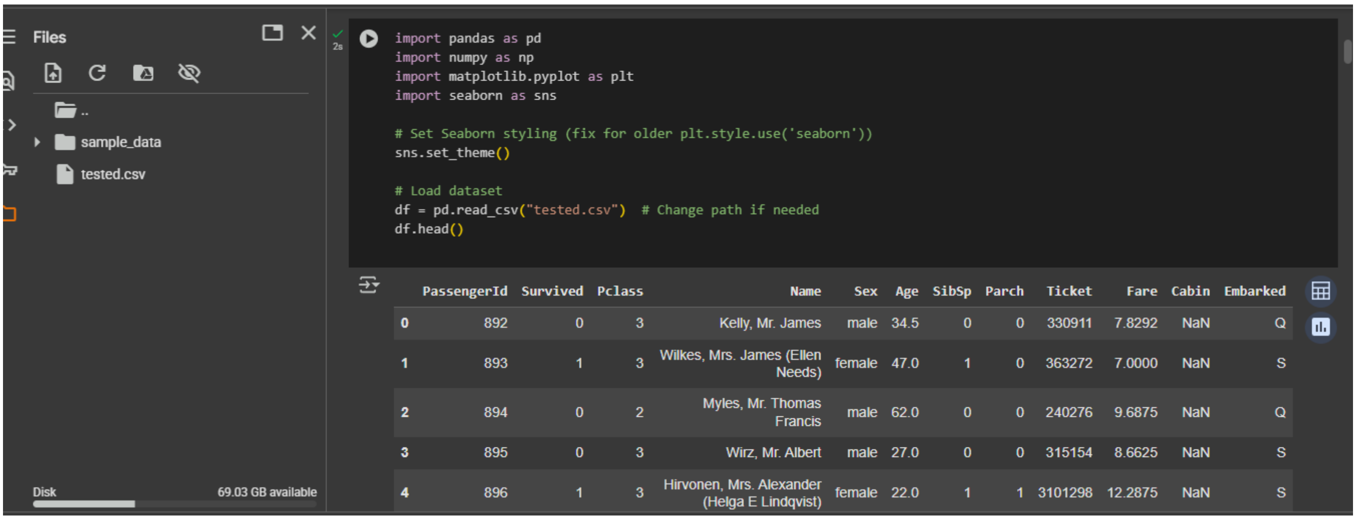
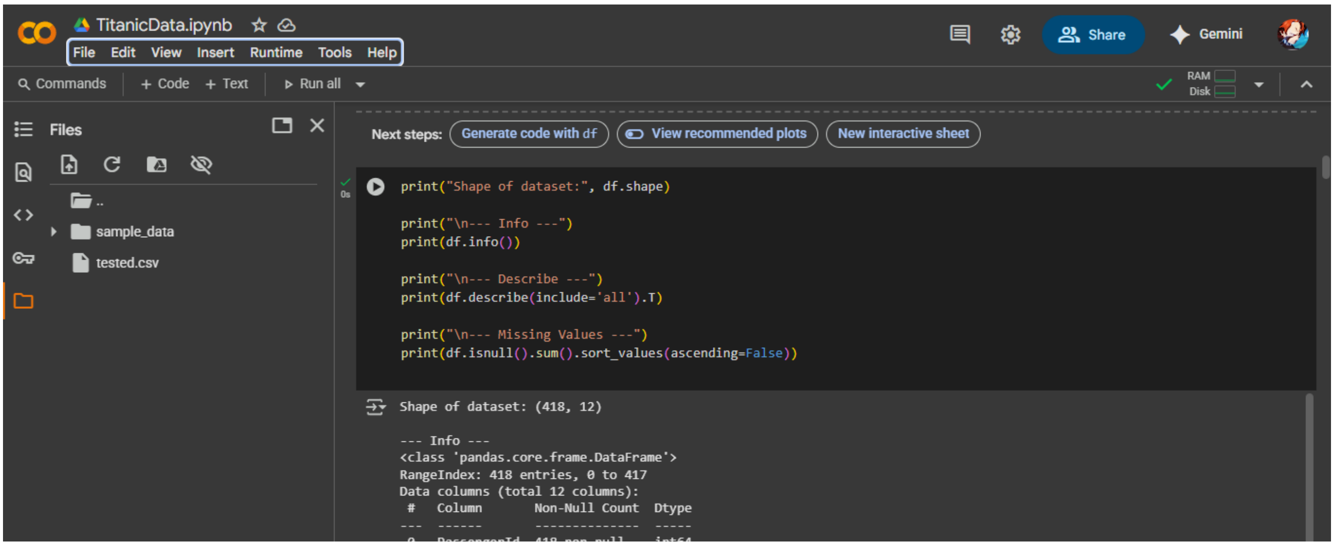
Conducting Data analysis on Titanic passenger data on a file named Tested.csv via Jupyter notebook file (.ipynb file).



```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# Set Seaborn styling (fix for older plt.style.use('seaborn'))
sns.set_theme()

# Load dataset
df = pd.read_csv("tested.csv")  # Change path if needed
df.head()
```

	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

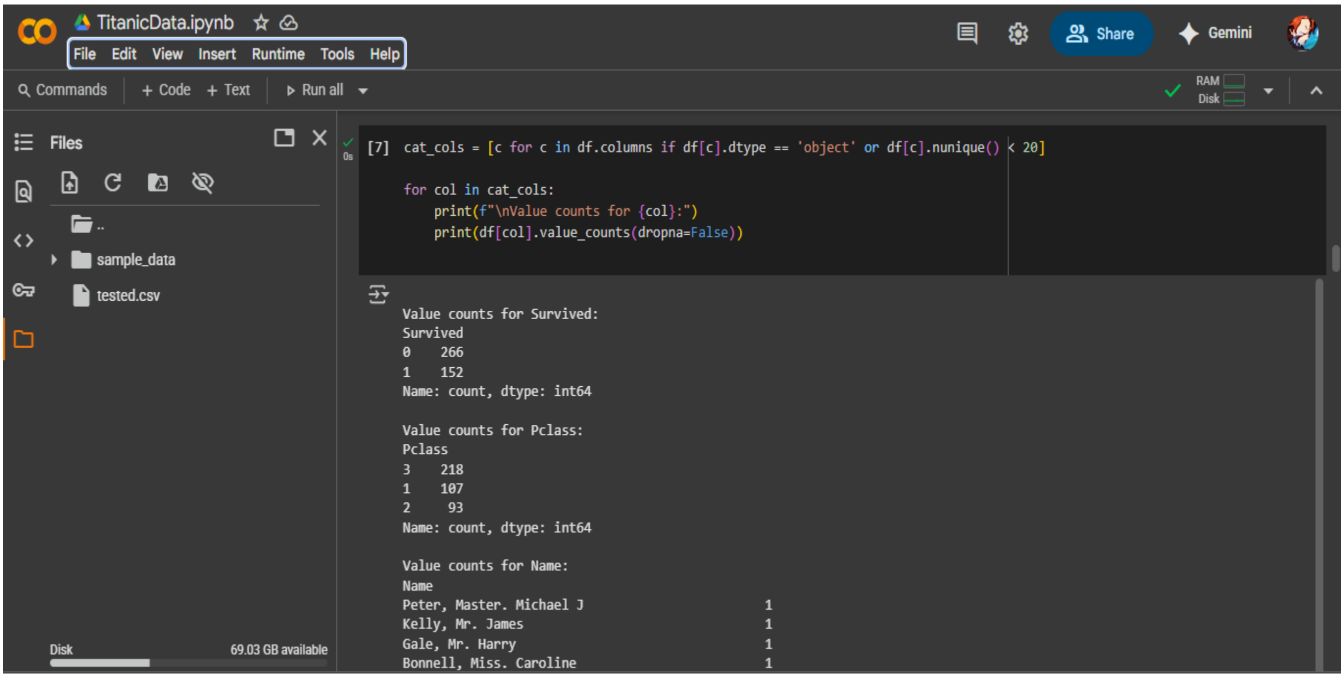
Here, we uploaded the CSV file on Google Colab and call the necessary data and import the libraries needed.



```
print("Shape of dataset:", df.shape)

print("\n--- Info ---")
print(df.info())

print("\n--- Describe ---")
print(df.describe(include='all').T)

print("\n--- Missing Values ---")
print(df.isnull().sum().sort_values(ascending=False))
```

```
Shape of dataset: (418, 12)

--- Info ---
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 418 entries, 0 to 417
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  418 non-null    int64
```

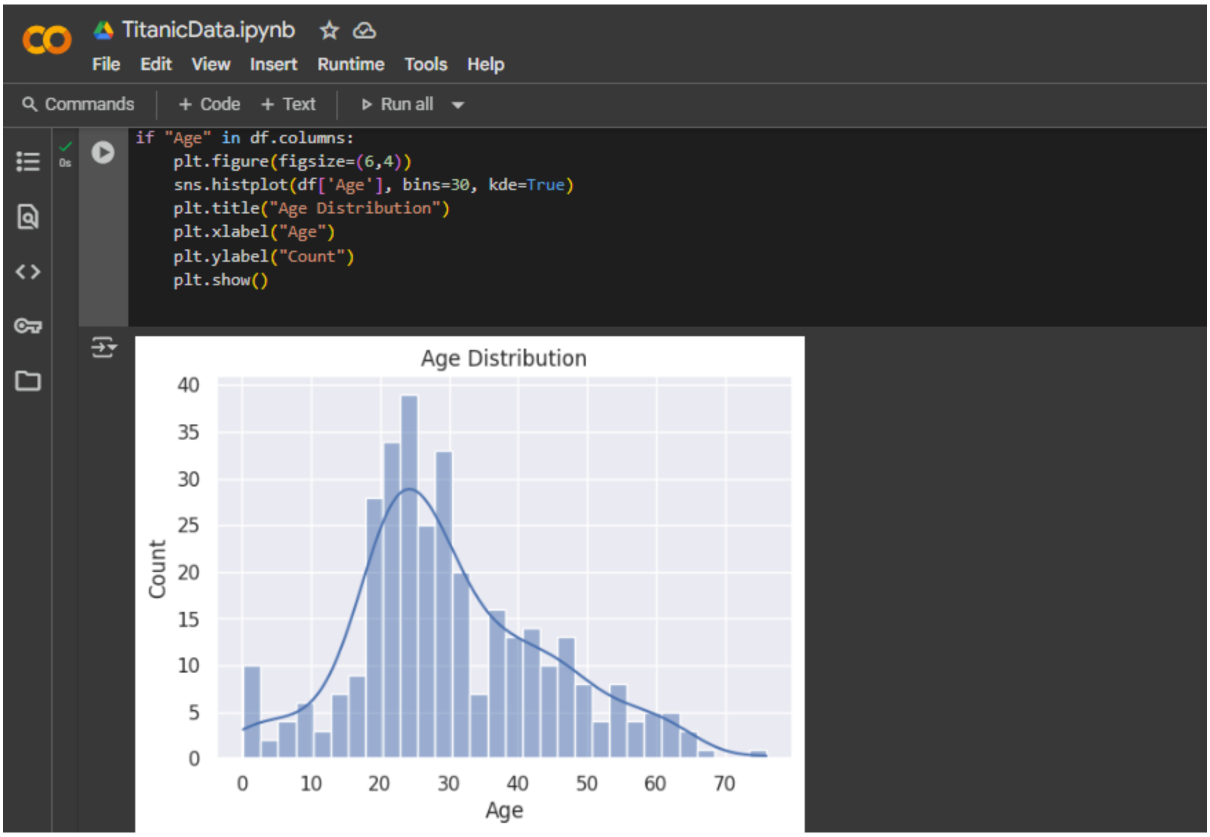
Here, We check the size of the data provided by using df.info() and df.describe() and df.shape() function.



```
[7] cat_cols = [c for c in df.columns if df[c].dtype == 'object' or df[c].nunique() < 20]

    for col in cat_cols:
        print(f"\nValue counts for {col}:")
        print(df[col].value_counts(dropna=False))
```

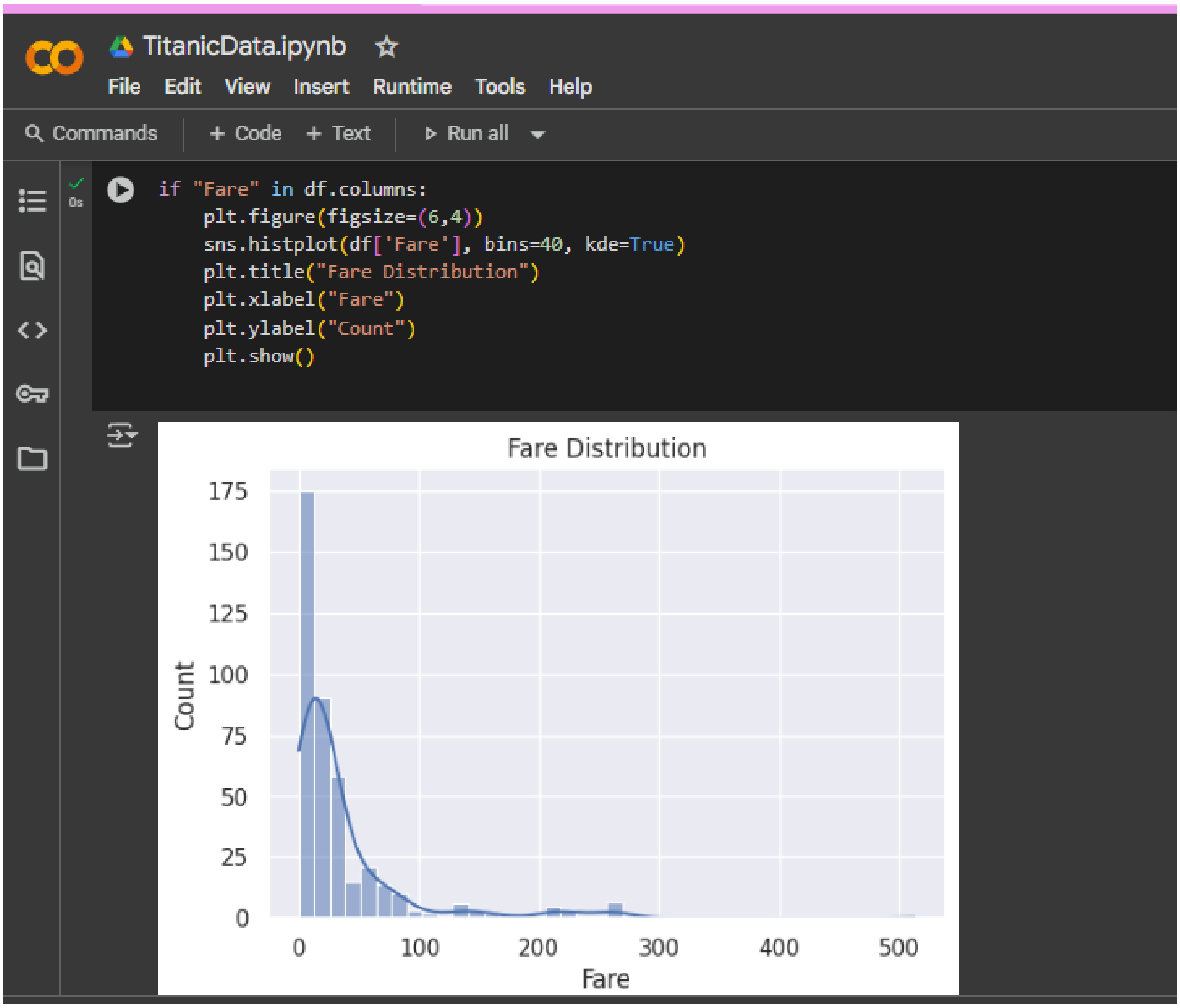
```
Value counts for Survived:
Survived
0    266
1    152
Name: count, dtype: int64

Value counts for Pclass:
Pclass
3    218
1    107
2     93
Name: count, dtype: int64

Value counts for Name:
Name
Peter, Master. Michael J                      1
Kelly, Mr. James                              1
Gale, Mr. Harry                               1
Bonnell, Miss. Caroline                       1
```

Here, we count the number of survivors of the titanic incident and the number one indicates that the passenger has survived.



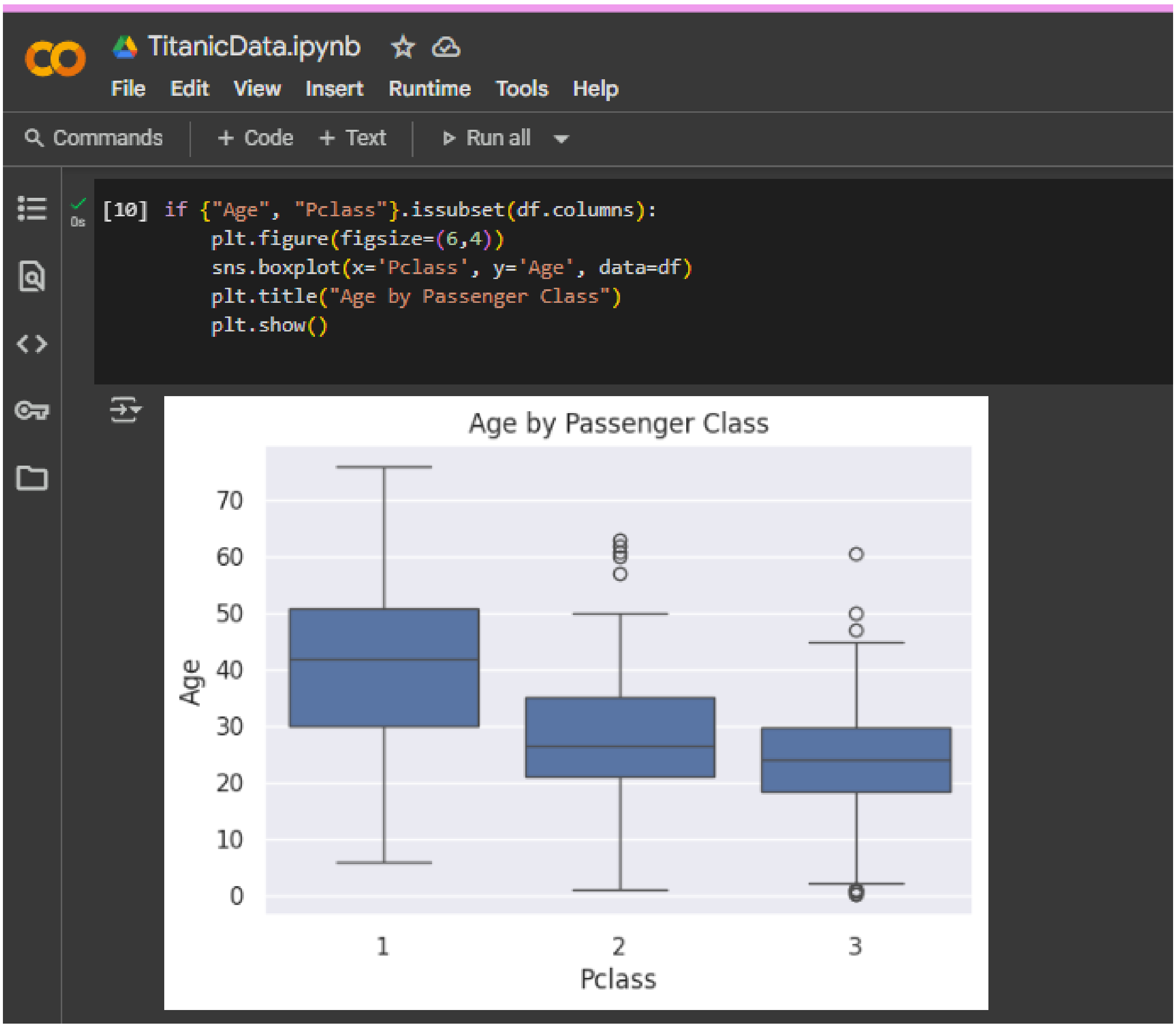
```
if "Age" in df.columns:
    plt.figure(figsize=(6,4))
    sns.histplot(df['Age'], bins=30, kde=True)
    plt.title("Age Distribution")
    plt.xlabel("Age")
    plt.ylabel("Count")
    plt.show()
```

The histogram shown above is the distribution of the number of passengers based on their age group.



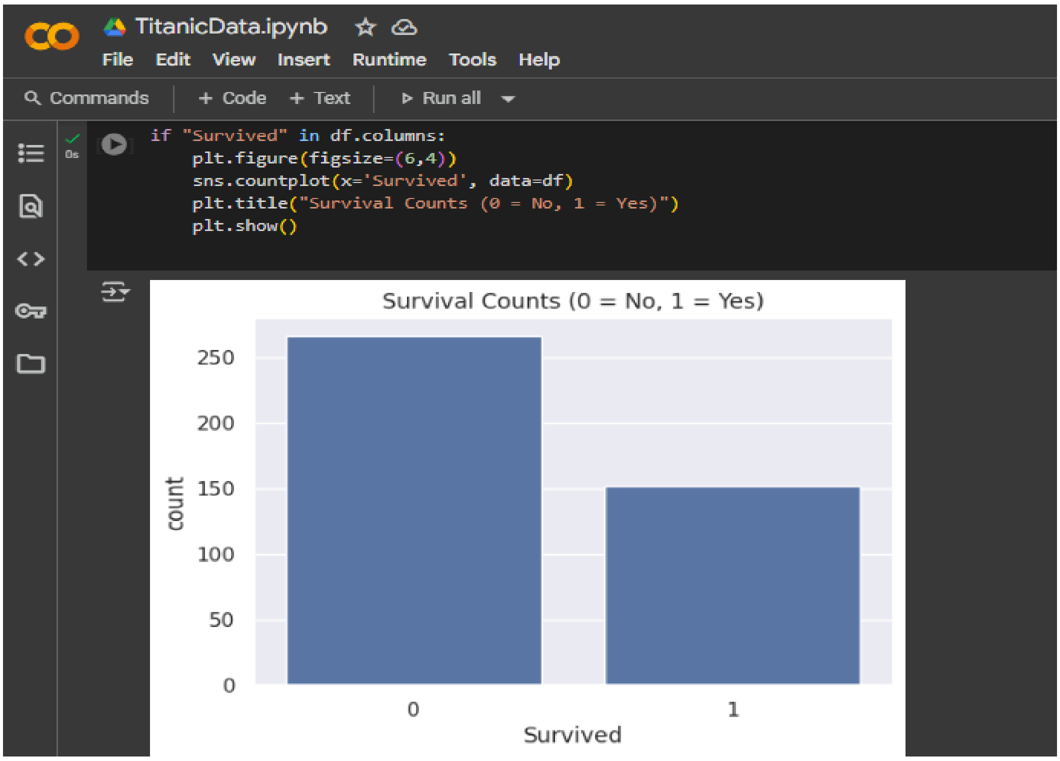
```
if "Fare" in df.columns:
    plt.figure(figsize=(6,4))
    sns.histplot(df['Fare'], bins=40, kde=True)
    plt.title("Fare Distribution")
    plt.xlabel("Fare")
    plt.ylabel("Count")
    plt.show()
```

The histogram shown above is the distribution of the price of passenger ticket.



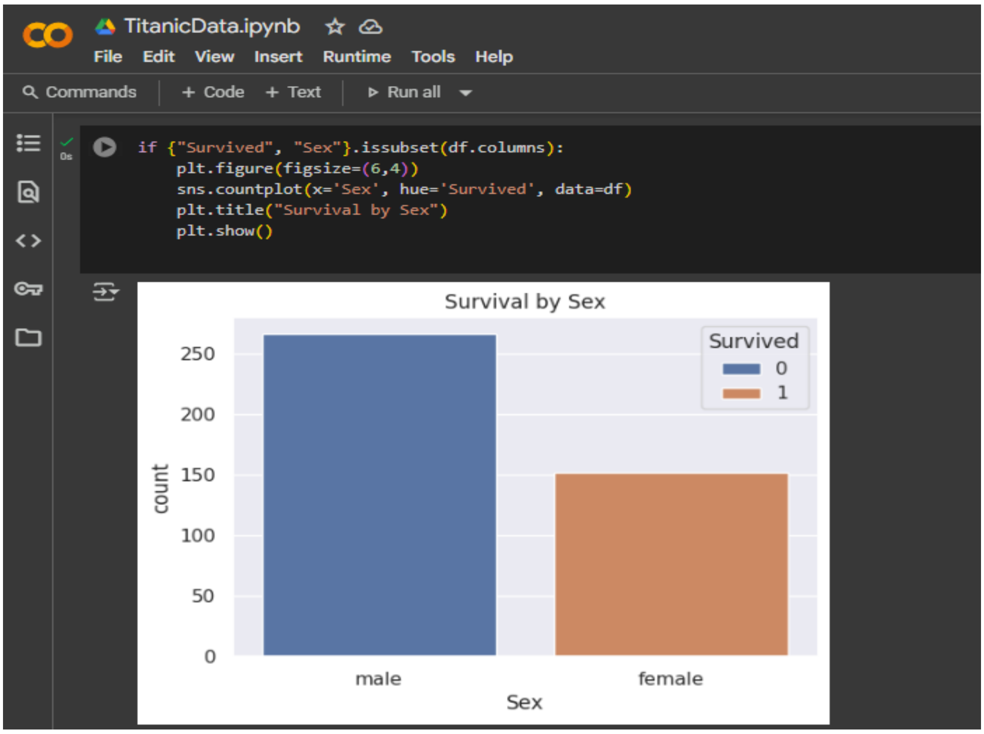
```
[10] if {"Age", "Pclass"}.issubset(df.columns):
         plt.figure(figsize=(6,4))
         sns.boxplot(x='Pclass', y='Age', data=df)
         plt.title("Age by Passenger Class")
         plt.show()
```

This boxplot shows us the classification of which all passengers of which age group took tickets of which class.



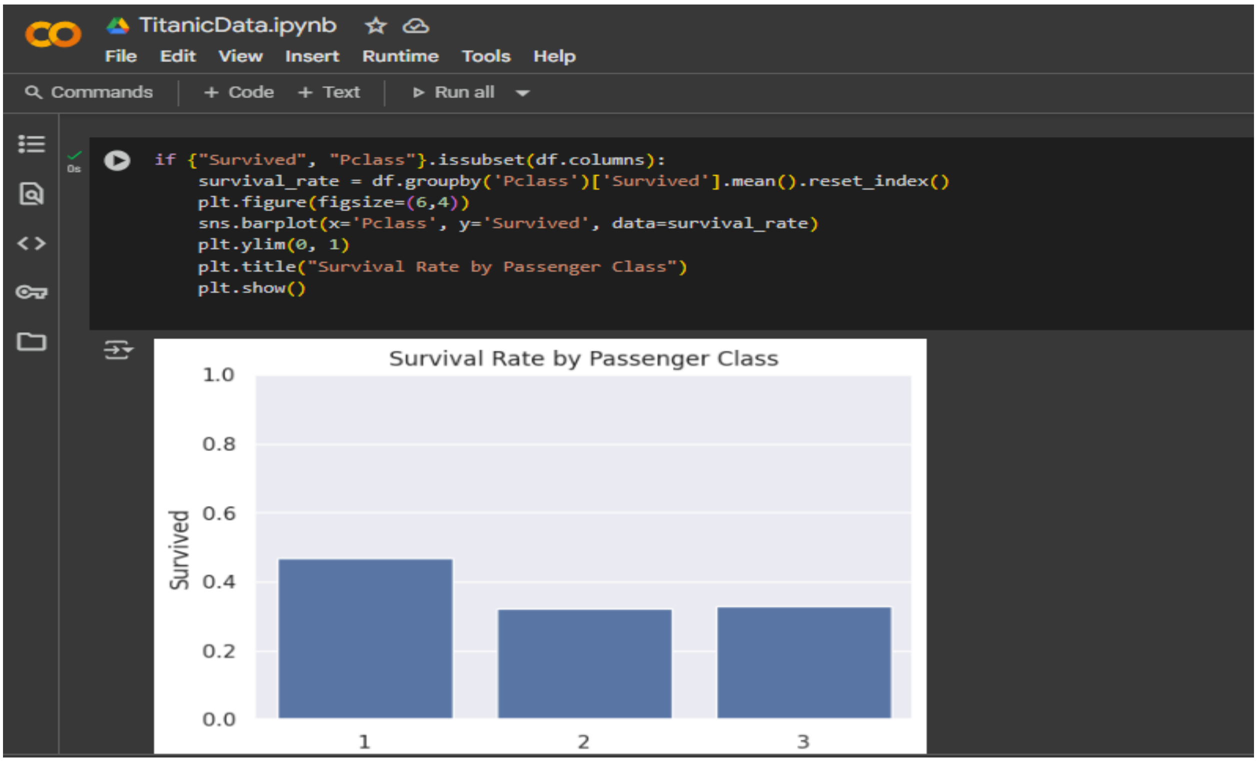
```
if "Survived" in df.columns:
    plt.figure(figsize=(6,4))
    sns.countplot(x='Survived', data=df)
    plt.title("Survival Counts (0 = No, 1 = Yes)")
    plt.show()
```

This is the visualization of the count of passengers who survived.



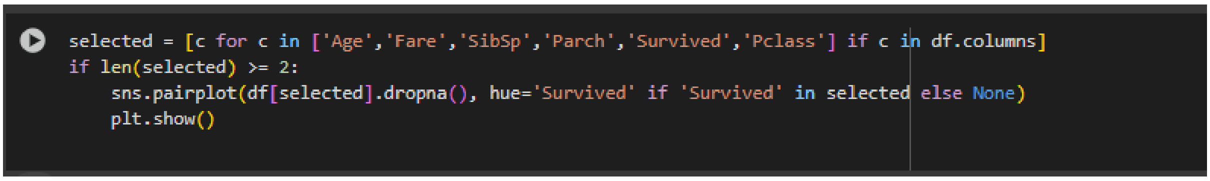
```
if {"Survived", "Sex"}.issubset(df.columns):
    plt.figure(figsize=(6,4))
    sns.countplot(x='Sex', hue='Survived', data=df)
    plt.title("Survival by Sex")
    plt.show()
```

This is the classification of the people who survived by gender.

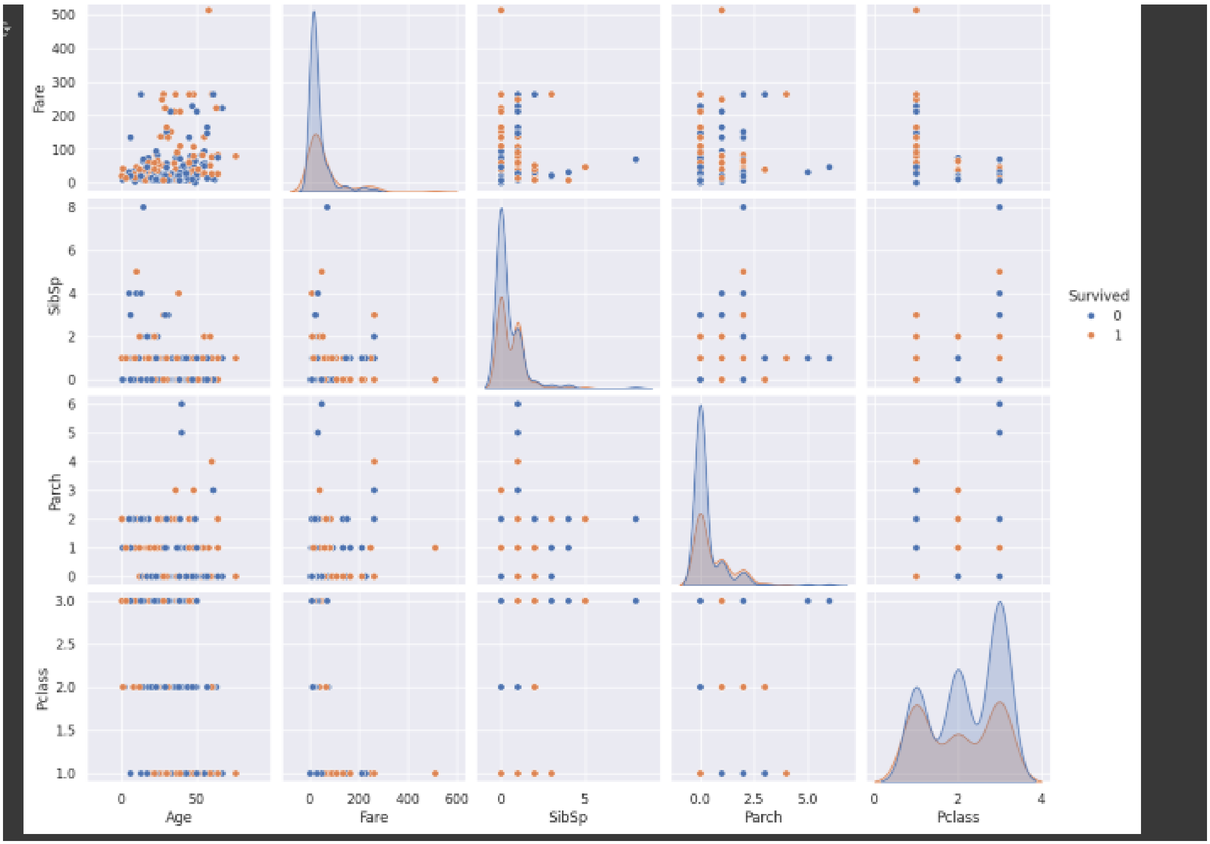


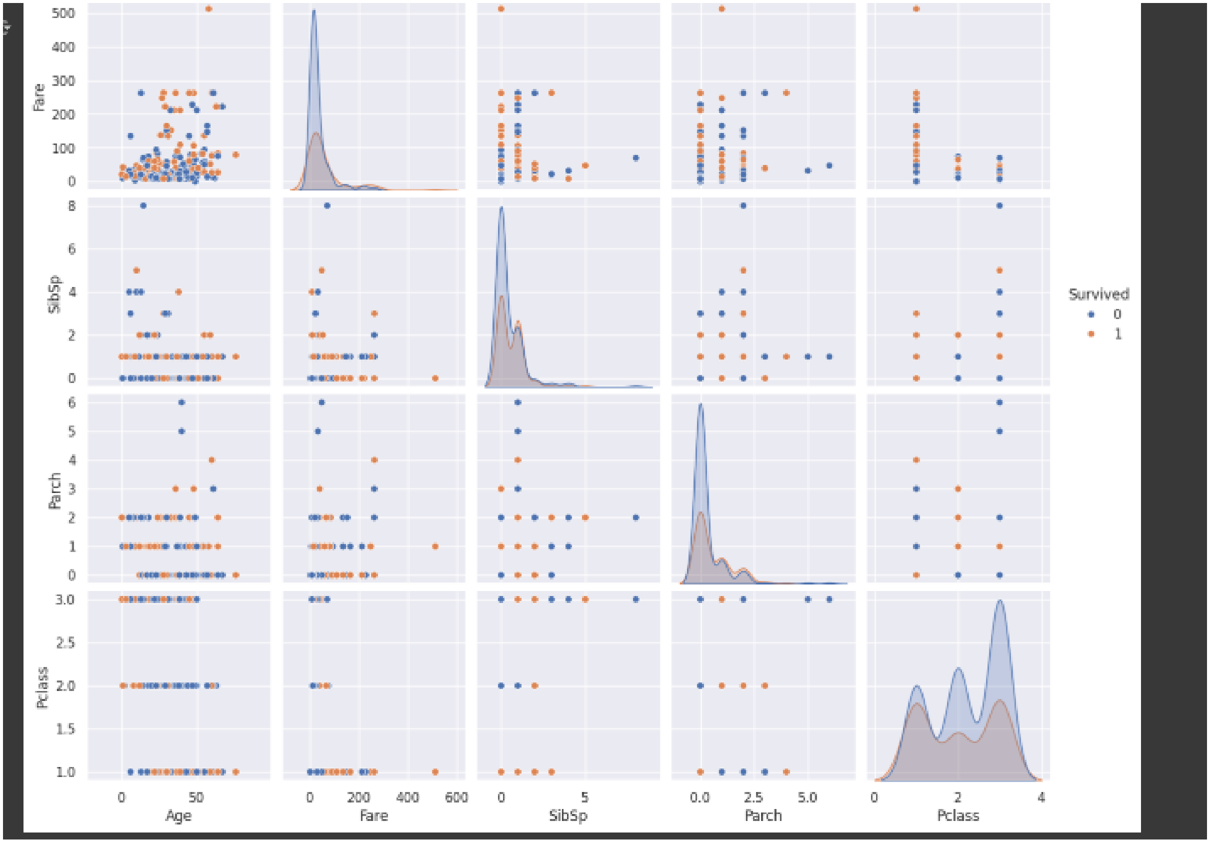
```
if {"Survived", "Pclass"}.issubset(df.columns):
    survival_rate = df.groupby('Pclass')['Survived'].mean().reset_index()
    plt.figure(figsize=(6,4))
    sns.barplot(x='Pclass', y='Survived', data=survival_rate)
    plt.ylim(0, 1)
    plt.title("Survival Rate by Passenger Class")
    plt.show()
```

This is the visualization of the survival rate of the passengers based on which Pclass they belong to.



```
selected = [c for c in ['Age','Fare','SibSp','Parch','Survived','Pclass'] if c in df.columns]
if len(selected) >= 2:
    sns.pairplot(df[selected].dropna(), hue='Survived' if 'Survived' in selected else None)
    plt.show()
```





This is a pair plot that shows all graphs together that gives us several insights. They will be mentioned in the summary below.

Summary:

-Age distribution peaks between 20–40 years old.

-Fare distribution is highly skewed; some very expensive tickets exist.

-Females had a higher survival rate than males.

-Higher class passengers had higher survival rates.