**Task 2:** Exploratory Data Analysis (EDA) Objective: Understand data using statistics and visualizations. Tools: Pandas, Matplotlib, Seaborn, Plotly**.**

1. **Generate summary statistics (mean, median, std, etc.)?**

**Answer:-**

import pandas as pd

# Load the dataset

df = pd.read\_csv("Titanic.csv")

# Handle missing values (optional but recommended before statistics)

df['Age'] = df['Age'].fillna(df['Age'].mean())

df['Fare'] = df['Fare'].fillna(df['Fare'].median())

# Generate summary statistics using describe()

print("Summary statistics (mean, std, min, max, etc.):\n")

print(df.describe())

# Additionally, you can print median separately

print("\nMedian of numerical columns:\n")

print(df.median(numeric\_only=True))

**Output:-**

Summary statistics (mean, std, min, max, etc.):

PassengerId Survived Pclass ... SibSp Parch Fare

count 891.000000 891.000000 891.000000 ... 891.000000 891.000000 891.000000

mean 446.000000 0.383838 2.308642 ... 0.523008 0.381594 32.204208

std 257.353842 0.486592 0.836071 ... 1.102743 0.806057 49.693429

min 1.000000 0.000000 1.000000 ... 0.000000 0.000000 0.000000

25% 223.500000 0.000000 2.000000 ... 0.000000 0.000000 7.910400

50% 446.000000 0.000000 3.000000 ... 0.000000 0.000000 14.454200

75% 668.500000 1.000000 3.000000 ... 1.000000 0.000000 31.000000

max 891.000000 1.000000 3.000000 ... 8.000000 6.000000 512.329200

[8 rows x 7 columns]

Median of numerical columns:

PassengerId 446.000000

Survived 0.000000

Pclass 3.000000

Age 29.699118

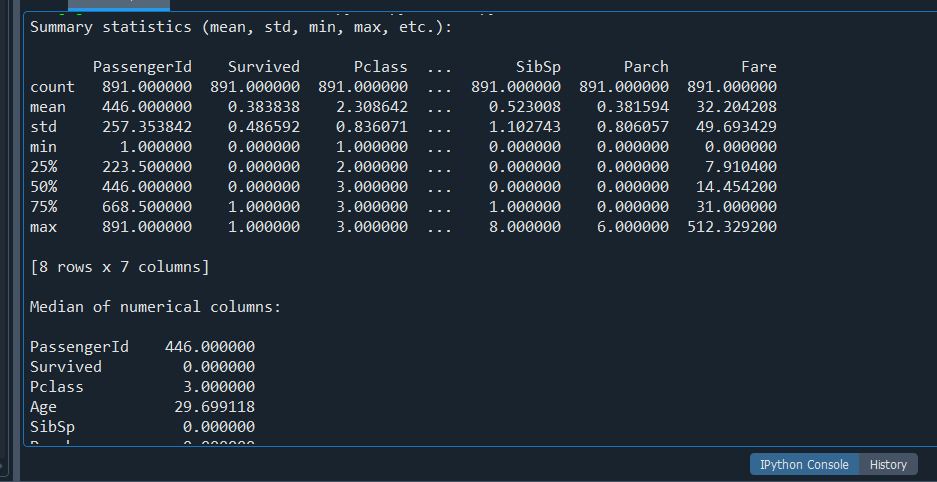
SibSp 0.000000

Parch 0.000000

Fare 14.454200

dtype: float64

**screenshot:-**



**2).Create histograms and boxplots for numeric features?**

**Answer:-**

import pandas as pd

import matplotlib.pyplot as plt

# Load dataset

df = pd.read\_csv("Titanic.csv")

# Handle missing values

df['Age'] = df['Age'].fillna(df['Age'].mean())

df['Fare'] = df['Fare'].fillna(df['Fare'].median())

# Select numeric columns

numeric\_cols = ['Age', 'Fare', 'SibSp', 'Parch']

# Create histograms

plt.figure(figsize=(12, 6))

for i, col in enumerate(numeric\_cols):

plt.subplot(2, 4, i + 1)

plt.hist(df[col], bins=20, color='skyblue', edgecolor='black')

plt.title(f'Histogram of {col}')

plt.xlabel(col)

plt.ylabel('Frequency')

# Create boxplots

for i, col in enumerate(numeric\_cols):

plt.subplot(2, 4, i + 5)

plt.boxplot(df[col])

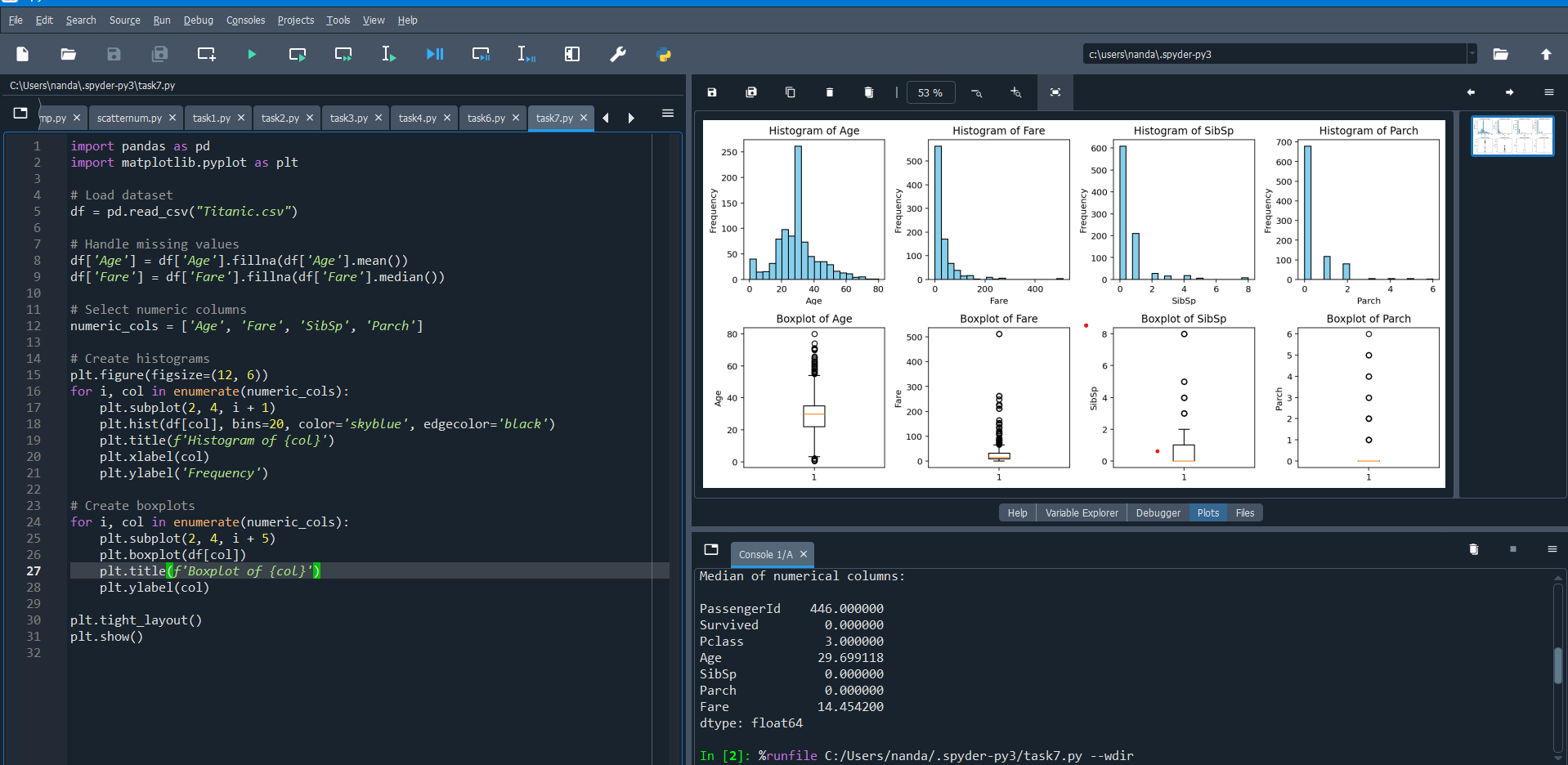
plt.title(f'Boxplot of {col}')

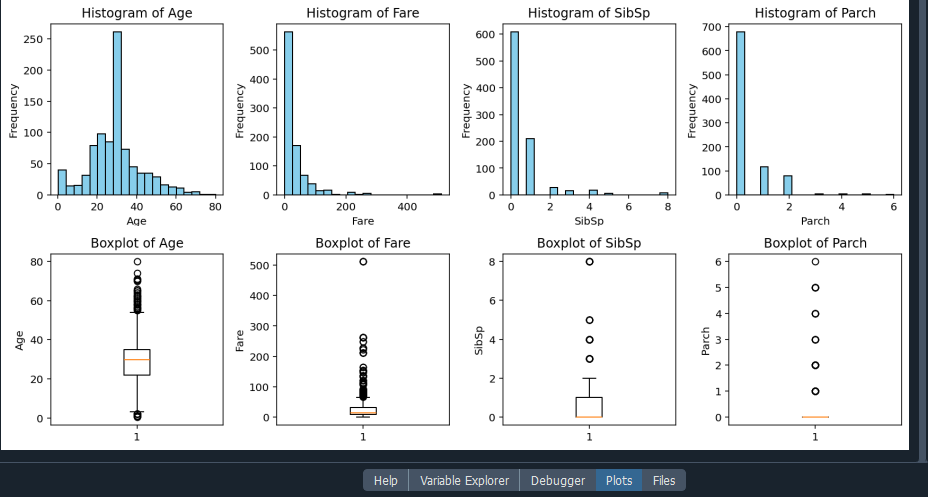
plt.ylabel(col)

plt.tight\_layout()

plt.show()

Output:-





1. **Use pairplot/correlation matrix for feature relationship?**

**Answer:-**

import pandas as pd

import matplotlib.pyplot as plt

import numpy as np

# Load dataset

df = pd.read\_csv("Titanic.csv")

# Fill missing values

df['Age'] = df['Age'].fillna(df['Age'].mean())

df['Fare'] = df['Fare'].fillna(df['Fare'].median())

df['Embarked'] = df['Embarked'].fillna(df['Embarked'].mode()[0])

df['Sex'] = df['Sex'].map({'male': 0, 'female': 1})

df['Embarked'] = df['Embarked'].map({'S': 0, 'C': 1, 'Q': 2})

# Drop unused columns

df = df.drop(columns=['Name', 'Ticket', 'Cabin'])

# Compute correlation matrix

corr\_matrix = df.corr(numeric\_only=True)

# Plot heatmap

plt.figure(figsize=(10, 6))

plt.imshow(corr\_matrix, cmap='coolwarm', interpolation='none')

plt.colorbar(label='Correlation Coefficient')

# Show labels

tick\_marks = np.arange(len(corr\_matrix.columns))

plt.xticks(tick\_marks, corr\_matrix.columns, rotation=45, ha='right')

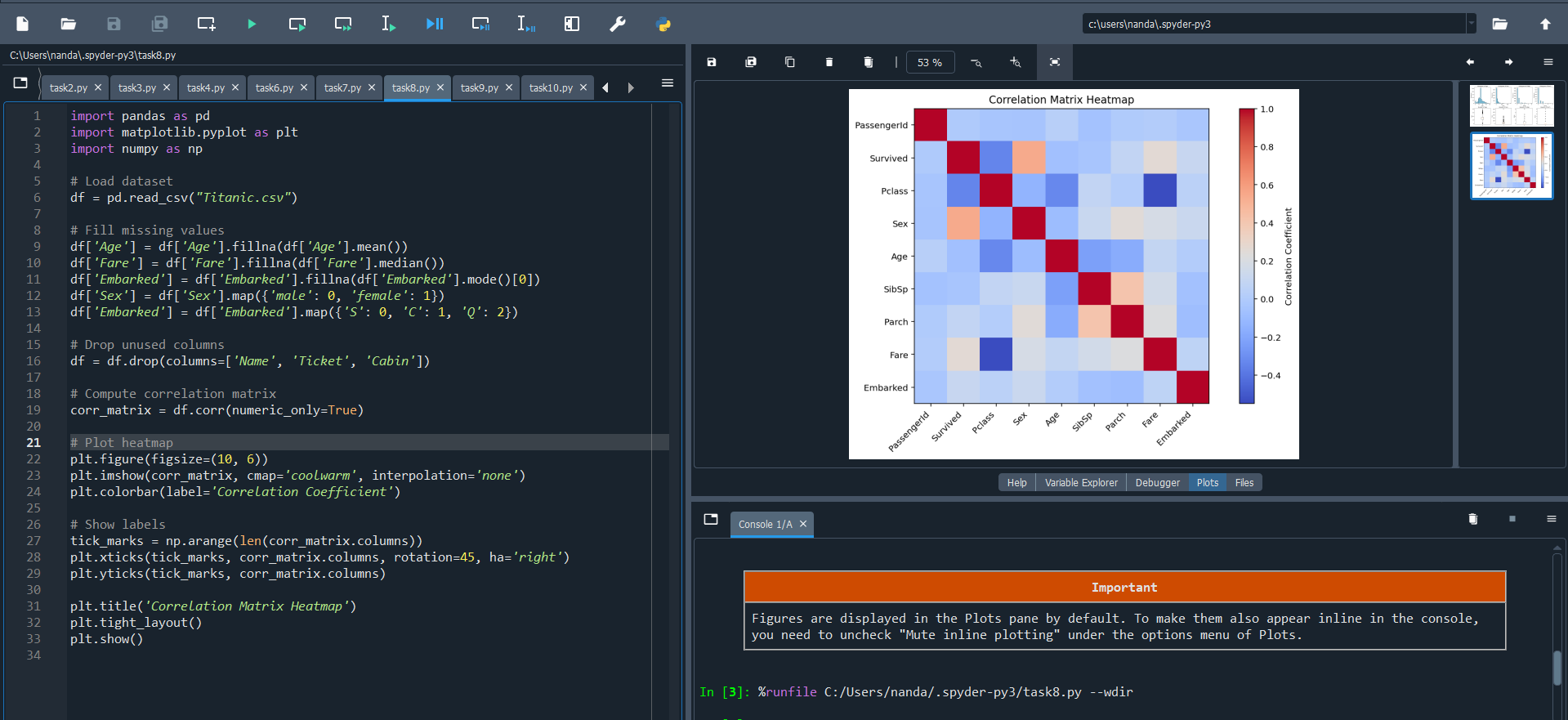
plt.yticks(tick\_marks, corr\_matrix.columns)

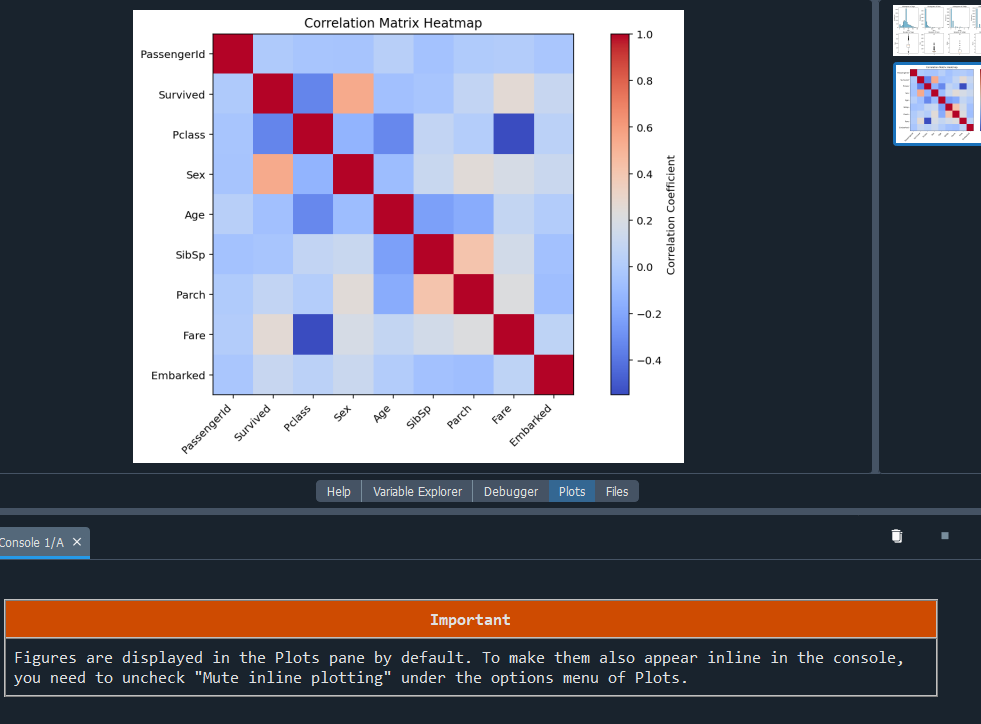
plt.title('Correlation Matrix Heatmap')

plt.tight\_layout()

plt.show()

**Output:-**

****

****

**4)** **Identify patterns, trends, or anomalies in the data?**

**Answer:-**

import pandas as pd

import matplotlib.pyplot as plt

# Load and clean dataset

df = pd.read\_csv("Titanic.csv")

df['Age'] = df['Age'].fillna(df['Age'].mean())

df['Fare'] = df['Fare'].fillna(df['Fare'].median())

df['Embarked'] = df['Embarked'].fillna(df['Embarked'].mode()[0])

df['Cabin'] = df['Cabin'].fillna('Unknown')

# Encode categorical features

df['Sex'] = df['Sex'].map({'male': 0, 'female': 1})

df['Embarked'] = df['Embarked'].map({'S': 0, 'C': 1, 'Q': 2})

# Drop non-essential columns

df = df.drop(columns=['Name', 'Ticket', 'Cabin'])

**# Pattern 1: Survival rate by Sex**

survival\_by\_sex = df.groupby('Sex')['Survived'].mean()

plt.figure()

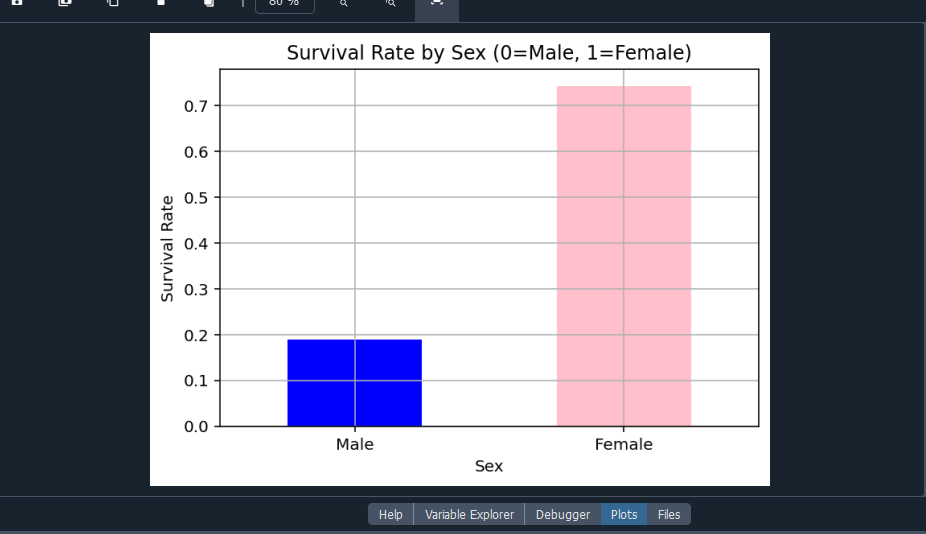
survival\_by\_sex.plot(kind='bar', color=['blue', 'pink'])

plt.title('Survival Rate by Sex (0=Male, 1=Female)')

plt.ylabel('Survival Rate')

plt.xticks([0, 1], ['Male', 'Female'], rotation=0)

plt.grid(True)



**# Pattern 2: Survival rate by Pclass**

survival\_by\_class = df.groupby('Pclass')['Survived'].mean()

plt.figure()

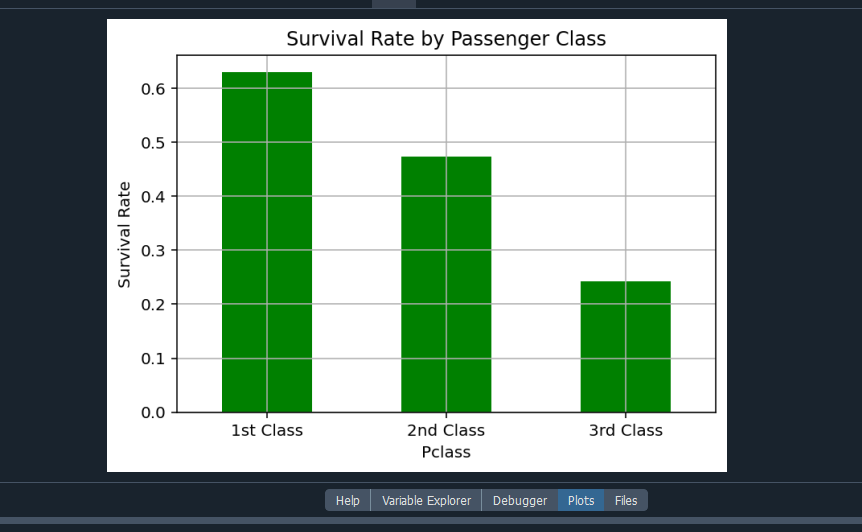
survival\_by\_class.plot(kind='bar', color='green')

plt.title('Survival Rate by Passenger Class')

plt.ylabel('Survival Rate')

plt.xticks([0, 1, 2], ['1st Class', '2nd Class', '3rd Class'], rotation=0)

plt.grid(True)



**# Trend/Anomaly: Fare distribution**

plt.figure()

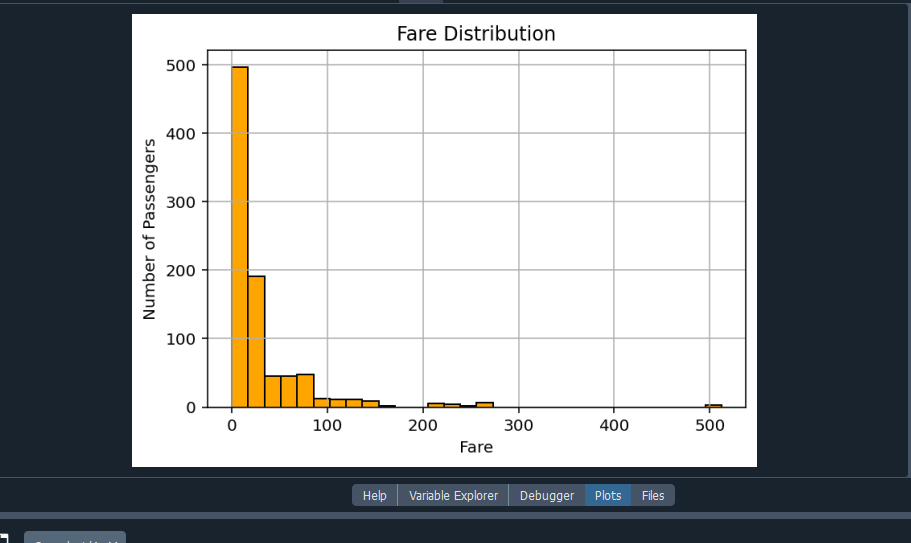
plt.hist(df['Fare'], bins=30, color='orange', edgecolor='black')

plt.title('Fare Distribution')

plt.xlabel('Fare')

plt.ylabel('Number of Passengers')

plt.grid(True)



**# Trend/Anomaly: Boxplot of Age to detect outliers**

plt.figure()

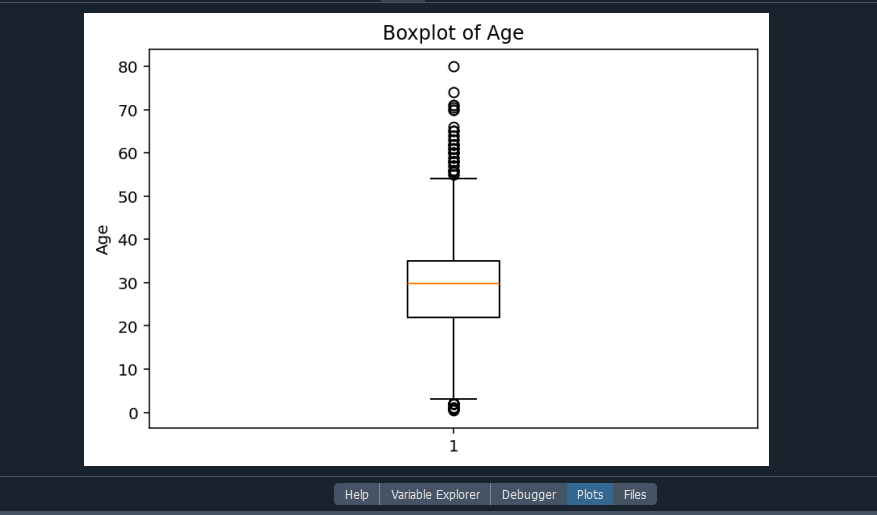
plt.boxplot(df['Age'])

plt.title('Boxplot of Age')

plt.ylabel('Age')

plt.tight\_layout()

plt.show()



**5)Make basic feature-level inferences from visuals?**

**Answer:-**

import pandas as pd

import matplotlib.pyplot as plt

# Load and preprocess the dataset

df = pd.read\_csv("Titanic.csv")

# Fill missing values

df['Age'] = df['Age'].fillna(df['Age'].mean())

df['Fare'] = df['Fare'].fillna(df['Fare'].median())

df['Embarked'] = df['Embarked'].fillna(df['Embarked'].mode()[0])

df['Cabin'] = df['Cabin'].fillna('Unknown')

# Encode categorical variables

df['Sex'] = df['Sex'].map({'male': 0, 'female': 1})

df['Embarked'] = df['Embarked'].map({'S': 0, 'C': 1, 'Q': 2})

# Drop irrelevant columns

df = df.drop(columns=['Name', 'Ticket', 'Cabin'])

# Histograms for numeric features

df[['Age', 'Fare', 'SibSp', 'Parch']].hist(bins=20, figsize=(10, 6))

plt.suptitle("Histograms of Numeric Features")

plt.tight\_layout()

plt.show()

# Boxplots to identify outliers

numeric\_cols = ['Age', 'Fare', 'SibSp', 'Parch']

plt.figure(figsize=(10, 8))

for i, col in enumerate(numeric\_cols):

plt.subplot(2, 2, i + 1)

plt.boxplot(df[col])

plt.title(f'Boxplot of {col}')

plt.ylabel(col)

plt.tight\_layout()

plt.show()

# Correlation matrix

correlation = df.corr()

print("Correlation Matrix:\n", correlation)

# Correlation heatmap (matplotlib only)

plt.figure(figsize=(8, 6))

plt.imshow(correlation, cmap='coolwarm', interpolation='none', aspect='auto')

plt.colorbar()

plt.xticks(range(len(correlation)), correlation.columns, rotation=90)

plt.yticks(range(len(correlation)), correlation.columns)

plt.title("Correlation Matrix Heatmap")

plt.tight\_layout()

plt.show()

(note:- From the visual analysis of the Titanic dataset, several key inferences can be drawn. Genderplayed a significant role in survival, as females had a much higher survival rate compared to males. Passenger class was also a strong predictor, with first-class passengers more likely to survive than those in second or third class, suggesting socio-economic status influenced rescue priority. The Fare distribution revealed a right-skewed pattern, indicating that while most passengers paid lower fares, a few paid exceptionally high amounts—possibly for luxury cabins. Boxplots of features like Age and Fare exposed potential outliers, with some individuals being significantly older or having paid unusually high fares. These insights highlight how certain features like sex, class, age, and fare relate closely to survival and data quality.)

**Output:-**

