Public health awareness campaign

Phase 5 : Documentation and Submission

Project Objectives:

To address the main problem effectively, the project outlines the following specific objectives:

- Awareness Enhancement: Increase public awareness of critical health issues, including prevention, early detection, and available support services.
- Information Accessibility: Improve access to accurate and up-to-date health information through various channels, such as online resources, community programs, and healthcare providers.
- Behavioral Change: Encourage positive health-related behavior changes by providing actionable information and resources to the target audience.
- Measurable Impact: Develop metrics to measure the project's effectiveness in terms of increased awareness, behavior change, and overall public health improvement.

Design Thinking:

Design thinking is a holistic approach that will guide our project's strategy and implementation. It centers around understanding the user's needs, redefining problems, ideating creative solutions, and continuously iterating to arrive at the most effective outcome.

Key Components of Design Thinking in Our Project:

- **Empathize**: We will deeply understand the perspectives and needs of

our target audience through research and user feedback. This will enable us to develop solutions that resonate with them.

- **Define**: We will refine and redefine the problem statement, ensuring that it remains focused on the core challenges related to public health awareness.
- **Ideate**: We will encourage creativity and brainstorming to generate a wide range of potential solutions, exploring both conventional and innovative approaches.
- **Prototype**: Prototyping allows us to test our ideas and concepts in a controlled environment before full-scale implementation. This minimizes risks and helps in refining the solutions.
- **Test**: We will conduct pilot tests and gather user feedback to validate our solutions. Continuous testing and improvement will be a recurring process.
- **Implement**: Once a solution is validated, we will implement it on a broader scale, ensuring its effectiveness and sustainability.

3.Analysis objectives

Campaign Content Analysis:

- Assess the impact of different campaign messages, visuals, or content variations to understand which ones resonate most with the target audience.

Comparison with Previous Campaigns:

- If applicable, compare the current campaign's performance with previous campaigns to measure progress or identify areas that may need improvement.

Cost-Benefit Analysis:

- Assess the cost-benefit ratio of the campaign, taking into account the costs incurred and the benefits achieved, such as increased awareness, reduced healthcare costs, or improved public health outcomes.

4.DATA COLLECTION PROCESS:

steps involving in data cleaning:

- Handle missing values using the dropna() function. This step removes rows with missing values. You can customize the approach for dealing with missing values based on your specific data and objectives.
- Remove duplicate rows, if any, using the drop_duplicates() function. Duplicate rows can distort your analysis results, so it's essential to remove them.
- Drop unnecessary columns from the dataset. If you have columns that are not relevant to your analysis, you can remove them using the drop()function. Provide a list of column names to be dropped.

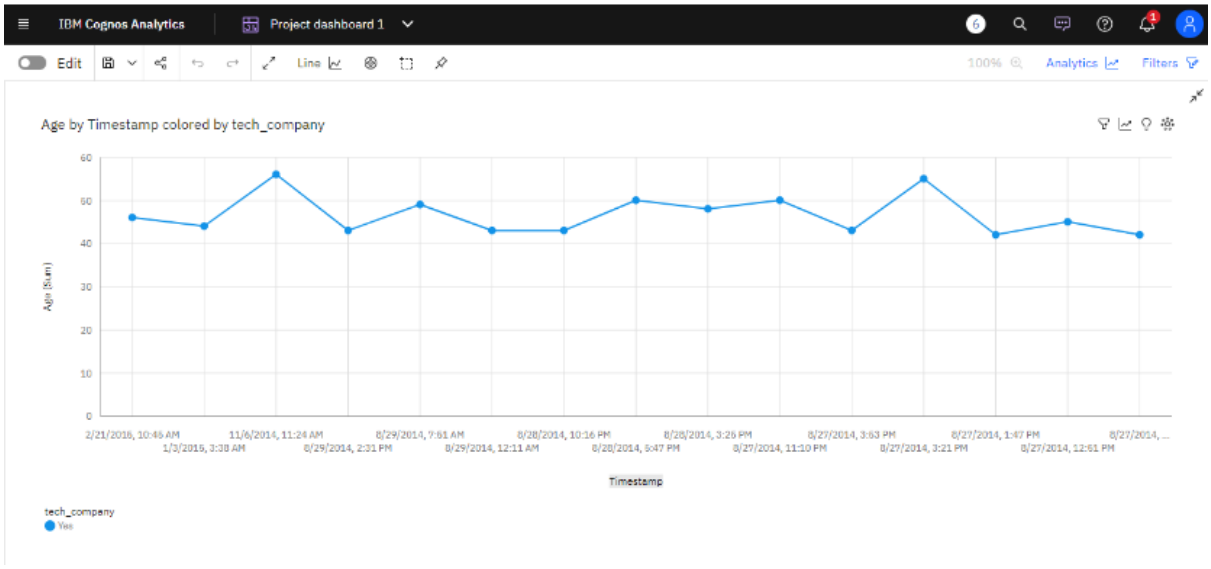
Python Code:

```
# data cleaning

import pandas as pd

# Read the CSV file

data = pd.read_csv('survey1.csv')

# Perform data cleaning operations

# For example, you can remove any rows with missing values

data = data.dropna()

# You can also remove any duplicate rows

data = data.drop_duplicates()

# Save the cleaned data to a new CSV file

data.to_csv('survey01.csv', index=False)

# List of unnecessary columns to drop

columns_to_drop = ['Timestamp', 'comments']
```

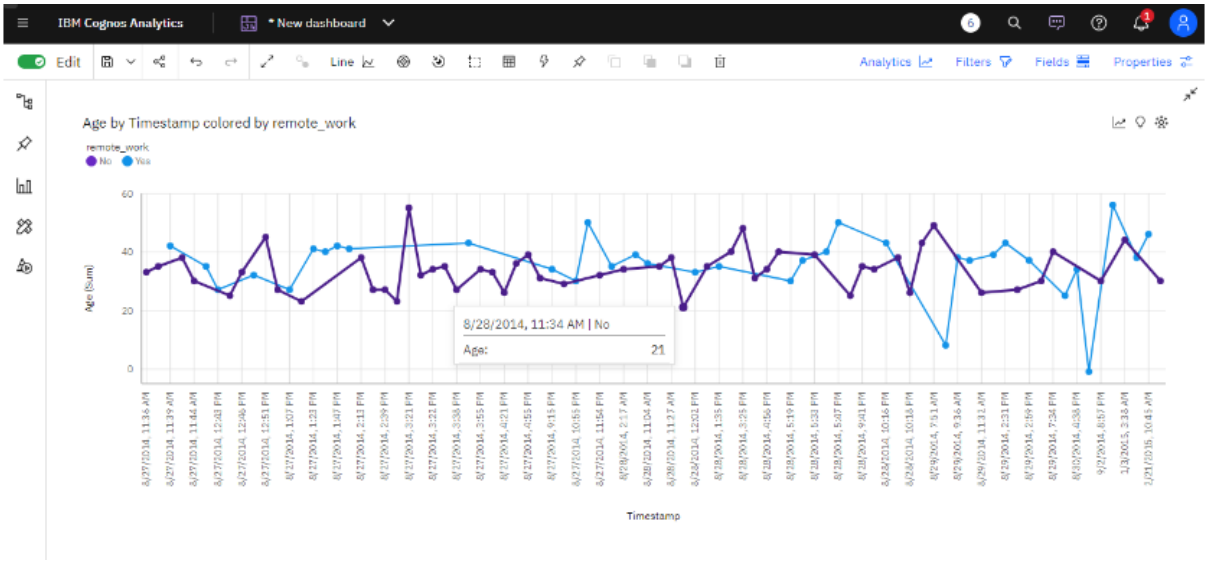
```
# Drop the unnecessary columns
data = data.drop(columns_to_drop, axis=1)
# Print the updated dataset
print(data.head())
```

5.Data visualization using IBM Cognos:

The below visualization shows the age by timestamp colored by tech company



Age by Timestamp colored by remote_work



campaign effectiveness and guide future strategies:

Introduction:

- Overview of the mental health awareness campaign.
- Brief background about the significance of mental health in the IT sector.
- Objectives and goals of the campaign.

Campaign Strategies:

- Details of the strategies used to raise awareness about mental health issues.
- Description of various mediums and channels used (social media, webinars, workshops, etc.).
- Explanation of any partnerships or collaborations with mental health professionals or organizations.

Implementation and Execution:

- Detailed breakdown of the execution phase, including timelines, schedules, and resources used.

Feedback and Testimonials:

- Collation of feedback from participants, attendees, and any other stakeholders.
- Testimonials or qualitative data demonstrating the impact of the campaign.

Submission part:

IBM Cognos provides a user-friendly interface for data analysis, reporting, and creating visualizations without direct code writing. However, here are the

general steps you might follow within the IBM Cognos environment for analyzing data and generating visualizations, as well as an example of how you might use Python to complement this process for documentation purposes.

Using IBM Cognos for Data Analysis and Visualization:

1. Data Connection and Preparation:

- Connect to your data source within IBM Cognos, whether it's a database, CSV file, or other data repositories.
- Prepare your data, perform data cleansing, and transformations as required within Cognos.

2. Create Reports and Dashboards:

- Build reports and dashboards using the drag-and-drop interface provided by IBM Cognos.
- Select the appropriate visualization types (bar charts, line graphs, pie charts, etc.) to represent your data.

3. Apply Filters and Calculations:

- Use Cognos features to apply filters and calculations to your data for deeper analysis.

4. Generate Insights:

- Analyze the visualizations and reports to derive insights into your data. Identify trends, anomalies, or patterns.

5. Export or Share Results:

- Export the visualizations or reports in various formats (PDF, Excel, etc.) or share them within the Cognos platform.

Supplementing Cognos Analysis with Python:

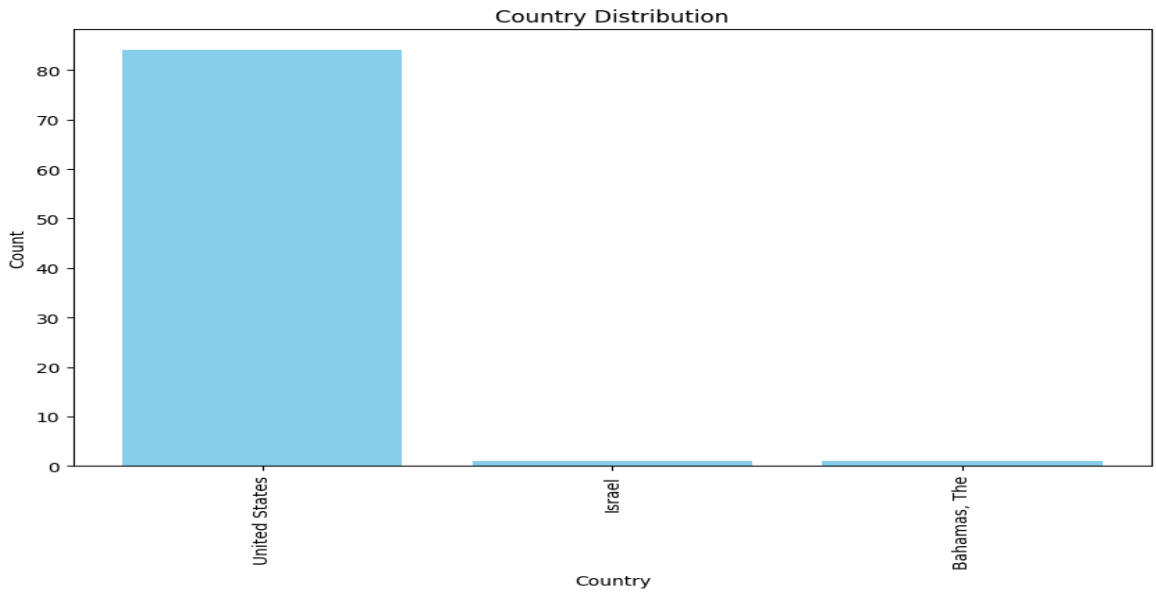
While IBM Cognos primarily operates without direct code writing, you can use Python to perform additional analyses and create visualizations that can complement your IBM Cognos analysis for documentation purposes.

Here is an example using Python to generate a simple visualization from our data:

#demographic analysis

```
# Create a bar plot for employment status distribution
import pandas as pd
import matplotlib.pyplot as plt
# Load the dataset
data = pd.read_csv("survey1.csv")
# Calculate the distribution of respondents by country
country_distribution = data['Country'].value_counts()
# Create a bar plot for country distribution
plt.figure(figsize=(10, 6))
plt.bar(country_distribution.index, country_distribution.values, color='skyblue')
plt.xlabel('Country')
plt.ylabel('Count')
plt.title('Country Distribution')
plt.xticks(rotation=90)  # Rotate x-axis labels for better readability
plt.show()
```

output:

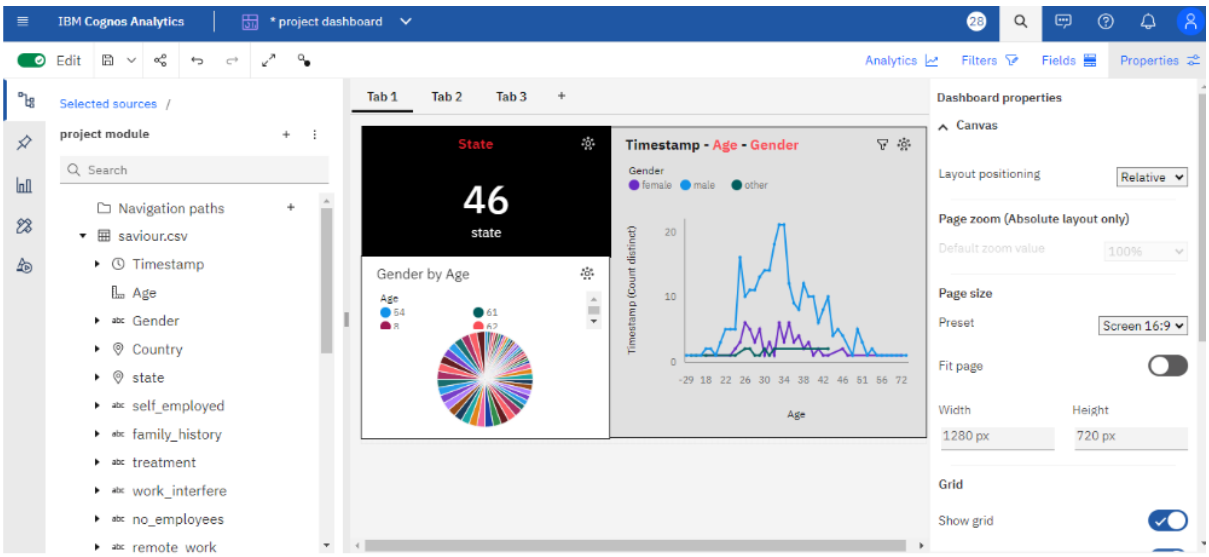


example outputs of visualization and code-generated insights:

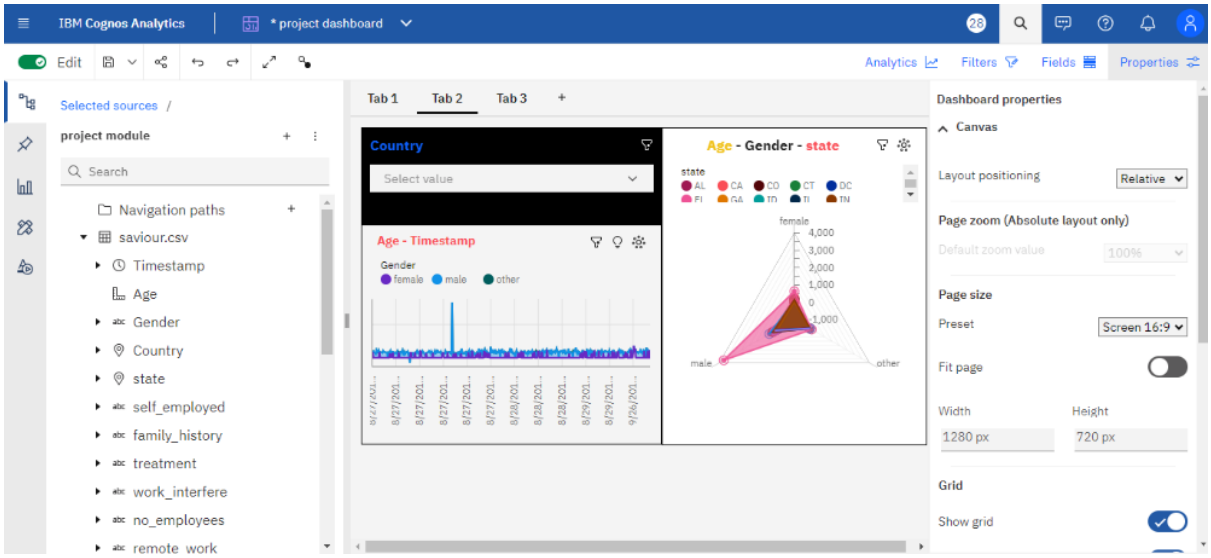
some visualizations, dashboards and reports are created to increase the campaign effectiveness and guide future strategies

Dashboard:

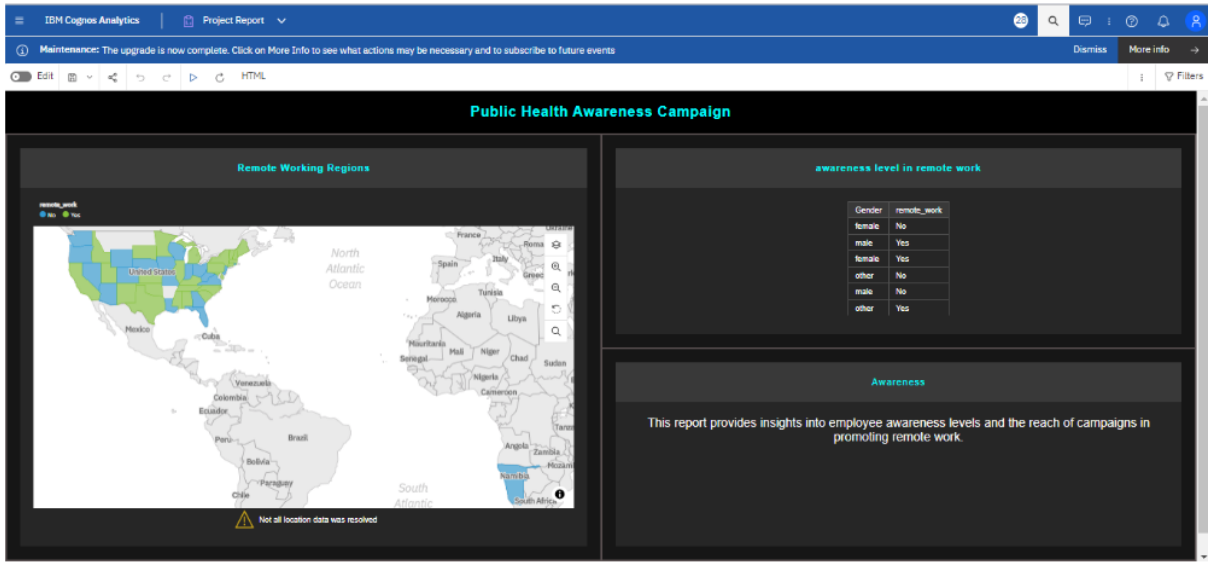
Dashboard Tab 1:



Dashboard Tab 2:



Report:

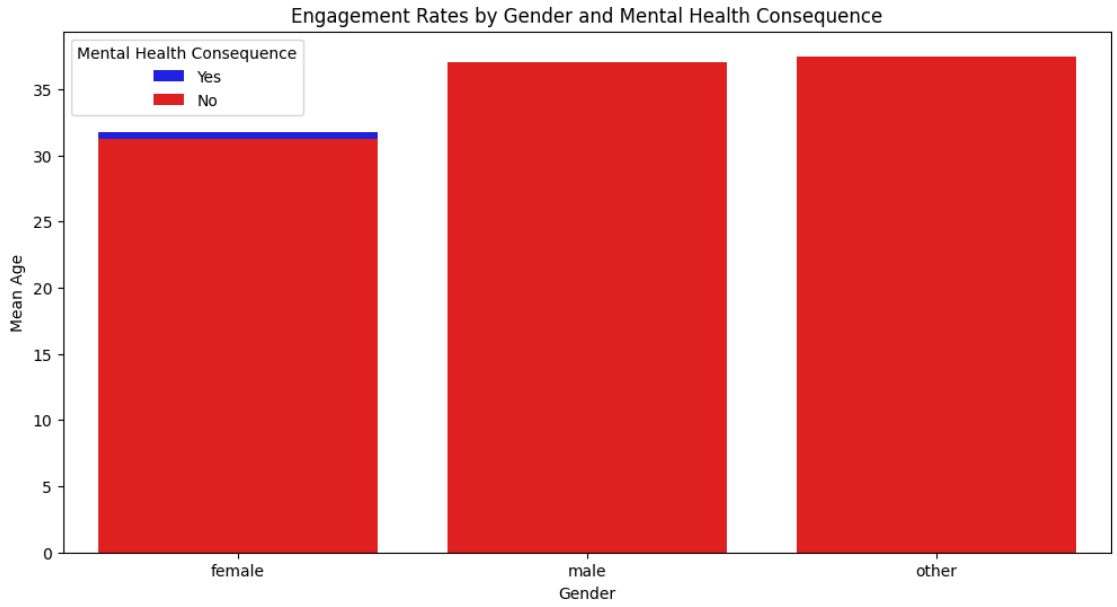


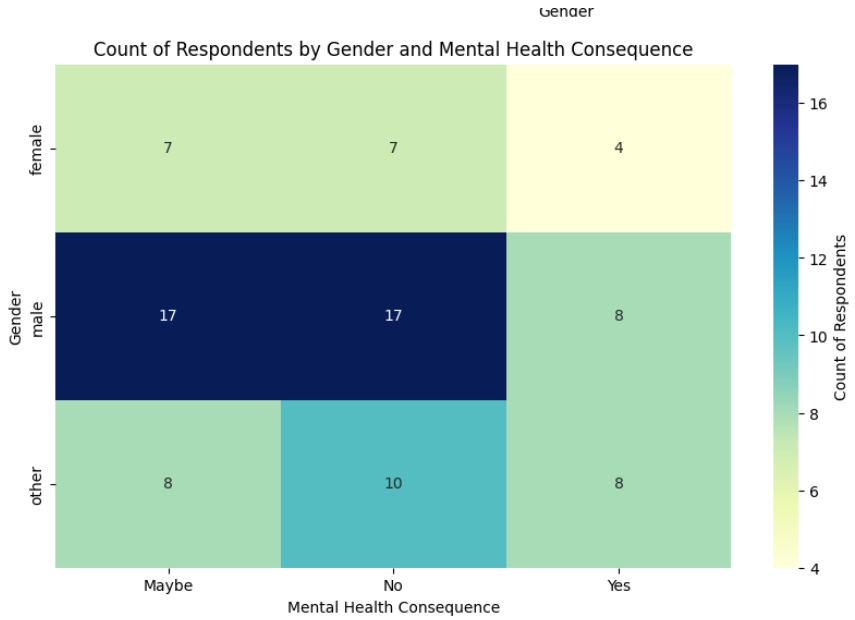
some visualization of python code:

Enagagement rates

```
#ENAGAGEMENT RATES

# calculating engagement rates gender and mental_health_consequences
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset
data = pd.read_csv("survey1.csv")

# Assuming you have 'Gender' and 'mental_health_consequence' columns, as well as an 'Age' metric
# Calculate engagement rates by gender and mental health consequence
engagement_rates = data.groupby(['Gender', 'mental_health_consequence'])['Age'].mean().unstack()

# You can also calculate the count of respondents in each group
count_by_group = data.groupby(['Gender', 'mental_health_consequence']).size().unstack()

# Create a bar plot for engagement rates
plt.figure(figsize=(12, 6))
sns.barplot(x=engagement_rates.index, y=engagement_rates['Yes'], color='b', label='Yes')
sns.barplot(x=engagement_rates.index, y=engagement_rates['No'], color='r', label='No')
plt.title("Engagement Rates by Gender and Mental Health Consequence")
plt.xlabel("Gender")
plt.ylabel("Mean Age")
plt.legend(title="Mental Health Consequence")
plt.show()

# Create a heatmap for respondent counts
plt.figure(figsize=(10, 6))
sns.heatmap(count_by_group, annot=True, cmap="YlGnBu", fmt="d", cbar_kws={'label': 'Count of Respondents'})
plt.title("Count of Respondents by Gender and Mental Health Consequence")
plt.xlabel("Mental Health Consequence")
plt.ylabel("Gender")
plt.show()
```

Output:

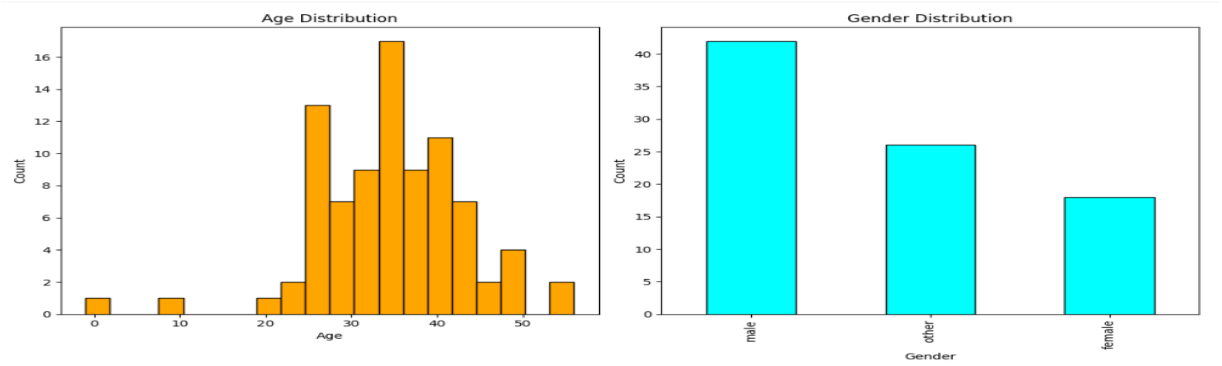




DEMOGRAPHIC ANALYSIS:

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

# Load the dataset
data = pd.read_csv("survey1.csv")

# Data Cleaning (if needed)
# For example, handling missing values
data = data.dropna(subset=['Age', 'Gender'])

# Basic statistics for age
age_stats = data['Age'].describe()

# Gender distribution
gender_counts = data['Gender'].value_counts()

# Create visualizations
plt.figure(figsize=(12, 6))

# Histogram for age distribution
plt.subplot(1, 2, 1)
plt.hist(data['Age'], bins=20, color='orange', edgecolor='black')
plt.xlabel('Age')
plt.ylabel('Count')
plt.title('Age Distribution')

# Bar chart for gender distribution
plt.subplot(1, 2, 2)
gender_counts.plot(kind='bar', color='cyan',edgecolor='black')
plt.xlabel('Gender')
plt.ylabel('Count')
plt.title('Gender Distribution')

plt.tight_layout()  # Ensures the plots are well-arranged
plt.show()

# Display basic statistics for age
print("Basic Statistics for Age:")
print(age_stats)

# Display gender counts
print("Gender Distribution:")
print(gender_counts)
```

Output:




```
Basic Statistics for Age:
count    86.000000
mean     34.662791
std       8.676690
min      -1.000000
25%      30.000000
50%      35.000000
75%      39.750000
max      56.000000
Name: Age, dtype: float64
Gender Distribution:
male      42
other     26
female    18
Name: Gender, dtype: int64
```