

▼ Q1. Download the Titanic dataset and perform the Exploratory data analysis using pandas.

Read the dataset (df= pd.read_csv(r'.....\Titanic.csv')

Display the first and last 10 instances from the dataset

Acquire the necessary information using the df.info() and df. Describe().

Retrieve the number of columns and rows. (using shape)

```
import pandas as pd
df = pd.read_csv('/content/Titanic - Titanic.csv')
```

```
print("First 10 instances from data set")
print(df.head(10))
```

```
First 10 instances from data set
PassengerId  Survived  Pclass \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3
5            6         0       3
6            7         0       1
7            8         0       3
8            9         1       3
9           10         1       2
```

```

                                Name    Sex  Age  SibSp \
0                Braund, Mr. Owen Harris  male  22.0    1
1  Cumings, Mrs. John Bradley (Florence Briggs Th... female  38.0    1
2                Heikkinen, Miss. Laina  female  26.0    0
3  Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0    1
4                Allen, Mr. William Henry  male  35.0    0
5                Moran, Mr. James  male  NaN    0
6      McCarthy, Mr. Timothy J  male  54.0    0
7      Palsson, Master. Gosta Leonard  male   2.0    3
8  Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)  female  27.0    0
9      Nasser, Mrs. Nicholas (Adele Achem)  female  14.0    1
```

```

Parch  Ticket    Fare  Cabin  Embarked
0      0   A/5 21171   7.2500   NaN      S
1      0    PC 17599  71.2833   C85      C
2      0  STON/O2. 3101282   7.9250   NaN      S
3      0   113803  53.1000  C123      S
4      0   373450   8.0500   NaN      S
5      0   330877   8.4583   NaN      Q
6      0   17463   51.8625   E46      S
7      1   349909  21.0750   NaN      S
8      2   347742  11.1333   NaN      S
9      0   237736  30.0708   NaN      C
```

```
print("\nlast 10 instances from data set")
print(df.tail(10))
```

```
last 10 instances from data set
PassengerId  Survived  Pclass                                Name \
881          882         0       3                Markun, Mr. Johann
882          883         0       3      Dahlberg, Miss. Gerda Ulrika
883          884         0       2      Banfield, Mr. Frederick James
884          885         0       3      Sutehall, Mr. Henry Jr
885          886         0       3      Rice, Mrs. William (Margaret Norton)
886          887         0       2      Montvila, Rev. Juozas
887          888         1       1      Graham, Miss. Margaret Edith
888          889         0       3  Johnston, Miss. Catherine Helen "Carrie"
889          890         1       1      Behr, Mr. Karl Howell
890          891         0       3      Dooley, Mr. Patrick
```

```

Sex  Age  SibSp  Parch  Ticket    Fare  Cabin  Embarked
881  male  33.0    0     0   349257   7.8958   NaN      S
882  female  22.0    0     0     7552  10.5167   NaN      S
883  male  28.0    0     0  C.A./SOTON 34068  10.5000   NaN      S
884  male  25.0    0     0  SOTON/OQ 392076   7.0500   NaN      S
885  female  39.0    0     5   382652  29.1250   NaN      Q
886  male  27.0    0     0   211536  13.0000   NaN      S
887  female  19.0    0     0   112053  30.0000  B42      S
888  female  NaN    1     2    W./C. 6607  23.4500   NaN      S
```

889	male	26.0	0	0	111369	30.0000	C148	C
890	male	32.0	0	0	370376	7.7500	NaN	Q

```
print("\ndf.info()")
print(df.info())
```

```
df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
#   Column      Non-Null Count  Dtype
---  -
0   PassengerId  891 non-null    int64
1   Survived     891 non-null    int64
2   Pclass       891 non-null    int64
3   Name         891 non-null    object
4   Sex          891 non-null    object
5   Age          714 non-null    float64
6   SibSp        891 non-null    int64
7   Parch        891 non-null    int64
8   Ticket       891 non-null    object
9   Fare         891 non-null    float64
10  Cabin        204 non-null    object
11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
None
```

```
print("\ndf.describe()")
print(df.describe())
```

```
df.describe()
      PassengerId  Survived  Pclass     Age  SibSp  \
count  891.000000  891.000000  891.000000  714.000000  891.000000
mean    446.000000    0.383838    2.308642   29.699118    0.523008
std     257.353842    0.486592    0.836071   14.526497    1.102743
min       1.000000    0.000000    1.000000    0.420000    0.000000
25%     223.500000    0.000000    2.000000   20.125000    0.000000
50%     446.000000    0.000000    3.000000   28.000000    0.000000
75%     668.500000    1.000000    3.000000   38.000000    1.000000
max     891.000000    1.000000    3.000000   80.000000    8.000000

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

```
print("\nRetrieve the number of columns and rows. (using shape)")
print(df.shape)
```

```
Retrieve the number of columns and rows. (using shape)
(891, 12)
```

▼ Q2. Create the data visualization using the matplotlib.

Visualize the Gender of Passengers using the Bar graph.

Visualize the Survival Count of Passengers using the Bar graph.

Visualize the Age of Passengers using the Bar/Histogram graph.

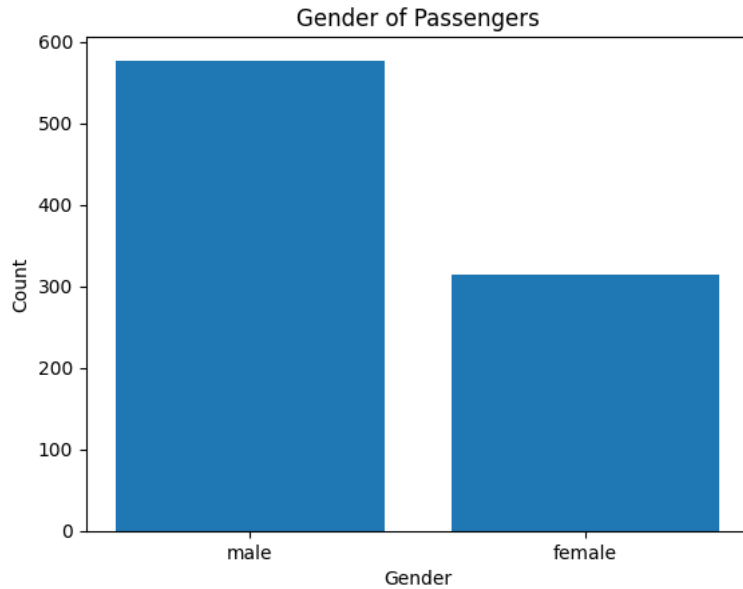
Visualize the comparison of Age and Fare of Passengers using the Scatterplot.

```
import matplotlib.pyplot as plt

print("Visualize the Gender of Passengers using the Bar graph.")
gender_counts = df['Sex'].value_counts()
plt.bar(gender_counts.index, gender_counts.values)
plt.title('Gender of Passengers')
plt.xlabel('Gender')
plt.ylabel('Count')
```

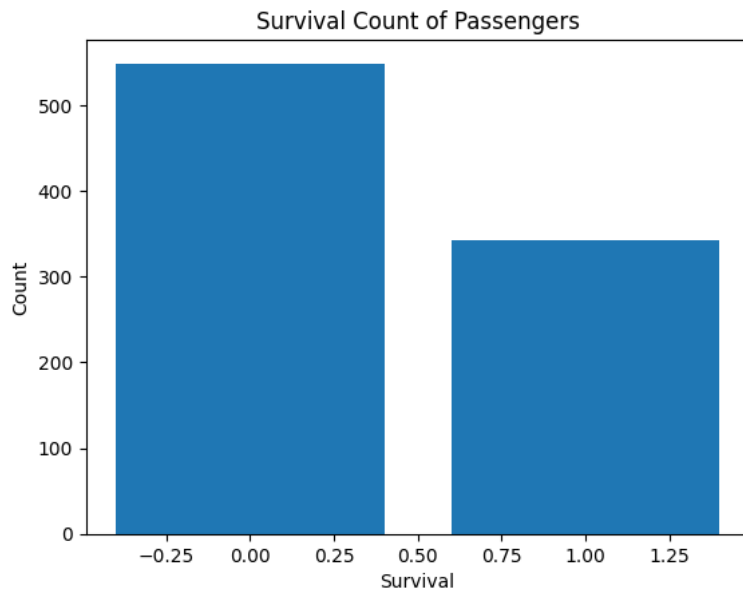
```
plt.show()
```

Visualize the Gender of Passengers using the Bar graph.



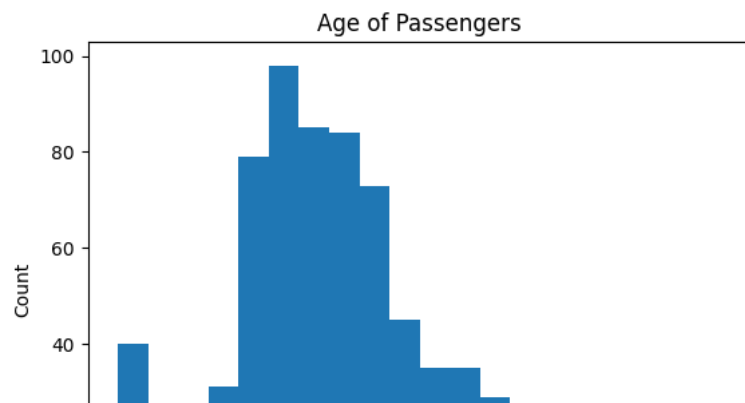
```
print("\nVisualize the Survival Count of Passengers using the Bar graph.")
survival_counts = df['Survived'].value_counts()
plt.bar(survival_counts.index, survival_counts.values)
plt.title('Survival Count of Passengers')
plt.xlabel('Survival')
plt.ylabel('Count')
plt.show()
```

Visualize the Survival Count of Passengers using the Bar graph.



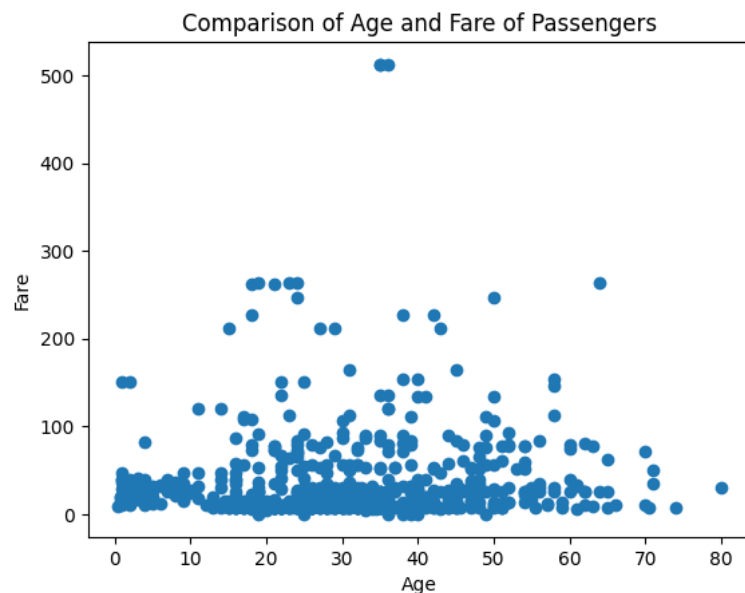
```
print("\nVisualize the Age of Passengers using the Bar/Histogram graph.")
plt.hist(df['Age'].dropna(), bins=20)
plt.title('Age of Passengers')
plt.xlabel('Age')
plt.ylabel('Count')
plt.show()
```

Visualize the Age of Passengers using the Bar/Histogram graph.



```
print("\nVisualize the comparison of Age and Fare of Passengers using the Scatterplot.")
plt.scatter(df['Age'], df['Fare'])
plt.title('Comparison of Age and Fare of Passengers')
plt.xlabel('Age')
plt.ylabel('Fare')
plt.show()
```

Visualize the comparison of Age and Fare of Passengers using the Scatterplot.



✓ 0s completed at 10:32 AM

● ×