

```
import pandas as pd # pip install pandas
import matplotlib.pyplot as plt # pip install matplotlib
import seaborn as sns # pip install seaborn

# Load the dataset
# Assuming the dataset is saved as 'titanic.csv'
df = pd.read_csv('titanic.csv')

# Display the first few rows of the dataframe
print(df.head())

# Basic statistics of numeric columns
print(df.describe())

# Check for missing values
print(df.isnull().sum())

''' Data Visualization and EDA Techniques
1. Distribution of Numeric Features
To understand the distribution of numeric
features, such as ages of passengers or fare prices, histograms
are very useful.'''

# Histogram of the Age column
plt.figure(figsize=(10, 6))
sns.distplot(df['Age'].dropna(), kde=True, bins=30)
plt.title('Distribution of Age')
plt.xlabel('Age')
plt.ylabel('Density')
plt.show()

'''2. Categorical Data Analysis
Bar charts and count plots can show the distribution of categorical data,
like the number of passengers in each class or the number of survivors.'''
# Count plot for the 'Pclass' column
plt.figure(figsize=(10, 6))
sns.countplot(x='Pclass', data=df)
plt.title('Passenger Class Distribution')
plt.xlabel('Passenger Class')
plt.ylabel('Count')
plt.show()

'''3. Correlation Heatmap
Understanding how different features are related to each other can be crucial.
A heatmap can visualize the correlation matrix.'''
# Correlation heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
plt.title('Correlation between Features')
plt.show()

''' 4. Comparing Survival Rates
Comparing the survival rates across different groups can provide insights into
what factors might have influenced survival on the Titanic.'''

# Survival rate by passenger class
plt.figure(figsize=(10, 6))
sns.barplot(x='Pclass', y='Survived', data=df)
plt.title('Survival Rate by Passenger Class')
plt.xlabel('Passenger Class')
plt.ylabel('Survival Rate')
plt.show()

# Survival rate by gender
plt.figure(figsize=(10, 6))
sns.barplot(x='Sex', y='Survived', data=df)
plt.title('Survival Rate by Gender')
plt.xlabel('Gender')
plt.ylabel('Survival Rate')
plt.show()

''' 5. Pairplot
A pairplot can help visualize the distribution of individual variables and
the relationships between them, for a subset of features.'''

# Pairplot for a subset of variables
sns.pairplot(df[['Survived', 'Pclass', 'Age', 'Fare']], hue='Survived')
plt.show()
```

PassengerId	Survived	Pclass	\
0	1	0	3
1	2	1	1
2	3	1	3
3	4	1	1
4	5	0	3

Name	Sex	Age	SibSp	\
Braund, Mr. Owen Harris	male	22.0	1	
Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	
Heikkinen, Miss. Laina	female	26.0	0	
Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	
Allen, Mr. William Henry	male	35.0	0	

Parch	Ticket	Fare	Cabin	Embarked
0	A/5 21171	7.2500	NaN	S
1	PC 17599	71.2833	C85	C
2	STON/O2. 3101282	7.9250	NaN	S
3	113803	53.1000	C123	S
4	373450	8.0500	NaN	S

count	mean	std	min	25%	50%	75%	max
891.000000	446.000000	257.353842	1.000000	223.500000	446.000000	668.500000	891.000000
891.000000	0.383838	0.486592	0.000000	0.000000	0.000000	1.000000	1.000000
891.000000	2.308642	0.836071	1.000000	2.000000	3.000000	3.000000	3.000000
714.000000	29.699118	14.526497	0.420000	20.125000	28.000000	38.000000	80.000000
891.000000	0.523008	1.102743	0.000000	0.000000	0.000000	1.000000	8.000000

count	mean	std	min	25%	50%	75%	max
891.000000	0.381594	0.806057	0.000000	0.000000	0.000000	0.000000	6.000000
891.000000	32.204208	49.693429	0.000000	7.910400	14.454200	31.000000	512.329200

PassengerId	0
Survived	0
Pclass	0
Name	0
Sex	0
Age	177
SibSp	0
Parch	0
Ticket	0
Fare	0
Cabin	687
Embarked	2

dtype: int64

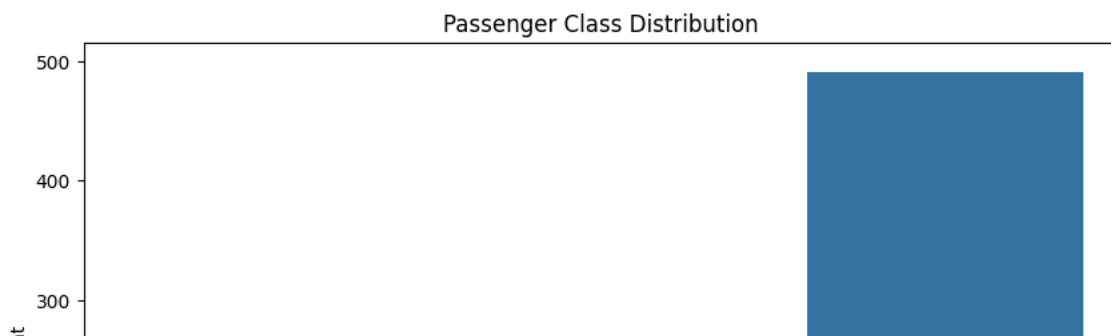
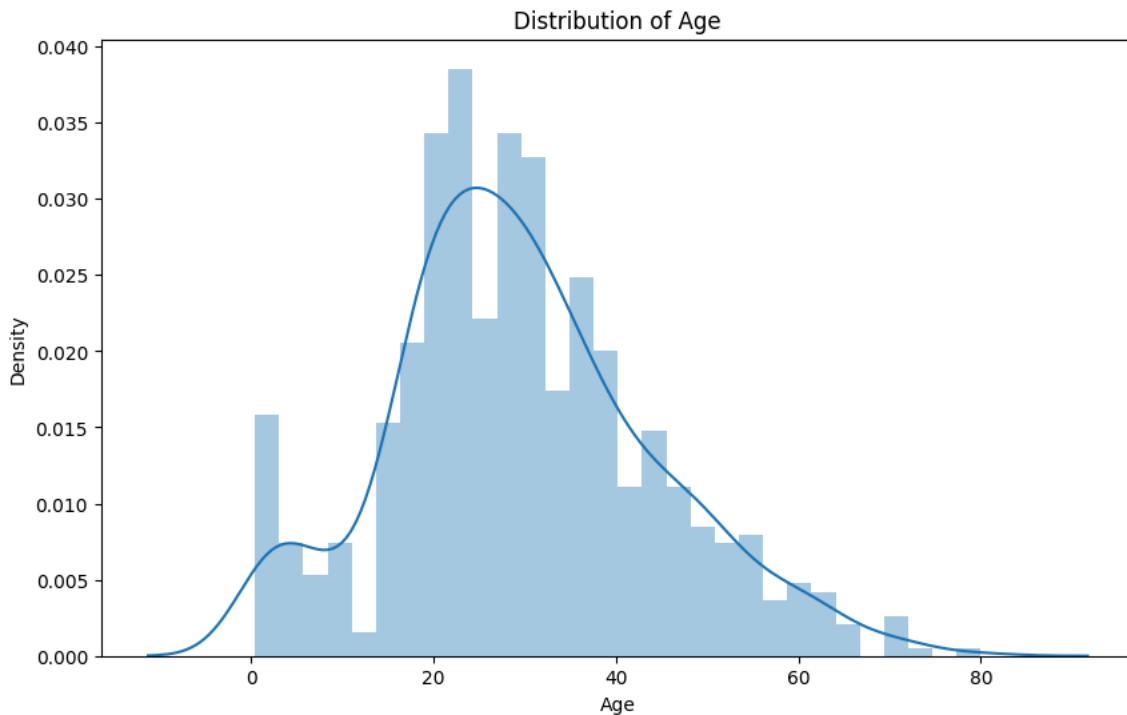
<ipython-input-2-b06cb08e993d>:28: UserWarning:

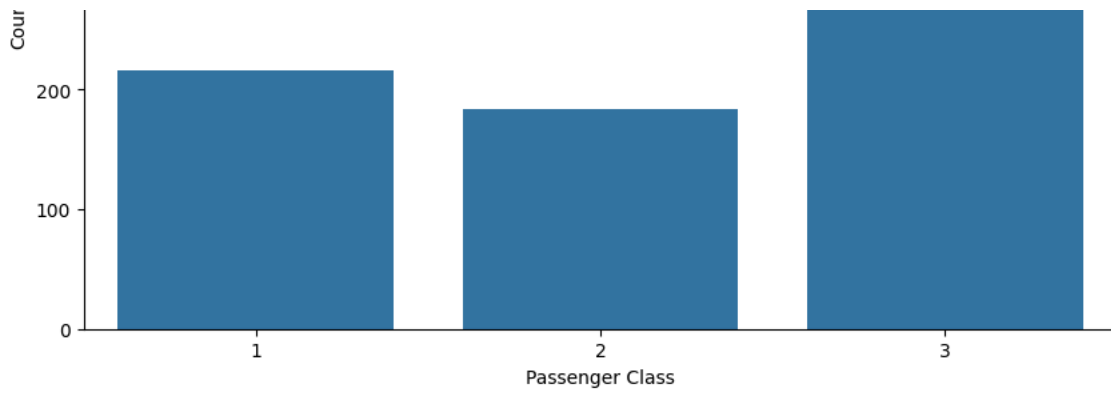
`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see <https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751>

```
sns.distplot(df['Age'].dropna(), kde=True, bins=30)
```





<ipython-input-2-b06cb08e993d>:52: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version sns.heatmap(df.corr(), annot=True, cmap='coolwarm')

