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)

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.

```python
In [ ]:  import pandas as pd
         import numpy as np
         import matplotlib.pyplot as plt
         import seaborn as sns
         from IPython.display import Image, display
         %matplotlib inline
         titan = pd.read_csv('/content/Titanic - Titanic.csv')
         display(titan.head(10))
         display(titan.tail(10))
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.4583
6	7	0	1	McCarthy, Mr. Timothy J	male	54.0	0	0	17463	51.8625
7	8	0	3	Palsson, Master. Gosta Leonard	male	2.0	3	1	349909	21.0750
8	9	1	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27.0	0	2	347742	11.1333
9	10	1	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14.0	1	0	237736	30.0708

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fa
881	882	0	3	Markun, Mr. Johann	male	33.0	0	0	349257	7.89
882	883	0	3	Dahlberg, Miss. Gerda Ulrika	female	22.0	0	0	7552	10.51
883	884	0	2	Banfield, Mr. Frederick James	male	28.0	0	0	C.A./SOTON 34068	10.50
884	885	0	3	Sutehall, Mr. Henry Jr	male	25.0	0	0	SOTON/OQ 392076	7.05
885	886	0	3	Rice, Mrs. William (Margaret Norton)	female	39.0	0	5	382652	29.12
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.00
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.00
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.45
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.00
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.75

```
In [ ]:  titan.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
```

In [ ]: `titan.describe()`

Out[ ]:

|	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

In [ ]:
```python
rownum = np.shape(titan)[0]
colnum = np.shape(titan)[1]
print(rownum)
print(colnum)
```

```
891
12
```

In [ ]:
```python
gender_counts = titan['Sex'].value_counts()
plt.figure(figsize=(6, 4))
sns.barplot(x=gender_counts.index, y=gender_counts.values)
plt.title('Gender of Passengers')
plt.xlabel('Gender')
plt.ylabel('Count')
plt.show()
```

Gender of Passengers
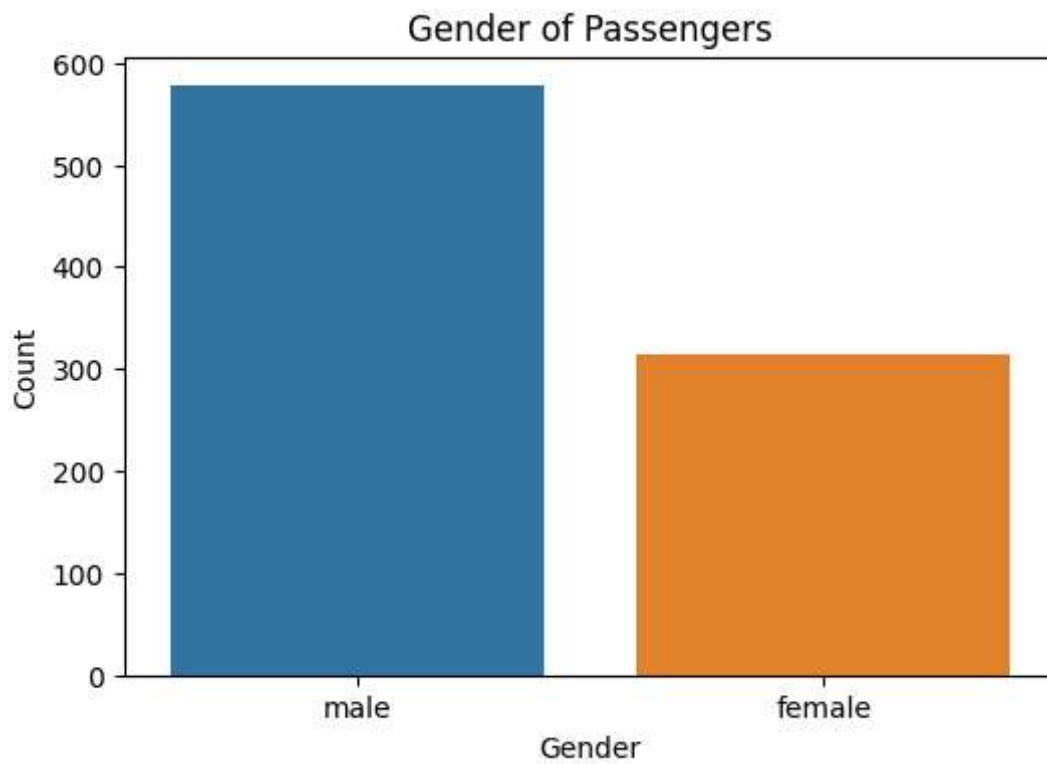

```
In [ ]:  survival_counts = titan['Survived'].value_counts()
         plt.figure(figsize=(6, 4))
         sns.barplot(x=survival_counts.index, y=survival_counts.values)
         plt.title('Survival Count of Passengers')
         plt.xlabel('Survived')
         plt.ylabel('Count')
         plt.show()
```

Survival Count of Passengers
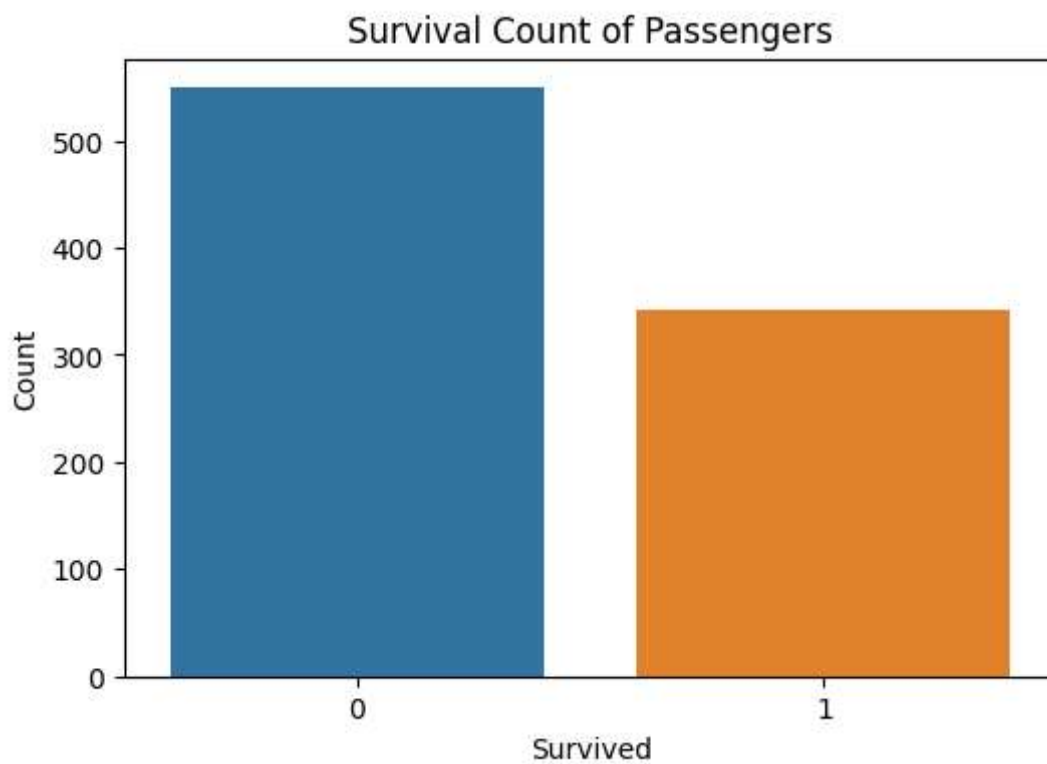
```
In [ ]: plt.figure(figsize=(8, 6))
        plt.hist(titan['Age'], bins=20, edgecolor='black')
        plt.title('Age of Passengers')
        plt.xlabel('Age')
        plt.ylabel('Count')
        plt.show()
```
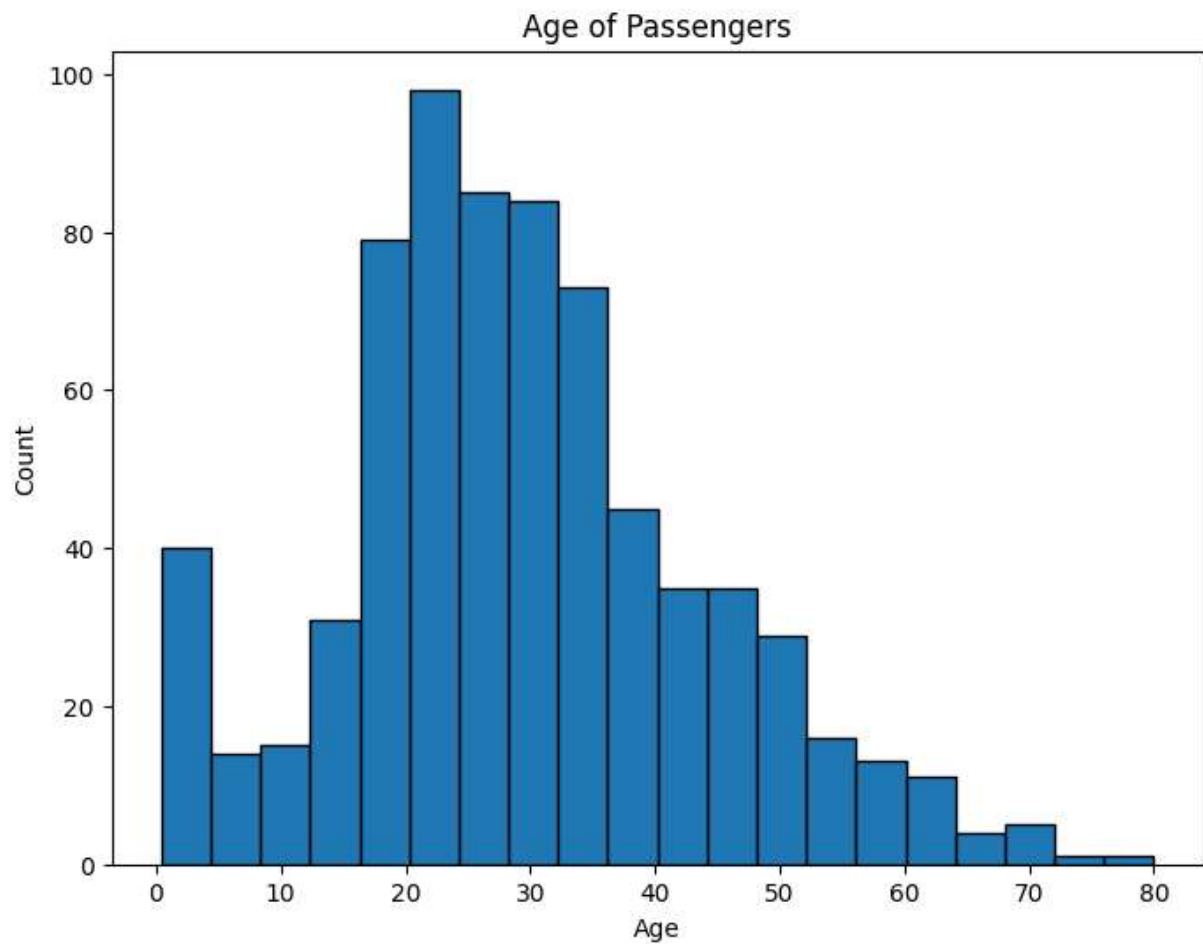


```
In [ ]: plt.figure(figsize=(8, 6))
        sns.scatterplot(x='Age', y='Fare', data=titan)
        plt.title('Age vs Fare of Passengers')
        plt.xlabel('Age')
        plt.ylabel('Fare')
        plt.show()
```

Age vs Fare of Passengers