```
import pandas as pd
df=pd.read_csv('/content/marriage.csv')
df.head()
```

	Area Name	Total	Educational level	Age at marriage	Number of ever married persons	Unnamed: 5
0	NaN	NaN	NaN	NaN	Males	Females
1	INDIA	Total	Total	Less than 10	4276588	7849859
2	INDIA	Total	Total	10-11	979213	3434492
3	INDIA	Total	Total	12-13	1144889	7717216
4	INDIA	Total	Total	14-15	3900762	28124694

Next steps: Generate code with df View recommended plots New interactive sheet

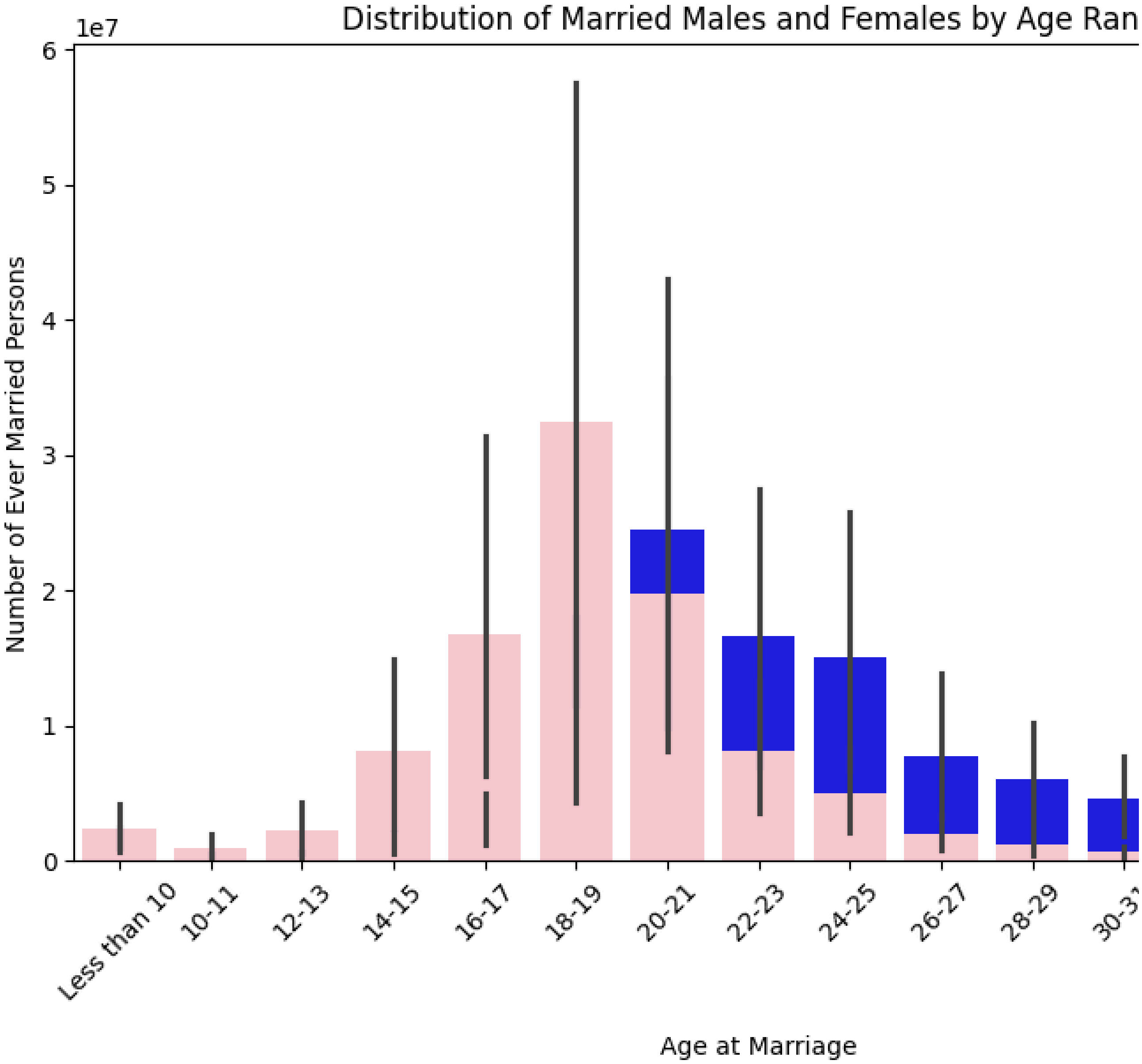
```
# Data cleansing and Data transformation
df_cleaned = df.drop(0).reset_index(drop=True)
df_cleaned.columns = [
    "Area_Name",
    "Total",
    "Educational_Level",
    "Age_at_Marriage",
    "Married_Males",
    "Married_Females"
]
df_cleaned["Married_Males"] = pd.to_numeric(df_cleaned["Married_Males"], errors='
df_cleaned["Married_Females"] = pd.to_numeric(df_cleaned["Married_Females"], erro
```

```
df_cleaned_info = df_cleaned.info()
df_cleaned_head = df_cleaned.head()

df_cleaned_info, df_cleaned_head
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 120 entries, 0 to 119
Data columns (total 6 columns):
 #   Column             Non-Null Count  Dtype 
---  ------             --------------  ----- 
 0   Area_Name          120 non-null    object
 1   Total              120 non-null    object
 2   Educational_Level  120 non-null    object
 3   Age_at_Marriage    120 non-null    object
 4   Married_Males      120 non-null    int64 
 5   Married_Females    120 non-null    int64 
dtypes: int64(2), object(4)
memory usage: 5.8+ KB
(None,
   Area_Name  Total Educational_Level Age_at_Marriage  Married_Males  \
 0     INDIA  Total             Total    Less than 10        4276588   
```

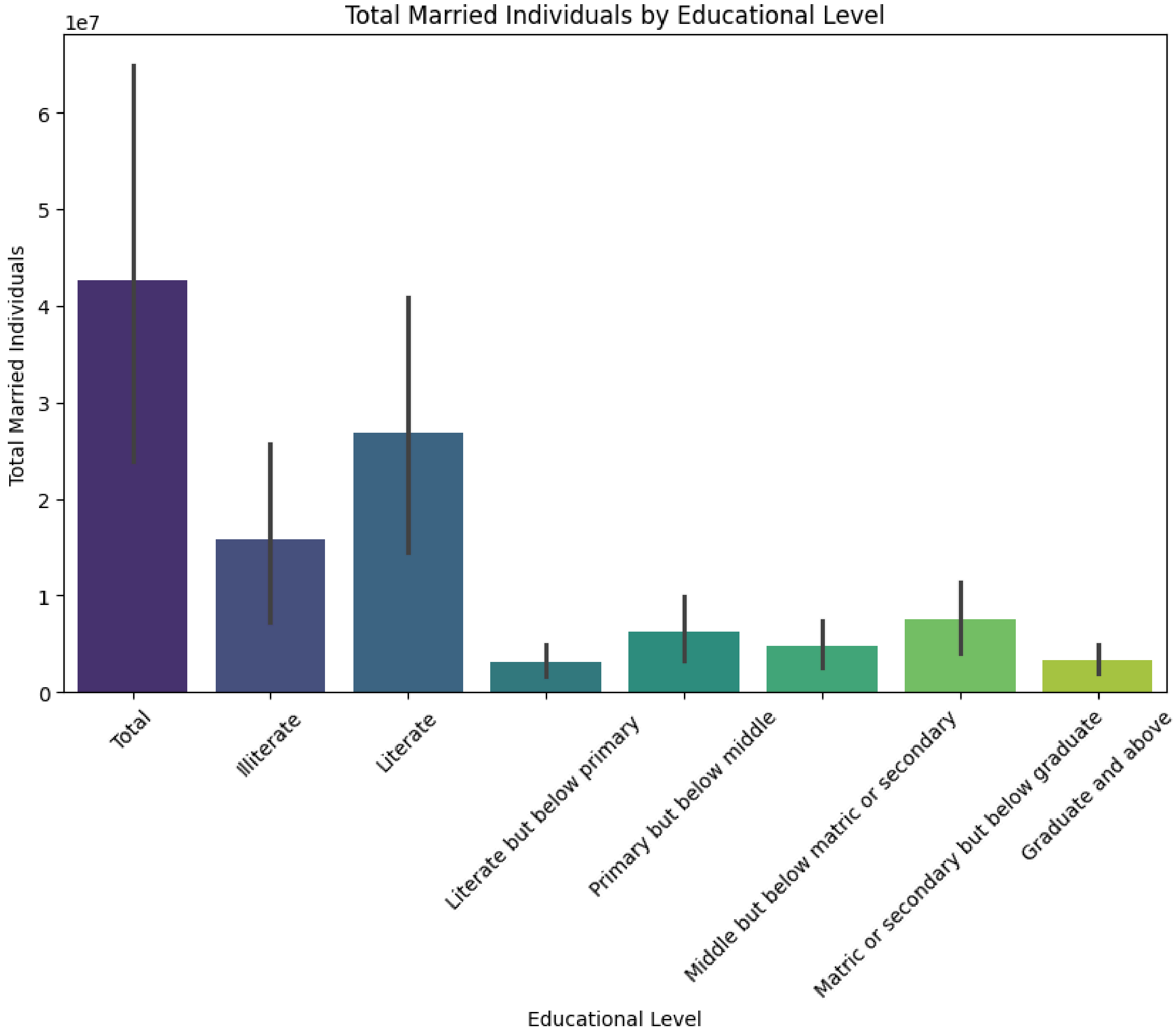
```
      1     INDIA  Total             Total           10–11         979213
      2     INDIA  Total             Total           12–13        1144889
      3     INDIA  Total             Total           14–15        3900762
      4     INDIA  Total             Total           16–17        9169907

         Married_Females
      0          7849859
      1          3434492
      2          7717216
      3         28124694
      4         55489428  )
```

```
#Data visualization
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(10, 6))
sns.barplot(data=df_cleaned, x='Age_at_Marriage', y='Married_Males', color='blue'
sns.barplot(data=df_cleaned, x='Age_at_Marriage', y='Married_Females', color='pin
plt.xticks(rotation=45)
plt.xlabel('Age at Marriage')
plt.ylabel('Number of Ever Married Persons')
plt.legend()
plt.title('Distribution of Married Males and Females by Age Range')
plt.show()
```

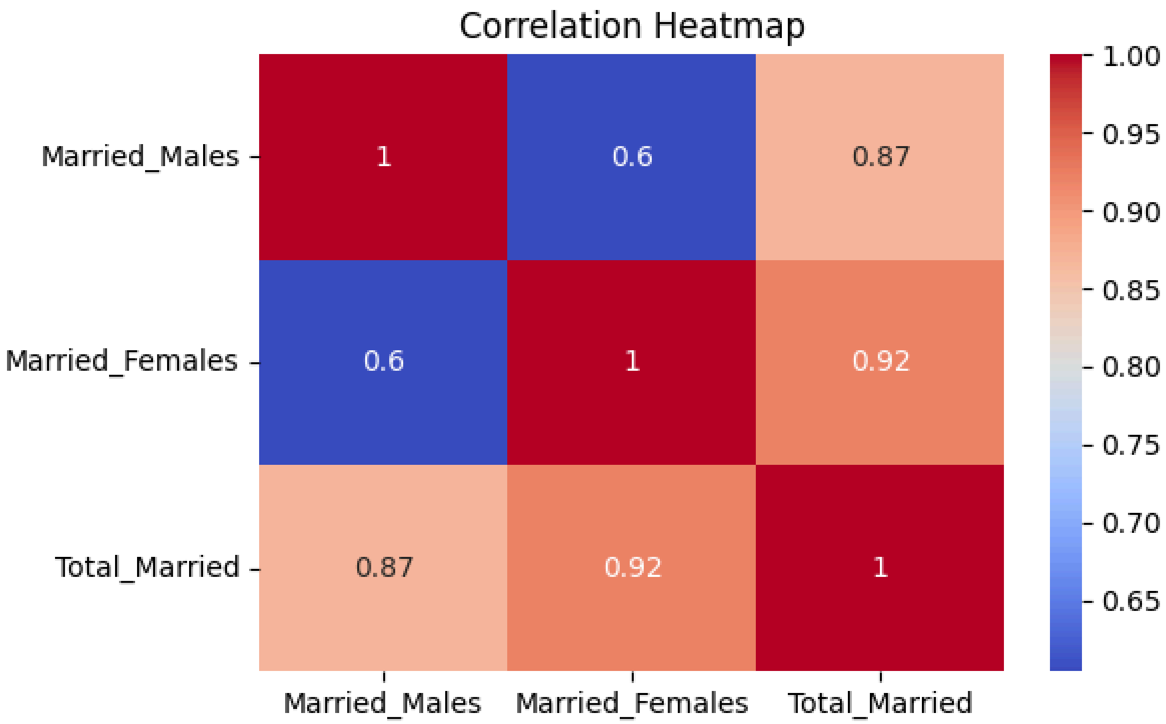


```
df_cleaned['Total_Married'] = df_cleaned['Married_Males'] + df_cleaned['Married_F
plt.figure(figsize=(10, 6))
sns.barplot(data=df_cleaned, x='Educational_Level', y='Total_Married', palette='v
plt.xticks(rotation=45)
plt.xlabel('Educational Level')
plt.ylabel('Total Married Individuals')
plt.title('Total Married Individuals by Educational Level')
plt.show()
```

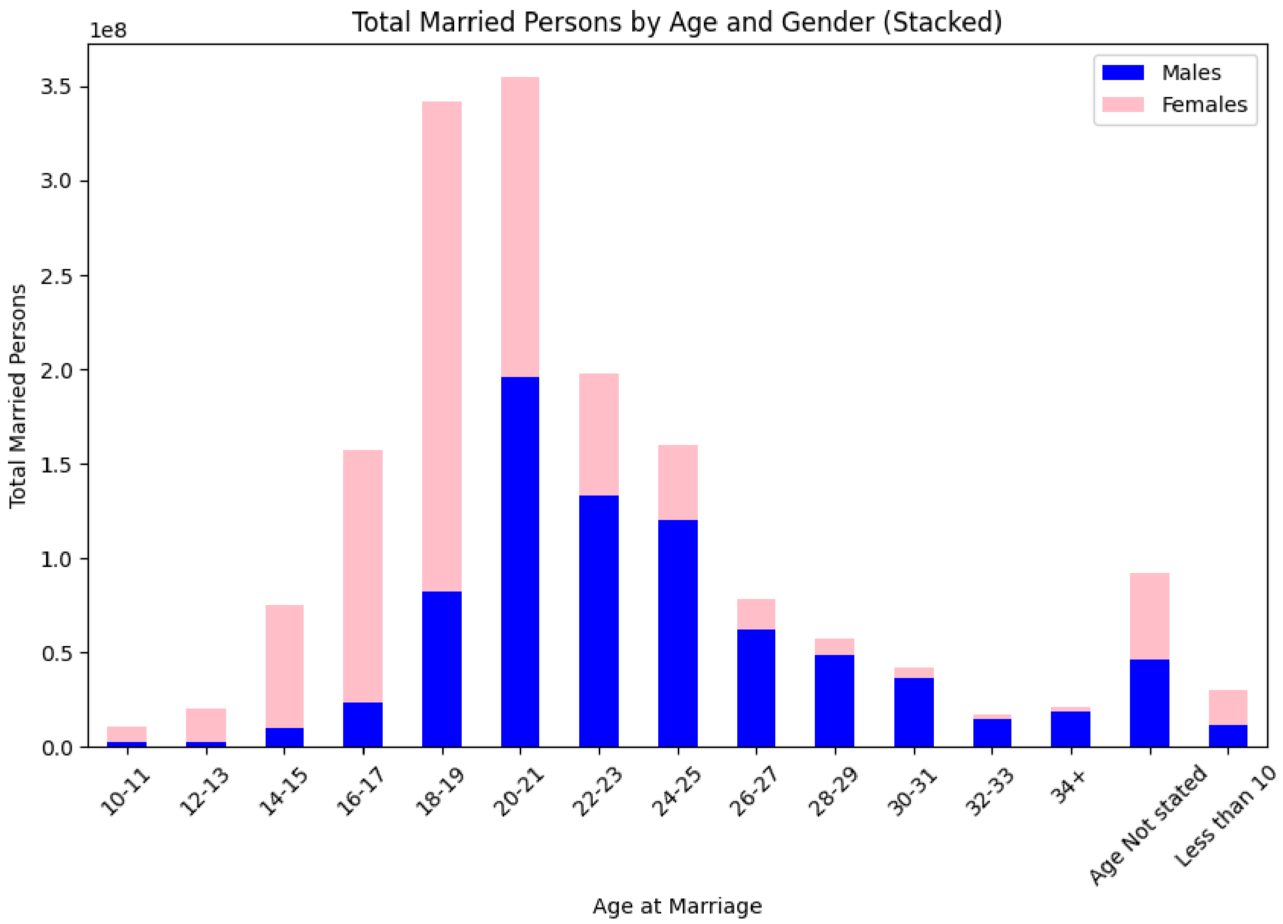
```
<ipython-input-5-f90561edc72b>:4: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed i

  sns.barplot(data=df_cleaned, x='Educational_Level', y='Total_Married', pale
```



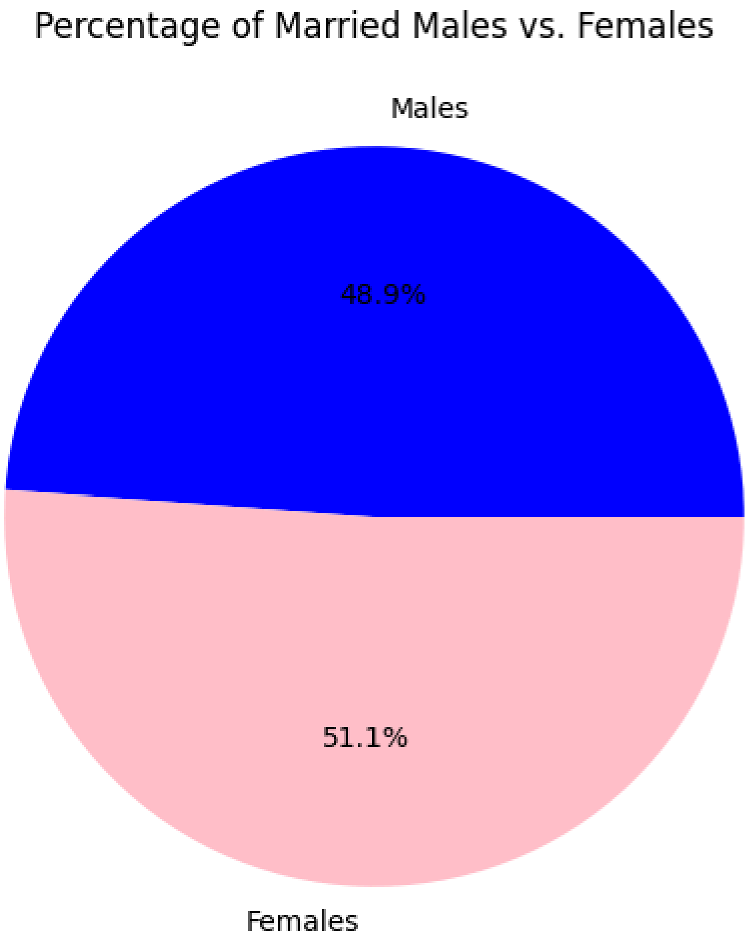
```
plt.figure(figsize=(6, 4))
sns.heatmap(df_cleaned[['Married_Males', 'Married_Females', 'Total_Married']].cor
plt.title('Correlation Heatmap')
plt.show()
```



```
df_age_grouped = df_cleaned.groupby('Age_at_Marriage')[['Married_Males', 'Married
df_age_grouped.plot(kind='bar', x='Age_at_Marriage', stacked=True, color=['blue',
plt.xlabel('Age at Marriage')
plt.ylabel('Total Married Persons')
plt.title('Total Married Persons by Age and Gender (Stacked)')
plt.xticks(rotation=45)
plt.legend(['Males', 'Females'])
plt.show()
```

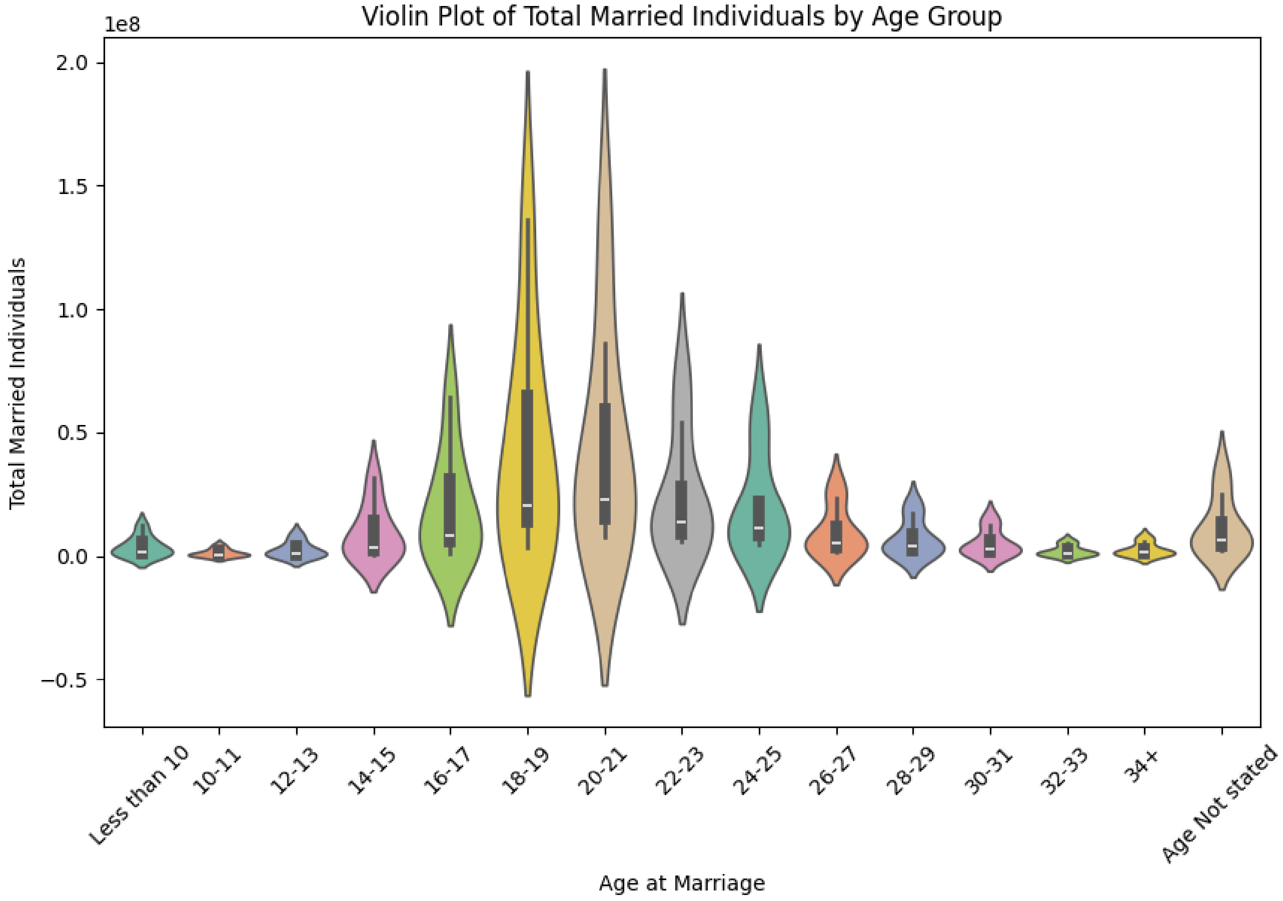


```
total_males = df_cleaned['Married_Males'].sum()
total_females = df_cleaned['Married_Females'].sum()
plt.figure(figsize=(6, 6))
plt.pie([total_males, total_females], labels=['Males', 'Females'], autopct='%1.1f
plt.title('Percentage of Married Males vs. Females')
plt.show()
```

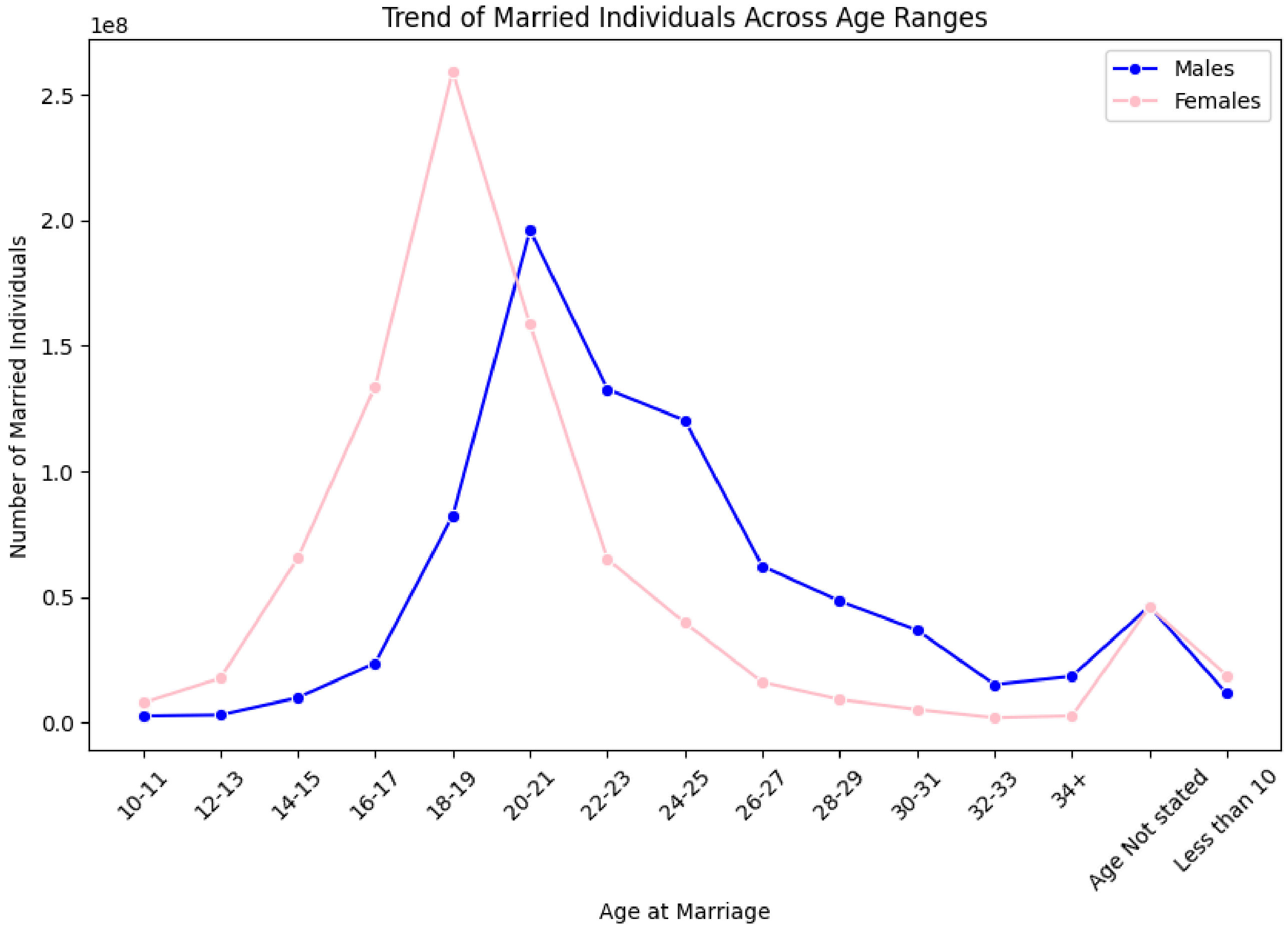


```
plt.figure(figsize=(10, 6))
sns.violinplot(data=df_cleaned, x='Age_at_Marriage', y='Total_Married', palette='
plt.xticks(rotation=45)
plt.xlabel('Age at Marriage')
plt.ylabel('Total Married Individuals')
plt.title('Violin Plot of Total Married Individuals by Age Group')
plt.show()
```

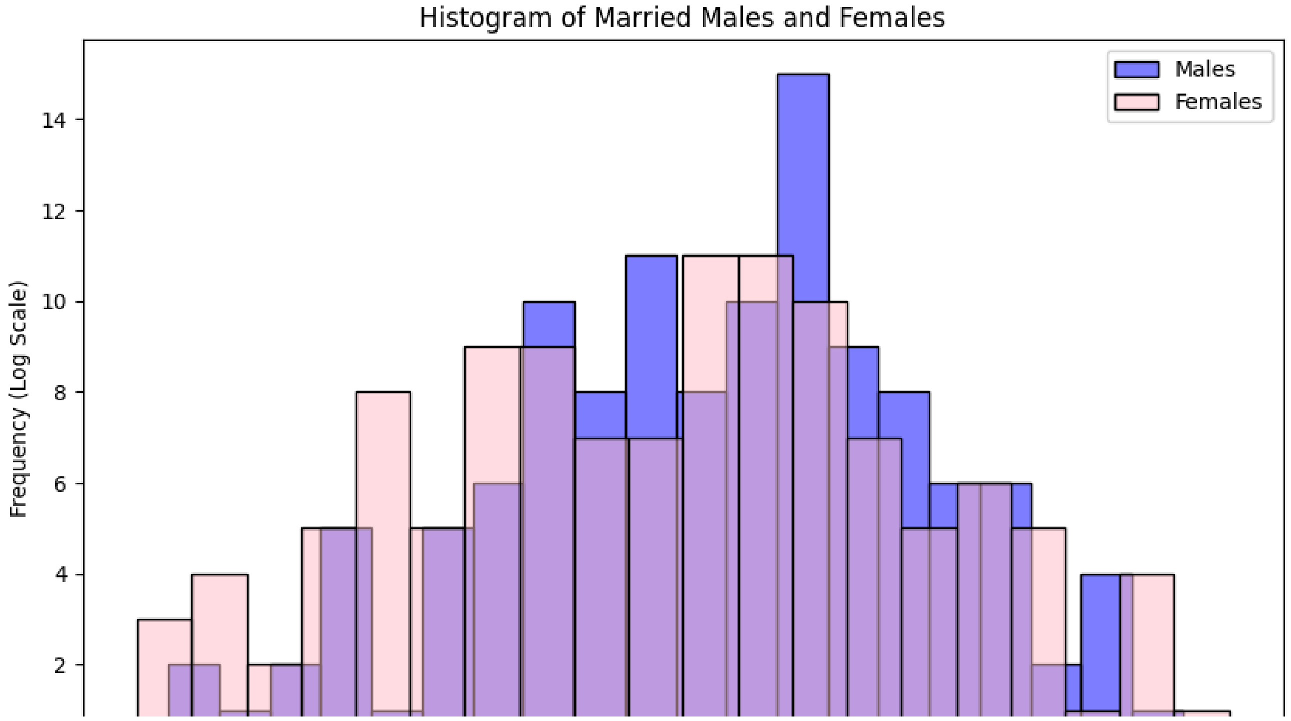
```
<ipython-input-10-8717221d0e66>:2: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed i

  sns.violinplot(data=df_cleaned, x='Age_at_Marriage', y='Total_Married', pal
```



```
df_age_trend = df_cleaned.groupby('Age_at_Marriage')[['Married_Males', 'Married_F
plt.figure(figsize=(10, 6))
sns.lineplot(data=df_age_trend, x='Age_at_Marriage', y='Married_Males', marker='o
sns.lineplot(data=df_age_trend, x='Age_at_Marriage', y='Married_Females', marker=
plt.xticks(rotation=45)
plt.xlabel('Age at Marriage')
plt.ylabel('Number of Married Individuals')
plt.title('Trend of Married Individuals Across Age Ranges')
plt.legend()
plt.show()
```



```
plt.figure(figsize=(10, 6))
sns.histplot(df_cleaned['Married_Males'], bins=20, color='blue', alpha=0.5, label
sns.histplot(df_cleaned['Married_Females'], bins=20, color='pink', alpha=0.5, lab
plt.xlabel('Number of Married Individuals')
plt.ylabel('Frequency (Log Scale)')
plt.title('Histogram of Married Males and Females')
plt.legend()
plt.show()
```



```
plt.figure(figsize=(10, 6))
sns.scatterplot(data=df_cleaned, x='Married_Males', y='Married_Females', hue='Are
plt.xlabel('Married Males')
plt.ylabel('Married Females')
plt.title('Scatter Plot Comparing Married Males and Females by Area')
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left')
plt.show()
```