Name- Tanvi Paygude

Elevate Labs Internship Task 5

Dataset Used: titanic.csv (File Attached in the Repository)

Objective: The main goal is to explore a dataset using Python tools such as Pandas, Matplotlib, and Seaborn. By doing this, we aim to find patterns, trends, insights, and any unusual points in the data through exploration and visualizations.

1. **Read the dataset**
   df = pd.read_csv('titanic.csv')

```
In [14]: df
```

Out[14]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	892	0	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN	Q
1	893	1	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	S
2	894	0	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN	Q
3	895	0	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN	S
4	896	1	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
413	1305	0	3	Spector, Mr. Woolf	male	NaN	0	0	A.5. 3236	8.0500	NaN	S
414	1306	1	1	Oliva y Ocana, Dona. Fermina	female	39.0	0	0	PC 17758	108.9000	C105	C
415	1307	0	3	Saether, Mr. Simon Sivertsen	male	38.5	0	0	SOTON/O.Q. 3101262	7.2500	NaN	S
416	1308	0	3	Ware, Mr. Frederick	male	NaN	0	0	359309	8.0500	NaN	S
417	1309	0	3	Peter, Master. Michael J	male	NaN	1	1	2668	22.3583	NaN	C

418 rows × 12 columns

2. **Preprocessing**
   print("Missing values before cleaning:\n", df.isnull().sum())
   df=df.dropna()

```
In [8]: print("Missing values before cleaning:\n", data.isnull().sum())
        data=data.dropna()
```

```
Out[8]: PassengerId      0
        Pclass           0
        Name             0
        Sex              0
        Age             86
        SibSp            0
        Parch            0
        Ticket           0
        Fare             1
        Cabin          327
        Embarked         0
        dtype: int64
```

3. **Basic Functions**
   - df.info()

```
In [18]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 87 entries, 12 to 414
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  87 non-null     int64
 1   Survived     87 non-null     int64
 2   Pclass       87 non-null     int64
 3   Name         87 non-null     object
 4   Sex          87 non-null     object
 5   Age          87 non-null     float64
 6   SibSp        87 non-null     int64
 7   Parch        87 non-null     int64
 8   Ticket       87 non-null     object
 9   Fare         87 non-null     float64
 10  Cabin        87 non-null     object
 11  Embarked     87 non-null     object
dtypes: float64(2), int64(5), object(5)
memory usage: 8.8+ KB
```

- df.describe()

```
In [19]: df.describe()
```

Out[19]:

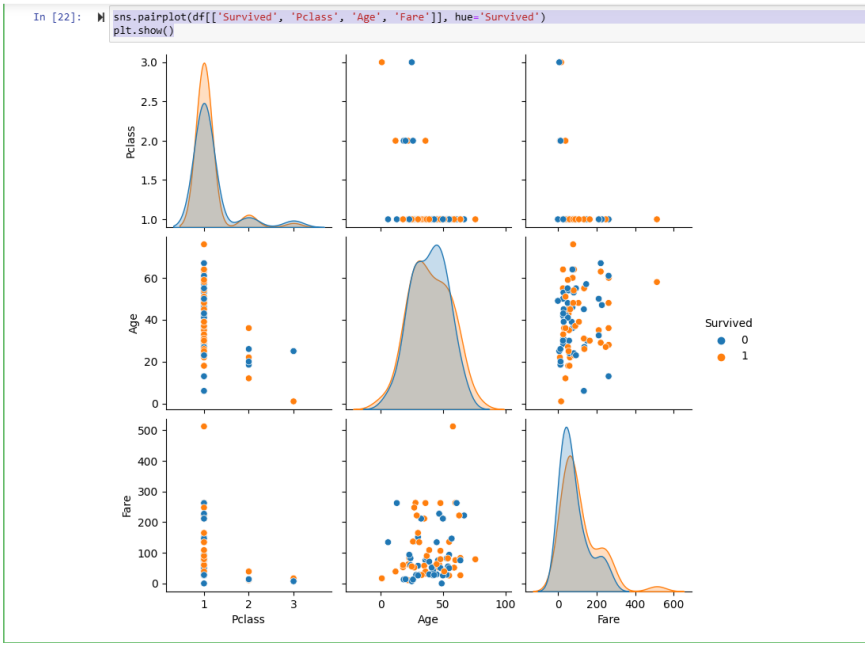
	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	87.000000	87.000000	87.000000	87.000000	87.000000	87.000000	87.000000
mean	1102.712644	0.505747	1.137931	39.247126	0.597701	0.482759	98.109198
std	126.751901	0.502865	0.435954	15.218730	0.637214	0.860801	88.177319
min	904.000000	0.000000	1.000000	1.000000	0.000000	0.000000	0.000000
25%	986.000000	0.000000	1.000000	27.000000	0.000000	0.000000	35.339600
50%	1094.000000	1.000000	1.000000	39.000000	1.000000	0.000000	71.283300
75%	1216.000000	1.000000	1.000000	50.000000	1.000000	1.000000	135.066650
max	1306.000000	1.000000	3.000000	76.000000	3.000000	4.000000	512.329200

- df['Survived'].value_counts()

```
In [20]: df['Survived'].value_counts()
```

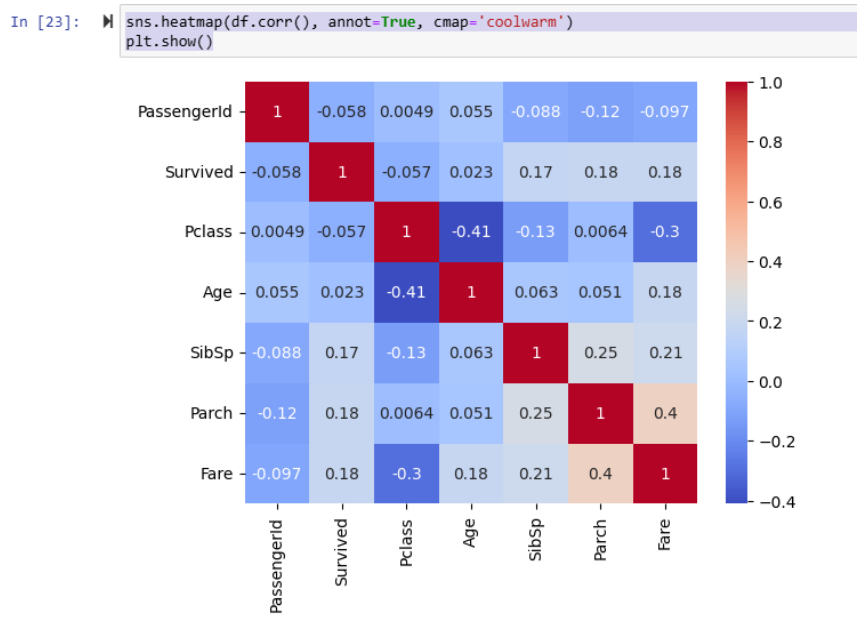
```
Out[20]: 1    44
         0    43
         Name: Survived, dtype: int64
```

4. **Visualization**
   - **Pairplot:** sns.pairplot(df[['Survived', 'Pclass', 'Age', 'Fare']], hue='Survived')
     plt.show()



Observation: The pairplot shows the relationships between survival status, passenger class (Pclass), age, and fare. We can observe that passengers who survived (colored differently) were generally from the higher classes (Pclass = 1), had paid higher fares, and were often younger. Non-survivors were mostly from lower classes and paid lower fares. There is also a visible spread in age and fare values among the survivors, suggesting that these factors played a role in survival chances.

- **Heatmap:** sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
  plt.show()

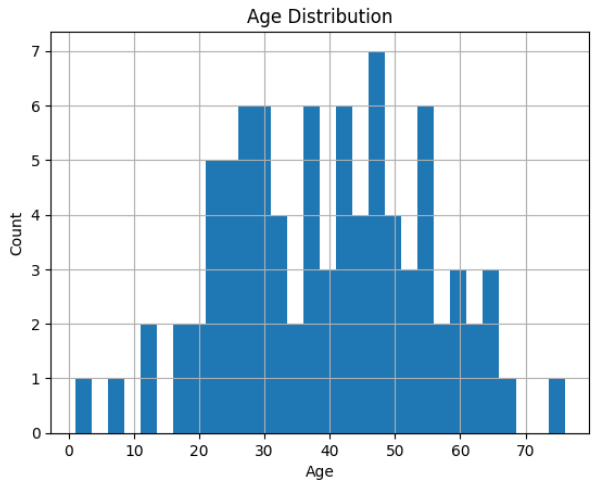


Observation: The heatmap illustrates the correlation between the numerical variables. There is a strong negative correlation between Pclass and Fare, meaning higher-class passengers (lower Pclass values) generally paid more. The Survived column shows a positive correlation with Fare,

indicating that passengers who paid higher fares were more likely to survive. A mild positive correlation is also visible between SibSp and Parch, suggesting that passengers with siblings/spouses were also likely traveling with parents/children (families traveling together).

- **Histogram** : df['Age'].hist(bins=30)
  plt.title('Age Distribution')
  plt.xlabel('Age')
  plt.ylabel('Count')
  plt.show()

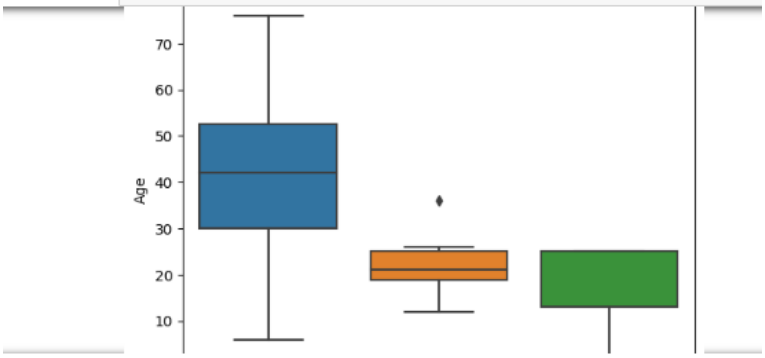
```
In [24]: df['Age'].hist(bins=30)
         plt.title('Age Distribution')
         plt.xlabel('Age')
         plt.ylabel('Count')
         plt.show()
```



Observation: The histogram of Age shows that most passengers were between 20 and 40 years old. There is a peak around the 20–30 age range, indicating that the majority of the travelers were young adults. Fewer older passengers (above 60 years) were on board, and there is a small number of very young children. This gives an idea about the general age demographics of the passengers on the Titanic.

- **Boxplot:** sns.boxplot(x='Pclass', y='Age', data=df)
  plt.title('Age distribution across Passenger Classes')
  plt.show()

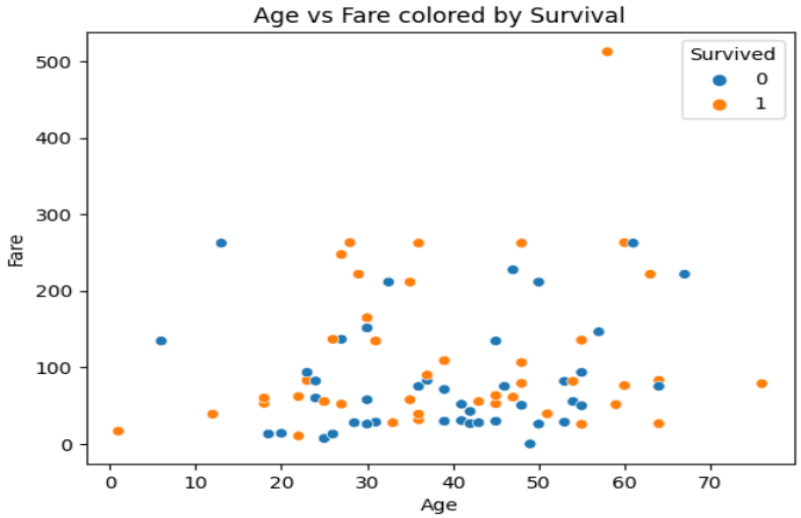
```
In [25]: sns.boxplot(x='Pclass', y='Age', data=df)
         plt.title('Age distribution across Passenger Classes')
         plt.show()
```



Observation: The boxplot shows how the age distribution varies across the different passenger classes. First-class passengers were generally older compared to those in the second and third classes. Third-class passengers included many young individuals, including children. The presence of outliers, particularly in the lower classes, suggests that some very young or very old individuals traveled even in the cheaper classes.

- Scatterplot: sns.scatterplot(x='Age', y='Fare', hue='Survived', data=df)
  plt.title('Age vs Fare colored by Survival')
  plt.show()

```
In [26]: sns.scatterplot(x='Age', y='Fare', hue='Survived', data=df)
         plt.title('Age vs Fare colored by Survival')
         plt.show()
```



Observation: The scatterplot between Age and Fare colored by survival shows that passengers who paid higher fares had a greater chance of survival. Survivors are clustered more toward higher fare values, especially for fares above 100. Age does not show a very strong relationship with Fare, but it can be observed that regardless of age, passengers paying more had better survival chances. The majority of lower-fare passengers, regardless of age, did not survive.