



---

# 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.

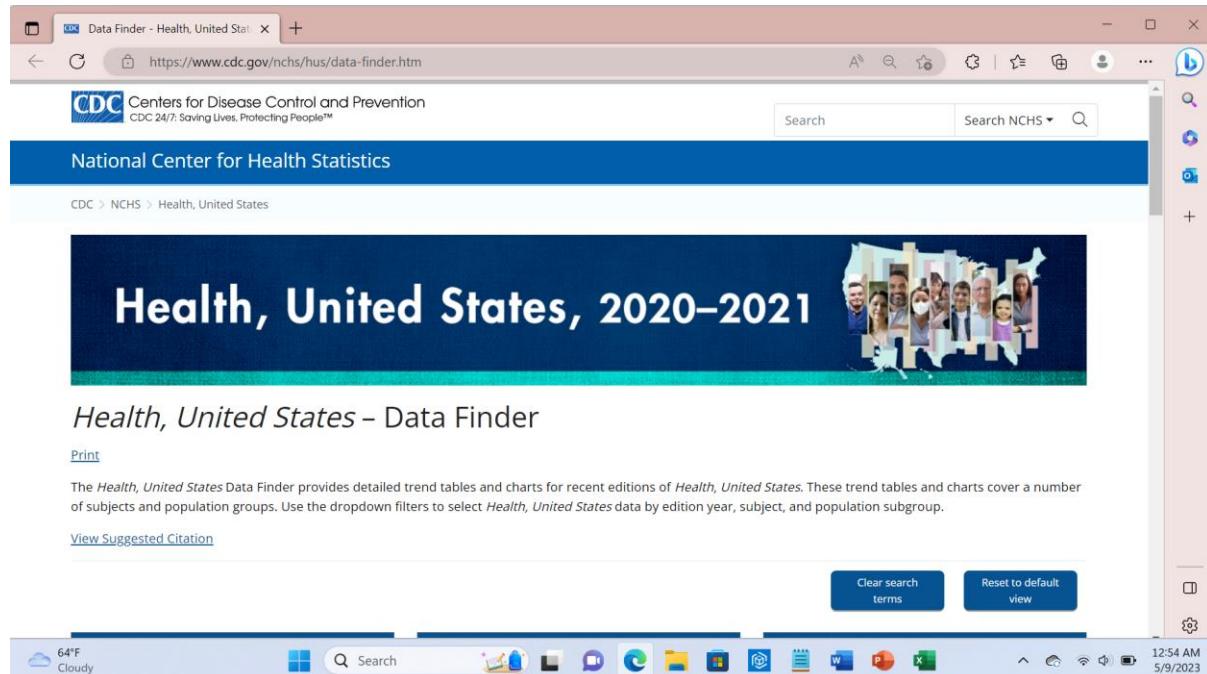
## SOLUTION DESIGN

### 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<sub>1c</sub> 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 HbA<sub>1c</sub>, 2nd HbA<sub>1c</sub> 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

	A	B	C	D	E	F	G	H	I	J	K
1	Patient Number	Year	Age	Sex	Race	Federal Poverty Level	Medical Status	1st HbA1c	2nd HbA1c	Extent of Glycaemic Control	
2	1	2015	45	Female	White	100	Physician - Diagnosed	9.7	11.9	-2.2	
3	2	2015	67	Female	White	200	Physician - Diagnosed	6.9	10.8	-3.9	
4	3	2015	75	Female	White	300	Undiagnosed	7.3	9.7	-2.4	
5	4	2015	65	Female	White	400	Undiagnosed	8.2	8.6	-0.4	
6	5	2015	54	Female	African-American	200	Physician - Diagnosed	10.4	7.5	2.9	
7	6	2015	66	Female	African-American	300	Undiagnosed	6.4	6.4	0	
8	7	2015	57	Female	African-American	300	Undiagnosed	7.7	5.3	2.4	
9	8	2015	41	Female	African-American	400	Undiagnosed	8.8	11.5	-2.7	
10	9	2015	72	Female	African-American	400	Undiagnosed	7.9	10.4	-2.5	
11	10	2015	43	Female	Hispanic	200	Physician - Diagnosed	9.9	9.3	0.6	
12	11	2015	85	Female	Hispanic	200	Physician - Diagnosed	7.7	8.2	-0.5	
13	12	2015	45	Male	White	100	Physician - Diagnosed	8.8	7.1	1.7	
14	13	2015	32	Male	White	100	Physician - Diagnosed	7.1	6	1.1	
15	14	2015	28	Male	White	200	Physician - Diagnosed	10.1	4.9	5.2	
16	15	2015	37	Male	White	200	Physician - Diagnosed	9.1	11.7	-2.6	
17	16	2015	49	Male	White	200	Physician - Diagnosed	8.4	10.6	-2.2	
18	17	2015	50	Male	African-American	300	Undiagnosed	7.4	9.5	-2.1	
19	18	2015	28	Male	African-American	300	Undiagnosed	9.9	8.4	1.5	
20	19	2015	45	Male	African-American	300	Undiagnosed	7.3	7.3	0	
21	20	2015	38	Male	African-American	300	Undiagnosed	8.4	6.2	2.2	

### 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

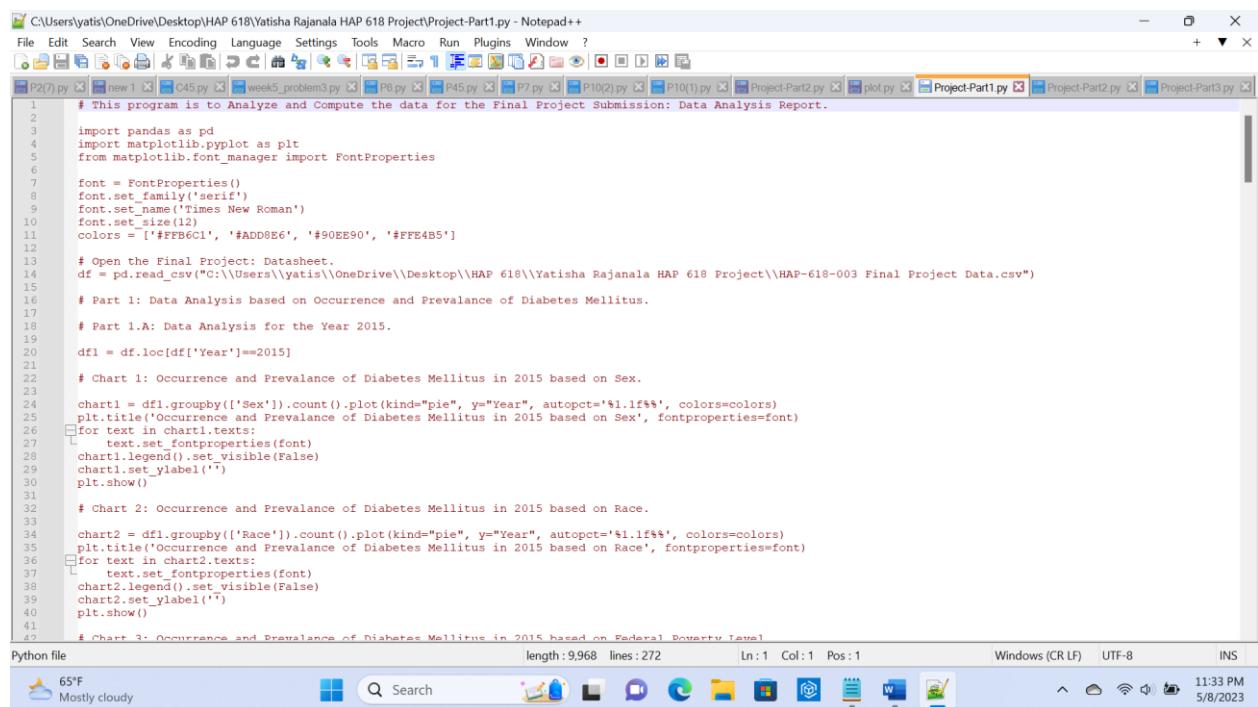
## Part 1: Data Analysis based on Occurrences and Prevalance of Diabetes Mellitus

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)**



```

C:\Users\yatis\OneDrive\Desktop\HAP 618\Yatisha Rajanala HAP 618 Project\Project-Part1.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
P2(7).py new 1 C45.py week5_problem3.py P8.py P45.py P7.py P10(2).py P10(1).py Project-Part2.py plot.py Project-Part1.py Project-Part2.py Project-Part3.py
1 # This program is to Analyze and Compute the data for the Final Project Submission: Data Analysis Report.
2
3 import pandas as pd
4 import matplotlib.pyplot as plt
5 from matplotlib.font_manager import FontProperties
6
7 font = FontProperties()
8 font.set_family('serif')
9 font.set_name('Times New Roman')
10 font.set_size(12)
11 colors = ['#FFB6C1', '#ADD8E6', '#90EE90', '#FFE4B5']
12
13 # Open the Final Project: Datasheet.
14 df = pd.read_csv("C:\\\\Users\\\\yatis\\\\OneDrive\\\\Desktop\\\\HAP 618\\\\Yatisha Rajanala HAP 618 Project\\\\HAP-618-003 Final Project Data.csv")
15
16 # Part 1: Data Analysis based on Occurrence and Prevalance of Diabetes Mellitus.
17
18 # Part 1.A: Data Analysis for the Year 2015.
19
20 dfl = df.loc[df['Year']==2015]
21
22 # Chart 1: Occurrence and Prevalance of Diabetes Mellitus in 2015 based on Sex.
23
24 chart1 = dfl.groupby(['Sex']).count().plot(kind="pie", y="Year", autopct='%.1f%%', colors=colors)
25 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2015 based on Sex', fontproperties=font)
26 for text in chart1texts:
27     text.set_fontproperties(font)
28     chart1.legend().set_visible(False)
29     chart1.set_ylabel('')
30     plt.show()
31
32 # Chart 2: Occurrence and Prevalance of Diabetes Mellitus in 2015 based on Race.
33
34 chart2 = dfl.groupby(['Race']).count().plot(kind="pie", y="Year", autopct='%.1f%%', colors=colors)
35 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2015 based on Race', fontproperties=font)
36 for text in chart2texts:
37     text.set_fontproperties(font)
38     chart2.legend().set_visible(False)
39     chart2.set_ylabel('')
40     plt.show()
41
42 # Chart 3: Occurrence and Prevalance of Diabetes Mellitus in 2015 based on Federal Poverty Level.

```

Python file length:9,968 lines:272 Ln:1 Col:1 Pos:1 Windows (CR LF) UTF-8 INS

65°F Mostly cloudy Search 11:33 PM 5/8/2023

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

```

C:\Users\yatis\OneDrive\Desktop\HAP 618\Yatisha Rajanala HAP 618 Project\Project-Part1.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
P2(7).py new 1 C45.py week5_problem3.py P6.py P45.py P7.py P10(2).py P10(1).py Project-Part2.py plot.py Project-Part1.py Project-Part2.py Project-Part3.py
41 # Chart 3: Occurrence and Prevalance of Diabetes Mellitus in 2015 based on Federal Poverty Level.
42 def FPL_Group1(Federal_Poverty_Level):
43     if Federal_Poverty_Level <299:
44         return '< 200%'
45     elif Federal_Poverty_Level <399:
46         return '201-300%'
47     else:
48         return '> 300%'
49
50 df1.loc[:, 'FPL_Groups1'] = df1['Federal Poverty Level'].apply(FPL_Group1)
51
52 chart3 = df1.groupby(['FPL_Groups1']).count().plot(kind="pie", y="Year", autopct='%.1f%%', colors=colors)
53 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2015 based on Federal Poverty Level', fontproperties=font)
54 for text in chart3.texts:
55     text.set_fontproperties(font)
56     chart3.legend().set_visible(False)
57     chart3.set_ylabel('')
58     plt.show()
59
60 # Chart 4: Occurrence and Prevalance of Diabetes Mellitus in 2015 based on Age.
61
62 def Age_Group1(Age):
63     if Age < 40:
64         return '20-40 Years'
65     elif Age < 60:
66         return '41-60 Years'
67     else:
68         return '> 60 Years'
69
70 df1.loc[:, 'Age_Groups1'] = df1['Age'].apply(Age_Group1)
71
72 chart4 = df1.groupby(['Age_Groups1']).count().plot(kind="pie", y="Year", autopct='%.1f%%', colors=colors)
73 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2015 based on Age', fontproperties=font)
74 for text in chart4.texts:
75     text.set_fontproperties(font)
76     chart4.legend().set_visible(False)
77     chart4.set_ylabel('')
78     plt.show()
79
80 # Part 1 B: Data Analysis for the Year 2016
81
Python file length: 9.968 lines : 272 Ln: 1 Col: 1 Pos: 1 Windows (CR LF) UTF-8 INS
82 65°F Mostly cloudy 11:34 PM 5/8/2023

```

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

```

C:\Users\yatis\OneDrive\Desktop\HAP 618\Yatisha Rajanala HAP 618 Project\Project-Part1.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
P2(7).py new 1 C45.py week5_problem3.py P6.py P45.py P7.py P10(2).py P10(1).py Project-Part2.py plot.py Project-Part1.py Project-Part2.py Project-Part3.py
81 # Part 1.B: Data Analysis for the Year 2016.
82 df2 = df.loc[df['Year']==2016]
83
84 # Chart 5: Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Sex.
85
86 chart5 = df2.groupby(['Sex']).count().plot(kind="pie", y="Year", autopct='%.1f%%', colors=colors)
87 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Sex', fontproperties=font)
88 for text in chart5.texts:
89     text.set_fontproperties(font)
90     chart5.legend().set_visible(False)
91     chart5.set_ylabel('')
92     plt.show()
93
94 # Chart 6: Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Race.
95
96 chart6 = df2.groupby(['Race']).count().plot(kind="pie", y="Year", autopct='%.1f%%', colors=colors)
97 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Race', fontproperties=font)
98 for text in chart6.texts:
99     text.set_fontproperties(font)
100    chart6.legend().set_visible(False)
101    chart6.set_ylabel('')
102    plt.show()
103
104 # Chart 7: Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Federal Poverty Level.
105
106 def FPL_Group2(Federal_Poverty_Level):
107     if Federal_Poverty_Level <299:
108         return '< 200%'
109     elif Federal_Poverty_Level <399:
110         return '201-300%'
111     else:
112         return '> 300%'
113
114 df2.loc[:, 'FPL_Groups2'] = df2['Federal Poverty Level'].apply(FPL_Group2)
115
116 chart7 = df2.groupby(['FPL_Groups2']).count().plot(kind="pie", y="Year", autopct='%.1f%%', colors=colors)
117 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Federal Poverty Level', fontproperties=font)
118 for text in chart7.texts:
119     text.set_fontproperties(font)
120     chart7.legend().set_visible(False)
121     chart7.set_ylabel('')
122     plt.show()
123
Python file length: 9.968 lines : 272 Ln: 1 Col: 1 Pos: 1 Windows (CR LF) UTF-8 INS
124 65°F Mostly cloudy 11:34 PM 5/8/2023

```

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

```

C:\Users\yatis\OneDrive\Desktop\HAP 618\Yatisha Rajanala HAP 618 Project\Project-Part1.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
P2(7).py new1 045.py week5_problem3.py P6.py P45.py P7.py P10(2).py P10(1).py Project-Part2.py plot.py Project-Part1.py Project-Part2.py Project-Part3.py
120     for text in chart7.texts:
121         text.set_fontproperties(font)
122         chart7.legend().set_visible(False)
123         chart7.set_ylabel('')
124         plt.show()
125
126     # Chart 8: Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Age.
127
128     def Age_Group2(Age):
129         if Age < 40:
130             return '20-40 Years'
131         elif Age < 60:
132             return '41-60 Years'
133         else:
134             return '> 60 Years'
135
136     df2.loc[:, 'Age_Groups2'] = df2['Age'].apply(Age_Group2)
137
138     chart8 = df2.groupby(['Age_Groups2']).count().plot(kind="pie", y="Year", autopct='%.1f%%', colors=colors)
139     plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2016 based on Age', fontproperties=font)
140     for text in chart8.texts:
141         text.set_fontproperties(font)
142         chart8.legend().set_visible(False)
143         chart8.set_ylabel('')
144         plt.show()
145
146     # Part 1.C: Data Analysis for the Year 2017.
147
148     df3 = df.loc[df['Year']==2017]
149
150     # Chart 9: Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Sex.
151
152     chart9 = df3.groupby(['Sex']).count().plot(kind="pie", y="Year", autopct='%.1f%%', colors=colors)
153     plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Sex', fontproperties=font)
154     for text in chart9.texts:
155         text.set_fontproperties(font)
156         chart9.legend().set_visible(False)
157         chart9.set_ylabel('')
158         plt.show()
159
160     # Chart 10: Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Race.
161
Python file length: 9,968 lines : 272 Ln: 1 Col: 1 Pos: 1 Windows (CR LF) UTF-8 INS
65°F Mostly cloudy 11:34 PM 5/8/2023

```

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

```

C:\Users\yatis\OneDrive\Desktop\HAP 618\Yatisha Rajanala HAP 618 Project\Project-Part1.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
P2(7).py new1 045.py week5_problem3.py P6.py P45.py P7.py P10(2).py P10(1).py Project-Part2.py plot.py Project-Part1.py Project-Part2.py Project-Part3.py
160     # Chart 10: Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Race.
161
162     chart10 = df3.groupby(['Race']).count().plot(kind="pie", y="Year", autopct='%.1f%%', colors=colors)
163     plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Race', fontproperties=font)
164     for text in chart10.texts:
165         text.set_fontproperties(font)
166         chart10.legend().set_visible(False)
167         chart10.set_ylabel('')
168         plt.show()
169
170     # Chart 11: Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Federal Poverty Level.
171
172     def FPL_Group3(Federal_Poverty_Level):
173         if Federal_Poverty_Level <299:
174             return '< 200%'
175         elif Federal_Poverty_Level <399:
176             return '201-300%'
177         else:
178             return '> 300%'
179
180     df3.loc[:, 'FPL_Groups3'] = df3['Federal Poverty Level'].apply(FPL_Group3)
181
182     chart11 = df3.groupby(['FPL_Groups3']).count().plot(kind="pie", y="Year", autopct='%.1f%%', colors=colors)
183     plt.title('Occurrences and Prevalance of Diabetes Mellitus in 2017 based on Federal Poverty Level', fontproperties=font)
184     for text in chart11.texts:
185         text.set_fontproperties(font)
186         chart11.legend().set_visible(False)
187         chart11.set_ylabel('')
188         plt.show()
189
190     # Chart 12: Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Age.
191
192     def Age_Group3(Age):
193         if Age < 40:
194             return '20-40 Years'
195         elif Age < 60:
196             return '41-60 Years'
197         else:
198             return '> 60 Years'
199
200     df3.loc[:, 'Age_Groups3'] = df3['Age'].apply(Age_Group3)
201
Python file length: 9,968 lines : 272 Ln: 1 Col: 1 Pos: 1 Windows (CR LF) UTF-8 INS
65°F Mostly cloudy 11:35 PM 5/8/2023

```

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

```

201 chart12 = df3.groupby(['Age_Groups3']).count().plot(kind="pie", y="Year", autopct='%.1lf%%', colors=colors)
202 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2017 based on Age', fontproperties=font)
203 for text in chart12.texts:
204     text.set_fontproperties(font)
205 chart12.legend().set_visible(False)
206 chart12.set_ylabel('')
207 plt.show()
208
209 # Part 1.D: Data Analysis for the Year 2018.
210
211 df4 = df.loc[df['Year']==2018]
212
213 # Chart 13: Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Sex.
214
215 chart13 = df4.groupby(['Sex']).count().plot(kind="pie", y="Year", autopct='%.1lf%%', colors=colors)
216 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Sex', fontproperties=font)
217 for text in chart13.texts:
218     text.set_fontproperties(font)
219 chart13.legend().set_visible(False)
220 chart13.set_ylabel('')
221 plt.show()
222
223 # Chart 14: Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Race.
224
225 chart14 = df4.groupby(['Race']).count().plot(kind="pie", y="Year", autopct='%.1lf%%', colors=colors)
226 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Race', fontproperties=font)
227 for text in chart14.texts:
228     text.set_fontproperties(font)
229 chart14.legend().set_visible(False)
230 chart14.set_ylabel('')
231 plt.show()
232
233 # Chart 15: Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Federal Poverty Level.
234
235 def FPL_Group4(Federal_Poverty_Level):
236     if Federal_Poverty_Level <299:
237         return '< 200$'
238     elif Federal_Poverty_Level <399:
239         return '201-300$'
240     else:
241         return '> 300$'
242
243
244 # Chart 15: Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Federal Poverty Level.
245
246 def FPL_Group4(Federal_Poverty_Level):
247     if Federal_Poverty_Level <299:
248         return '< 200$'
249     elif Federal_Poverty_Level <399:
250         return '201-300$'
251     else:
252         return '> 300$'
253
254 df4.loc[:, 'FPL_Groups4'] = df4['Federal Poverty Level'].apply(FPL_Group4)
255
256 chart15 = df4.groupby(['FPL_Groups4']).count().plot(kind="pie", y="Year", autopct='%.1lf%%', colors=colors)
257 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Federal Poverty Level', fontproperties=font)
258 for text in chart15.texts:
259     text.set_fontproperties(font)
260 chart15.legend().set_visible(False)
261 chart15.set_ylabel('')
262 plt.show()
263
264 # Chart 4: Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Age.
265
266 def Age_Group4(Age):
267     if Age < 40:
268         return '20-40 Years'
269     elif Age < 60:
270         return '41-60 Years'
271     else:
272         return '> 60 Years'
273
274 df4.loc[:, 'Age_Groups4'] = df4['Age'].apply(Age_Group4)
275
276 chart16 = df4.groupby(['Age_Groups4']).count().plot(kind="pie", y="Year", autopct='%.1lf%%', colors=colors)
277 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Age', fontproperties=font)
278 for text in chart16.texts:
279     text.set_fontproperties(font)
280 chart16.legend().set_visible(False)
281 chart16.set_ylabel('')
282 plt.show()

```

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

```

234 # Chart 15: Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Federal Poverty Level.
235
236 def FPL_Group4(Federal_Poverty_Level):
237     if Federal_Poverty_Level <299:
238         return '< 200$'
239     elif Federal_Poverty_Level <399:
240         return '201-300$'
241     else:
242         return '> 300$'
243
244 df4.loc[:, 'FPL_Groups4'] = df4['Federal Poverty Level'].apply(FPL_Group4)
245
246 chart15 = df4.groupby(['FPL_Groups4']).count().plot(kind="pie", y="Year", autopct='%.1lf%%', colors=colors)
247 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Federal Poverty Level', fontproperties=font)
248 for text in chart15.texts:
249     text.set_fontproperties(font)
250 chart15.legend().set_visible(False)
251 chart15.set_ylabel('')
252 plt.show()
253
254 # Chart 4: Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Age.
255
256 def Age_Group4(Age):
257     if Age < 40:
258         return '20-40 Years'
259     elif Age < 60:
260         return '41-60 Years'
261     else:
262         return '> 60 Years'
263
264 df4.loc[:, 'Age_Groups4'] = df4['Age'].apply(Age_Group4)
265
266 chart16 = df4.groupby(['Age_Groups4']).count().plot(kind="pie", y="Year", autopct='%.1lf%%', colors=colors)
267 plt.title('Occurrence and Prevalance of Diabetes Mellitus in 2018 based on Age', fontproperties=font)
268 for text in chart16.texts:
269     text.set_fontproperties(font)
270 chart16.legend().set_visible(False)
271 chart16.set_ylabel('')
272 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 occurrence and prevalence of Diabetes Mellitus were created; including those on the basis of Sex, Race, Federal Poverty Level and Age.
7. For the Pie Charts based on Sex and Race:
  - 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. For the Pie Charts based on Federal Poverty Level and Age:
  - 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 Demographic 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)**

```

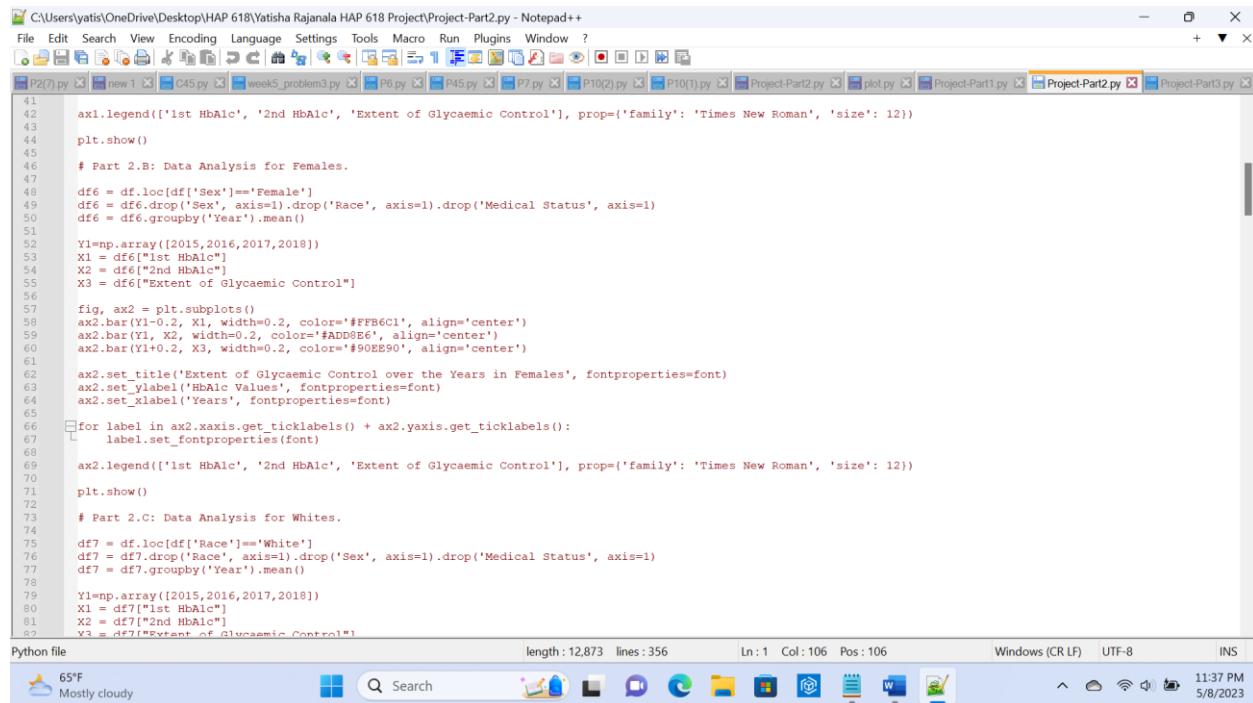
C:\Users\yatish\OneDrive\Desktop\HAP 618\Yatisha Rajanala HAP 618 Project\Project-Part2.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
P2(7).py new 1 C45.py week5_problem3.py P6.py P45.py P7.py P10(1).py Project-Part2.py plot.py Project-Part1.py Project-Part2.py Project-Part3.py
1 # This program is to Analyze and Compute the data for the Final Project Submission: Data Analysis Report.
2
3 import pandas as pd
4 import numpy as np
5 import matplotlib.pyplot as plt
6 from matplotlib.font_manager import FontProperties
7
8 font = FontProperties()
9 font.set_family('serif')
10 font.set_name('Times New Roman')
11 font.set_size(12)
12 colors = ['#FFB6C1', '#ADD8E6', '#90EE90', '#FFE4B5']
13
14 # Open the Final Project: Datasheet.
15 df = pd.read_csv("C:\\\\Users\\\\yatish\\\\OneDrive\\\\HAP 618\\\\Yatisha Rajanala HAP 618 Project\\\\HAP-618-003 Final Project Data.csv")
16
17 # Part 2: Data Analysis based on Extent of Glycaemic Control, using the HbA1c Measured Values.
18
19 # Part 2.A: Data Analysis for Males.
20
21 df5 = df.loc[df['Sex']=='Male']
22 df5 = df5.drop(['Sex'], axis=1).drop('Race', axis=1).drop('Medical Status', axis=1)
23 df5 = df5.groupby('Year').mean()
24
25 Y1=np.array([2015,2016,2017,2018])
26 X1 = df5["1st HbA1c"]
27 X2 = df5["2nd HbA1c"]
28 X3 = df5["Extent of Glycaemic Control"]
29
30 fig, ax1 = plt.subplots()
31 ax1.bar(Y1-0.2, X1, width=0.2, color="#FFB6C1", align='center')
32 ax1.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
33 ax1.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
34
35 ax1.set_title('Extent of Glycaemic Control over the Years in Males', fontproperties=font)
36 ax1.set_ylabel('HbA1c Values', fontproperties=font)
37 ax1.set_xlabel('Years', fontproperties=font)
38
39 for label in ax1.xaxis.get_ticklabels() + ax1.yaxis.get_ticklabels():
40     label.set_fontproperties(font)
41
42 ax1.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], ['serif', 'Times New Roman', 'civa', '12'])

```

Python file length: 12,873 lines: 356 Ln:1 Col:106 Pos: 106 Windows (CR LF) UTF-8 INS

65°F Mostly cloudy 11:37 PM 5/8/2023

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

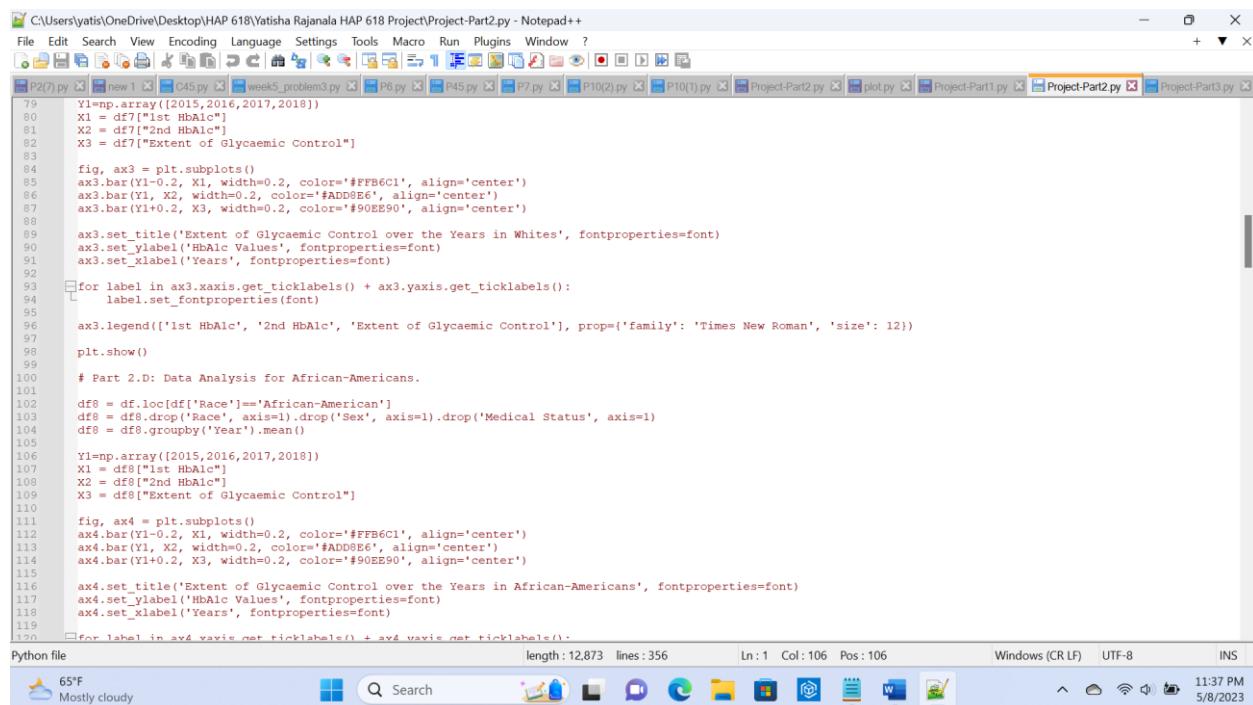


```

41 ax1.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
42 plt.show()
43
44 # Part 2.B: Data Analysis for Females.
45
46 df6 = df.loc[df['Sex']=='Female']
47 df6 = df6.drop('Sex', axis=1).drop('Race', axis=1).drop('Medical Status', axis=1)
48 df6 = df6.groupby('Year').mean()
49
50 Y1=np.array([2015,2016,2017,2018])
51 X1 = df6["1st HbA1c"]
52 X2 = df6["2nd HbA1c"]
53 X3 = df6["Extent of Glycaemic Control"]
54
55 fig, ax2 = plt.subplots()
56 ax2.bar(Y1[-0.2, X1, width=0.2, color="#FFB6C1", align='center')
57 ax2.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
58 ax2.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
59
60 ax2.set_title('Extent of Glycaemic Control over the Years in Females', fontproperties=font)
61 ax2.set_ylabel('HbA1c Values', fontproperties=font)
62 ax2.set_xlabel('Years', fontproperties=font)
63
64 for label in ax2.xaxis.get_ticklabels() + ax2.yaxis.get_ticklabels():
65     label.set_fontproperties(font)
66
67 ax2.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
68 plt.show()
69
70 # Part 2.C: Data Analysis for Whites.
71
72 df7 = df.loc[df['Race']=='White']
73 df7 = df7.drop('Race', axis=1).drop('Sex', axis=1).drop('Medical Status', axis=1)
74 df7 = df7.groupby('Year').mean()
75
76 Y1=np.array([2015,2016,2017,2018])
77 X1 = df7["1st HbA1c"]
78 X2 = df7["2nd HbA1c"]
79 X3 = df7["Extent of Glycaemic Control"]
80
81 fig, ax3 = plt.subplots()
82 ax3.bar(Y1[-0.2, X1, width=0.2, color="#FFB6C1", align='center')
83 ax3.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
84 ax3.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
85
86 ax3.set_title('Extent of Glycaemic Control over the Years in Whites', fontproperties=font)
87 ax3.set_ylabel('HbA1c Values', fontproperties=font)
88 ax3.set_xlabel('Years', fontproperties=font)
89
90 for label in ax3.xaxis.get_ticklabels() + ax3.yaxis.get_ticklabels():
91     label.set_fontproperties(font)
92
93 ax3.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
94 plt.show()
95
96 # Part 2.D: Data Analysis for African-Americans.
97
98 df8 = df.loc[df['Race']=='African-American']
99 df8 = df8.drop('Race', axis=1).drop('Sex', axis=1).drop('Medical Status', axis=1)
100 df8 = df8.groupby('Year').mean()
101
102 Y1=np.array([2015,2016,2017,2018])
103 X1 = df8["1st HbA1c"]
104 X2 = df8["2nd HbA1c"]
105 X3 = df8["Extent of Glycaemic Control"]
106
107 fig, ax4 = plt.subplots()
108 ax4.bar(Y1[-0.2, X1, width=0.2, color="#FFB6C1", align='center')
109 ax4.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
110 ax4.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
111
112 ax4.set_title('Extent of Glycaemic Control over the Years in African-Americans', fontproperties=font)
113 ax4.set_ylabel('HbA1c Values', fontproperties=font)
114 ax4.set_xlabel('Years', fontproperties=font)
115
116 for label in ax4.xaxis.get_ticklabels() + ax4.yaxis.get_ticklabels():
117     label.set_fontproperties(font)
118
119 ax4.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
120 plt.show()

```

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



```

79 Y1=np.array([2015,2016,2017,2018])
80 X1 = df7["1st HbA1c"]
81 X2 = df7["2nd HbA1c"]
82 X3 = df7["Extent of Glycaemic Control"]
83
84 fig, ax3 = plt.subplots()
85 ax3.bar(Y1[-0.2, X1, width=0.2, color="#FFB6C1", align='center')
86 ax3.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
87 ax3.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
88
89 ax3.set_title('Extent of Glycaemic Control over the Years in Whites', fontproperties=font)
90 ax3.set_ylabel('HbA1c Values', fontproperties=font)
91 ax3.set_xlabel('Years', fontproperties=font)
92
93 for label in ax3.xaxis.get_ticklabels() + ax3.yaxis.get_ticklabels():
94     label.set_fontproperties(font)
95
96 ax3.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
97 plt.show()
98
99 # Part 2.D: Data Analysis for African-Americans.
100
101 df8 = df.loc[df['Race']=='African-American']
102 df8 = df8.drop('Race', axis=1).drop('Sex', axis=1).drop('Medical Status', axis=1)
103 df8 = df8.groupby('Year').mean()
104
105 Y1=np.array([2015,2016,2017,2018])
106 X1 = df8["1st HbA1c"]
107 X2 = df8["2nd HbA1c"]
108 X3 = df8["Extent of Glycaemic Control"]
109
110 fig, ax4 = plt.subplots()
111 ax4.bar(Y1[-0.2, X1, width=0.2, color="#FFB6C1", align='center')
112 ax4.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
113 ax4.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
114
115 ax4.set_title('Extent of Glycaemic Control over the Years in African-Americans', fontproperties=font)
116 ax4.set_ylabel('HbA1c Values', fontproperties=font)
117 ax4.set_xlabel('Years', fontproperties=font)
118
119 for label in ax4.xaxis.get_ticklabels() + ax4.yaxis.get_ticklabels():
120     label.set_fontproperties(font)
121
122 ax4.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
123 plt.show()

```

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

```

C:\Users\yatis\OneDrive\Desktop\HAP 618\Yatisha Rajanala HAP 618 Project\Project-Part2.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
P2(7).py new 1 C45.py week5_problem3.py P6.py P45.py P7.py P10(2).py P10(1).py Project-Part2.py plot.py Project-Part1.py Project-Part2.py Project-Part3.py
120     for label in ax4.xaxis.get_ticklabels() + ax4.yaxis.get_ticklabels():
121         label.set_fontproperties(font)
122
123     ax4.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
124
125     plt.show()
126
127 # Part 2.E: Data Analysis for Hispanics.
128
129 df9 = df.loc[df['Race']=='Hispanic']
130 df9 = df9.drop('Race', axis=1).drop('Sex', axis=1).drop('Medical Status', axis=1)
131 df9 = df9.groupby('Year').mean()
132
133 Y1=np.array([2015,2016,2017,2018])
134 X1 = df9["1st HbA1c"]
135 X2 = df9["2nd HbA1c"]
136 X3 = df9["Extent of Glycaemic Control"]
137
138 fig, ax5 = plt.subplots()
139 ax5.bar(Y1-0.2, X1, width=0.2, color="#FFB6C1", align='center')
140 ax5.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
141 ax5.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
142
143 ax5.set_title('Extent of Glycaemic Control over the Years in Hispanics', fontproperties=font)
144 ax5.set_ylabel('HbA1c Values', fontproperties=font)
145 ax5.set_xlabel('Years', fontproperties=font)
146
147 for label in ax5.xaxis.get_ticklabels() + ax5.yaxis.get_ticklabels():
148     label.set_fontproperties(font)
149
150 ax5.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
151
152 plt.show()
153
154 # Part 2.F: Data Analysis for different Federal Poverty Levels.
155
156 def FPL_Group5(Federal_Poverty_Level):
157     if Federal_Poverty_Level <299:
158         return '< 200%'
159     elif Federal_Poverty_Level <399:
160         return '201-300%'
161     else:
162         return '300%'
163
164 df['FPL_Group5'] = df['Federal Poverty Level'].apply(FPL_Group5)
165
166 # Part 2.F.1: Data Analysis for Federal Poverty Level <200%.
167
168 df10 = df.loc[df['FPL_Group5']=='< 200%']
169 df10 = df10.drop('Sex', axis=1).drop('Race', axis=1).drop('Medical Status', axis=1).drop('FPL_Group5', axis=1)
170
171 df10 = df10.groupby(['Year']).mean()
172 df10 = df10.reset_index()
173
174 Y1=np.array([2015,2016,2017,2018])
175
176 X1 = df10["1st HbA1c"]
177 X2 = df10["2nd HbA1c"]
178 X3 = df10["Extent of Glycaemic Control"]
179
180 fig, ax6 = plt.subplots()
181 ax6.bar(Y1-0.2, X1, width=0.2, color="#FFB6C1", align='center')
182 ax6.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
183 ax6.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
184
185 ax6.set_title('Extent of Glycaemic Control over the Years for Federal Poverty Level <200%', fontproperties=font)
186 ax6.set_ylabel('HbA1c Values', fontproperties=font)
187 ax6.set_xlabel('Years', fontproperties=font)
188
189 for label in ax6.xaxis.get_ticklabels() + ax6.yaxis.get_ticklabels():
190     label.set_fontproperties(font)
191
192 ax6.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
193
194 plt.show()
195
196 # Part 2.F.2: Data Analysis for Federal Poverty Level 201-300%

```

Python file length : 12,873 lines : 356 Ln: 1 Col: 106 Pos: 106 Windows (CR LF) UTF-8 INS

65°F Mostly cloudy 11:38 PM 5/8/2023

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

```

C:\Users\yatis\OneDrive\Desktop\HAP 618\Yatisha Rajanala HAP 618 Project\Project-Part2.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
P2(7).py new 1 C45.py week5_problem3.py P6.py P45.py P7.py P10(2).py P10(1).py Project-Part2.py plot.py Project-Part1.py Project-Part2.py Project-Part3.py
155
156 def FPL_Group5(Federal_Poverty_Level):
157     if Federal_Poverty_Level <299:
158         return '< 200%'
159     elif Federal_Poverty_Level <399:
160         return '201-300%'
161     else:
162         return '300%'
163
164 df['FPL_Group5'] = df['Federal Poverty Level'].apply(FPL_Group5)
165
166 # Part 2.F.1: Data Analysis for Federal Poverty Level <200%.
167
168 df10 = df.loc[df['FPL_Group5']=='< 200%']
169 df10 = df10.drop('Sex', axis=1).drop('Race', axis=1).drop('Medical Status', axis=1).drop('FPL_Group5', axis=1)
170
171 df10 = df10.groupby(['Year']).mean()
172 df10 = df10.reset_index()
173
174 Y1=np.array([2015,2016,2017,2018])
175
176 X1 = df10["1st HbA1c"]
177 X2 = df10["2nd HbA1c"]
178 X3 = df10["Extent of Glycaemic Control"]
179
180 fig, ax6 = plt.subplots()
181 ax6.bar(Y1-0.2, X1, width=0.2, color="#FFB6C1", align='center')
182 ax6.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
183 ax6.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
184
185 ax6.set_title('Extent of Glycaemic Control over the Years for Federal Poverty Level <200%', fontproperties=font)
186 ax6.set_ylabel('HbA1c Values', fontproperties=font)
187 ax6.set_xlabel('Years', fontproperties=font)
188
189 for label in ax6.xaxis.get_ticklabels() + ax6.yaxis.get_ticklabels():
190     label.set_fontproperties(font)
191
192 ax6.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
193
194 plt.show()
195
196 # Part 2.F.2: Data Analysis for Federal Poverty Level 201-300%

```

Python file length : 12,873 lines : 356 Ln: 1 Col: 106 Pos: 106 Windows (CR LF) UTF-8 INS

65°F Mostly cloudy 11:38 PM 5/8/2023

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

```

C:\Users\yatis\OneDrive\Desktop\HAP 618\Yatisha Rajanala HAP 618 Project\Project-Part2.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
P2(7) py new 1 C45 py week5_problem3.py P6.py P45.py P7.py P10(2).py P10(1).py Project-Part2.py plot.py Project-Part1.py Project-Part2.py Project-Part3.py
196 # Part 2.F.2: Data Analysis for Federal Poverty Level 201-300%.
197 df11 = df.loc[df['FPL_Group5']=='201-300%']
198 df11 = df11.drop('Sex', axis=1).drop('Race', axis=1).drop('Medical Status', axis=1).drop('FPL_Group5', axis=1)
199
200 df11 = df11.groupby(['Year']).mean()
201 df11 = df11.reset_index()
202
203 Y1=np.array([2015,2016,2017,2018])
204
205 X1 = df11["1st HbA1c"]
206 X2 = df11["2nd HbA1c"]
207 X3 = df11["Extent of Glycaemic Control"]
208
209 fig, ax7 = plt.subplots()
210 ax7.bar(Y1-0.2, X1, width=0.2, color="#FFB6C1", align='center')
211 ax7.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
212 ax7.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
213
214 ax7.set_title('Extent of Glycaemic Control over the Years for Federal Poverty Level 201-300%', fontproperties=font)
215 ax7.set_xlabel('Years', fontproperties=font)
216 ax7.set_ylabel('HbA1c Values', fontproperties=font)
217
218 for label in ax7.xaxis.get_ticklabels() + ax7.yaxis.get_ticklabels():
219     label.set_fontproperties(font)
220
221 ax7.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
222
223 plt.show()
224
225 # Part 2.F.2: Data Analysis for Federal Poverty Level >300%.
226
227 df12 = df.loc[df['FPL_Group5']==">> 300%"]
228 df12 = df12.drop('Sex', axis=1).drop('Race', axis=1).drop('Medical Status', axis=1).drop('FPL_Group5', axis=1)
229
230 df12 = df12.groupby(['Year']).mean()
231 df12 = df12.reset_index()
232
233 Y1=np.array([2015,2016,2017,2018])
234
235 X1 = df12["1st HbA1c"]
236 X2 = df12["2nd HbA1c"]
237

```

Python file length : 12,873 lines : 356 Ln : 1 Col : 106 Pos : 106 Windows (CR LF) UTF-8 INS

65°F Mostly cloudy 11:38 PM 5/8/2023

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

```

C:\Users\yatis\OneDrive\Desktop\HAP 618\Yatisha Rajanala HAP 618 Project\Project-Part2.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
P2(7) py new 1 C45 py week5_problem3.py P6.py P45.py P7.py P10(2).py P10(1).py Project-Part2.py plot.py Project-Part1.py Project-Part2.py Project-Part3.py
236 X1 = df12["1st HbA1c"]
237 X2 = df12["2nd HbA1c"]
238 X3 = df12["Extent of Glycaemic Control"]
239
240 fig, ax8 = plt.subplots()
241 ax8.bar(Y1-0.2, X1, width=0.2, color="#FFB6C1", align='center')
242 ax8.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
243 ax8.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
244
245 ax8.set_title('Extent of Glycaemic Control over the Years for Federal Poverty Level > 300%', fontproperties=font)
246 ax8.set_xlabel('HbA1c Values', fontproperties=font)
247 ax8.set_ylabel('Years', fontproperties=font)
248
249 for label in ax8.xaxis.get_ticklabels() + ax8.yaxis.get_ticklabels():
250     label.set_fontproperties(font)
251
252 ax8.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
253
254 plt.show()
255
256 # Part 2.G: Data Analysis for different Age Groups.
257
258 def Age_Group5(Age):
259     if Age < 40:
260         return '20-40 Years'
261     elif Age < 60:
262         return '41-60 Years'
263     else:
264         return '> 60 Years'
265
266 df['Age_Group5'] = df['Age'].apply(Age_Group5)
267
268 # Part 2.G.1: Data Analysis for Age 20-40 Years.
269
270 df13 = df.loc[df['Age_Group5']=='20-40 Years']
271 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)
272
273 df13 = df13.groupby(['Year']).mean()
274 df13 = df13.reset_index()
275
276 Y1=np.array([2015,2016,2017,2018])
277

```

Python file length : 12,873 lines : 356 Ln : 1 Col : 106 Pos : 106 Windows (CR LF) UTF-8 INS

65°F Mostly cloudy 11:38 PM 5/8/2023

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

```

C:\Users\yatis\OneDrive\Desktop\HAP 618\Yatisha Rajanala HAP 618 Project\Project-Part2.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
P2(7).py new 1 C45.py week5_problem3.py P6.py P45.py P7.py P10(2).py P10(1).py Project-Part2.py plot.py Project-Part1.py Project-Part2.py Project-Part3.py
276 Y1=np.array([2015,2016,2017,2018])
277 X1 = df13["1st HbA1c"]
278 X2 = df13["2nd HbA1c"]
279 X3 = df13["Extent of Glycaemic Control"]
280
281 fig, ax9 = plt.subplots()
282 ax9.bar(Y1-0.2, X1, width=0.2, color="#FFB6C1", align='center')
283 ax9.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
284 ax9.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
285
286 ax9.set_title('Extent of Glycaemic Control over the Years for Age 20-40 Years', fontproperties=font)
287 ax9.set_ylabel('HbA1c Values', fontproperties=font)
288 ax9.set_xlabel('Years', fontproperties=font)
289
290 for label in ax9.xaxis.get_ticklabels() + ax9.yaxis.get_ticklabels():
291     label.set_fontproperties(font)
292
293 ax9.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
294
295 plt.show()
296
297 # Part 2.G.1: Data Analysis for Age 41-60 Years.
298
299 df14 = df.loc[df['Age_Group5']=='41-60 Years']
300 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)
301
302 df14 = df14.groupby(['Year']).mean()
303 df14 = df14.reset_index()
304
305 Y1=np.array([2015,2016,2017,2018])
306
307 X1 = df14["1st HbA1c"]
308 X2 = df14["2nd HbA1c"]
309 X3 = df14["Extent of Glycaemic Control"]
310
311 fig, ax10 = plt.subplots()
312 ax10.bar(Y1-0.2, X1, width=0.2, color="#FFB6C1", align='center')
313 ax10.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
314 ax10.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
315
316 ax10.set_title('Extent of Glycaemic Control over the Years for Age 41-60 Years', fontproperties=font)

```

Python file length : 12,873 lines : 356 Ln : 1 Col : 106 Pos : 106 Windows (CR LF) UTF-8 INS

65°F Mostly cloudy 11:39 PM 5/8/2023

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

```

C:\Users\yatis\OneDrive\Desktop\HAP 618\Yatisha Rajanala HAP 618 Project\Project-Part2.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
P2(7).py new 1 C45.py week5_problem3.py P6.py P45.py P7.py P10(2).py P10(1).py Project-Part2.py plot.py Project-Part1.py Project-Part2.py Project-Part3.py
316
317 ax10.set_title('Extent of Glycaemic Control over the Years for Age 41-60 Years', fontproperties=font)
318 ax10.set_ylabel('HbA1c Values', fontproperties=font)
319 ax10.set_xlabel('Years', fontproperties=font)
320
321 for label in ax10.xaxis.get_ticklabels() + ax10.yaxis.get_ticklabels():
322     label.set_fontproperties(font)
323
324 ax10.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
325
326 plt.show()
327
328 # Part 2.G.1: Data Analysis for Age >60 Years.
329
330 df15 = df.loc[df['Age_Group5']=='> 60 Years']
331 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)
332
333 df15 = df15.groupby(['Year']).mean()
334 df15 = df15.reset_index()
335
336 Y1=np.array([2015,2016,2017,2018])
337
338 X1 = df15["1st HbA1c"]
339 X2 = df15["2nd HbA1c"]
340 X3 = df15["Extent of Glycaemic Control"]
341
342 fig, ax11 = plt.subplots()
343 ax11.bar(Y1-0.2, X1, width=0.2, color="#FFB6C1", align='center')
344 ax11.bar(Y1, X2, width=0.2, color="#ADD8E6", align='center')
345 ax11.bar(Y1+0.2, X3, width=0.2, color="#90EE90", align='center')
346
347 ax11.set_title('Extent of Glycaemic Control over the Years for Age >60 Years', fontproperties=font)
348 ax11.set_ylabel('HbA1c Values', fontproperties=font)
349 ax11.set_xlabel('Years', fontproperties=font)
350
351 for label in ax11.xaxis.get_ticklabels() + ax11.yaxis.get_ticklabels():
352     label.set_fontproperties(font)
353
354 ax11.legend(['1st HbA1c', '2nd HbA1c', 'Extent of Glycaemic Control'], prop={'family': 'Times New Roman', 'size': 12})
355
356 plt.show()

```

Python file length : 12,873 lines : 356 Ln : 1 Col : 106 Pos : 106 Windows (CR LF) UTF-8 INS

65°F Mostly cloudy 11:39 PM 5/8/2023

## 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. For the Bar Graphs based on Sex and Race:
  - 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. For the Bar Graphs based on Federal Poverty Level and Age:
  - 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 1<sup>st</sup> HbA1c, 2<sup>nd</sup> 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.

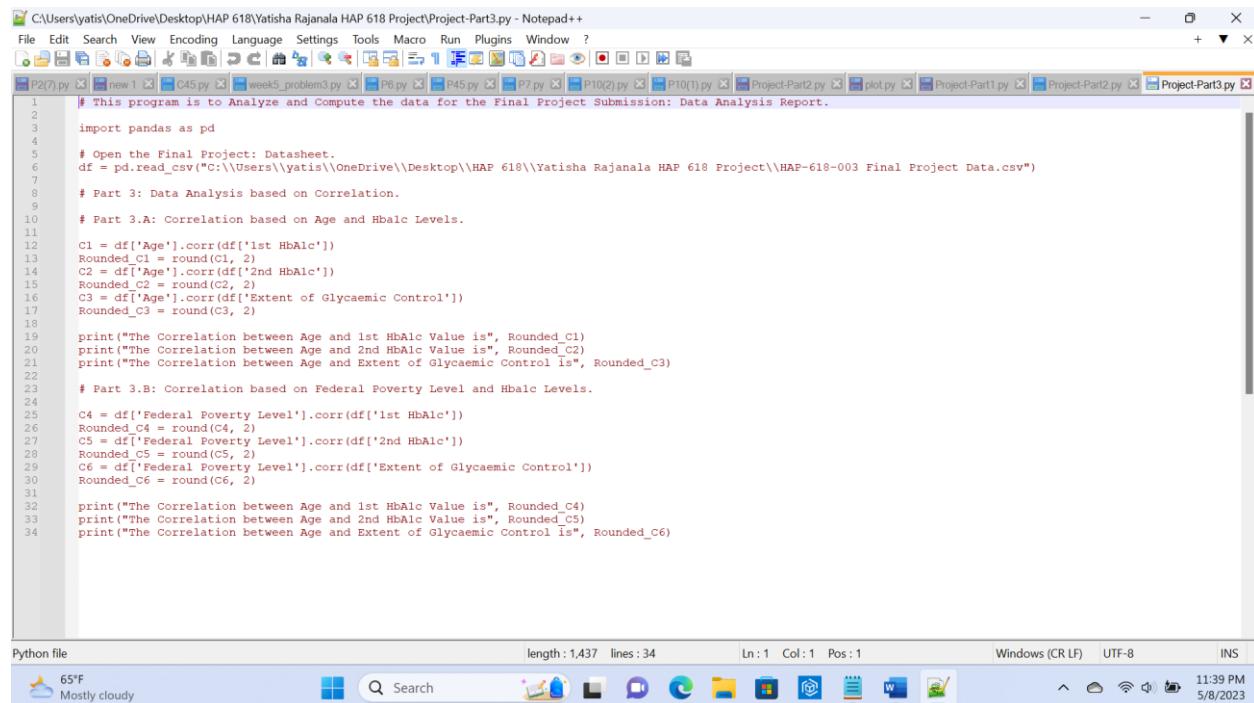
### Part 3: Data Analysis based on Correlation

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 1<sup>st</sup> HbA1c values, 2<sup>nd</sup> 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



```

C:\Users\yatis\OneDrive\Desktop\HAP 618\Yatisha Rajanala HAP 618 Project\Project-Part3.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
P2(7).py week1.py C45.py week5_problem3.py P6.py P45.py P7.py P10(2).py P10(1).py Project-Part2.py plot.py Project-Part1.py Project-Part1.py Project-Part2.py Project-Part3.py

1 # This program is to Analyze and Compute the data for the Final Project Submission: Data Analysis Report.
2
3 import pandas as pd
4
5 # Open the Final Project: Datasheet.
6 df = pd.read_csv("C:\\\\Users\\\\yatis\\\\OneDrive\\\\Desktop\\\\HAP 618\\\\Yatisha Rajanala HAP 618 Project\\\\HAP-618-003 Final Project Data.csv")
7
8 # Part 3: Data Analysis based on Correlation.
9
10 # Part 3.A: Correlation based on Age and HbA1c Levels.
11
12 C1 = df['Age'].corr(df['1st HbA1c'])
13 Rounded_C1 = round(C1, 2)
14 C2 = df['Age'].corr(df['2nd HbA1c'])
15 Rounded_C2 = round(C2, 2)
16 C3 = df['Age'].corr(df['Extent of Glycaemic Control'])
17 Rounded_C3 = round(C3, 2)
18
19 print("The Correlation between Age and 1st HbA1c Value is", Rounded_C1)
20 print("The Correlation between Age and 2nd HbA1c Value is", Rounded_C2)
21 print("The Correlation between Age and Extent of Glycaemic Control is", Rounded_C3)
22
23 # Part 3.B: Correlation based on Federal Poverty Level and HbA1c Levels.
24
25 C4 = df['Federal Poverty Level'].corr(df['1st HbA1c'])
26 Rounded_C4 = round(C4, 2)
27 C5 = df['Federal Poverty Level'].corr(df['2nd HbA1c'])
28 Rounded_C5 = round(C5, 2)
29 C6 = df['Federal Poverty Level'].corr(df['Extent of Glycaemic Control'])
30 Rounded_C6 = round(C6, 2)
31
32 print("The Correlation between Age and 1st HbA1c Value is", Rounded_C4)
33 print("The Correlation between Age and 2nd HbA1c Value is", Rounded_C5)
34 print("The Correlation between Age and Extent of Glycaemic Control is", Rounded_C6)

Python file length : 1,437 lines : 34 Ln : 1 Col : 1 Pos : 1 Windows (CR LF) UTF-8 INS
65°F Mostly cloudy Search 11:39 PM 5/8/2023

```

### **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 Command 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 1<sup>st</sup> HbA1c values, 2<sup>nd</sup> 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 professionals 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 visualization 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.

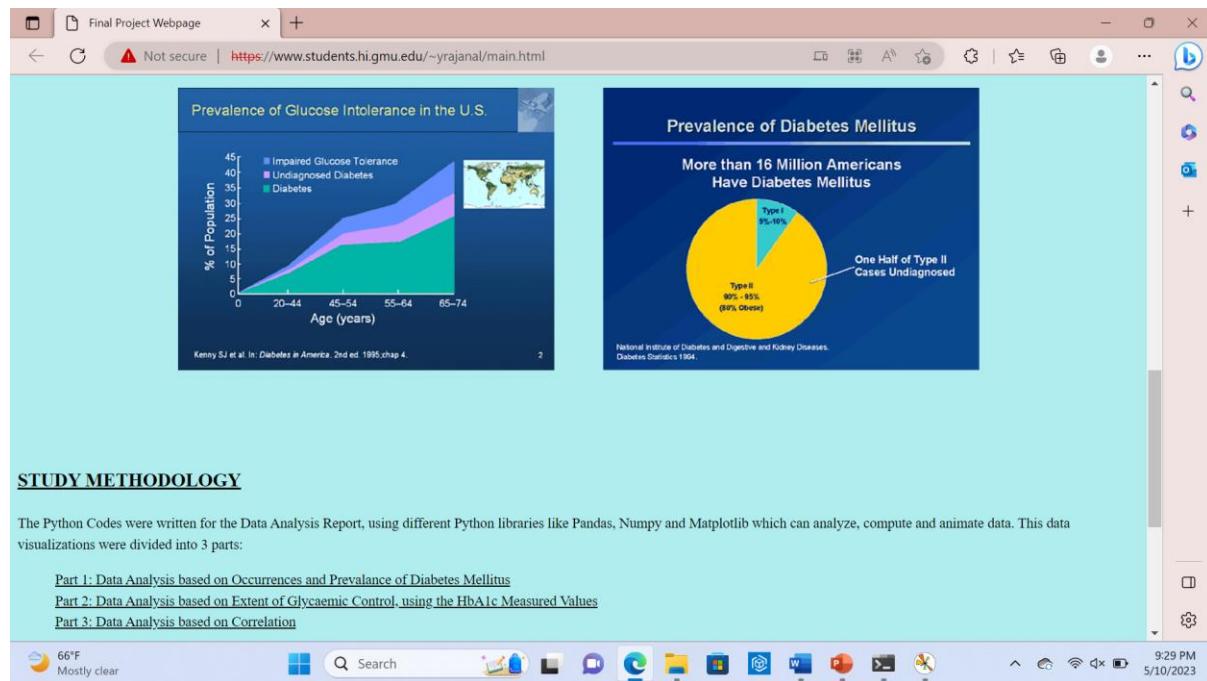
### Image 20: Results: Main Webpage (a)

The screenshot shows a Microsoft Edge browser window with the title bar "Final Project Webpage". The address bar displays the URL <https://www.students.hi.gmu.edu/~yrajanal/main.html>. A warning message "Not secure" is visible. The main content area features a circular graphic with the word "DATA" in the center, surrounded by various medical icons like a brain, heart, stethoscope, and pills. Below this graphic is the section title **DATA ANALYSIS REPORT**. Underneath, there is a subtitle: **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**. At the bottom of the page, the section title **OVERVIEW OF DATA ANALYSIS REPORT** is centered. The Windows taskbar at the bottom shows the date and time as 5/10/2023 and 9:28 PM.

### Image 21: Results: Main Webpage (b)

The screenshot shows the same Microsoft Edge browser window as in Image 20. The main content area now displays the **OVERVIEW OF DATA ANALYSIS REPORT** section. Below it, the **BACKGROUND** section is visible. The text in the background section discusses Diabetes Mellitus (DM) as an irreversible chronic metabolic disorder, its prevalence, complications, and economic impact. It also mentions the high prevalence of DM in the United States and the need for effective prevention and management strategies. The Windows taskbar at the bottom shows the date and time as 5/10/2023 and 9:28 PM.

## Image 22: Results: Main Webpage (c)

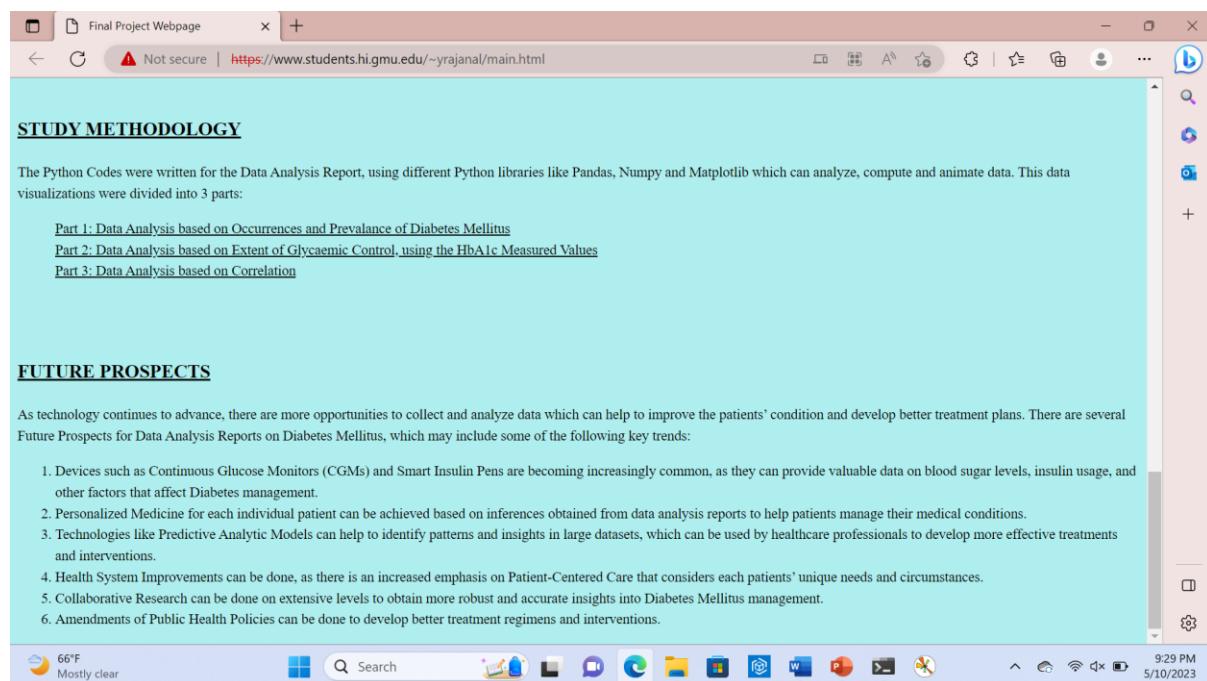


### STUDY METHODOLOGY

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:

- [Part 1: Data Analysis based on Occurrences and Prevalance of Diabetes Mellitus](#)
- [Part 2: Data Analysis based on Extent of Glycaemic Control, using the HbA1c Measured Values](#)
- [Part 3: Data Analysis based on Correlation](#)

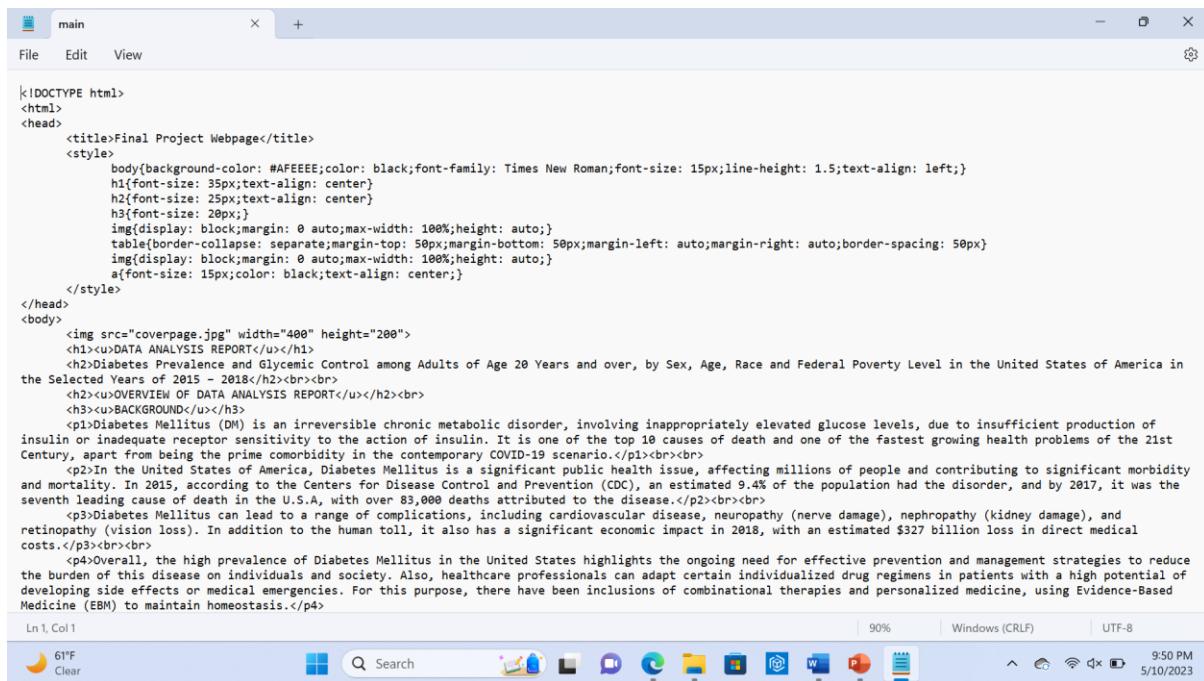
## Image 23: Results: Main Webpage (d)



### FUTURE PROSPECTS

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:

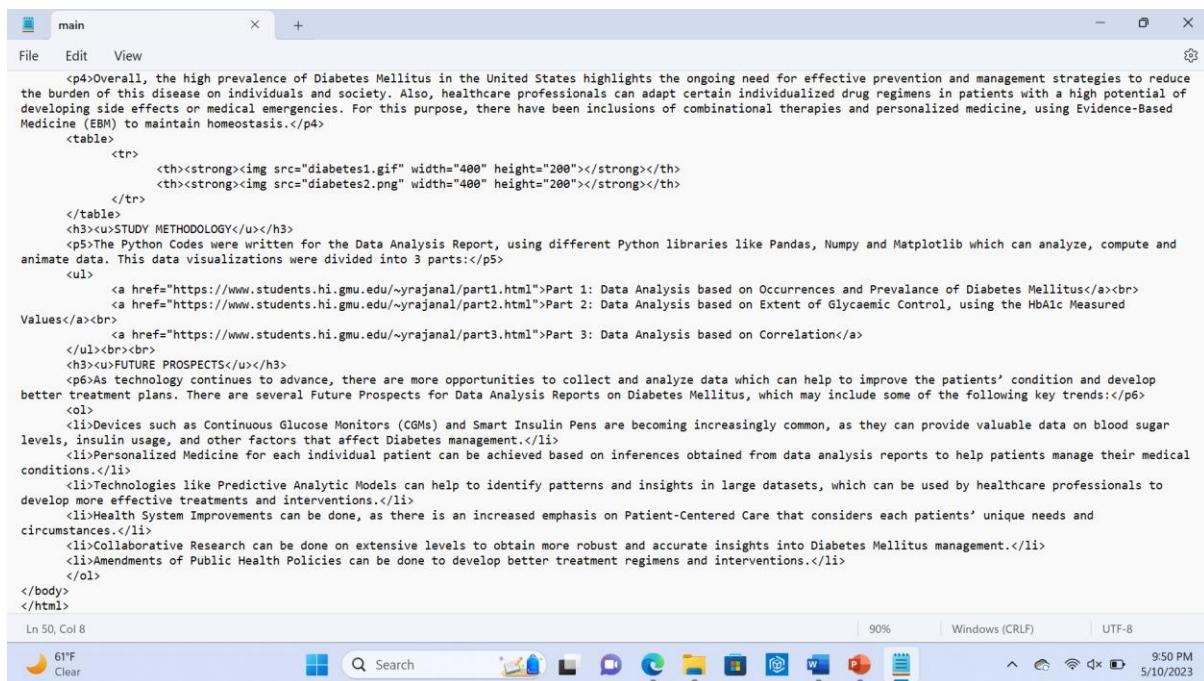
1. Devices such as Continuous Glucose Monitors (CGMs) and Smart Insulin Pens are becoming increasingly common, as they can provide valuable data on blood sugar levels, insulin usage, and other factors that affect Diabetes management.
2. Personalized Medicine for each individual patient can be achieved 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 in large datasets, which can be used by healthcare professionals to develop more effective treatments and interventions.
4. Health System Improvements can be done, 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.

**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>
    
    <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>
</body>

```

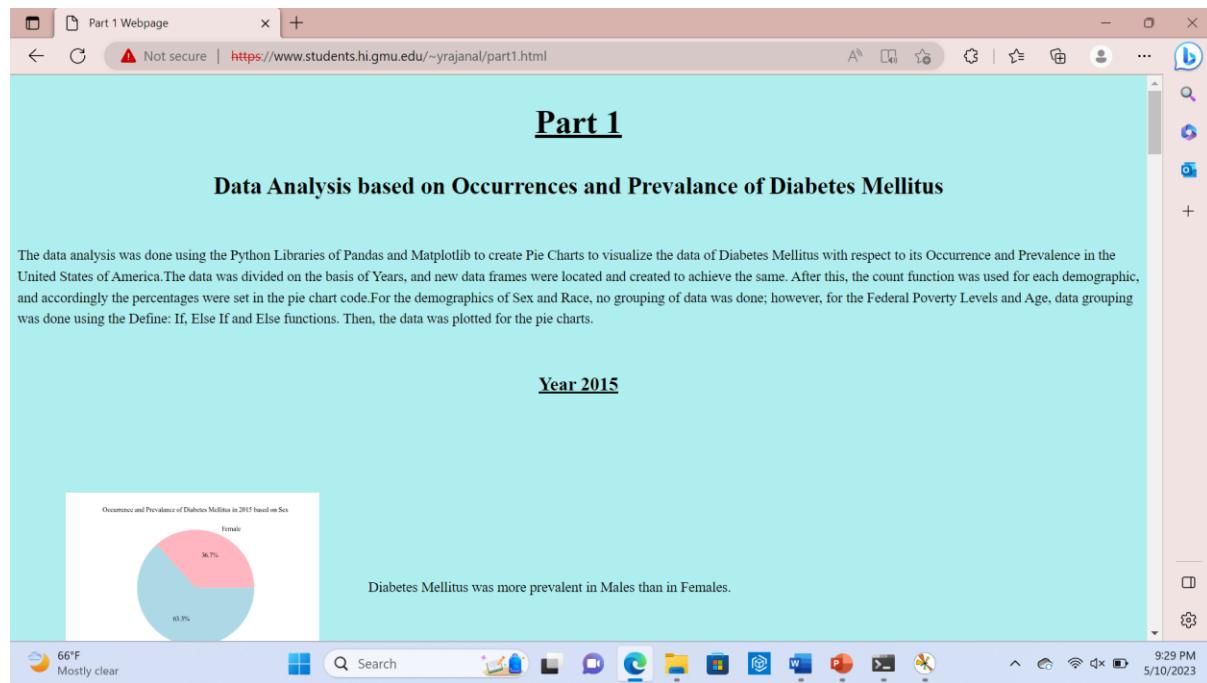
**Image 25: Code for Main Webpage (b)**


```

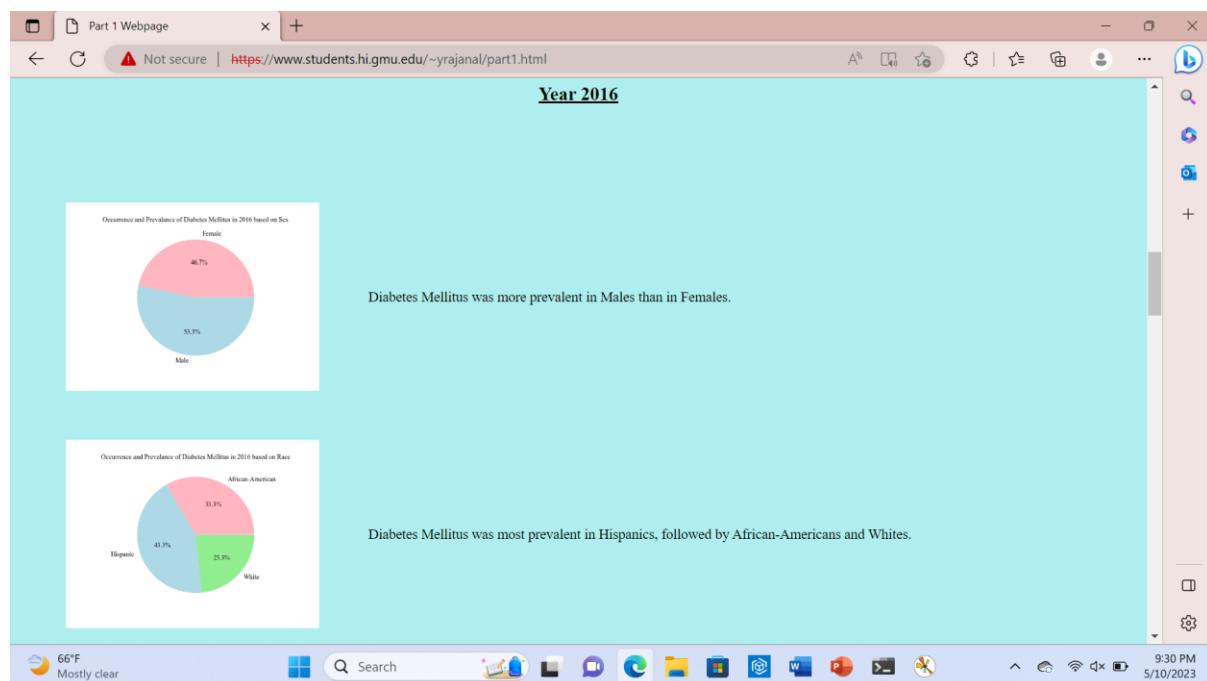
<p4>Overall, the high prevalence of Diabetes Mellitus in the United States highlights the ongoing need for effective prevention and management strategies to reduce the burden of this disease on individuals and society. Also, healthcare professionals can adapt certain individualized drug regimens in patients 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></strong></th>
        <th><strong></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>
<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>

```

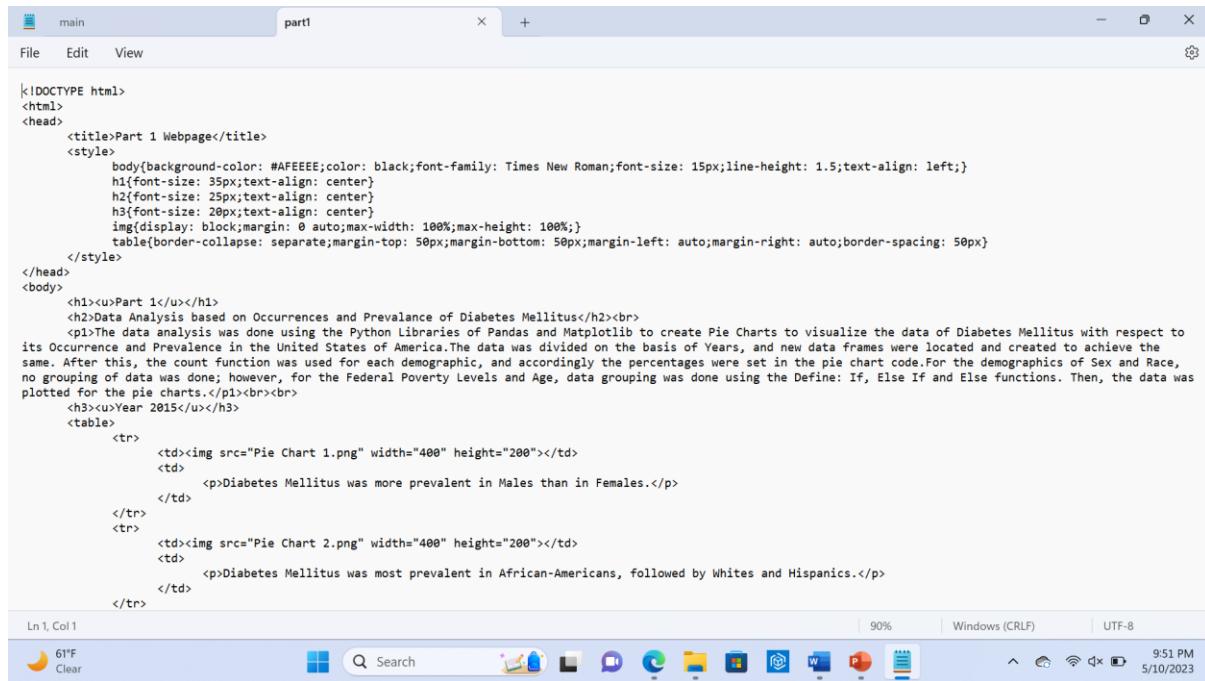
### Image 26: Results: Part 1 Webpage (a)



### Image 27: Results: Part 1 Webpage (b)



**Image 28: Code for Part 1 Webpage (a)**

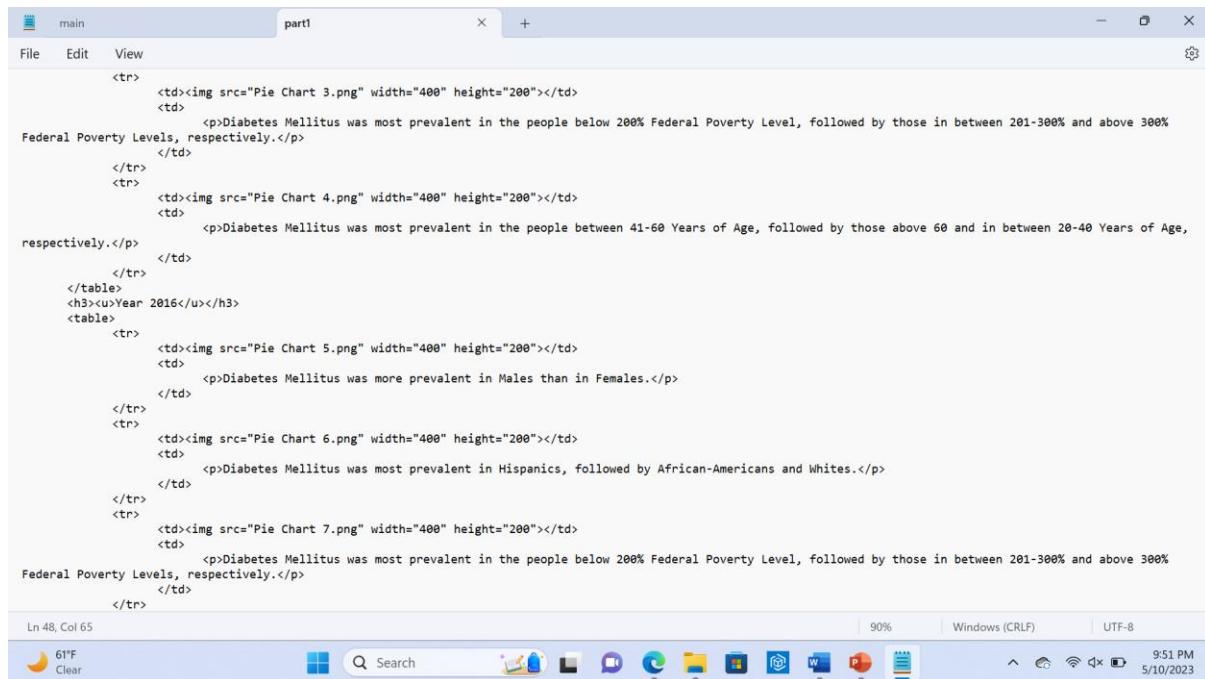


```

<!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>
    <p>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.</p><br>
    <h3><u>Year 2015</u></h3>
    <table>
        <tr>
            <td></td>
            <td>
                <p>Diabetes Mellitus was more prevalent in Males than in Females.</p>
            </td>
        </tr>
        <tr>
            <td></td>
            <td>
                <p>Diabetes Mellitus was most prevalent in African-Americans, followed by Whites and Hispanics.</p>
            </td>
        </tr>
    </table>
    Ln 1, Col 1

```

**Image 29: Code for Part 1 Webpage (b)**



```

        <td></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></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></td>
        <td>
            <p>Diabetes Mellitus was more prevalent in Males than in Females.</p>
        </td>
    </tr>
    <tr>
        <td></td>
        <td>
            <p>Diabetes Mellitus was most prevalent in Hispanics, followed by African-Americans and Whites.</p>
        </td>
    </tr>
    <tr>
        <td></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>
    Ln 48, Col 65

```

**Image 30: Code for Part 1 Webpage (c)**

```

<tr>
<td></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></td>
<td>
<p>Diabetes Mellitus was more prevalent in Females than in Males.</p>
</td>
</tr>
<tr>
<td></td>
<td>
<p>Diabetes Mellitus was most prevalent in African-Americans, followed by Whites and Hispanics.</p>
</td>
</tr>
<tr>
<td></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></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>
</tr>

```

Ln 48, Col 65      90%      Windows (CRLF)      UTF-8

61°F Clear      9:52 PM      5/10/2023

**Image 31: Code for Part 1 Webpage (d)**

```

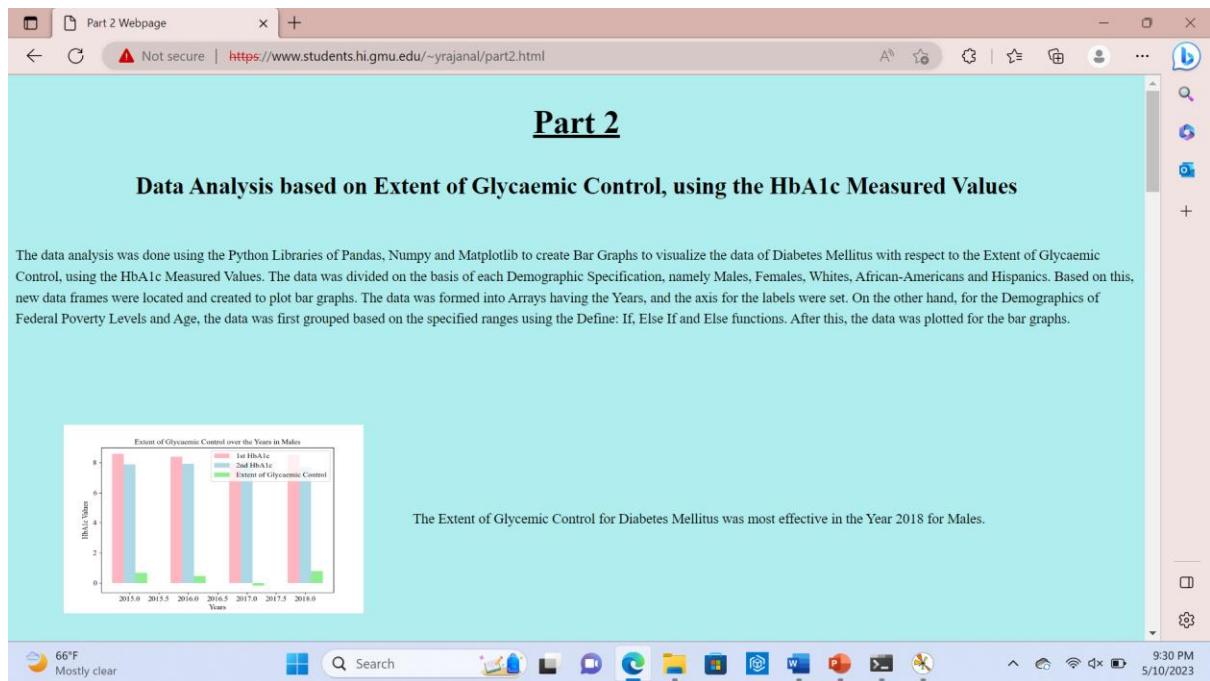
<tr>
<td></td>
<td>
<p>Diabetes Mellitus was more prevalent in Females than in Males.</p>
</td>
</tr>
<tr>
<td></td>
<td>
<p>Diabetes Mellitus was most prevalent in African-Americans, followed by Whites and Hispanics.</p>
</td>
</tr>
<tr>
<td></td>
<td>
<p>Diabetes Mellitus was most prevalent in the 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></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>

```

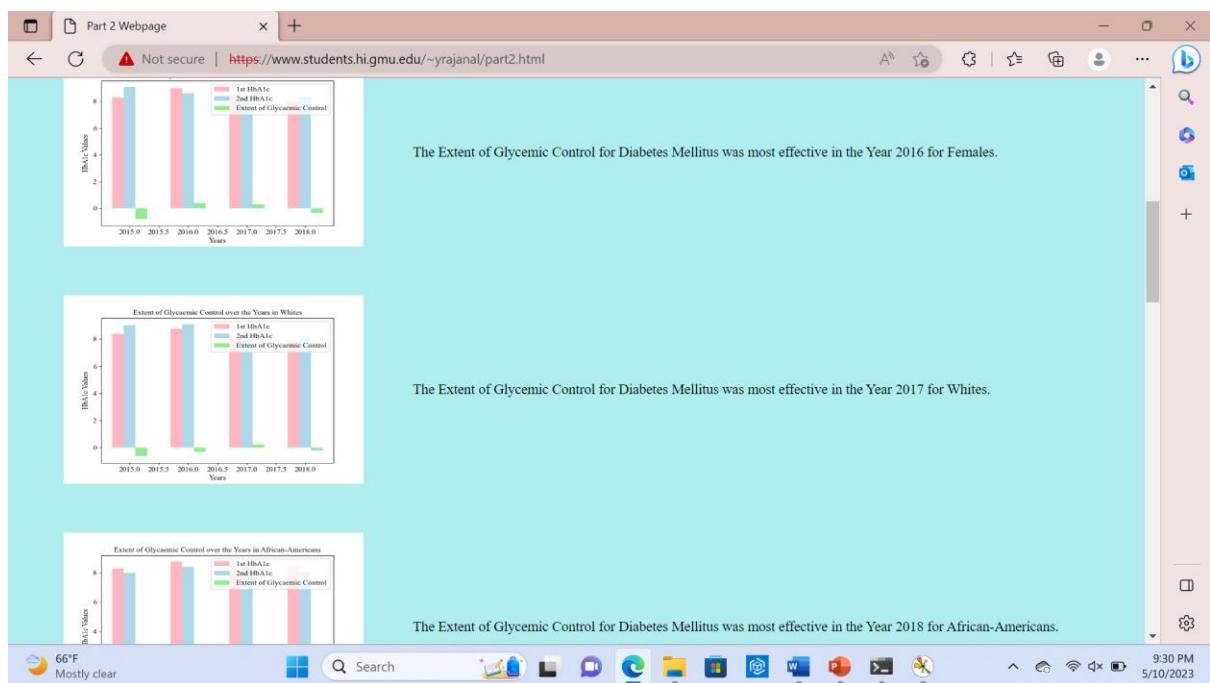
Ln 48, Col 65      90%      Windows (CRLF)      UTF-8

61°F Clear      9:52 PM      5/10/2023

### 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>
    <p>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.</p>
    </p1><br>
    <table>
        <tr>
            <td></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></td>
            <td>
                <p>The Extent of Glycemic Control for Diabetes Mellitus was most effective in the Year 2016 for Females.</p>
            </td>
        </tr>
        <tr>
    </table>

```

Ln 1, Col 1 | 90% | Windows (CRLF) | UTF-8  
61°F Clear | Search | Icons | 9:52 PM | 5/10/2023

### Image 35: Code for Part 2 Webpage (b)

```

        <td></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></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></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></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></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></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>

```

Ln 1, Col 1 | 90% | Windows (CRLF) | UTF-8  
61°F Clear | Search | Icons | 9:53 PM | 5/10/2023

**Image 36: Code for Part 2 Webpage (c)**

```

<tr>
<td></td>
</tr>
<tr>
<td></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></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></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></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></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>

```

Ln 1, Col 1      90%      Windows (CRLF)      UTF-8

61°F Clear      9:53 PM      5/10/2023

**Image 37: Results: Part 3 Webpage (a)**

## Part 3

### Data Analysis based on Correlation

Code For Correlation

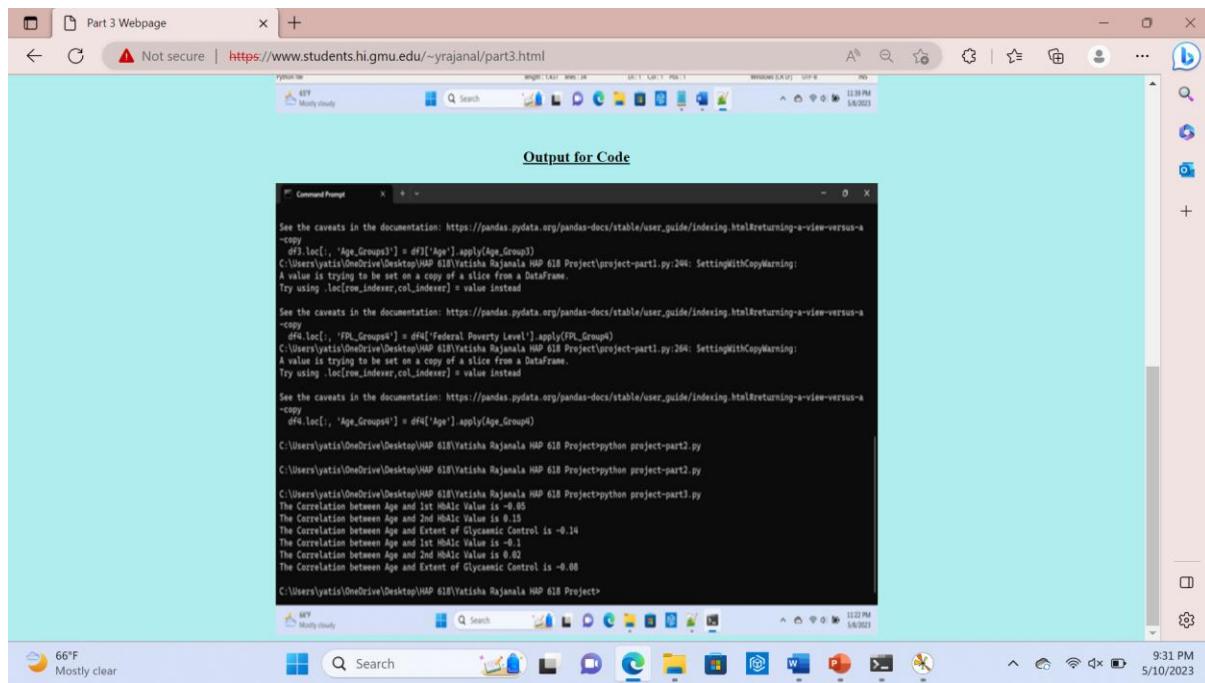
```

# This program is to Analyze and Compute the data for the Final Project Submission: Data Analysis Input.
import pandas as pd
# Open the Final Project Database.
df = pd.read_csv(r'C:\Users\yrajanal\OneDrive\Desktop\HAP-618\Yatisha Rajanala HAP-618\Project\Part 3 - Notepad\Part 3 Final Project Data.csv')
# Part 3: Data Analysis based on Correlation.
# Part 3.1: Correlation based on Age and Blood Sugar.
CG = df['Age'].corr(df['1st MBG'])
Rounded_CG = round(CG, 2)
print("The Correlation between Age and 1st MBG Value is", Rounded_CG)
print("The Correlation between Age and 2nd MBG Value is", Rounded_CG)
print("The Correlation between Age and Extent of Glycemic Control is", Rounded_CG)
# Part 3.2: Correlation based on Federal Poverty Level, and Blood Sugar.
CG = df['Federal Poverty Level'].corr(df['1st MBG'])
Rounded_CG = round(CG, 2)
print("The Correlation between Age and 1st MBG Value is", Rounded_CG)
print("The Correlation between Age and 2nd MBG Value is", Rounded_CG)
print("The Correlation between Age and Extent of Glycemic Control is", Rounded_CG)
# Part 3.3: Correlation based on Age, and Blood Sugar.
CG = df['Age'].corr(df['Extent of Glycemic Control'])
Rounded_CG = round(CG, 2)
print("The Correlation between Age and 1st MBG Value is", Rounded_CG)
print("The Correlation between Age and 2nd MBG Value is", Rounded_CG)
print("The Correlation between Age and Extent of Glycemic Control is", Rounded_CG)

```

66°F Mostly clear      9:31 PM      5/10/2023

### Image 38: Results: Part 2 Webpage (b)



### Image 39: Code for Part 3 Webpage

```

<!DOCTYPE html>
<html>
<head>
<title>Part 3 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 3</u></h1>
<h2>Data Analysis based on Correlation</h2><br>
<h3><u>Code For Correlation</u></h3>
<br>
<h3><u>Output for Code</u></h3>

</body>
</html>

```

## 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 Command Prompt.
2. Importation 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). <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>