference. Python complies with the Health Insurance Portability and Accountability Act of 1996 (HIPAA) checklist for ensuring Medical Data Safety, and thus, it is the best option for data analysis.

This information, along with the report of present trends, can then be passed onto policymakers, healthcare providers, researchers, and the public in the United States, to provide data-driven insights into strategies for preventing, supporting evidence-based decisions, and developing effective interventions to reduce the burden of Diabetes Mellitus in the United States.

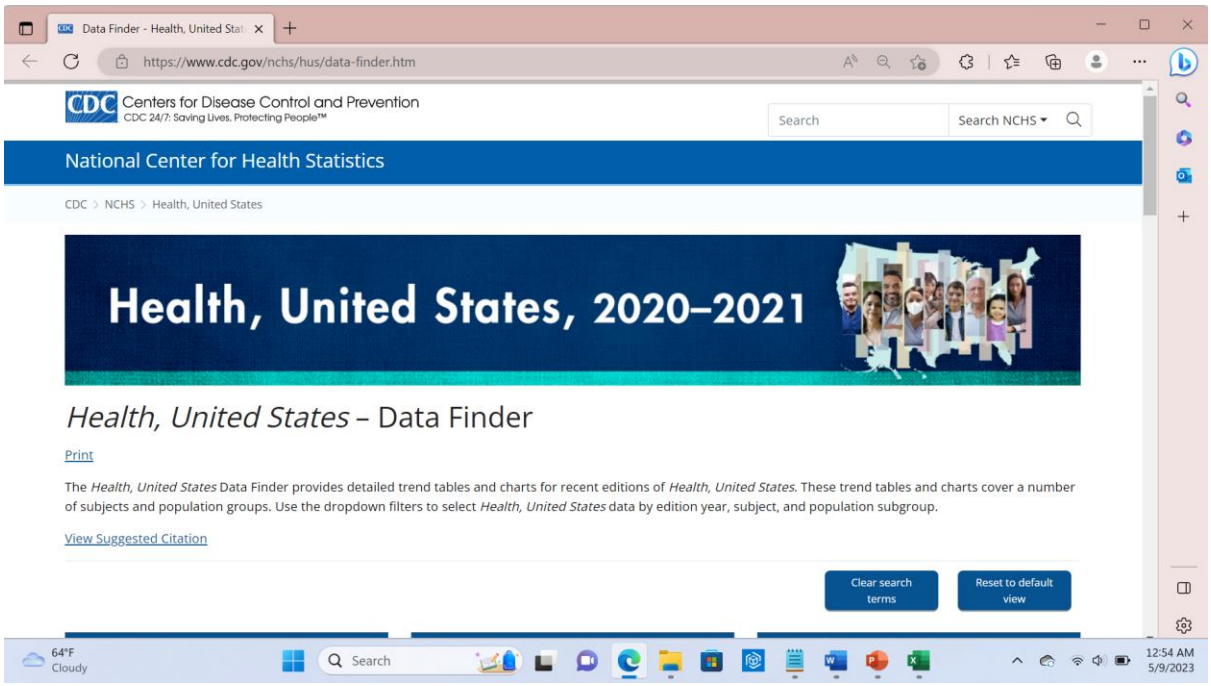
SOLUTION DESIGN

1. Data Collection:

The data was collected from the online dataset provided by Centers for Disease Control and Prevention (CDC) titled, "Diabetes Prevalence and Glycemic Control among Adults of Age 20 Years and over, by Sex, Age, Race and Federal Poverty Level in the United States of America in the Selected Years of 2015 – 2018".

The dataset consists of the different patients' demographic details, specifically their Sex, Age, Race, Percent of Poverty Level, Physician-Diagnosed or Undiagnosed Diabetes, and Extent of Glycemic Control based on the HbA_{1c} levels.

Image 1: Centers for Disease Control and Prevention (CDC) Website



2. Data Cleaning and Consolidation:

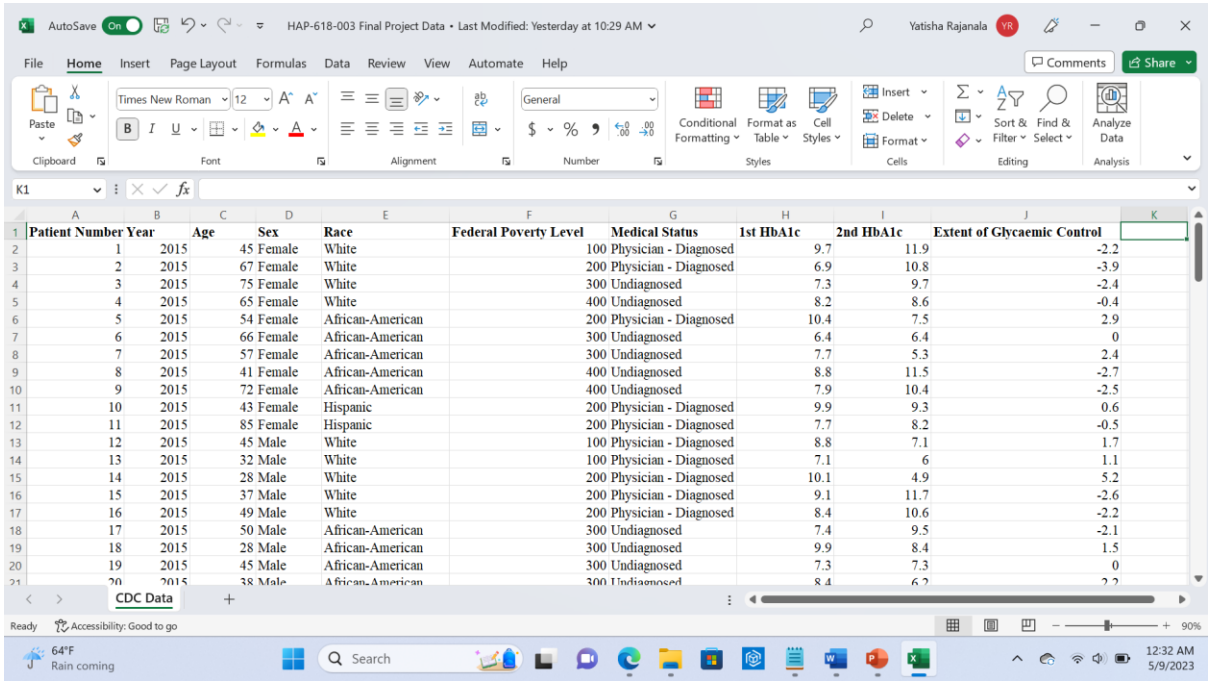
The data was cleaned and consolidated into a CSV File, with discrete rows and columns. The columns include, Patient Number, Year, Age, Sex, Race, Federal Poverty Level, Medical Status, 1st HbA1c, 2nd HbA1c and Extent of Glycemic Control.

3. Data Visualization:

The data from the datasheet is then used to calculate percentage of occurrences, mean and standard deviation, and further to plot pie charts, bar graphs and correlation charts, using

different Python libraries like Pandas, Numpy and Matplotlib which can analyze, compute and animate data, respectively.

Image 2: Cleaned and Consolidate Datasheet



Patient Number	Year	Age	Sex	Race	Federal Poverty Level	Medical Status	1st HbA1c	2nd HbA1c	Extent of Glycaemic Control
1	2015	45	Female	White	100	Physician - Diagnosed	9.7	11.9	-2.2
2	2015	67	Female	White	200	Physician - Diagnosed	6.9	10.8	-3.9
3	2015	75	Female	White	300	Undiagnosed	7.3	9.7	-2.4
4	2015	65	Female	White	400	Undiagnosed	8.2	8.6	-0.4
5	2015	54	Female	African-American	200	Physician - Diagnosed	10.4	7.5	2.9
6	2015	66	Female	African-American	300	Undiagnosed	6.4	6.4	0
7	2015	57	Female	African-American	300	Undiagnosed	7.7	5.3	2.4
8	2015	41	Female	African-American	400	Undiagnosed	8.8	11.5	-2.7
9	2015	72	Female	African-American	400	Undiagnosed	7.9	10.4	-2.5
10	2015	43	Female	Hispanic	200	Physician - Diagnosed	9.9	9.3	0.6
11	2015	85	Female	Hispanic	200	Physician - Diagnosed	7.7	8.2	-0.5
12	2015	45	Male	White	100	Physician - Diagnosed	8.8	7.1	1.7
13	2015	32	Male	White	100	Physician - Diagnosed	7.1	6	1.1
14	2015	28	Male	White	200	Physician - Diagnosed	10.1	4.9	5.2
15	2015	37	Male	White	200	Physician - Diagnosed	9.1	11.7	-2.6
16	2015	49	Male	White	200	Physician - Diagnosed	8.4	10.6	-2.2
17	2015	50	Male	African-American	300	Undiagnosed	7.4	9.5	-2.1
18	2015	28	Male	African-American	300	Undiagnosed	9.9	8.4	1.5
19	2015	45	Male	African-American	300	Undiagnosed	7.3	7.3	0

4. Draw Inferences and Conclusions:

Based on these results and visualizations, conclusions about the trends in data are observed and inferences are made. Also, the steps taken in implementing the python code are mentioned in the report.

5. Collect Results and Finalize Report Outcomes:

The final product is a detailed Data Analysis Report based on the above-mentioned parameters, consisting of percentage of occurrences, mean, standard deviations, and relevant correlations, in the form of visualizations like pie charts, bar graphs and pictures.

These results are then neatly represented on Webpages, so that the data can be comprehensive for both healthcare professionals and the public. The Main Website Link, where the Data Analysis Report is available:

https://www.students.hi.gmu.edu/~yrajanal/main.html

IMPLEMENTATION AND TESTING OF CODES

The Python Codes were written for the Data Analysis Report, using different Python libraries like Pandas, Numpy and Matplotlib which can analyze, compute and animate data. This data visualizations were divided into 3 parts:

1. Part 1: Data Analysis based on Occurrences and Prevalance of Diabetes Mellitus
2. Part 2: Data Analysis based on Extent of Glycaemic Control, using the HbA1c Measured Values
3. Part 3: Data Analysis based on Correlation

Part 1: Data Analysis based on Occurrences and Prevalance of Diabetes Mellitus

The data analysis was done using the Python Libraries of Pandas and Matplotlib to create Pie Charts to visualize the data of Diabetes Mellitus with respect to its Occurrence and Prevalence in the United States of America.

The data was divided on the basis of Years, and new data frames were located and created to achieve the same. After this, the count function was used for each demographic, and accordingly the percentages were set in the pie chart code.

For the demographics of Sex and Race, no grouping of data was done; however, for the Federal Poverty Levels and Age, data grouping was done using the Define: If, Else If and Else functions. Then, the data was plotted for the pie charts.

Image 3: Code for Data Analysis: Part 1 (a)

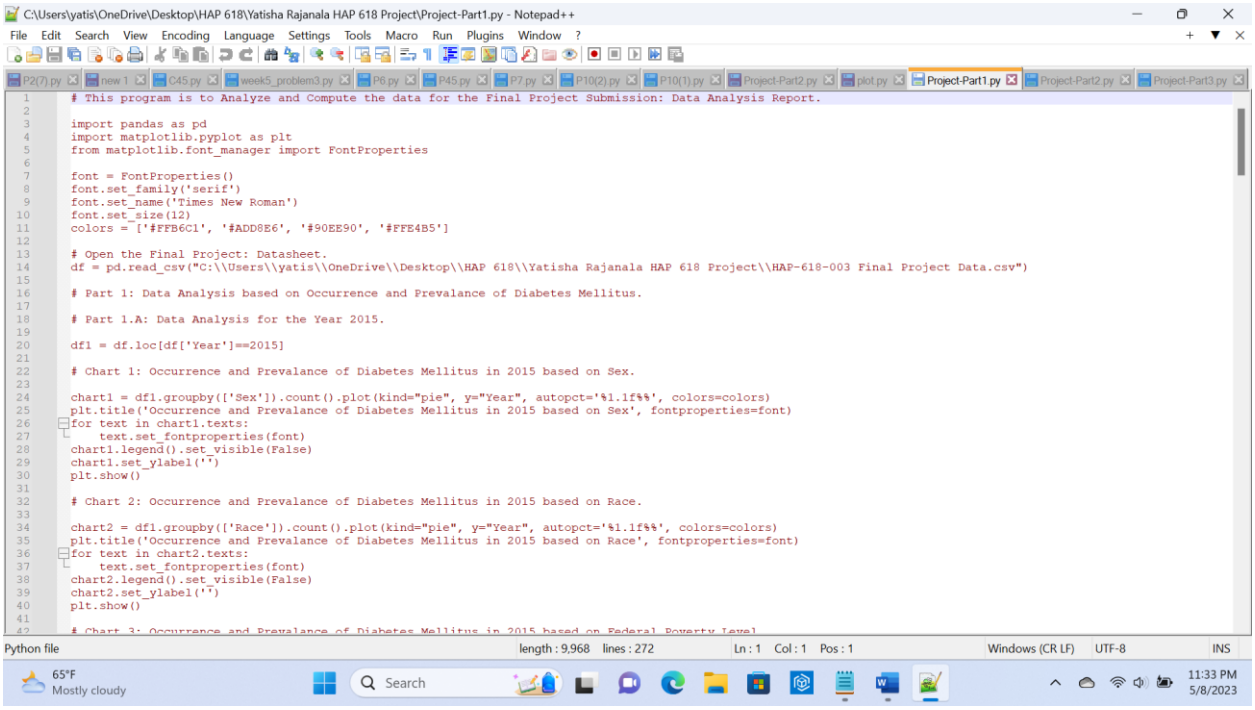
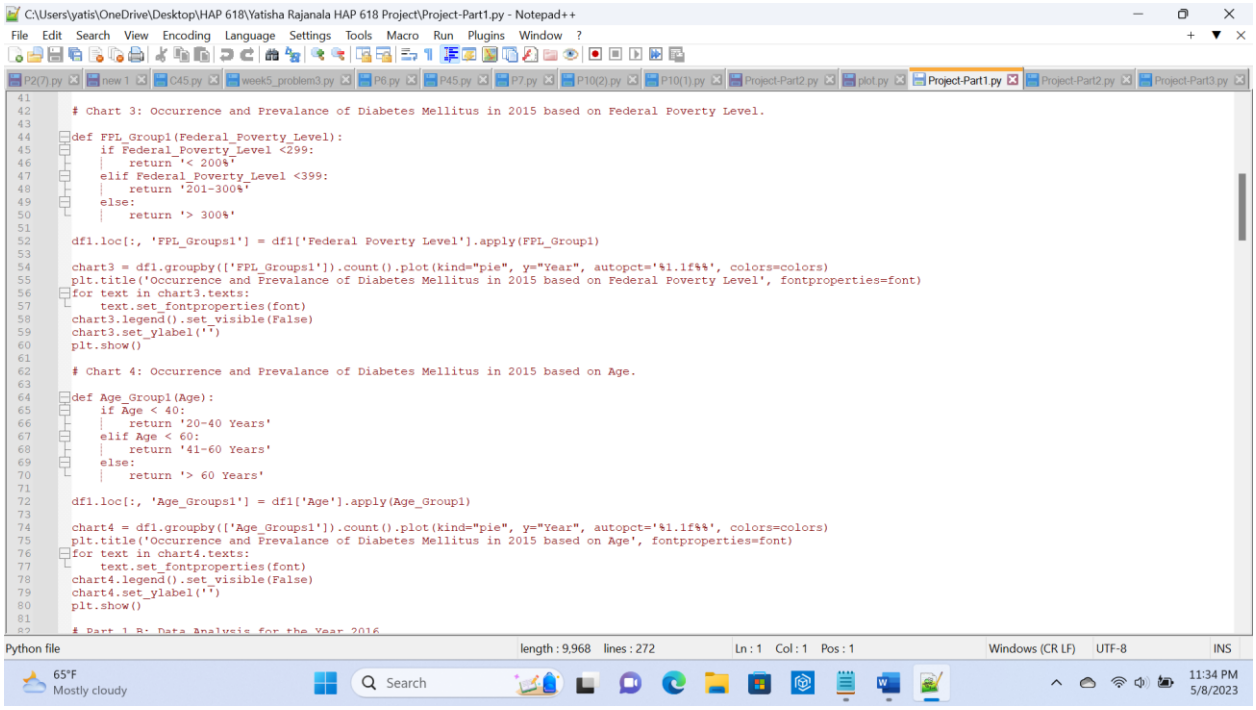
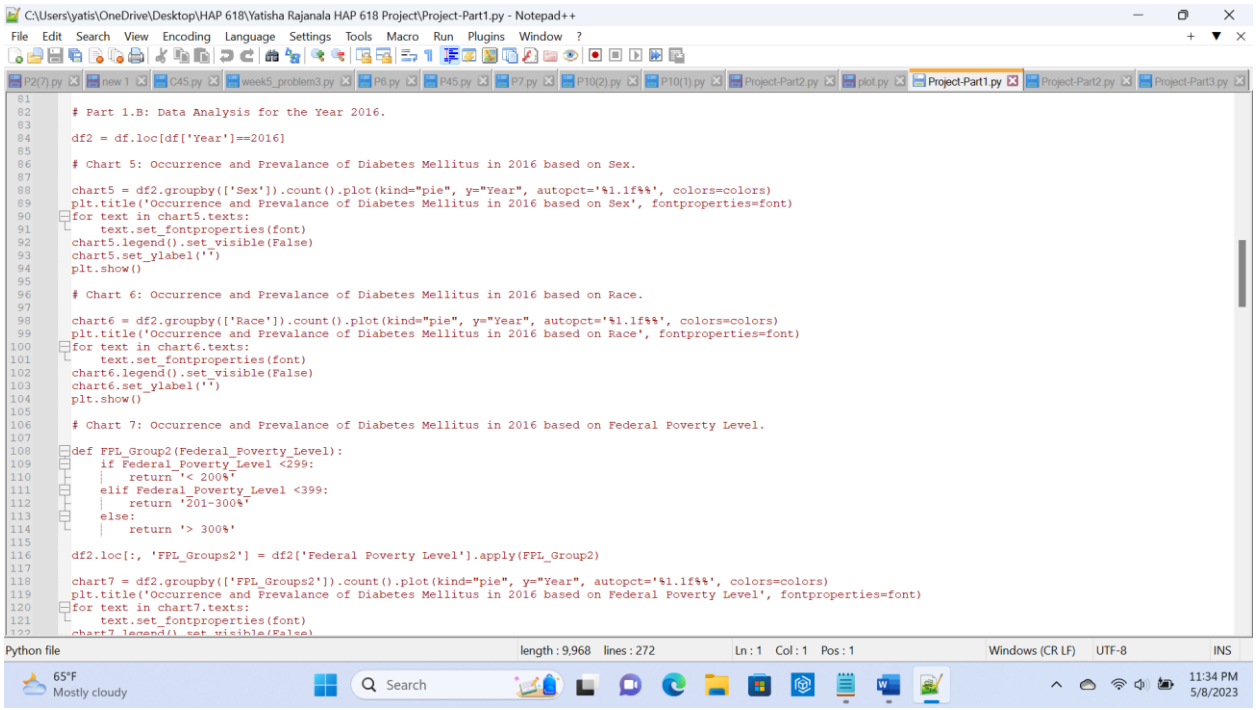


Image 4: Code for Data Analysis: Part 1 (b)



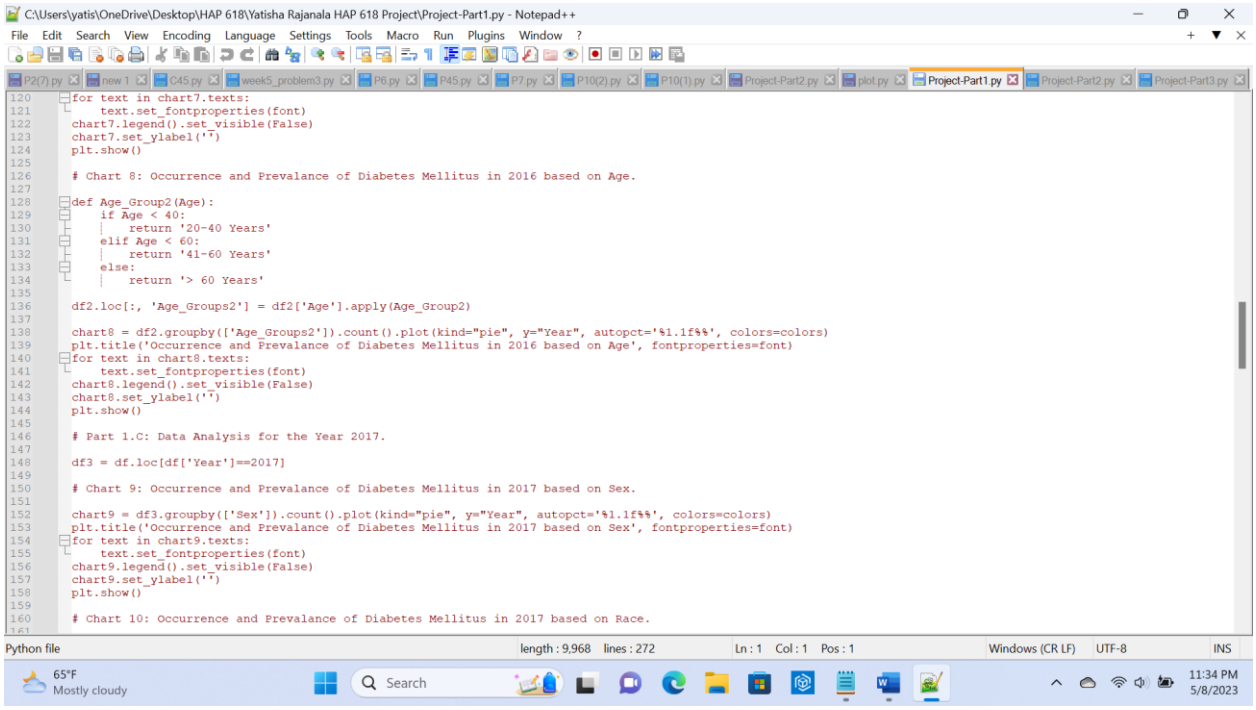
```
# Chart 3: Occurrence and Prevalance of Diabetes Mellitus in 2015 based on Federal Poverty Level.

def FPL_Group1(Federal_Poverty_Level):
    if Federal_Poverty_Level <299:
        return '< 200%'
    elif Federal_Poverty_Level <399:
        return '201-300%'
    else:
        return '> 300%'

df1.loc[:, 'FPL_Groups1'] = df1['Federal Poverty Level'].apply(FPL_Group1)

chart3 = df1.groupby(['FPL_Groups1']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2015 based on Federal Poverty Level', fontproperties=font)
for text in chart3.texts:
    text.set_fontproperties(font)
chart3.legend().set_visible(False)
chart3.set_ylabel('')
plt.show()

# Chart 4: Occurrence and Prevalance of Diabetes Mellitus in 2015 based on Age.

def Age_Group1(Age):
    if Age < 40:
        return '20-40 Years'
    elif Age < 60:
        return '41-60 Years'
    else:
        return '> 60 Years'

df1.loc[:, 'Age_Groups1'] = df1['Age'].apply(Age_Group1)

chart4 = df1.groupby(['Age_Groups1']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2015 based on Age', fontproperties=font)
for text in chart4.texts:
    text.set_fontproperties(font)
chart4.legend().set_visible(False)
chart4.set_ylabel('')
plt.show()
```

Image 5: Code for Data Analysis: Part 1 (c)



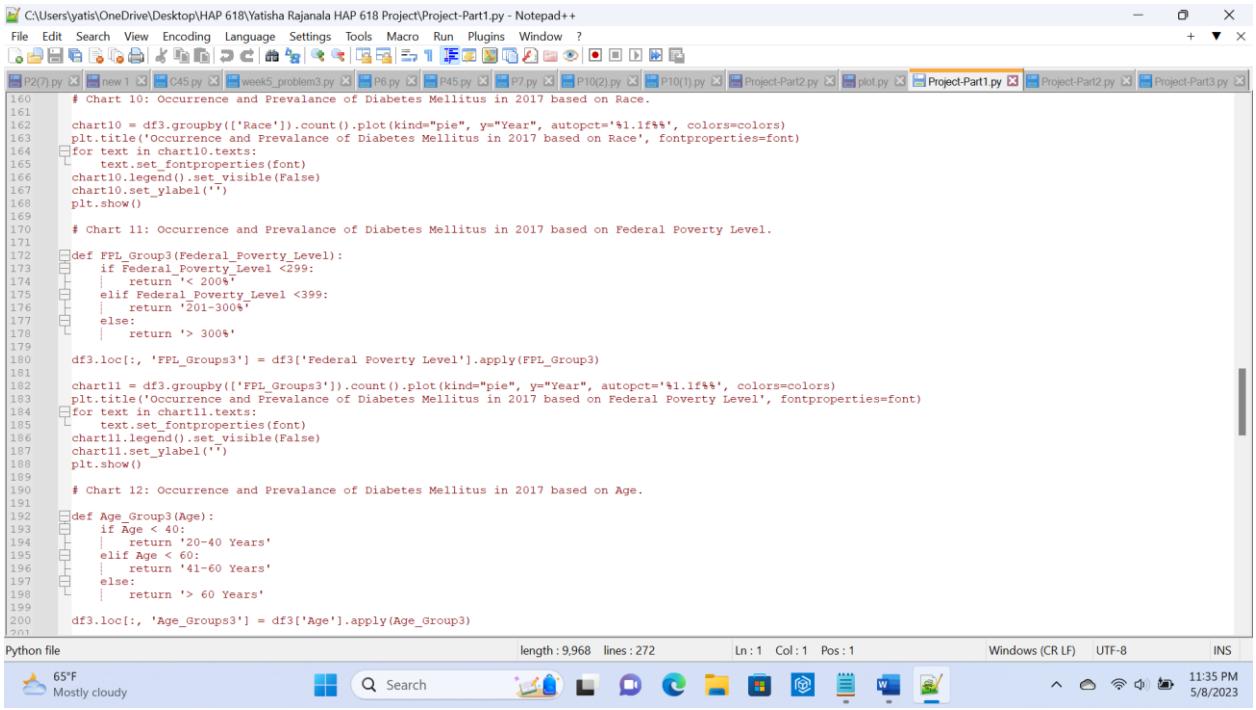
```
# Part 1.B: Data Analysis for the Year 2016.

df2 = df.loc[df['Year']==2016]

# Chart 5: Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Sex.

chart5 = df2.groupby(['Sex']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Sex', fontproperties=font)
for text in chart5.texts:
    text.set_fontproperties(font)
chart5.legend().set_visible(False)
chart5.set_ylabel('')
plt.show()

# Chart 6: Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Race.

chart6 = df2.groupby(['Race']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Race', fontproperties=font)
for text in chart6.texts:
    text.set_fontproperties(font)
chart6.legend().set_visible(False)
chart6.set_ylabel('')
plt.show()

# Chart 7: Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Federal Poverty Level.

def FPL_Group2(Federal_Poverty_Level):
    if Federal_Poverty_Level <299:
        return '< 200%'
    elif Federal_Poverty_Level <399:
        return '201-300%'
    else:
        return '> 300%'

df2.loc[:, 'FPL_Groups2'] = df2['Federal Poverty Level'].apply(FPL_Group2)

chart7 = df2.groupby(['FPL_Groups2']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Federal Poverty Level', fontproperties=font)
for text in chart7.texts:
    text.set_fontproperties(font)
```

Image 6: Code for Data Analysis: Part 1 (d)



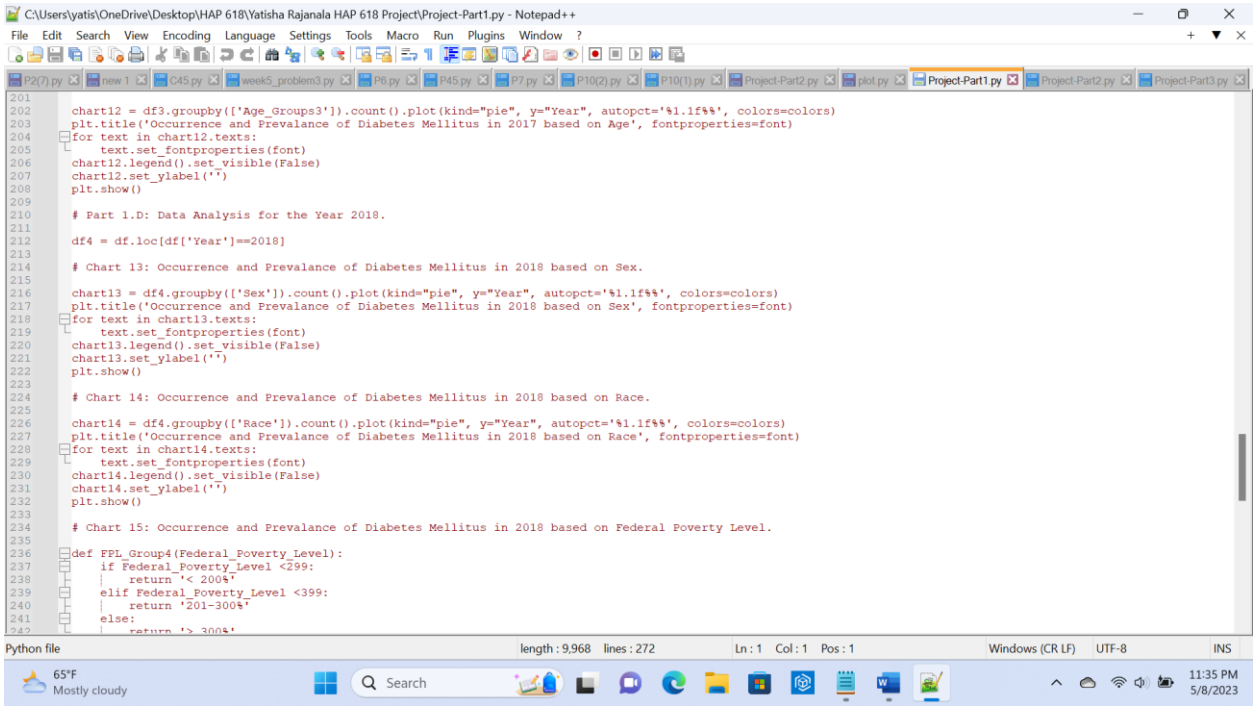
```
for text in chart7.texts:
    text.set_fontproperties(font)
chart7.legend().set_visible(False)
chart7.set_ylabel('')
plt.show()

# Chart 8: Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Age.

def Age_Group2(Age):
    if Age < 40:
        return '20-40 Years'
    elif Age < 60:
        return '41-60 Years'
    else:
        return '> 60 Years'

df2.loc[:, 'Age_Groups2'] = df2['Age'].apply(Age_Group2)

chart8 = df2.groupby(['Age_Groups2']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Age', fontproperties=font)
for text in chart8.texts:
    text.set_fontproperties(font)
chart8.legend().set_visible(False)
chart8.set_ylabel('')
plt.show()

# Part 1.C: Data Analysis for the Year 2017.

df3 = df.loc[df['Year']==2017]

# Chart 9: Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Sex.

chart9 = df3.groupby(['Sex']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Sex', fontproperties=font)
for text in chart9.texts:
    text.set_fontproperties(font)
chart9.legend().set_visible(False)
chart9.set_ylabel('')
plt.show()

# Chart 10: Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Race.
```

Image 7: Code for Data Analysis: Part 1 (e)



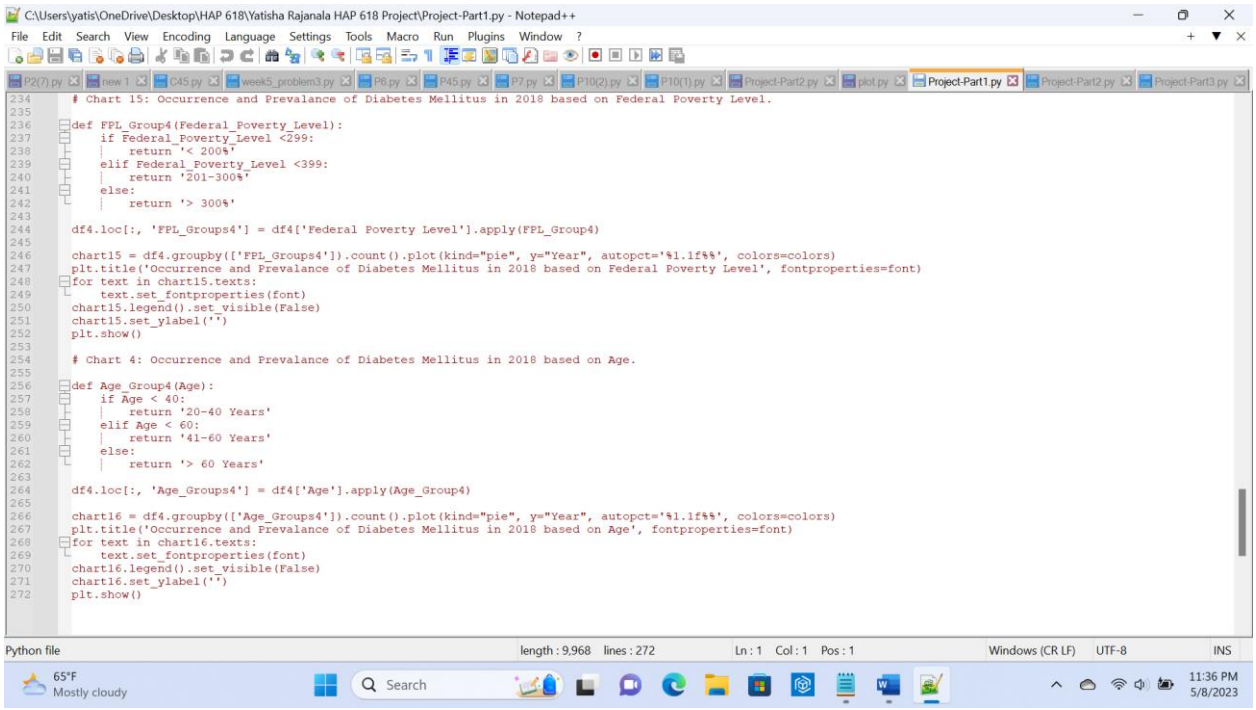
```
# Chart 10: Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Race.

chart10 = df3.groupby(['Race']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Race', fontproperties=font)
for text in chart10.texts:
    text.set_fontproperties(font)
chart10.legend().set_visible(False)
chart10.set_ylabel('')
plt.show()

# Chart 11: Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Federal Poverty Level.

def FPL_Group3(Federal_Poverty_Level):
    if Federal_Poverty_Level <299:
        return '< 200%'
    elif Federal_Poverty_Level <399:
        return '201-300%'
    else:
        return '> 300%'

df3.loc[:, 'FPL_Groups3'] = df3['Federal Poverty Level'].apply(FPL_Group3)

chart11 = df3.groupby(['FPL_Groups3']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Federal Poverty Level', fontproperties=font)
for text in chart11.texts:
    text.set_fontproperties(font)
chart11.legend().set_visible(False)
chart11.set_ylabel('')
plt.show()

# Chart 12: Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Age.

def Age_Group3(Age):
    if Age < 40:
        return '20-40 Years'
    elif Age < 60:
        return '41-60 Years'
    else:
        return '> 60 Years'

df3.loc[:, 'Age_Groups3'] = df3['Age'].apply(Age_Group3)
```

Image 8: Code for Data Analysis: Part 1 (f)



```
chart12 = df3.groupby(['Age_Groups3']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Age', fontproperties=font)
for text in chart12.texts:
    text.set_fontproperties(font)
chart12.legend().set_visible(False)
chart12.set_ylabel('')
plt.show()

# Part 1.D: Data Analysis for the Year 2018.

df4 = df.loc[df['Year']==2018]

# Chart 13: Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Sex.

chart13 = df4.groupby(['Sex']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Sex', fontproperties=font)
for text in chart13.texts:
    text.set_fontproperties(font)
chart13.legend().set_visible(False)
chart13.set_ylabel('')
plt.show()

# Chart 14: Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Race.

chart14 = df4.groupby(['Race']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Race', fontproperties=font)
for text in chart14.texts:
    text.set_fontproperties(font)
chart14.legend().set_visible(False)
chart14.set_ylabel('')
plt.show()

# Chart 15: Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Federal Poverty Level.

def FPL_Group4(Federal_Poverty_Level):
    if Federal_Poverty_Level <299:
        return '< 200%'
    elif Federal_Poverty_Level <399:
        return '201-300%'
    else:
        return '> 300%'
```

Image 9: Code for Data Analysis: Part 1 (g)



```
# Chart 15: Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Federal Poverty Level.

def FPL_Group4(Federal_Poverty_Level):
    if Federal_Poverty_Level <299:
        return '< 200%'
    elif Federal_Poverty_Level <399:
        return '201-300%'
    else:
        return '> 300%'

df4.loc[:, 'FPL_Groups4'] = df4['Federal Poverty Level'].apply(FPL_Group4)

chart15 = df4.groupby(['FPL_Groups4']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Federal Poverty Level', fontproperties=font)
for text in chart15.texts:
    text.set_fontproperties(font)
chart15.legend().set_visible(False)
chart15.set_ylabel('')
plt.show()

# Chart 4: Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Age.

def Age_Group4(Age):
    if Age < 40:
        return '20-40 Years'
    elif Age < 60:
        return '41-60 Years'
    else:
        return '> 60 Years'

df4.loc[:, 'Age_Groups4'] = df4['Age'].apply(Age_Group4)

chart16 = df4.groupby(['Age_Groups4']).count().plot(kind="pie", y="Year", autopct='%1.1f%%', colors=colors)
plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Age', fontproperties=font)
for text in chart16.texts:
    text.set_fontproperties(font)
chart16.legend().set_visible(False)
chart16.set_ylabel('')
plt.show()
```

Part 1: Python Code Analysis and Steps for Coding:

1. The Python code was written in Notepad++, saved as a .py file and executed on Command Prompt.
2. Firstly, the Python libraries of Pandas, Matplotlib and FontProperties from Matplotlib were imported.
3. The Font was set for the data visualizations, including font name, font size and colors.
4. The dataset in the form of CSV file was opened and read.
5. The data was grouped based on the Years, and accordingly the dataframes were extracted from the main dataset.
6. For each year, 4 different pie charts, based on occerence and prevalence of Diabetes Mellitus were created; including those on the basis of Sex, Race, Federal Poverty Level and Age.
7. <u>For the Pie Charts based on Sex and Race:</u>
    i. The dataframe was directly grouped using each individual column, as both these columns consist of string values.
    ii. The data in these columns was counted using the Count function.
8. <u>For the Pie Charts based on Federal Poverty Level and Age:</u>
    i. The dataframe was first grouped using the Define: If, Else If and Else functions, into each individual ranges, as both these columns consist of large varied values.
    ii. The obtained grouped data was then added to the dataframe as a new column with the various categories.
    iii. The data in this new column was counted using the Count function.
9. Then, the pie chart was plotted and the percentage of occurrence was set.
10. Using the Font Properties, the title, text, legend and labels were set.
11. Lastly, the pie chart was displayed using the function, plt.show().
12. The pie charts for each year from 2015-2018 were done, so a total of 16 pie charts were produced, when the code was Run on the Command Prompt.
13. These pie charts were saved as PNG Images on the server.

Part 2: Data Analysis based on Extent of Glycaemic Control, using the HbA1c Measured Values

The data analysis was done using the Python Libraries of Pandas, Numpy and Matplotlib to create Bar Graphs to visualize the data of Diabetes Mellitus with respect to the Extent of Glycaemic Control, using the HbA1c Measured Values.

The data was divided on the basis of each Demogrphic Specification, namely Males, Females, Whites, African-American and Hispanic and so on. Based on this, new data frames were located and created to plot bar graphs. The data was formed into arrays having the Years, and the axis for the labels was set.

On the other side, for the Demographics of Federal Poverty Levels and Age, the data was first grouped based on the specified ranges using the Define: If, Else If and Else functions. After this, the data was plotted for the bar graphs.

Image 10: Code for Data Analysis: Part 2 (a)

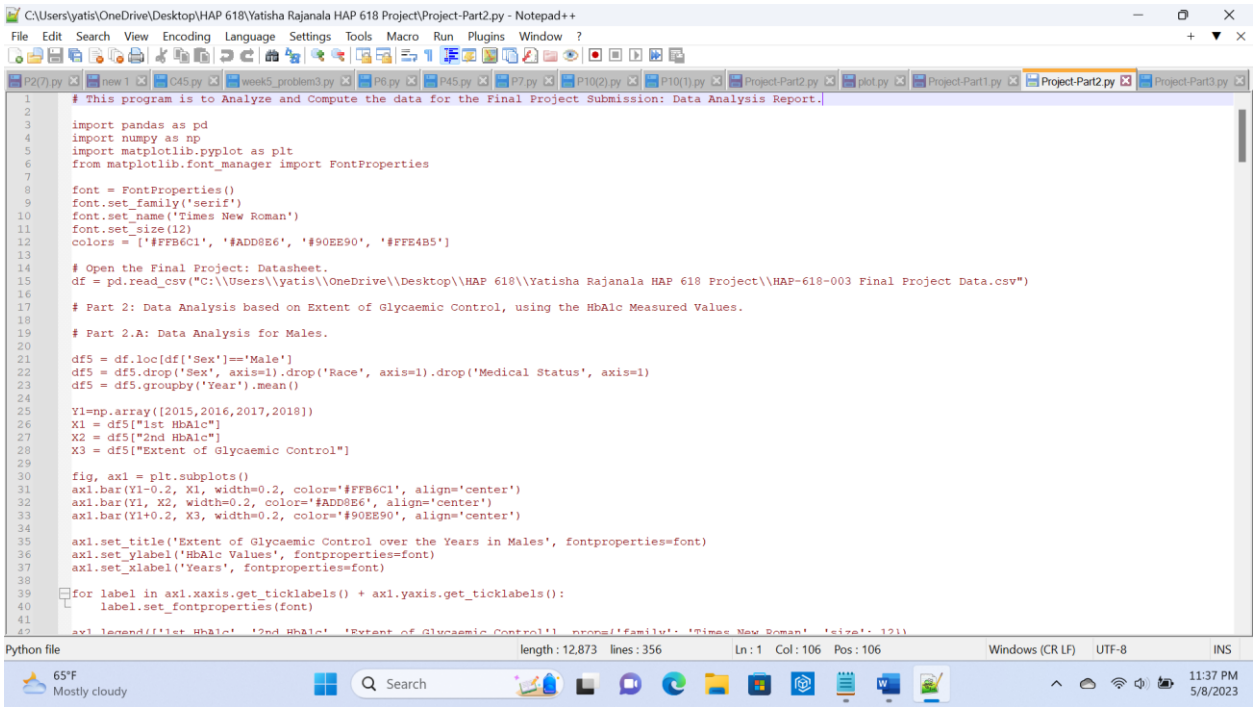
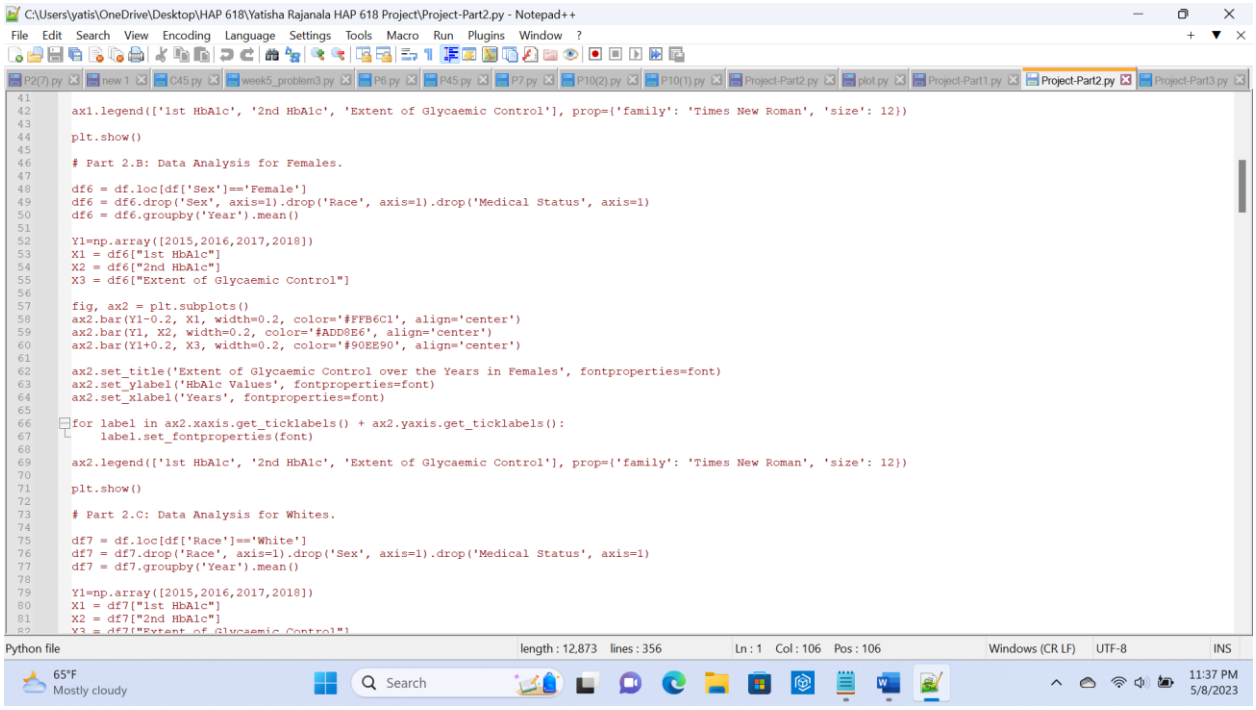
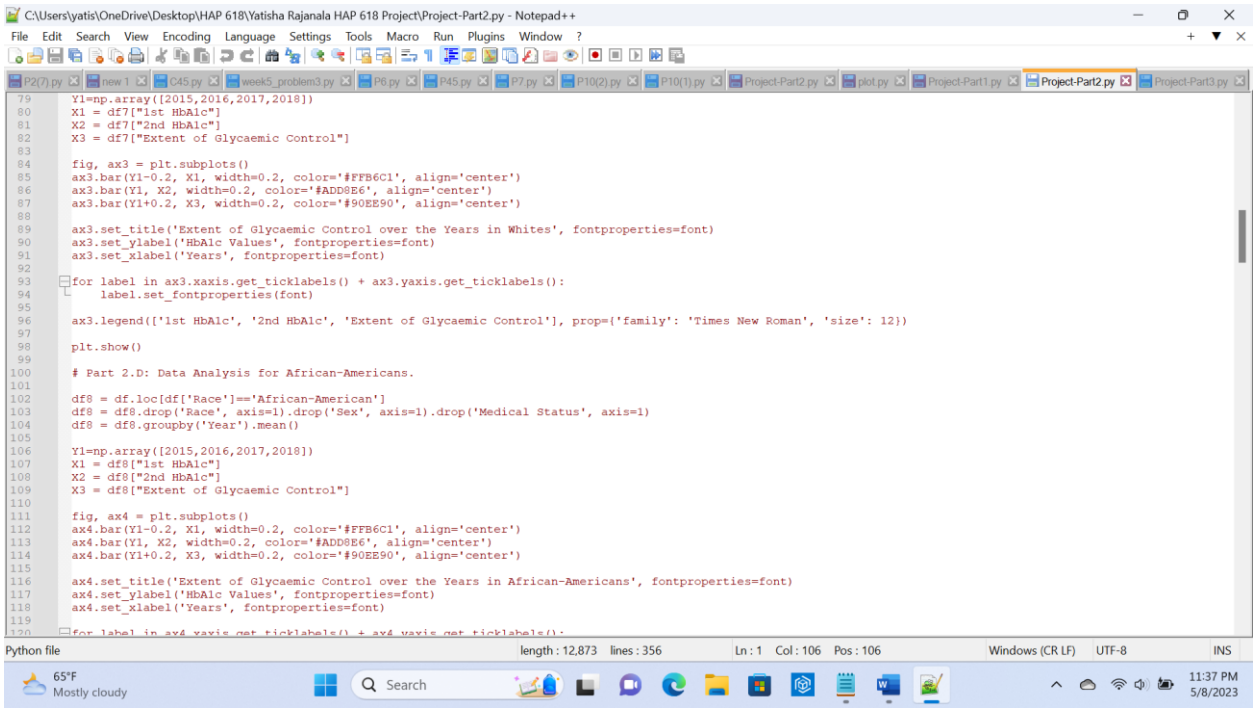


Image 11: Code for Data Analysis: Part 2 (b)



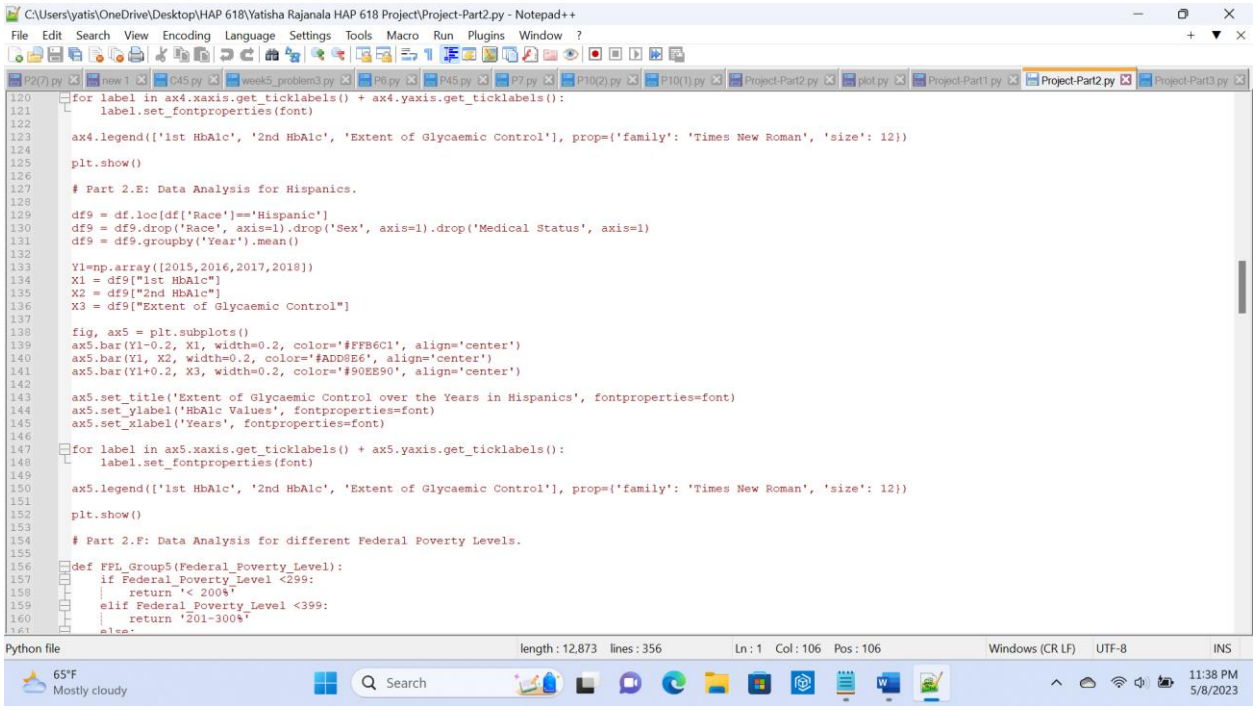
```
    ax1.legend(['1st HbAlc', '2nd HbAlc', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})

    plt.show()

    # Part 2.B: Data Analysis for Females.

    df6 = df.loc[df['Sex']=='Female']
    df6 = df6.drop('Sex', axis=1).drop('Race', axis=1).drop('Medical Status', axis=1)
    df6 = df6.groupby('Year').mean()

    Y1=np.array([2015,2016,2017,2018])
    X1 = df6["1st HbAlc"]
    X2 = df6["2nd HbAlc"]
    X3 = df6["Extent of Glycaemic Control"]

    fig, ax2 = plt.subplots()
    ax2.bar(Y1-0.2, X1, width=0.2, color='#FFB6C1', align='center')
    ax2.bar(Y1, X2, width=0.2, color='#ADD8E6', align='center')
    ax2.bar(Y1+0.2, X3, width=0.2, color='#90EE90', align='center')

    ax2.set_title('Extent of Glycaemic Control over the Years in Females', fontproperties=font)
    ax2.set_ylabel('HbAlc Values', fontproperties=font)
    ax2.set_xlabel('Years', fontproperties=font)

    for label in ax2.xaxis.get_ticklabels() + ax2.yaxis.get_ticklabels():
        label.set_fontproperties(font)

    ax2.legend(['1st HbAlc', '2nd HbAlc', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})

    plt.show()

    # Part 2.C: Data Analysis for Whites.

    df7 = df.loc[df['Race']=='White']
    df7 = df7.drop('Race', axis=1).drop('Sex', axis=1).drop('Medical Status', axis=1)
    df7 = df7.groupby('Year').mean()

    Y1=np.array([2015,2016,2017,2018])
    X1 = df7["1st HbAlc"]
    X2 = df7["2nd HbAlc"]
```

Image 12: Code for Data Analysis: Part 2 (c)



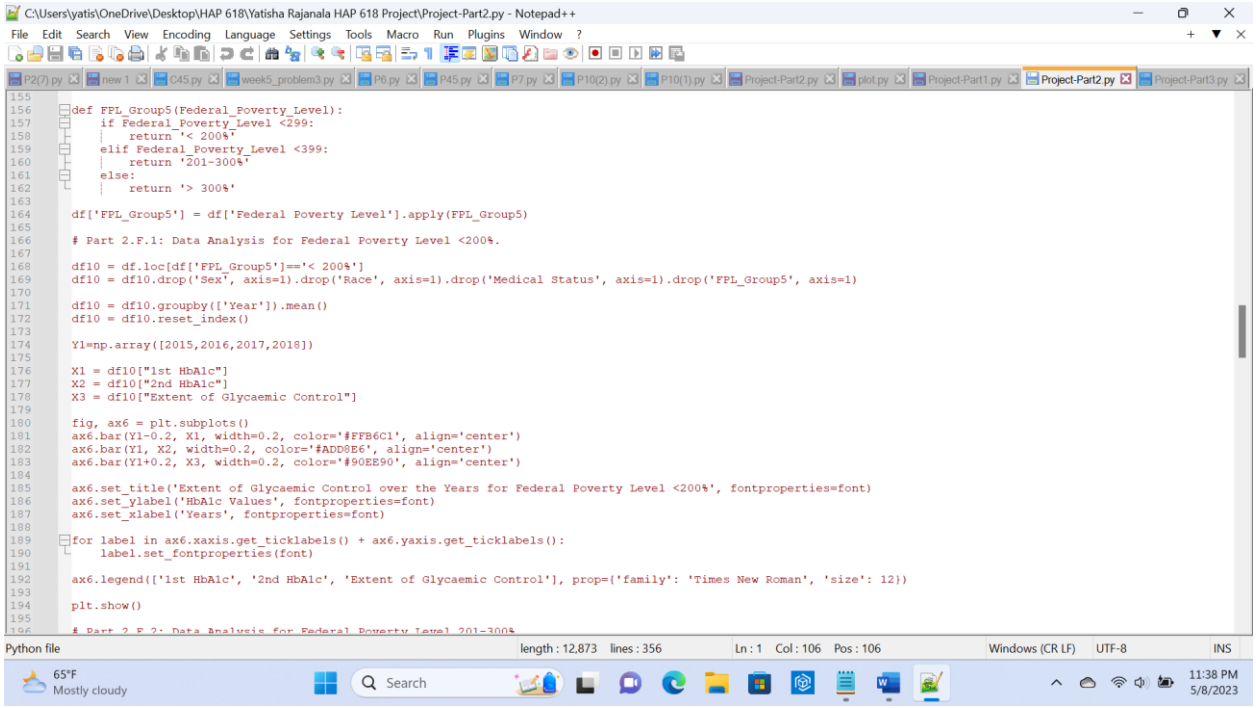
```
    Y1=np.array([2015,2016,2017,2018])
    X1 = df7["1st HbAlc"]
    X2 = df7["2nd HbAlc"]
    X3 = df7["Extent of Glycaemic Control"]

    fig, ax3 = plt.subplots()
    ax3.bar(Y1-0.2, X1, width=0.2, color='#FFB6C1', align='center')
    ax3.bar(Y1, X2, width=0.2, color='#ADD8E6', align='center')
    ax3.bar(Y1+0.2, X3, width=0.2, color='#90EE90', align='center')

    ax3.set_title('Extent of Glycaemic Control over the Years in Whites', fontproperties=font)
    ax3.set_ylabel('HbAlc Values', fontproperties=font)
    ax3.set_xlabel('Years', fontproperties=font)

    for label in ax3.xaxis.get_ticklabels() + ax3.yaxis.get_ticklabels():
        label.set_fontproperties(font)

    ax3.legend(['1st HbAlc', '2nd HbAlc', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})

    plt.show()

    # Part 2.D: Data Analysis for African-Americans.

    df8 = df.loc[df['Race']=='African-American']
    df8 = df8.drop('Race', axis=1).drop('Sex', axis=1).drop('Medical Status', axis=1)
    df8 = df8.groupby('Year').mean()

    Y1=np.array([2015,2016,2017,2018])
    X1 = df8["1st HbAlc"]
    X2 = df8["2nd HbAlc"]
    X3 = df8["Extent of Glycaemic Control"]

    fig, ax4 = plt.subplots()
    ax4.bar(Y1-0.2, X1, width=0.2, color='#FFB6C1', align='center')
    ax4.bar(Y1, X2, width=0.2, color='#ADD8E6', align='center')
    ax4.bar(Y1+0.2, X3, width=0.2, color='#90EE90', align='center')

    ax4.set_title('Extent of Glycaemic Control over the Years in African-Americans', fontproperties=font)
    ax4.set_ylabel('HbAlc Values', fontproperties=font)
    ax4.set_xlabel('Years', fontproperties=font)
```

Image 13: Code for Data Analysis: Part 2 (d)



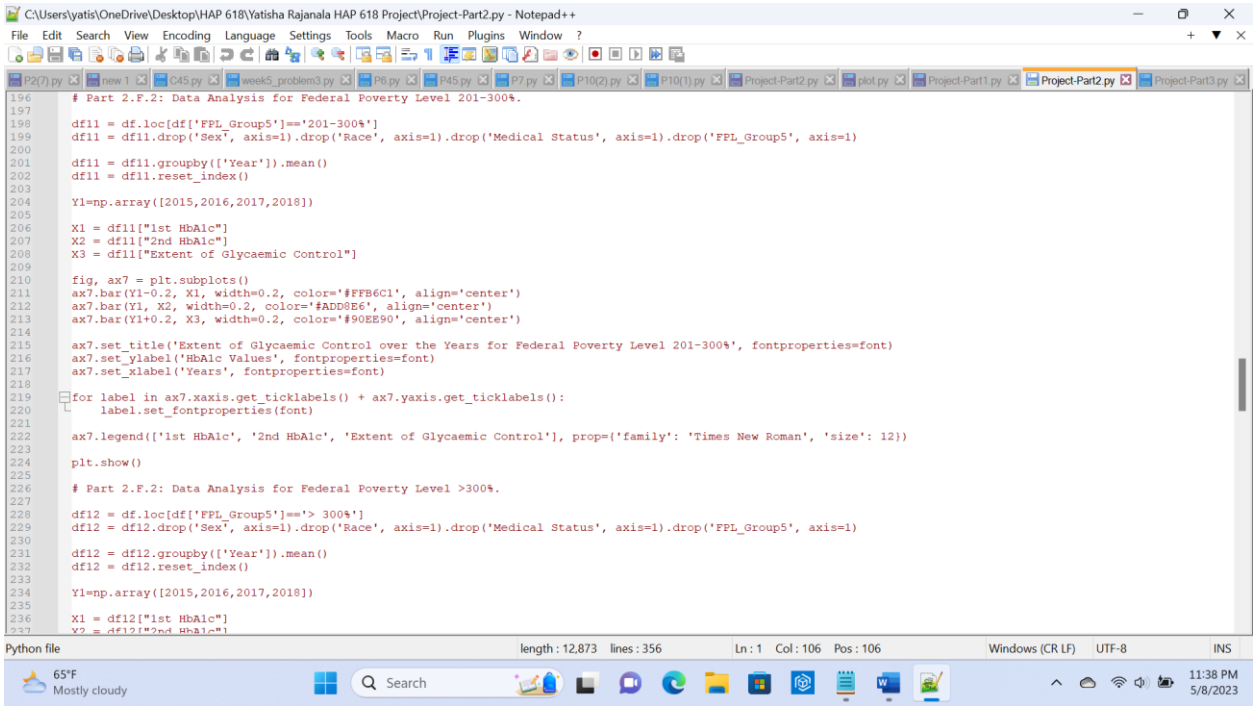
```
for label in ax4.xaxis.get_ticklabels() + ax4.yaxis.get_ticklabels():
    label.set_fontproperties(font)

ax4.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})

plt.show()

# Part 2.E: Data Analysis for Hispanics.

df9 = df.loc[df['Race']=='Hispanic']
df9 = df9.drop('Race', axis=1).drop('Sex', axis=1).drop('Medical Status', axis=1)
df9 = df9.groupby('Year').mean()

Y1=np.array([2015,2016,2017,2018])
X1 = df9["1st HbA1c"]
X2 = df9["2nd HbA1c"]
X3 = df9["Extent of Glycaemic Control"]

fig, ax5 = plt.subplots()
ax5.bar(Y1-0.2, X1, width=0.2, color='#FFB6C1', align='center')
ax5.bar(Y1, X2, width=0.2, color='#ADD8E6', align='center')
ax5.bar(Y1+0.2, X3, width=0.2, color='#90EE90', align='center')

ax5.set_title('Extent of Glycaemic Control over the Years in Hispanics', fontproperties=font)
ax5.set_ylabel('HbA1c Values', fontproperties=font)
ax5.set_xlabel('Years', fontproperties=font)

for label in ax5.xaxis.get_ticklabels() + ax5.yaxis.get_ticklabels():
    label.set_fontproperties(font)

ax5.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})

plt.show()

# Part 2.F: Data Analysis for different Federal Poverty Levels.

def FPL_Group5(Federal_Poverty_Level):
    if Federal_Poverty_Level <299:
        return '< 200%'
    elif Federal_Poverty_Level <399:
        return '201-300%'
```

Image 14: Code for Data Analysis: Part 2 (e)



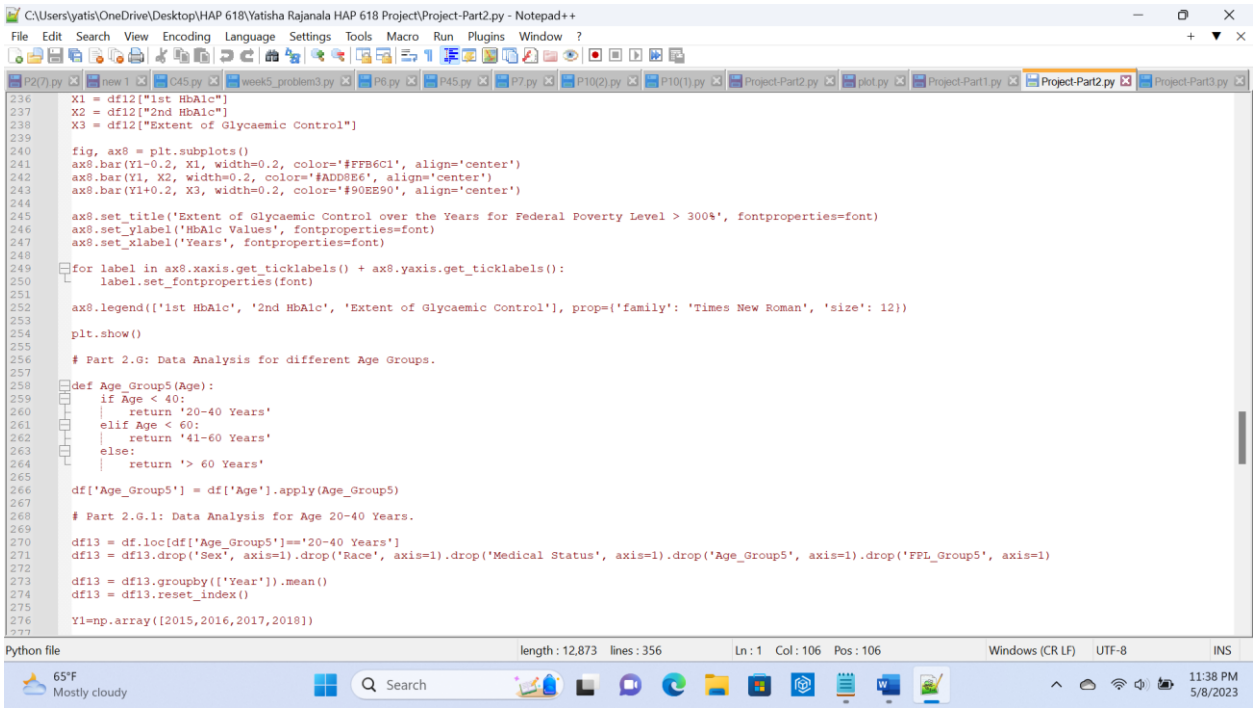
```
def FPL_Group5(Federal_Poverty_Level):
    if Federal_Poverty_Level <299:
        return '< 200%'
    elif Federal_Poverty_Level <399:
        return '201-300%'
    else:
        return '> 300%'

df['FPL_Group5'] = df['Federal Poverty Level'].apply(FPL_Group5)

# Part 2.F.1: Data Analysis for Federal Poverty Level <200%.

df10 = df.loc[df['FPL_Group5']=='< 200%']
df10 = df10.drop('Sex', axis=1).drop('Race', axis=1).drop('Medical Status', axis=1).drop('FPL_Group5', axis=1)

df10 = df10.groupby(['Year']).mean()
df10 = df10.reset_index()

Y1=np.array([2015,2016,2017,2018])

X1 = df10["1st HbA1c"]
X2 = df10["2nd HbA1c"]
X3 = df10["Extent of Glycaemic Control"]

fig, ax6 = plt.subplots()
ax6.bar(Y1-0.2, X1, width=0.2, color='#FFB6C1', align='center')
ax6.bar(Y1, X2, width=0.2, color='#ADD8E6', align='center')
ax6.bar(Y1+0.2, X3, width=0.2, color='#90EE90', align='center')

ax6.set_title('Extent of Glycaemic Control over the Years for Federal Poverty Level <200%', fontproperties=font)
ax6.set_ylabel('HbA1c Values', fontproperties=font)
ax6.set_xlabel('Years', fontproperties=font)

for label in ax6.xaxis.get_ticklabels() + ax6.yaxis.get_ticklabels():
    label.set_fontproperties(font)

ax6.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})

plt.show()
```

Image 15: Code for Data Analysis: Part 2 (f)



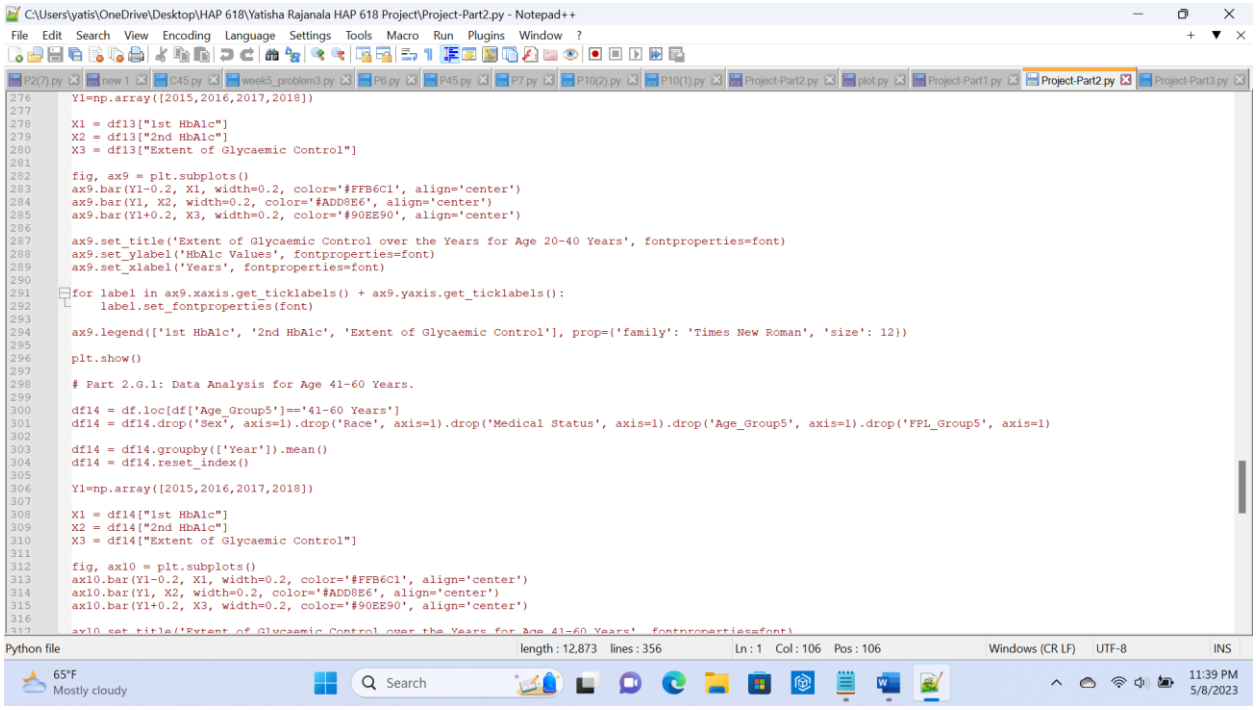
```
# Part 2.F.2: Data Analysis for Federal Poverty Level 201-300%.

df11 = df.loc[df['FPL_Group5']=='201-300%']
df11 = df11.drop('Sex', axis=1).drop('Race', axis=1).drop('Medical Status', axis=1).drop('FPL_Group5', axis=1)

df11 = df11.groupby(['Year']).mean()
df11 = df11.reset_index()

Y1=np.array([2015,2016,2017,2018])

X1 = df11["1st HbAlc"]
X2 = df11["2nd HbAlc"]
X3 = df11["Extent of Glycaemic Control"]

fig, ax7 = plt.subplots()
ax7.bar(Y1-0.2, X1, width=0.2, color='#FFB6C1', align='center')
ax7.bar(Y1, X2, width=0.2, color='#ADD8E6', align='center')
ax7.bar(Y1+0.2, X3, width=0.2, color='#90EE90', align='center')

ax7.set_title('Extent of Glycaemic Control over the Years for Federal Poverty Level 201-300%', fontproperties=font)
ax7.set_ylabel('HbAlc Values', fontproperties=font)
ax7.set_xlabel('Years', fontproperties=font)

for label in ax7.xaxis.get_ticklabels() + ax7.yaxis.get_ticklabels():
    label.set_fontproperties(font)

ax7.legend(['1st HbAlc', '2nd HbAlc', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})

plt.show()

# Part 2.F.2: Data Analysis for Federal Poverty Level >300%.

df12 = df.loc[df['FPL_Group5']=='> 300%']
df12 = df12.drop('Sex', axis=1).drop('Race', axis=1).drop('Medical Status', axis=1).drop('FPL_Group5', axis=1)

df12 = df12.groupby(['Year']).mean()
df12 = df12.reset_index()

Y1=np.array([2015,2016,2017,2018])

X1 = df12["1st HbAlc"]
```

Image 16: Code for Data Analysis: Part 2 (g)



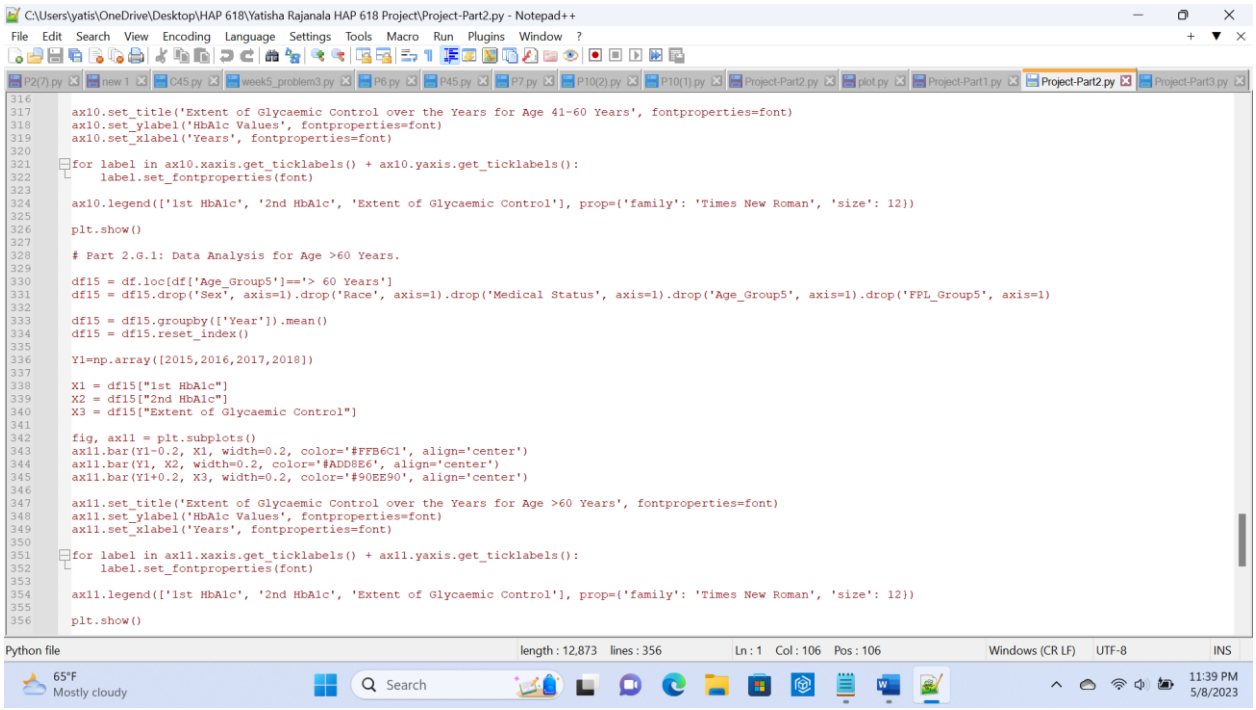
```
X1 = df12["1st HbAlc"]
X2 = df12["2nd HbAlc"]
X3 = df12["Extent of Glycaemic Control"]

fig, ax8 = plt.subplots()
ax8.bar(Y1-0.2, X1, width=0.2, color='#FFB6C1', align='center')
ax8.bar(Y1, X2, width=0.2, color='#ADD8E6', align='center')
ax8.bar(Y1+0.2, X3, width=0.2, color='#90EE90', align='center')

ax8.set_title('Extent of Glycaemic Control over the Years for Federal Poverty Level > 300%', fontproperties=font)
ax8.set_ylabel('HbAlc Values', fontproperties=font)
ax8.set_xlabel('Years', fontproperties=font)

for label in ax8.xaxis.get_ticklabels() + ax8.yaxis.get_ticklabels():
    label.set_fontproperties(font)

ax8.legend(['1st HbAlc', '2nd HbAlc', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})

plt.show()

# Part 2.G: Data Analysis for different Age Groups.

def Age_Group5(Age):
    if Age < 40:
        return '20-40 Years'
    elif Age < 60:
        return '41-60 Years'
    else:
        return '> 60 Years'

df['Age_Group5'] = df['Age'].apply(Age_Group5)

# Part 2.G.1: Data Analysis for Age 20-40 Years.

df13 = df.loc[df['Age_Group5']=='20-40 Years']
df13 = df13.drop('Sex', axis=1).drop('Race', axis=1).drop('Medical Status', axis=1).drop('Age_Group5', axis=1).drop('FPL_Group5', axis=1)

df13 = df13.groupby(['Year']).mean()
df13 = df13.reset_index()

Y1=np.array([2015,2016,2017,2018])
```

Image 17: Code for Data Analysis: Part 2 (h)



```
Y1=np.array([2015,2016,2017,2018])

X1 = df13["1st HbA1c"]
X2 = df13["2nd HbA1c"]
X3 = df13["Extent of Glycaemic Control"]

fig, ax9 = plt.subplots()
ax9.bar(Y1-0.2, X1, width=0.2, color='#FFB6C1', align='center')
ax9.bar(Y1, X2, width=0.2, color='#ADD8E6', align='center')
ax9.bar(Y1+0.2, X3, width=0.2, color='#90EE90', align='center')

ax9.set_title('Extent of Glycaemic Control over the Years for Age 20-40 Years', fontproperties=font)
ax9.set_ylabel('HbA1c Values', fontproperties=font)
ax9.set_xlabel('Years', fontproperties=font)

for label in ax9.xaxis.get_ticklabels() + ax9.yaxis.get_ticklabels():
    label.set_fontproperties(font)

ax9.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})

plt.show()

# Part 2.G.1: Data Analysis for Age 41-60 Years.

df14 = df.loc[df['Age_Group5']=='41-60 Years']
df14 = df14.drop('Sex', axis=1).drop('Race', axis=1).drop('Medical Status', axis=1).drop('Age_Group5', axis=1).drop('FPL_Group5', axis=1)

df14 = df14.groupby(['Year']).mean()
df14 = df14.reset_index()

Y1=np.array([2015,2016,2017,2018])

X1 = df14["1st HbA1c"]
X2 = df14["2nd HbA1c"]
X3 = df14["Extent of Glycaemic Control"]

fig, ax10 = plt.subplots()
ax10.bar(Y1-0.2, X1, width=0.2, color='#FFB6C1', align='center')
ax10.bar(Y1, X2, width=0.2, color='#ADD8E6', align='center')
ax10.bar(Y1+0.2, X3, width=0.2, color='#90EE90', align='center')
```

Image 18: Code for Data Analysis: Part 2 (i)



```
ax10.set_title('Extent of Glycaemic Control over the Years for Age 41-60 Years', fontproperties=font)
ax10.set_ylabel('HbA1c Values', fontproperties=font)
ax10.set_xlabel('Years', fontproperties=font)

for label in ax10.xaxis.get_ticklabels() + ax10.yaxis.get_ticklabels():
    label.set_fontproperties(font)

ax10.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})

plt.show()

# Part 2.G.1: Data Analysis for Age >60 Years.

df15 = df.loc[df['Age_Group5']=='> 60 Years']
df15 = df15.drop('Sex', axis=1).drop('Race', axis=1).drop('Medical Status', axis=1).drop('Age_Group5', axis=1).drop('FPL_Group5', axis=1)

df15 = df15.groupby(['Year']).mean()
df15 = df15.reset_index()

Y1=np.array([2015,2016,2017,2018])

X1 = df15["1st HbA1c"]
X2 = df15["2nd HbA1c"]
X3 = df15["Extent of Glycaemic Control"]

fig, ax11 = plt.subplots()
ax11.bar(Y1-0.2, X1, width=0.2, color='#FFB6C1', align='center')
ax11.bar(Y1, X2, width=0.2, color='#ADD8E6', align='center')
ax11.bar(Y1+0.2, X3, width=0.2, color='#90EE90', align='center')

ax11.set_title('Extent of Glycaemic Control over the Years for Age >60 Years', fontproperties=font)
ax11.set_ylabel('HbA1c Values', fontproperties=font)
ax11.set_xlabel('Years', fontproperties=font)

for label in ax11.xaxis.get_ticklabels() + ax11.yaxis.get_ticklabels():
    label.set_fontproperties(font)

ax11.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})

plt.show()
```

Part 2: Python Code Analysis and Steps for Coding:

1. The Python code was written in Notepad++, saved as a .py file and executed on Command Prompt.
2. Firstly, the Python libraries of Pandas, NumPy, Matplotlib and FontProperties from Matplotlib were imported.
3. The Font was set for the data visualizations, including font name, font size and colors.
4. The dataset in the form of CSV file was opened and read.
5. The data was grouped on the basis of the different Demographics, and accordingly the dataframes were extracted from the main dataset; including parameters like Males, Whites, Federal Poverty Level Range of <200%, Age Range of 20-40 Years.
6. For each demographic, 1 bar graph was designed to analyze the trends during the different years, using the HbA1c measured values of Diabetes Mellitus.
7. <u>For the Bar Graphs based on Sex and Race:</u>
    i. The dataframe was directly grouped using each individual dataset, as both these columns consist of string values.
    ii. However, other string values like Medical Status were removed from the new dataset to avoid data overlap, using the .drop() function.
8. <u>For the Bar Graphs based on Federal Poverty Level and Age:</u>
    i. The dataframe was first grouped using the Define: If, Else If and Else functions, into each individual ranges, as both these columns consist of large varied values.
    ii. The obtained grouped data was then added to the dataframe as a new column with the various categories, and each data range was located individually using the .loc[] function.
9. Then, the obtained new dataset was grouped on the basis of years, and an Array was created using np.array() NumPy.
10. The bar graph was plotted using the Years on the x-axis and the Measured Values of the 1st HbA1c, 2nd HbA1c and Extent of Glycemic Control bars on the y-axis as subplots.
11. Using the Font Properties, the title, text, legend, labels, bars and ticklabels were set.
12. Lastly, the bar graph was displayed using the function, plt.show().
13. The bar graphs for each demographic specification from 2015-2018 were done, so a total of 11 bar graphs were produced, when the code was Run on the Command Prompt.
14. These bar graphs were saved as PNG Images on the server.

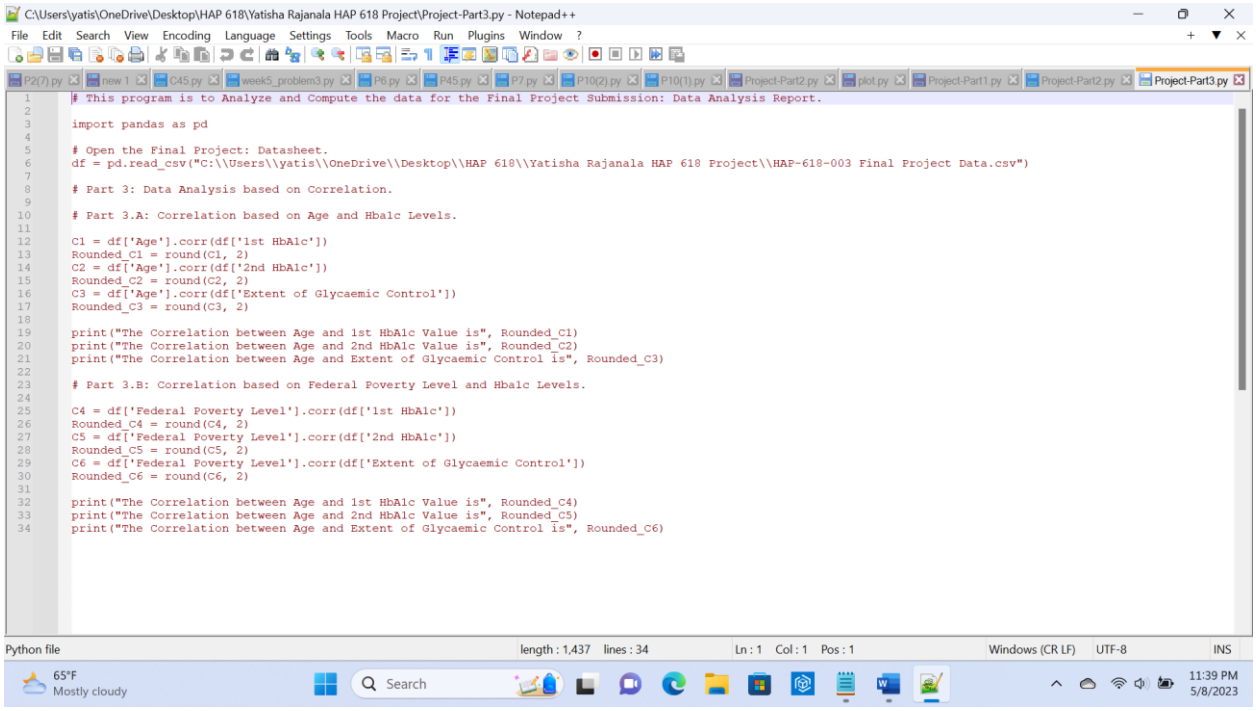
Part 3: Data Analysis based on Correlation

The data analysis was done using the Python Library of Pandas to analyze the data of Diabetes Mellitus with respect to its Correlation.

The data was divided into 2 sections for analysis on the basis of the Measured HbA1c Values, specifically 1st HbA1c values, 2nd HbA1c values and Extent of Glycemic Control.

These measured values were then correlated to the parameters of Federal Poverty Level and Age, as they were more significant and grouped into data ranges which may show correlation in the given dataset.

Image 19: Code for Data Analysis: Part 3



Part 3: Python Code Analysis and Steps for Coding:

1. The Python code was written in Notepad++, saved as a .py file and executed on Comand Prompt.
2. Firstly, the Python library of Pandas was imported.
3. The dataset in the form of CSV file was opened and read.
4. The data was divided into 2 sections for analysis on the basis of the Measured HbA1c Values, specifically 1st HbA1c values, 2nd HbA1c values and Extent of Glycemic Control.
5. These measured values were then correlated to the parameters of Federal Poverty Level and Age, using the .corr() function.
6. The obtained value was then rounded to only 2 decimal places using the .round() function.
7. Lastly, the correlation values were displayed using the function, print().
8. The correlation of the different parameters was done, so a total of 6 measured values were produced, when the code was Run on the Command Prompt.
9. These output was saved as PNG Images on the server.

RESULTS AND OUTCOMES

The Python Codes were run on the Command Prompt and the obtained results and visualizations were saved in the form of PNG Images on the server.

As the main moto of the Data Analysis Report is to make the analyzed data available for both healthcare proffessionals and public, the HTML Interface was used to display the data in an orderly manner. Thus, HTML codes were written for webpages for future references.

A total of 4 webpages were created, which are hyperlinked from the main webpage into 3 other branching webpages. These webpages consist information regarding the Data Analysis Report, including all of its data visualiation text and images.

HTML Code Analysis and Steps for Coding:

1. The HTML code was written in Notepad, saved as a .html file and executed on the Web Browser.
2. Firstly, the Data Analysis Report was divided into 4 sections, and thus, 4 different Webpages were created, namely Main Webpage, Part 1 Webpage, Part2 Webpage and Part 3 Webpage.
3. The Main Webpage consists of title and CSS style formats for the webpage, followed by the body of the page, which includes headings, paragraphs and tables with formatted images, unordered list of hyperlinks to the other 3 webpages and ordered list of data.
4. The Part 1 and Part 2 Webpages consists of title and CSS style formats for the webpage, followed by the body of the page, which includes headings, paragraphs and tables with formatted images on the left side and related data analysis summary texts on the right side.
5. The Part 3 Webpage consists of title and CSS style formats for the webpage, followed by the body of the page, which includes headings and images to show the data analysis results.

Image 20: Results: Main Webpage (a)

Image 21: Results: Main Webpage (b)

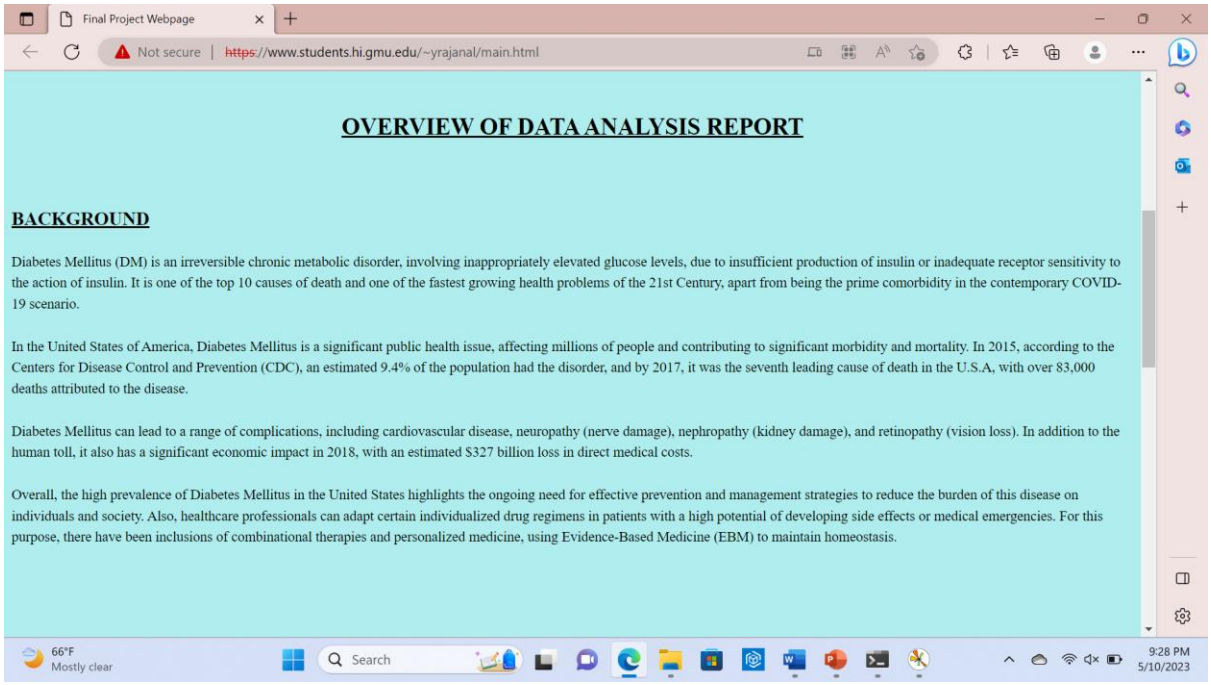


Image 22: Results: Main Webpage (c)

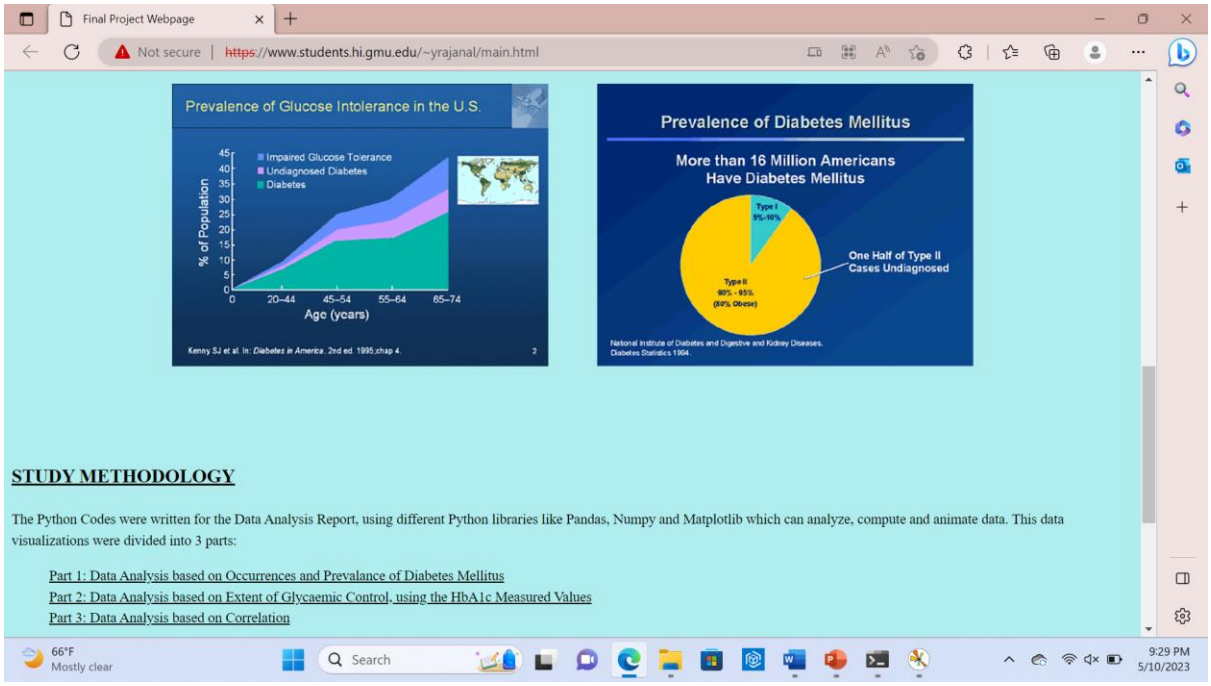


Image 23: Results: Main Webpage (d)

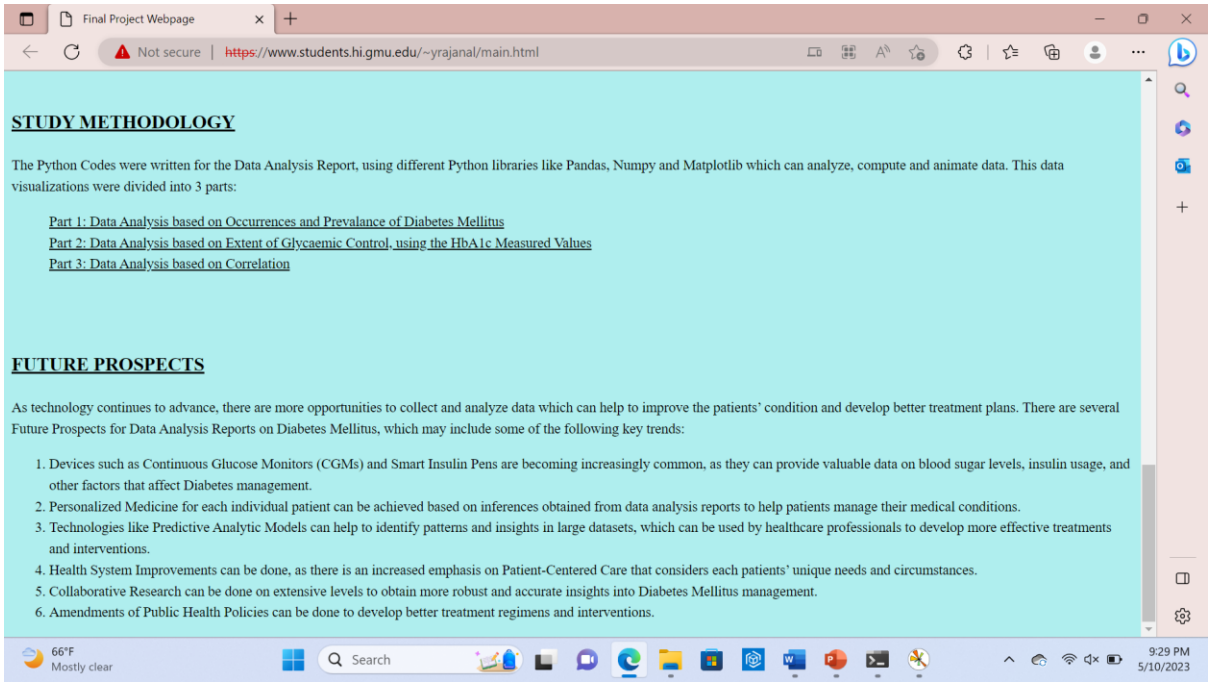
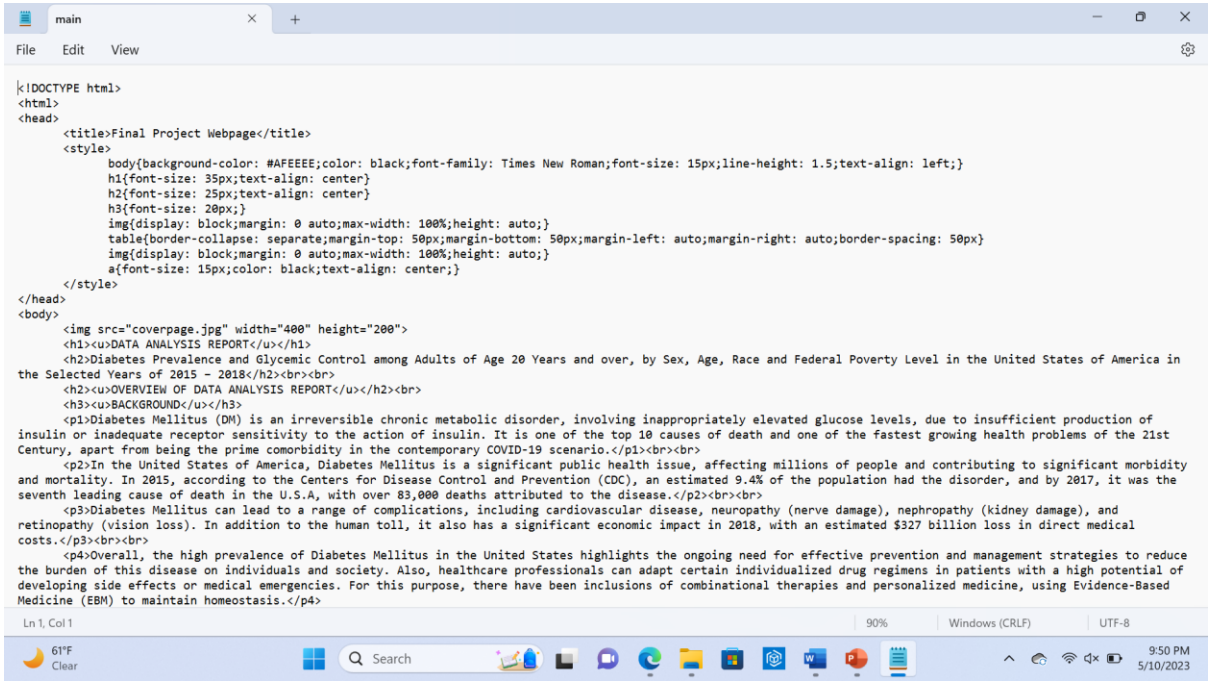
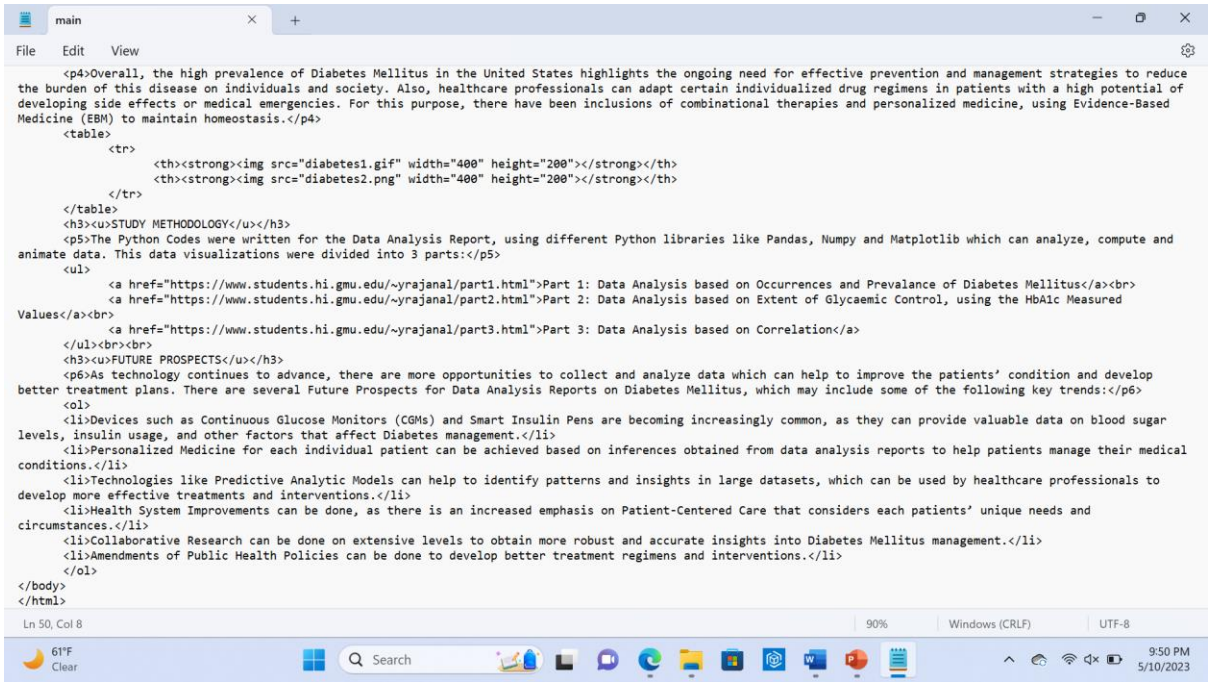


Image 24: Code for Main Webpage (a)



```
<!DOCTYPE html>
<html>
<head>
	<title>Final Project Webpage</title>
	<style>
		body{background-color: #AFEEEE;color: black;font-family: Times New Roman;font-size: 15px;line-height: 1.5;text-align: left;}
		h1{font-size: 35px;text-align: center}
		h2{font-size: 25px;text-align: center}
		h3{font-size: 20px;}
		img{display: block;margin: 0 auto;max-width: 100%;height: auto;}
		table{border-collapse: separate;margin-top: 50px;margin-bottom: 50px;margin-left: auto;margin-right: auto;border-spacing: 50px}
		img{display: block;margin: 0 auto;max-width: 100%;height: auto;}
		a{font-size: 15px;color: black;text-align: center;}
	</style>
</head>
<body>
	<img src="coverpage.jpg" width="400" height="200">
	<h1><u>DATA ANALYSIS REPORT</u></h1>
	<h2>Diabetes Prevalence and Glycemic Control among Adults of Age 20 Years and over, by Sex, Age, Race and Federal Poverty Level in the United States of America in the Selected Years of 2015 – 2018</h2><br><br>
	<h2><u>OVERVIEW OF DATA ANALYSIS REPORT</u></h2><br>
	<h3><u>BACKGROUND</u></h3>
	<p1>Diabetes Mellitus (DM) is an irreversible chronic metabolic disorder, involving inappropriately elevated glucose levels, due to insufficient production of insulin or inadequate receptor sensitivity to the action of insulin. It is one of the top 10 causes of death and one of the fastest growing health problems of the 21st Century, apart from being the prime comorbidity in the contemporary COVID-19 scenario.</p1><br><br>
	<p2>In the United States of America, Diabetes Mellitus is a significant public health issue, affecting millions of people and contributing to significant morbidity and mortality. In 2015, according to the Centers for Disease Control and Prevention (CDC), an estimated 9.4% of the population had the disorder, and by 2017, it was the seventh leading cause of death in the U.S.A, with over 83,000 deaths attributed to the disease.</p2><br><br>
	<p3>Diabetes Mellitus can lead to a range of complications, including cardiovascular disease, neuropathy (nerve damage), nephropathy (kidney damage), and retinopathy (vision loss). In addition to the human toll, it also has a significant economic impact in 2018, with an estimated $327 billion loss in direct medical costs.</p3><br><br>
	<p4>Overall, the high prevalence of Diabetes Mellitus in the United States highlights the ongoing need for effective prevention and management strategies to reduce the burden of this disease on individuals and society. Also, healthcare professionals can adapt certain individualized drug regimens in patients with a high potential of developing side effects or medical emergencies. For this purpose, there have been inclusions of combinational therapies and personalized medicine, using Evidence-Based Medicine (EBM) to maintain homeostasis.</p4>
```

Image 25: Code for Main Webpage (b)



```
	<p4>Overall, the high prevalence of Diabetes Mellitus in the United States highlights the ongoing need for effective prevention and management strategies to reduce the burden of this disease on individuals and society. Also, healthcare professionals can adapt certain individualized drug regimens in patients with a high potential of developing side effects or medical emergencies. For this purpose, there have been inclusions of combinational therapies and personalized medicine, using Evidence-Based Medicine (EBM) to maintain homeostasis.</p4>
	<table>
		<tr>
			<th><strong><img src="diabetes1.gif" width="400" height="200"></strong></th>
			<th><strong><img src="diabetes2.png" width="400" height="200"></strong></th>
		</tr>
	</table>
	<h3><u>STUDY METHODOLOGY</u></h3>
	<p5>The Python Codes were written for the Data Analysis Report, using different Python libraries like Pandas, Numpy and Matplotlib which can analyze, compute and animate data. This data visualizations were divided into 3 parts:</p5>
	<ul>
		<a href="https://www.students.hi.gmu.edu/~yrajanal/part1.html">Part 1: Data Analysis based on Occurrences and Prevalance of Diabetes Mellitus</a><br>
		<a href="https://www.students.hi.gmu.edu/~yrajanal/part2.html">Part 2: Data Analysis based on Extent of Glycaemic Control, using the HbA1c Measured Values</a><br>
		<a href="https://www.students.hi.gmu.edu/~yrajanal/part3.html">Part 3: Data Analysis based on Correlation</a>
	</ul><br><br>
	<h3><u>FUTURE PROSPECTS</u></h3>
	<p6>As technology continues to advance, there are more opportunities to collect and analyze data which can help to improve the patients' condition and develop better treatment plans. There are several Future Prospects for Data Analysis Reports on Diabetes Mellitus, which may include some of the following key trends:</p6>
	<ol>
	<li>Devices such as Continuous Glucose Monitors (CGMs) and Smart Insulin Pens are becoming increasingly common, as they can provide valuable data on blood sugar levels, insulin usage, and other factors that affect Diabetes management.</li>
	<li>Personalized Medicine for each individual patient can be achieved based on inferences obtained from data analysis reports to help patients manage their medical conditions.</li>
	<li>Technologies like Predictive Analytic Models can help to identify patterns and insights in large datasets, which can be used by healthcare professionals to develop more effective treatments and interventions.</li>
	<li>Health System Improvements can be done, as there is an increased emphasis on Patient-Centered Care that considers each patients' unique needs and circumstances.</li>
	<li>Collaborative Research can be done on extensive levels to obtain more robust and accurate insights into Diabetes Mellitus management.</li>
	<li>Amendments of Public Health Policies can be done to develop better treatment regimens and interventions.</li>
	</ol>
</body>
</html>
```

Image 26: Results: Part 1 Webpage (a)

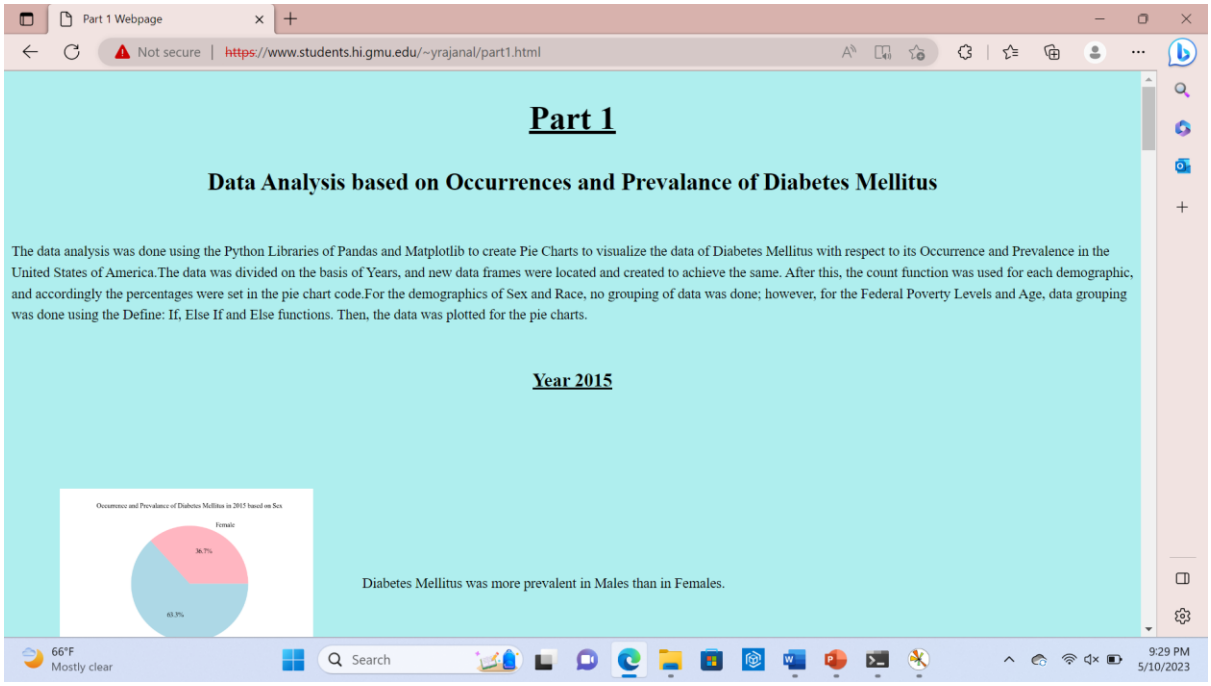


Image 27: Results: Part 1 Webpage (b)

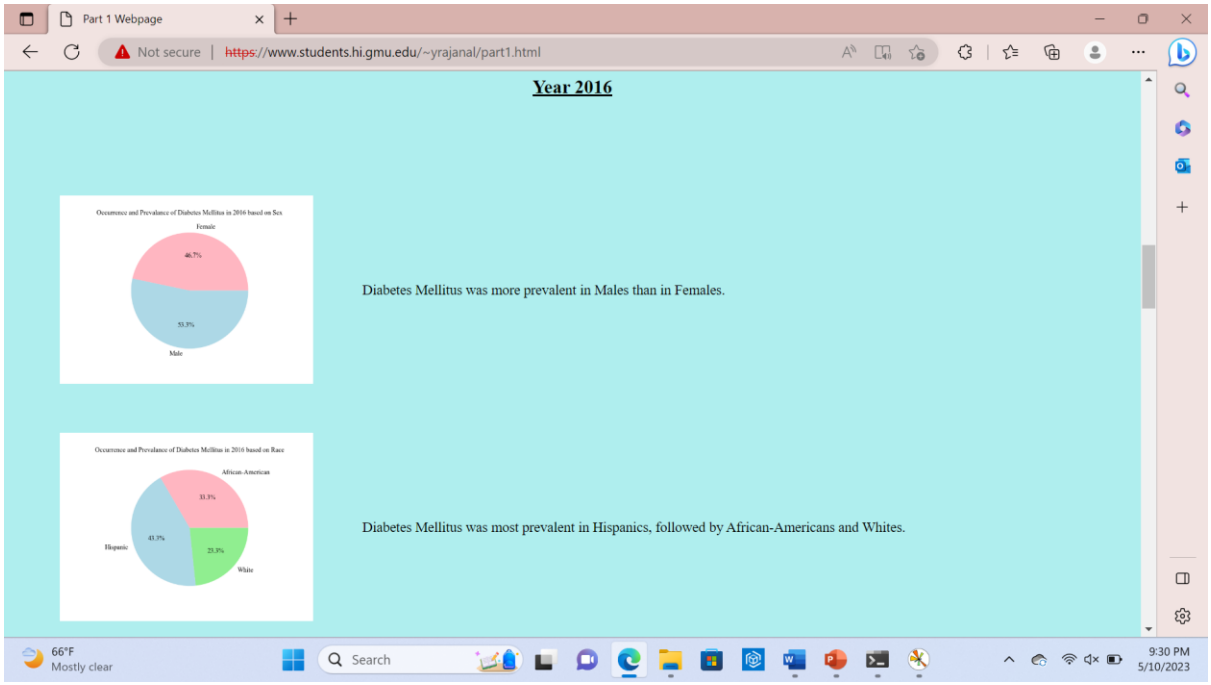
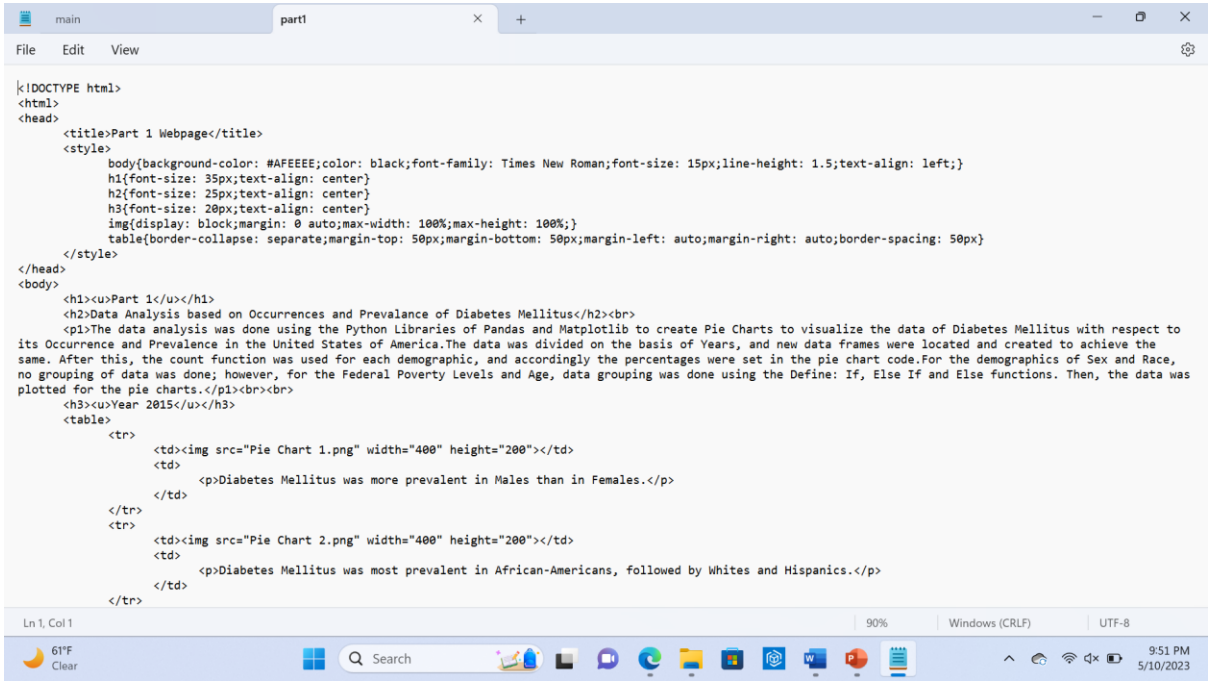
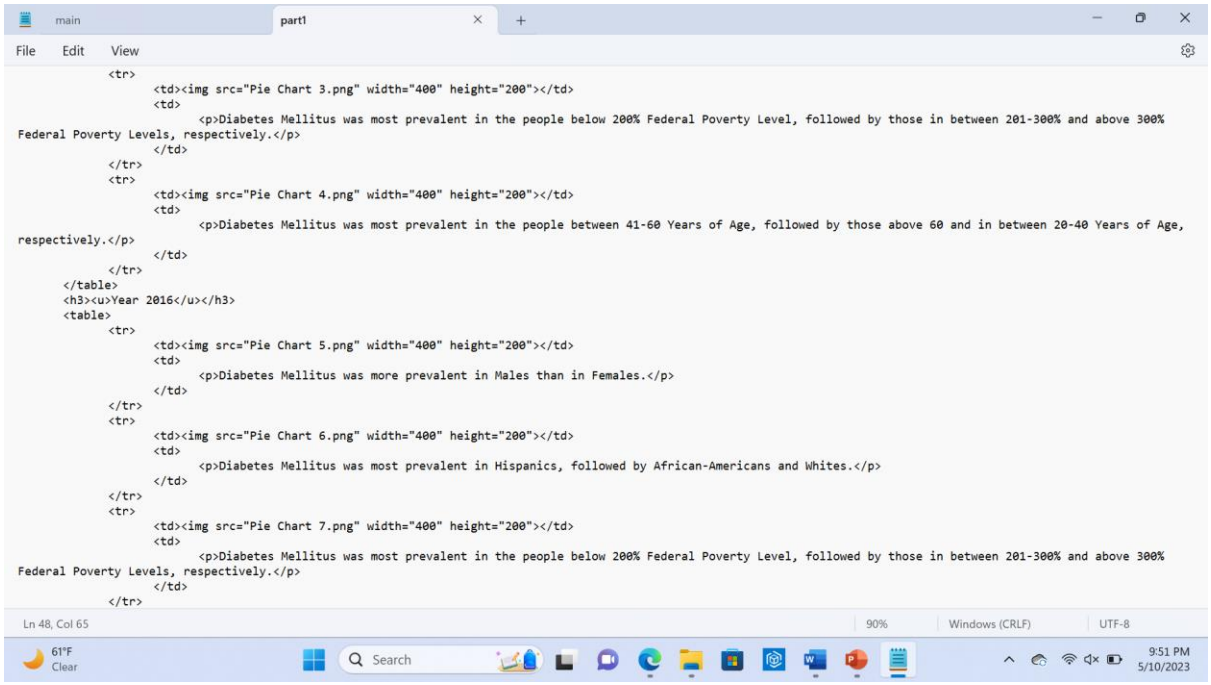


Image 28: Code for Part 1 Webpage (a)



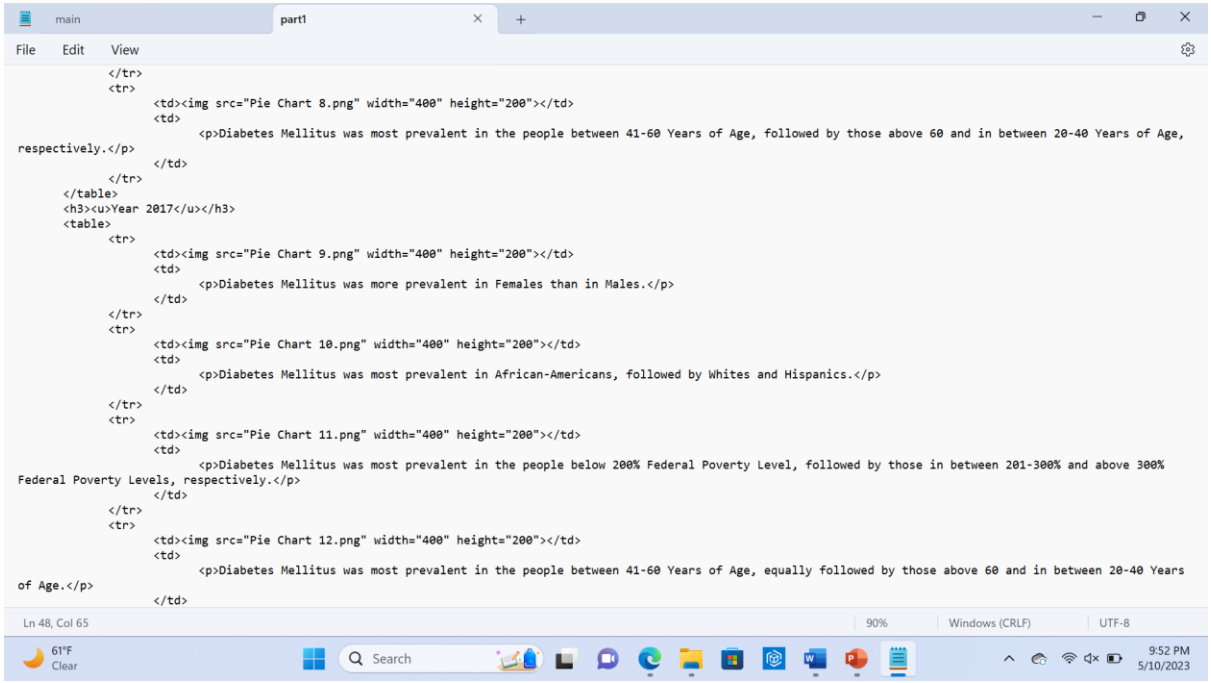
```
<!DOCTYPE html>
<html>
<head>
	<title>Part 1 Webpage</title>
	<style>
		body{background-color: #AFEEEE;color: black;font-family: Times New Roman;font-size: 15px;line-height: 1.5;text-align: left;}
		h1{font-size: 35px;text-align: center}
		h2{font-size: 25px;text-align: center}
		h3{font-size: 20px;text-align: center}
		img{display: block;margin: 0 auto;max-width: 100%;max-height: 100%;}
		table{border-collapse: separate;margin-top: 50px;margin-bottom: 50px;margin-left: auto;margin-right: auto;border-spacing: 50px}
	</style>
</head>
<body>
	<h1><u>Part 1</u></h1>
	<h2>Data Analysis based on Occurrences and Prevalance of Diabetes Mellitus</h2><br>
	<p1>The data analysis was done using the Python Libraries of Pandas and Matplotlib to create Pie Charts to visualize the data of Diabetes Mellitus with respect to
its Occurrence and Prevalence in the United States of America.The data was divided on the basis of Years, and new data frames were located and created to achieve the
same. After this, the count function was used for each demographic, and accordingly the percentages were set in the pie chart code.For the demographics of Sex and Race,
no grouping of data was done; however, for the Federal Poverty Levels and Age, data grouping was done using the Define: If, Else If and Else functions. Then, the data was
plotted for the pie charts.</p1><br><br>
	<h3><u>Year 2015</u></h3>
	<table>
		<tr>
			<td><img src="Pie Chart 1.png" width="400" height="200"></td>
			<td>
				<p>Diabetes Mellitus was more prevalent in Males than in Females.</p>
			</td>
		</tr>
		<tr>
			<td><img src="Pie Chart 2.png" width="400" height="200"></td>
			<td>
				<p>Diabetes Mellitus was most prevalent in African-Americans, followed by Whites and Hispanics.</p>
			</td>
		</tr>
```

Image 29: Code for Part 1 Webpage (b)



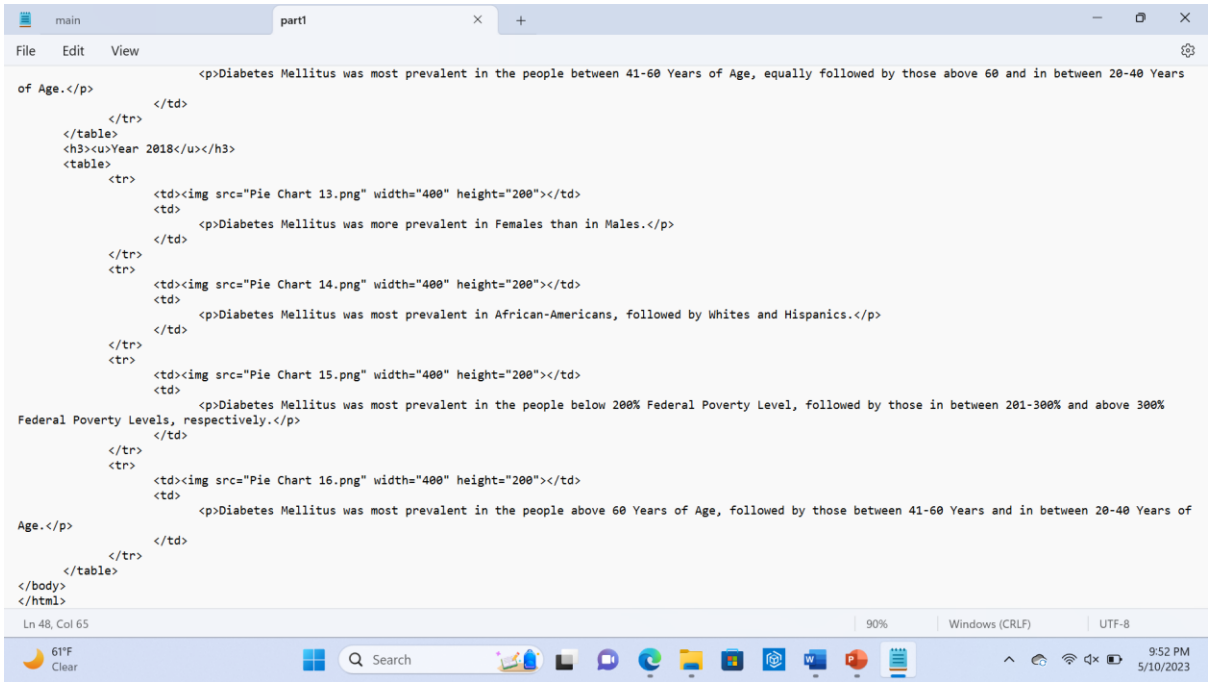
```
		<tr>
			<td><img src="Pie Chart 3.png" width="400" height="200"></td>
			<td>
				<p>Diabetes Mellitus was most prevalent in the people below 200% Federal Poverty Level, followed by those in between 201-300% and above 300%
Federal Poverty Levels, respectively.</p>
			</td>
		</tr>
		<tr>
			<td><img src="Pie Chart 4.png" width="400" height="200"></td>
			<td>
				<p>Diabetes Mellitus was most prevalent in the people between 41-60 Years of Age, followed by those above 60 and in between 20-40 Years of Age,
respectively.</p>
			</td>
		</tr>
	</table>
	<h3><u>Year 2016</u></h3>
	<table>
		<tr>
			<td><img src="Pie Chart 5.png" width="400" height="200"></td>
			<td>
				<p>Diabetes Mellitus was more prevalent in Males than in Females.</p>
			</td>
		</tr>
		<tr>
			<td><img src="Pie Chart 6.png" width="400" height="200"></td>
			<td>
				<p>Diabetes Mellitus was most prevalent in Hispanics, followed by African-Americans and Whites.</p>
			</td>
		</tr>
		<tr>
			<td><img src="Pie Chart 7.png" width="400" height="200"></td>
			<td>
				<p>Diabetes Mellitus was most prevalent in the people below 200% Federal Poverty Level, followed by those in between 201-300% and above 300%
Federal Poverty Levels, respectively.</p>
			</td>
		</tr>
```

Image 30: Code for Part 1 Webpage (c)



```
            </tr>
            <tr>
                  <td><img src="Pie Chart 8.png" width="400" height="200"></td>
                  <td>
                        <p>Diabetes Mellitus was most prevalent in the people between 41-60 Years of Age, followed by those above 60 and in between 20-40 Years of Age,
respectively.</p>
                  </td>
            </tr>
      </table>
      <h3><u>Year 2017</u></h3>
      <table>
            <tr>
                  <td><img src="Pie Chart 9.png" width="400" height="200"></td>
                  <td>
                        <p>Diabetes Mellitus was more prevalent in Females than in Males.</p>
                  </td>
            </tr>
            <tr>
                  <td><img src="Pie Chart 10.png" width="400" height="200"></td>
                  <td>
                        <p>Diabetes Mellitus was most prevalent in African-Americans, followed by Whites and Hispanics.</p>
                  </td>
            </tr>
            <tr>
                  <td><img src="Pie Chart 11.png" width="400" height="200"></td>
                  <td>
                        <p>Diabetes Mellitus was most prevalent in the people below 200% Federal Poverty Level, followed by those in between 201-300% and above 300%
Federal Poverty Levels, respectively.</p>
                  </td>
            </tr>
            <tr>
                  <td><img src="Pie Chart 12.png" width="400" height="200"></td>
                  <td>
                        <p>Diabetes Mellitus was most prevalent in the people between 41-60 Years of Age, equally followed by those above 60 and in between 20-40 Years
of Age.</p>
                  </td>
```

Image 31: Code for Part 1 Webpage (d)



```
                  <p>Diabetes Mellitus was most prevalent in the people between 41-60 Years of Age, equally followed by those above 60 and in between 20-40 Years
of Age.</p>
                  </td>
            </tr>
      </table>
      <h3><u>Year 2018</u></h3>
      <table>
            <tr>
                  <td><img src="Pie Chart 13.png" width="400" height="200"></td>
                  <td>
                        <p>Diabetes Mellitus was more prevalent in Females than in Males.</p>
                  </td>
            </tr>
            <tr>
                  <td><img src="Pie Chart 14.png" width="400" height="200"></td>
                  <td>
                        <p>Diabetes Mellitus was most prevalent in African-Americans, followed by Whites and Hispanics.</p>
                  </td>
            </tr>
            <tr>
                  <td><img src="Pie Chart 15.png" width="400" height="200"></td>
                  <td>
                        <p>Diabetes Mellitus was most prevalent in the people below 200% Federal Poverty Level, followed by those in between 201-300% and above 300%
Federal Poverty Levels, respectively.</p>
                  </td>
            </tr>
            <tr>
                  <td><img src="Pie Chart 16.png" width="400" height="200"></td>
                  <td>
                        <p>Diabetes Mellitus was most prevalent in the people above 60 Years of Age, followed by those between 41-60 Years and in between 20-40 Years of
Age.</p>
                  </td>
            </tr>
      </table>
</body>
</html>
```

Image 32: Results: Part 2 Webpage (a)

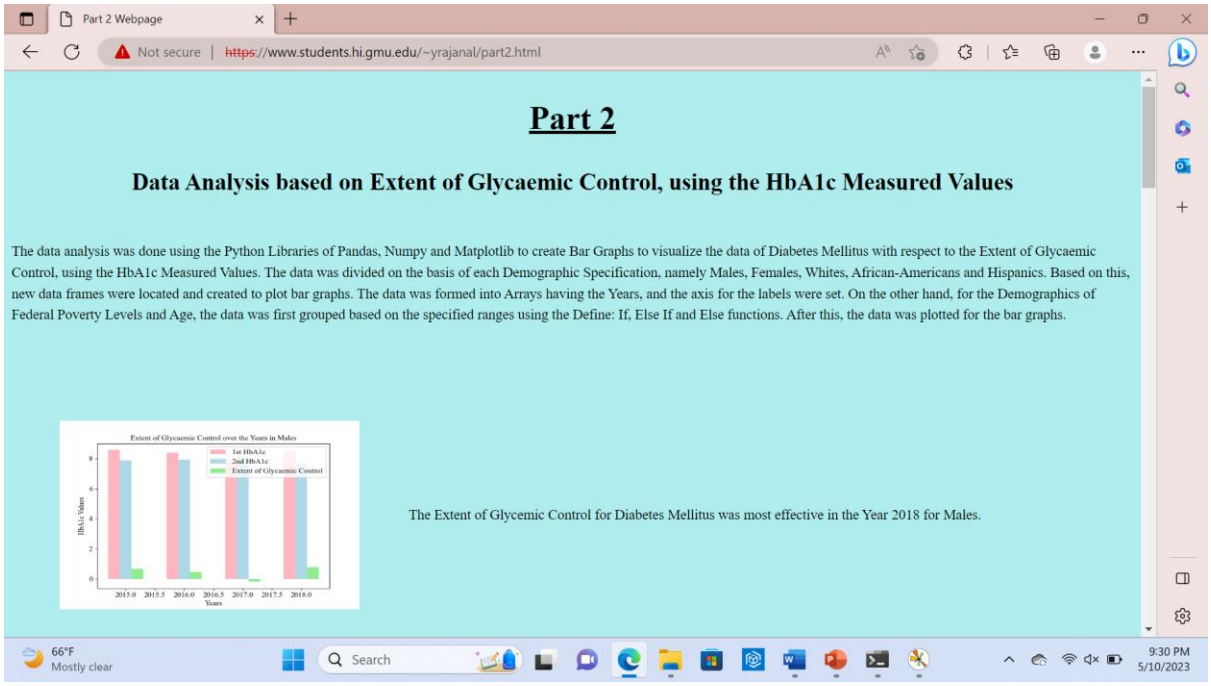


Image 33: Results: Part 2 Webpage (b)

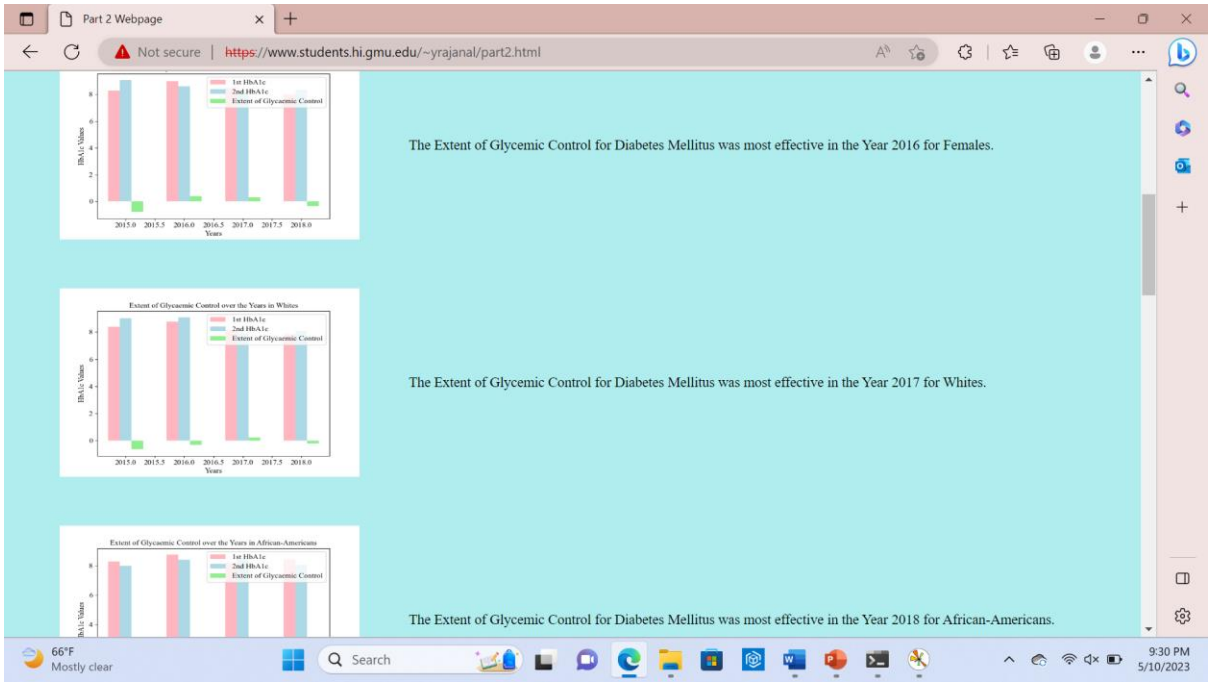
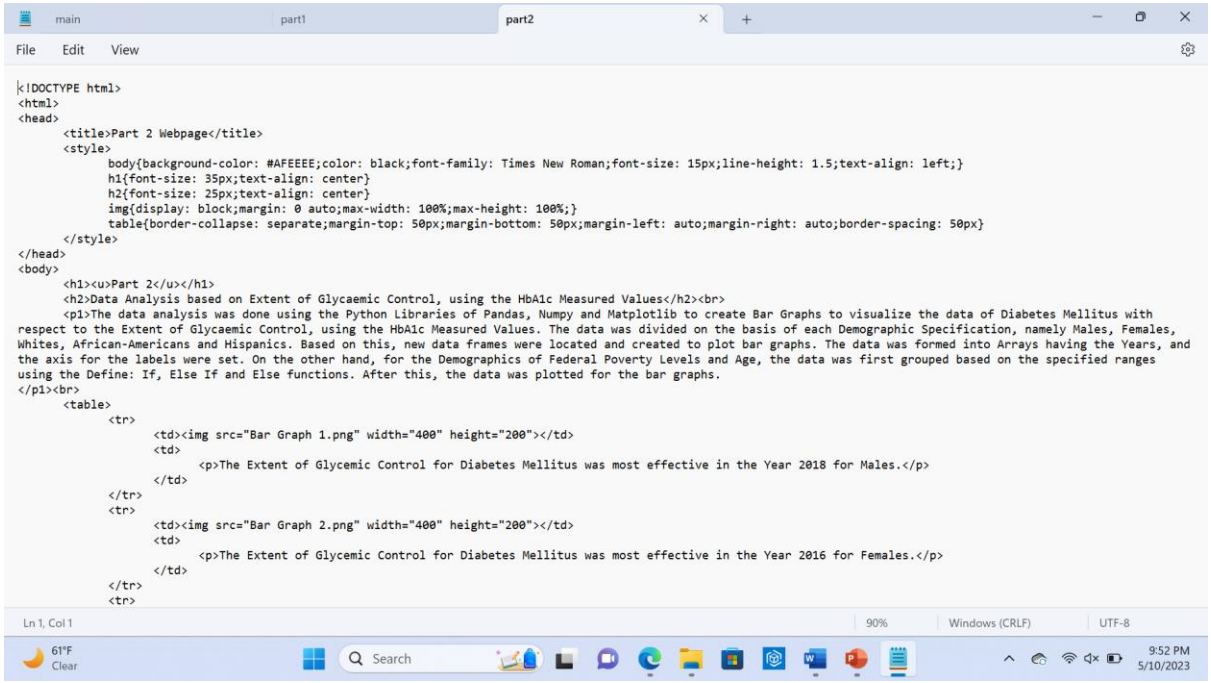
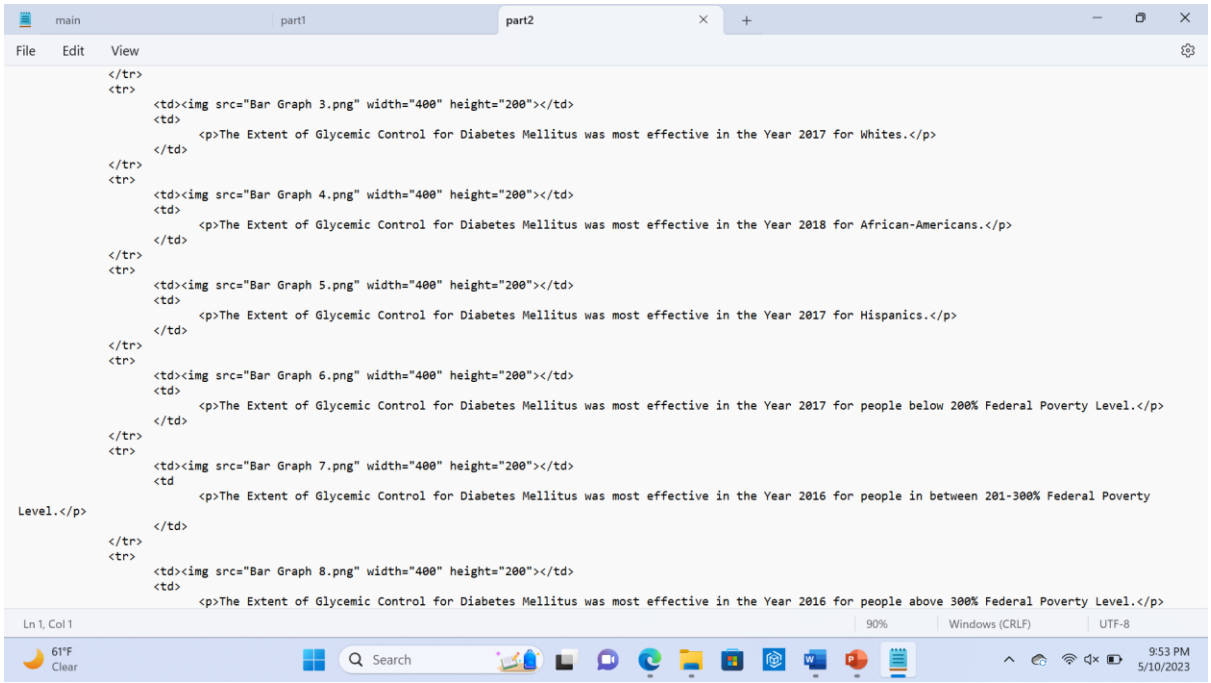


Image 34: Code for Part 2 Webpage (a)



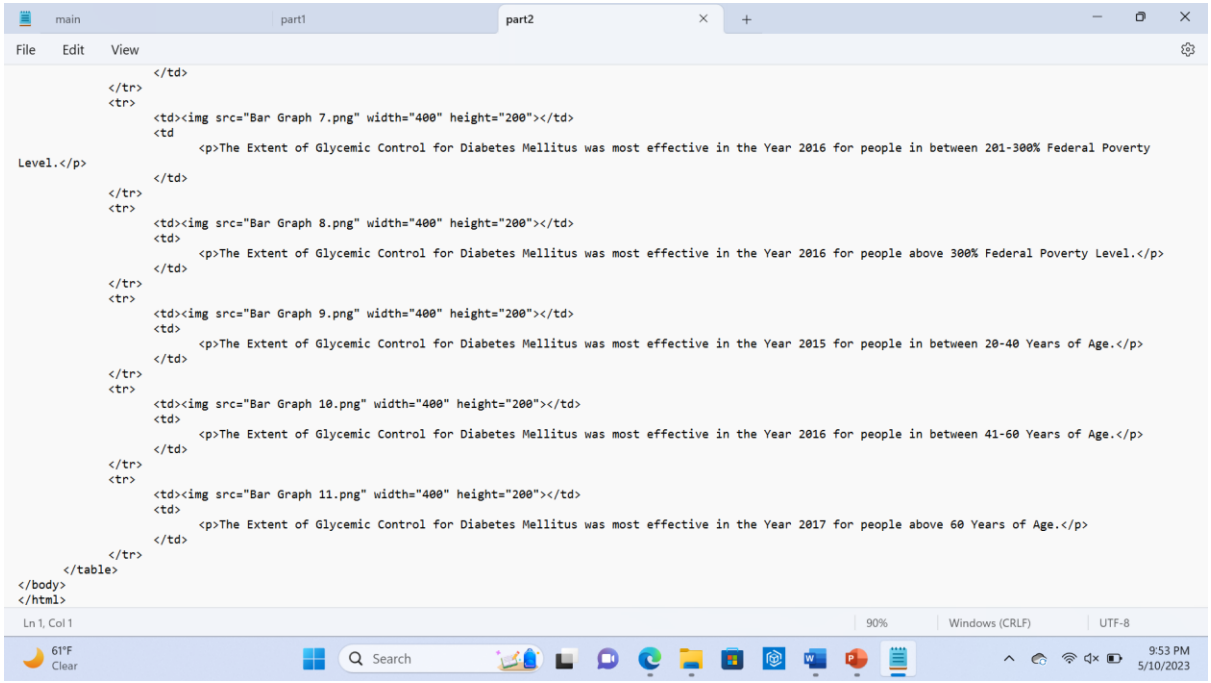
```
<!DOCTYPE html>
<html>
<head>
	<title>Part 2 Webpage</title>
	<style>
		body{background-color: #AFEEEE;color: black;font-family: Times New Roman;font-size: 15px;line-height: 1.5;text-align: left;}
		h1{font-size: 35px;text-align: center}
		h2{font-size: 25px;text-align: center}
		img{display: block;margin: 0 auto;max-width: 100%;max-height: 100%;}
		table{border-collapse: separate;margin-top: 50px;margin-bottom: 50px;margin-left: auto;margin-right: auto;border-spacing: 50px}
	</style>
</head>
<body>
	<h1><u>Part 2</u></h1>
	<h2>Data Analysis based on Extent of Glycaemic Control, using the HbA1c Measured Values</h2><br>
	<p1>The data analysis was done using the Python Libraries of Pandas, Numpy and Matplotlib to create Bar Graphs to visualize the data of Diabetes Mellitus with
respect to the Extent of Glycaemic Control, using the HbA1c Measured Values. The data was divided on the basis of each Demographic Specification, namely Males, Females,
Whites, African-Americans and Hispanics. Based on this, new data frames were located and created to plot bar graphs. The data was formed into Arrays having the Years, and
the axis for the labels were set. On the other hand, for the Demographics of Federal Poverty Levels and Age, the data was first grouped based on the specified ranges
using the Define: If, Else If and Else functions. After this, the data was plotted for the bar graphs.
</p1><br>
	<table>
		<tr>
			<td><img src="Bar Graph 1.png" width="400" height="200"></td>
			<td>
				<p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2018 for Males.</p>
			</td>
		</tr>
		<tr>
			<td><img src="Bar Graph 2.png" width="400" height="200"></td>
			<td>
				<p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2016 for Females.</p>
			</td>
		</tr>
		<tr>
```

Image 35: Code for Part 2 Webpage (b)



```
		</tr>
		<tr>
			<td><img src="Bar Graph 3.png" width="400" height="200"></td>
			<td>
				<p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2017 for Whites.</p>
			</td>
		</tr>
		<tr>
			<td><img src="Bar Graph 4.png" width="400" height="200"></td>
			<td>
				<p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2018 for African-Americans.</p>
			</td>
		</tr>
		<tr>
			<td><img src="Bar Graph 5.png" width="400" height="200"></td>
			<td>
				<p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2017 for Hispanics.</p>
			</td>
		</tr>
		<tr>
			<td><img src="Bar Graph 6.png" width="400" height="200"></td>
			<td>
				<p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2017 for people below 200% Federal Poverty Level.</p>
			</td>
		</tr>
		<tr>
			<td><img src="Bar Graph 7.png" width="400" height="200"></td>
			<td
				<p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2016 for people in between 201-300% Federal Poverty
Level.</p>
			</td>
		</tr>
		<tr>
			<td><img src="Bar Graph 8.png" width="400" height="200"></td>
			<td>
				<p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2016 for people above 300% Federal Poverty Level.</p>
```

Image 36: Code for Part 2 Webpage (c)



```
                </td>
            </tr>
            <tr>
                <td><img src="Bar Graph 7.png" width="400" height="200"></td>
                <td
                    <p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2016 for people in between 201-300% Federal Poverty
Level.</p>
                </td>
            </tr>
            <tr>
                <td><img src="Bar Graph 8.png" width="400" height="200"></td>
                <td>
                    <p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2016 for people above 300% Federal Poverty Level.</p>
                </td>
            </tr>
            <tr>
                <td><img src="Bar Graph 9.png" width="400" height="200"></td>
                <td>
                    <p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2015 for people in between 20-40 Years of Age.</p>
                </td>
            </tr>
            <tr>
                <td><img src="Bar Graph 10.png" width="400" height="200"></td>
                <td>
                    <p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2016 for people in between 41-60 Years of Age.</p>
                </td>
            </tr>
            <tr>
                <td><img src="Bar Graph 11.png" width="400" height="200"></td>
                <td>
                    <p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2017 for people above 60 Years of Age.</p>
                </td>
            </tr>
        </table>
</body>
</html>
```

Image 37: Results: Part 3 Webpage (a)

Image 38: Results: Part 2 Webpage (b)

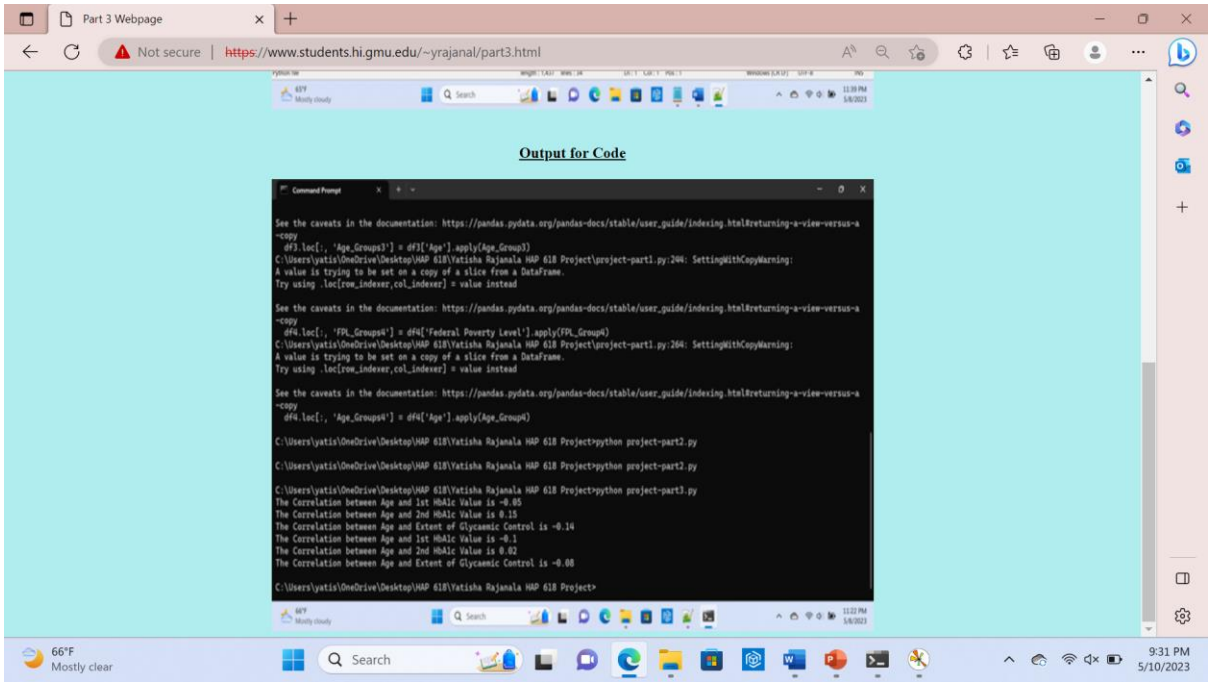
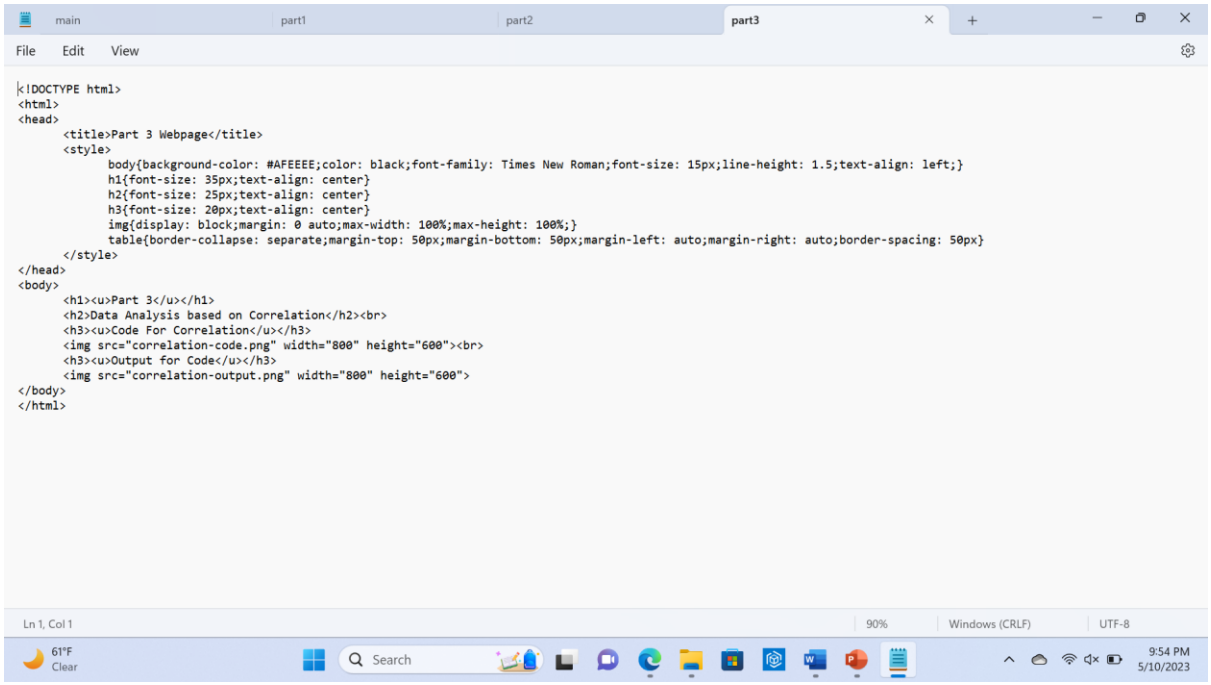


Image 39: Code for Part 3 Webpage



CONCLUSIONS

From the Data Analysis Report on the topic, "Diabetes Prevalence and Glycemic Control among Adults of Age 20 Years and over, by Sex, Age, Race, and Federal Poverty Level in the United States of America in the Selected Years of 2015 – 2018", I have learnt the following techniques in Computational Tools:

1. Writing a Python code in Notepad++, saving it as a .py file and its execution on Comand Prompt.
2. Importion of barried Python libraries like Pandas, NumPy, Matplotlib and FontProperties from Matplotlib.
3. Setting the Font for the data visualizations, including font name, font size and colors.
4. Opening the dataset in Python in the form of CSV file.
5. Creation of dataframes using Pandas.
6. Locating the dataframes using the .loc[] function
7. Grouping of the dataset ranges using the Define: If, Else If and Else functions.
8. Appending new columns to the dataframes.
9. Counting the data using the .count() function, and rounding off the values using the .round() function.
10. Assigning the Percentages to the dataset, and displaying the results using the print() function.
11. Removal of string values from the dataset using thr .drop() function.
12. Creation of an Array using np.array() from NumPy.
13. Correlating parameters using the .corr() function.
14. Plotting of the Pie Charts and Bar Graphs using Matplotlib, and saving them as PNG Images on the server.
15. Writing a HTML code in Notepad, saving it as a .html file and its execution on Web Page.
16. The inclusion of title, CSS style formats for the webpage, web page body, headings, paragraphs, tables with formatted images, unordered list of hyperlinks, ordered list of data.

FUTURE WORK

As technology continues to advance, there are more opportunities to collect and analyze data which can help to improve the patients' condition and develop better treatment plans. There are several Future Prospects for Data Analysis Reports on Diabetes Mellitus, which may include some of the following key trends:

1. Devices such as Continuous Glucose Monitors (CGMs) and Smart Insulin Pens are becoming increasingly common, as they can provide valuable data on blood sugar levels, insulin usage, and other factors that affect Diabetes Mellitus management.
2. Personalized Medicine for each individual patient can be achieved based on inferences obtained from data analysis reports to help patients manage their medical conditions.
3. Technologies like Predictive Analytic Models can help to identify patterns and insights regarding large datasets, which can be used by healthcare professionals to develop more effective treatments and interventions.
4. Health System Improvements can be made, as there is an increased emphasis on Patient-Centered Care that considers each patients' unique needs and circumstances.
5. Collaborative Research can be done on extensive levels to obtain more robust and accurate insights into Diabetes Mellitus management.
6. Amendments of Public Health Policies can be done to develop better treatment regimens and interventions.

REFERENCES

Barot, P. (2020, September 28). *Why use Python for building a Healthcare Application*. BoTree Technologies. https://www.botreetechnologies.com/blog/python-in-healthcare-application

Dsilva, D., Cudney, E. A., & Shah, H. A. (2018). Data analytics in healthcare: Past, present, and future. American Journal of Industrial and Business Management, 8(1), 147-157.

International Diabetes Federation. (2021). IDF Diabetes Atlas 10th edition 2021. Diabetesatlas.org. https://diabetesatlas.org/

Matplotlib: Python plotting — Matplotlib 3.1.1 documentation. (2012). Matplotlib.org. https://matplotlib.org

Professor Jinshan Tang (2023, January), PowerPoint Presentations and Class Notes, Master of Science Health Informatics Program, George Mason University.

Python in Healthcare: AI Applications in Hospitals. (n.d.). Www.datacamp.com. https://www.datacamp.com/blog/python-in-healthcare-ai-applications-in-hospitals

Selected Trend Table from Health, United States, 2011. Diabetes prevalence and glycemic control among adults 20 years of age and over, by sex, age, and race and Hispanic origin: United States, selected years 1988 - 1994 through 2015 - 2018 | HealthData.gov. (n.d.). Healthdata.gov. Retrieved May 2, 2023, from https://healthdata.gov/dataset/Selected-Trend-Table-from-Health-United-States-201/qkns-8p7f

Zafar, F., Raza, S., Khalid, M. U., & Tahir, M. A. (2019). Predictive Analytics in Healthcare for Diabetes Prediction. Proceedings of the 2019 9th International Conference on Biomedical Engineering and Technology - ICBET' 19. https://doi.org/10.1145/3326172.3326213