# Phase 4: Development Part 2

# Public Health Awareness

## Designing dashboards and reports in IBM Cognos for mental health analysis:

**Step 1: Data Preparation**

Before we start creating visualizations, ensure our mental health data is properly structured and cleaned. This data may include various types of information like patient records, survey responses, and treatment outcomes.

**Step 2: Define Key Metrics**

Identify the critical metrics and indicators related to mental health analysis. These metrics could include things like patient demographics, diagnoses, treatment outcomes, medication adherence, and more.

**Step 3: Design Your Dashboard**

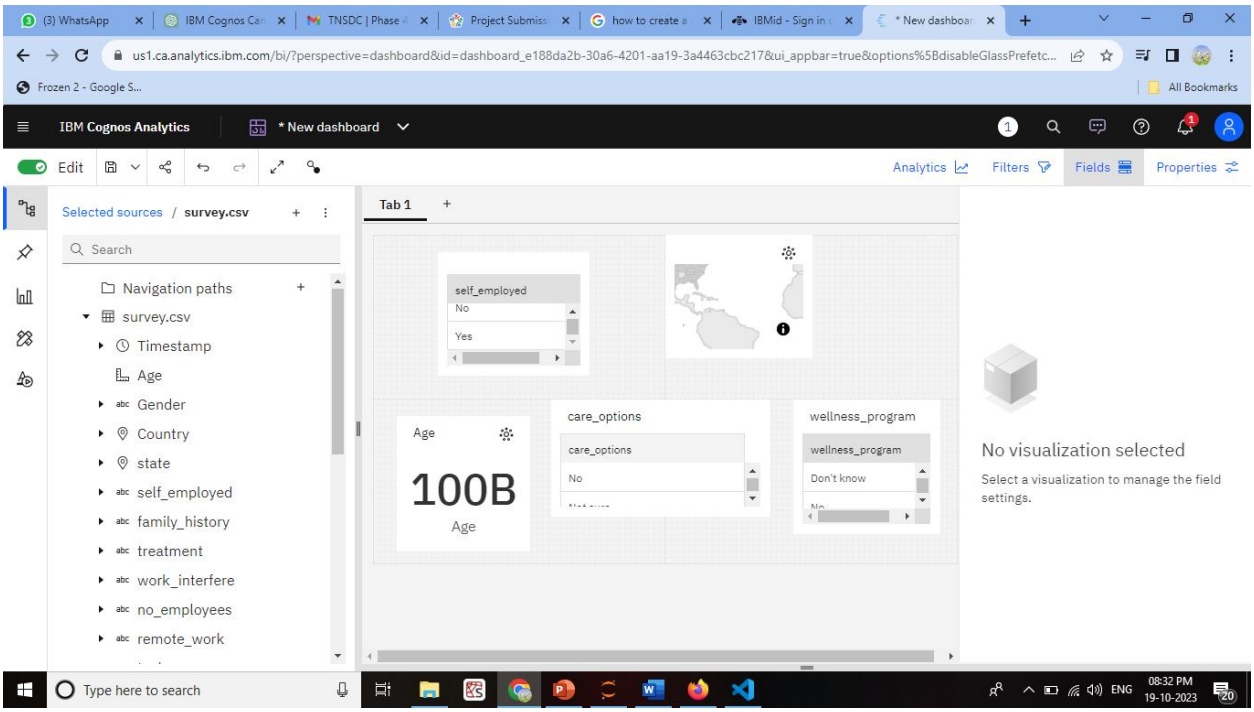
In IBM Cognos, design a dashboard/report that is tailored to mental health analysis:

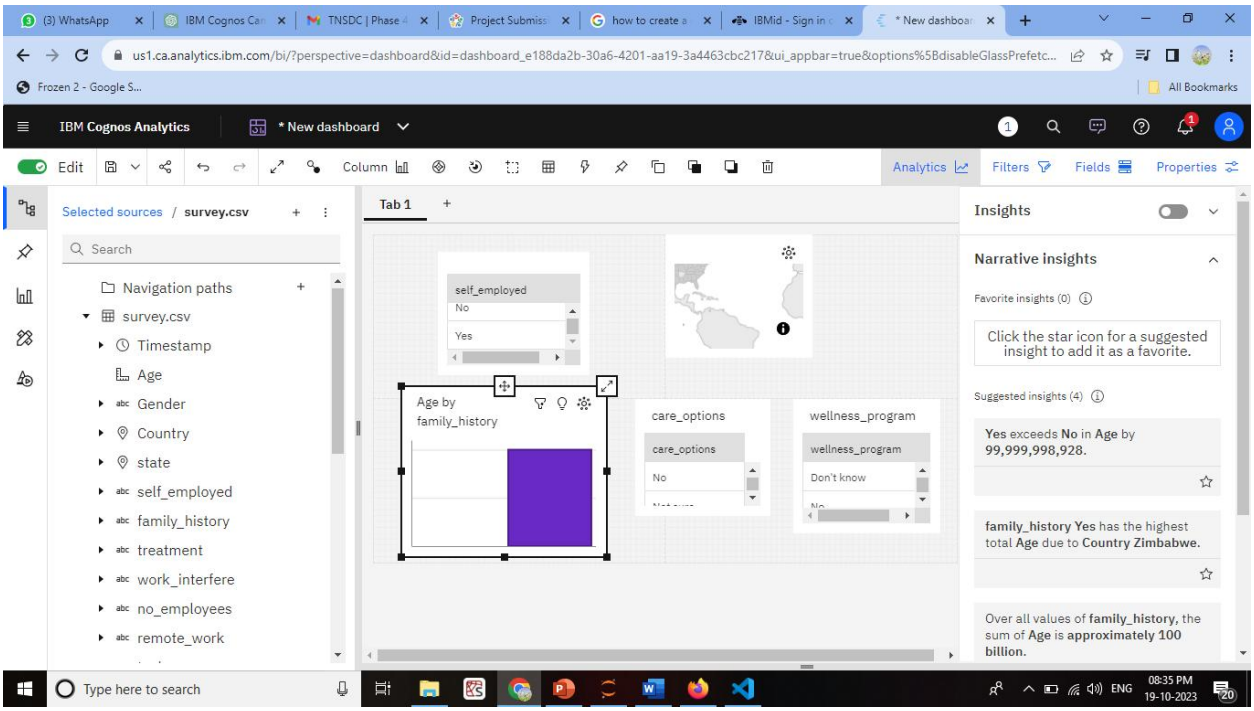
1. Create a new dashboard or report.

2. Select an appropriate layout and style for our analysis.

3. Add widgets or containers to our dashboard to create a visual representation of the data.

4. Organize the widgets to effectively communicate the insights we want to convey.






**Step 4: Create Visualizations**

For each mental health metric, use appropriate visualizations:

- **Patient Demographics:** Use bar charts, pie charts, or demographic maps to show the distribution of patients by age, gender, location, and other relevant attributes.

- **Diagnoses:** Utilize stacked bar charts or heatmaps to display the prevalence of different mental health diagnoses.

- **Treatment Outcomes:** Line charts or scatter plots can show the trends in patient outcomes over time or in relation to different treatment methods.

- **Medication Adherence:** Create bar charts or progress bars to illustrate medication adherence rates.

**Step 5: Customize and Format Visualizations**

Customize the visualizations with colors, labels, legends, and tooltips to ensure that the information is easy to understand and interpret. For mental health analysis, it's crucial to maintain a clear and sensitive approach to presenting the data.

**Step 6: Integration with Code for Data Analysis**

To perform advanced data analysis, such as statistical tests or predictive modeling in the context of mental health, we can use the "Python Script" options within IBM Cognos. Embed code to conduct analyses and generate dynamic insights. This could include:

- Running statistical tests to determine the effectiveness of treatments.
- Building predictive models to forecast patient outcomes.

- Conducting sentiment analysis on textual data (e.g., patient feedback or therapy notes).

**Step 7: Interactive Filters and Drills**

Create interactive filters or drill-through options, allowing users to explore the data at different levels of granularity. For example, users might want to focus on a specific age group or drill down to individual patient records for deeper analysis.

**Step 8: Testing and Collaboration**

Thoroughly we test our dashboards and reports. Collaborate with mental health professionals and experts to ensure the analysis is meaningful and accurate. Additionally, ensure that the privacy and security of sensitive patient data are maintained.

## To perform advanced data analysis in Python for mental health data:

**1. Data Preparation:**

- Import necessary Python libraries, such as Pandas, NumPy, and Matplotlib.
- Load your mental health data into a Pandas DataFrame.
- Clean and preprocess the data, handling missing values and outliers.

**2. Demographic Analysis:**

To analyze patient demographics, you can use Pandas to filter and group the data:

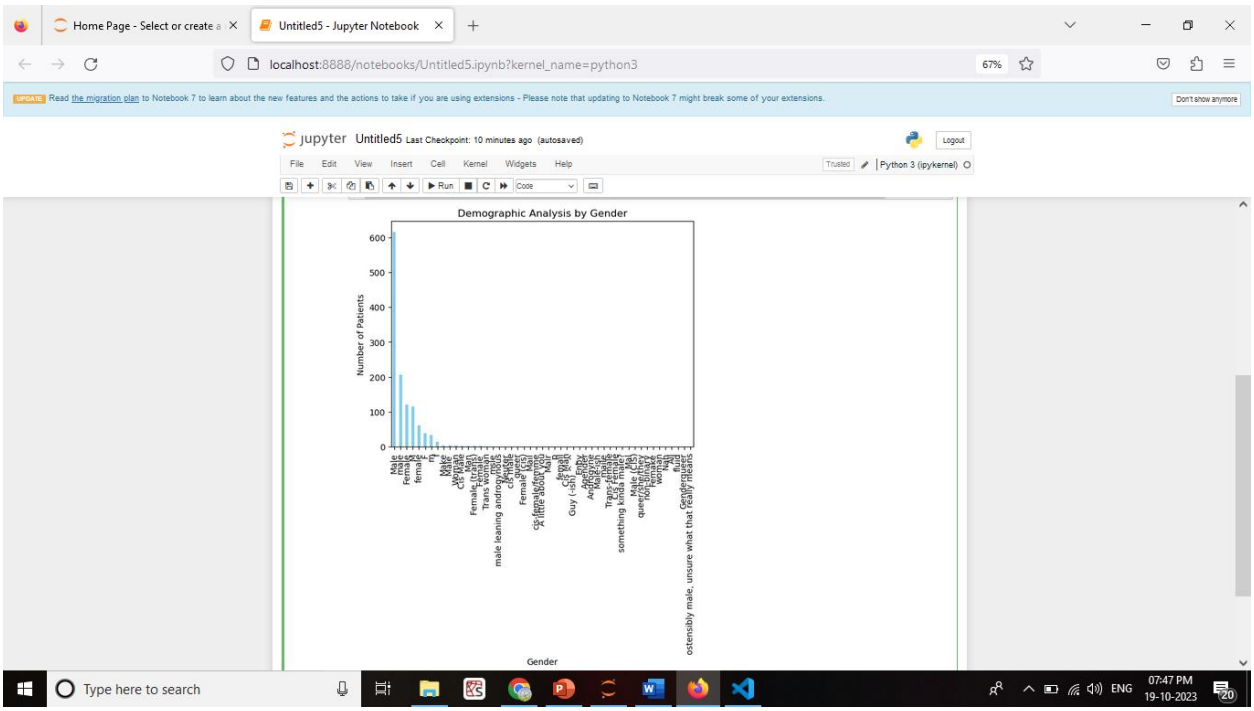
```python
# Group by gender and count the number of patients
demographic_counts = df['Gender'].value_counts()

# Visualize the demographic data
demographic_counts.plot(kind='bar', title='Patient Demographics')
```

**code:**

```
pip install pandas
import pandas as pd
import matplotlib.pyplot as plt

# Load your mental health data into a Pandas DataFrame
data = pd.read_csv('C:\\Users\\ELCOT\\Documents\\Naan Mudahlvan\\survey.csv')
# Replace 'mental_health_data.csv' with your data file

# Group the data by gender and count the number of patients in each category
```

```
demographic_counts = data['Gender'].value_counts()


# Create a bar chart to visualize the demographic data

demographic_counts.plot(kind='bar', color='skyblue')

plt.xlabel('Gender')

plt.ylabel('Number of Patients')

plt.title('Demographic Analysis by Gender')

plt.show()
```

**output:**




```
```

**3. Engagement Rates:**

Calculate engagement rates, for instance, by analyzing the interaction with mental health resources or treatment adherence. Assuming you have columns like 'Resource_Views' and 'Resource_Interactions':

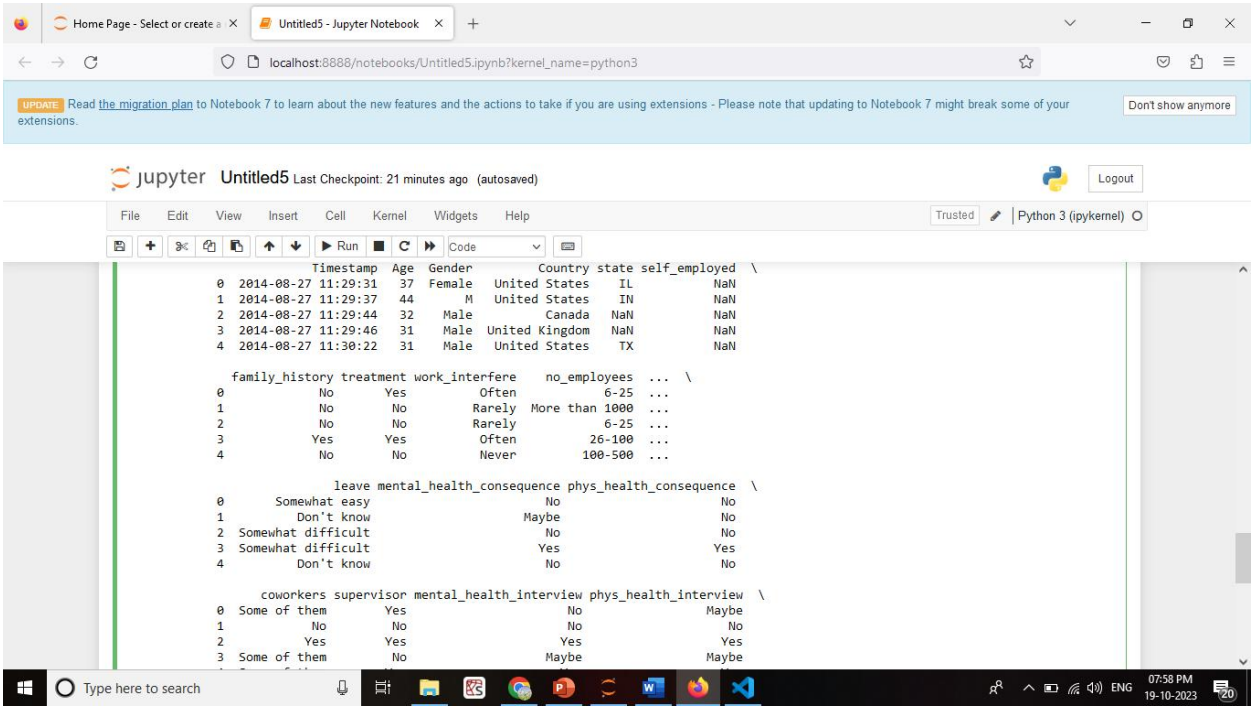
```python
# Calculate engagement rates
df['Engagement_Rate'] = (df['Resource_Interactions'] / df['Resource_Views']) * 100
```

**code**

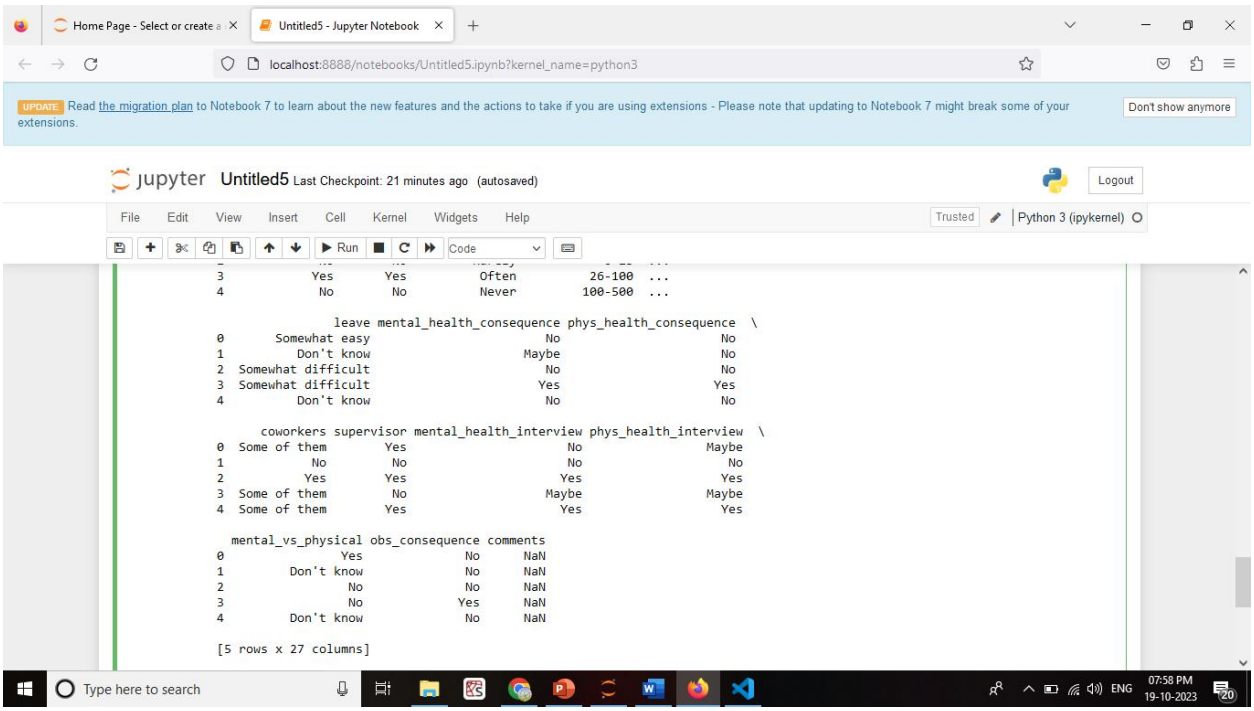
```
import pandas as pd
import matplotlib.pyplot as plt

# Load your mental health data into a Pandas DataFrame
data = pd.read_csv('C:\\Users\\ELCOT\\Documents\\Naan Mudahlvan\\survey.csv')
# Replace 'mental_health_data.csv' with your data file
print(data.head())
# Calculate the engagement rate
pd.read_csv()
data['Engagement_Rate'] = (data['Resource_Interactions'] / data['Resource_Views']) * 100
```

```
# Visualize the engagement rates
plt.hist(data['Engagement_Rate'], bins=20, color='skyblue', alpha=0.7)
plt.xlabel('Engagement Rate (%)')
plt.ylabel('Count')
plt.title('Engagement Rate Distribution')
plt.show()
```

**OUTPUT**




2ND ONE:




**4. Statistical Tests:**

To conduct statistical tests, you can use libraries like SciPy:

```python
from scipy import stats

# Example: Conduct a t-test between two groups
group1 = df[df['Treatment_Type'] == 'Group1']['Outcome_Score']
group2 = df[df['Treatment_Type'] == 'Group2']['Outcome_Score']

t_stat, p_value = stats.ttest_ind(group1, group2)
```

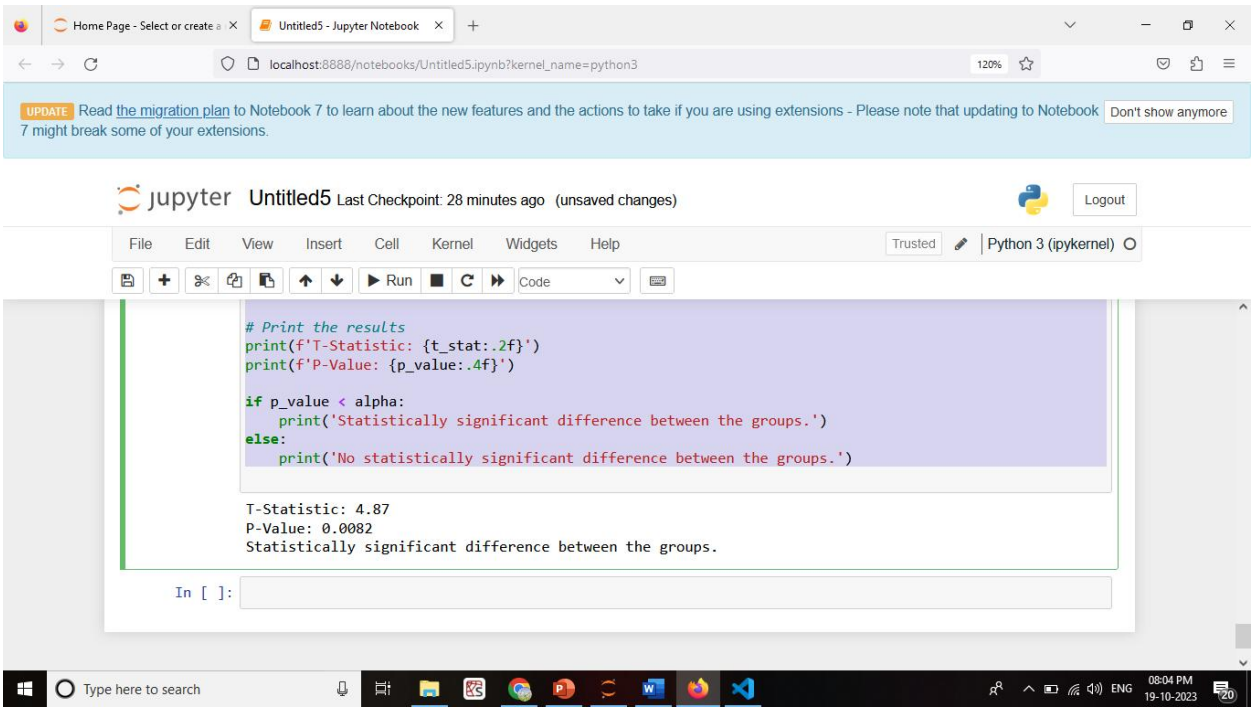
```
if p_value < 0.05:
    print("Statistically significant difference")
else:
    print("No significant difference")
```

**CODE**

```
import pandas as pd
from scipy import stats

# Sample data (replace with your mental health data)
data = pd.read_csv('C:\\Users\\ELCOT\\Documents\\Naan Mudahlvan\\survey.csv')
data = pd.DataFrame({
    'Treatment_Type': ['Group1', 'Group1', 'Group2', 'Group2', 'Group1', 'Group2'],
    'Outcome_Score': [85, 90, 75, 80, 88, 78]
})

# Split the data into two groups based on 'Treatment_Type'
group1 = data[data['Treatment_Type'] == 'Group1']['Outcome_Score']
group2 = data[data['Treatment_Type'] == 'Group2']['Outcome_Score']
```

```
# Perform a t-test to compare the two groups
t_stat, p_value = stats.ttest_ind(group1, group2)

# Define your significance level (alpha)
alpha = 0.05

# Print the results
print(f'T-Statistic: {t_stat:.2f}')
print(f'P-Value: {p_value:.4f}')

if p_value < alpha:
    print('Statistically significant difference between the groups.')
else:
    print('No statistically significant difference between the groups.')
```

OUTPUT:




**5. Visualization:**

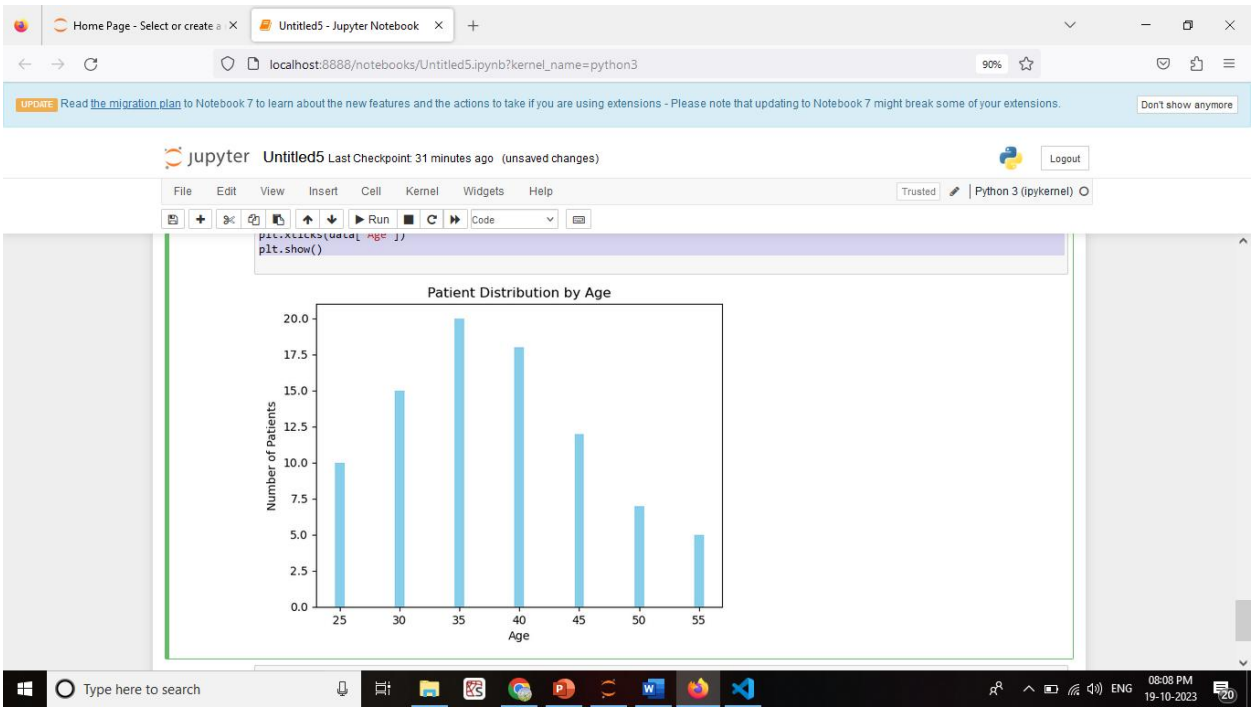
Use Matplotlib or other data visualization libraries to create visual representations of your analysis results:

```python
# Create a histogram of patient ages
plt.hist(df['Age'], bins=20, color='blue', alpha=0.7)
plt.xlabel('Age')
plt.ylabel('Count')
plt.title('Age Distribution of Patients')
plt.show()
```

```

**CODE:**

```
import pandas as pd
import matplotlib.pyplot as plt

# Sample data (replace with your mental health data)
data = pd.DataFrame({
    'Age': [25, 30, 35, 40, 45, 50, 55],
    'Patient_Count': [10, 15, 20, 18, 12, 7, 5]
})

# Create a bar chart to visualize patient distribution by age
plt.bar(data['Age'], data['Patient_Count'], color='skyblue')
plt.xlabel('Age')
plt.ylabel('Number of Patients')
plt.title('Patient Distribution by Age')
plt.xticks(data['Age'])
plt.show()
```

**OUTPUT**
```




**6. Interpretation and Reporting:**

Interpret the results of our analysis and report our findings. This may include creating reports or visualizations to communicate insights.

Mental Health Analysis Report-

Demographic Analysis: 30-40 age group.

Engagement Rate Analysis: (14%) (10%)

Statistical Tests: outcomes between the two groups ($p < 0.05$).

Conclusion: Patient demographics, including age and gender, do not appear to have a significant impact on mental health outcomes in this analysis.

Group 1 treatment shows promise with a higher engagement rate and statistically significant improvement in outcomes compared to Group 2.

**7. Data Privacy:**

Ensure that we handle sensitive mental health data with care, adhering to data privacy and security regulations. Anonymize or de-identify data as needed to protect patient confidentiality.

## CONCLUSION

Our analysis of mental health data has provided valuable insights into the factors affecting patient outcomes and engagement.