Q2.Perform data cleaning and exploratory data analysis (EDA) on a dataset of your choice, such as the Titanic dataset from Kaggle. Explore the relationships between variables and identify patterns and trends in the data.
Sample Dataset :- https://www.kaggle.com/c/titanic/data

Code-

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

# Generate a synthetic dataset similar to Titanic
np.random.seed(42)
n_samples = 891

synthetic_data = pd.DataFrame({
    'survived': np.random.randint(0, 2, size=n_samples),
    'pclass': np.random.randint(1, 4, size=n_samples),
    'sex': np.random.choice(['male', 'female'], size=n_samples),
    'age': np.random.uniform(1, 80, size=n_samples),
    'sibsp': np.random.randint(0, 5, size=n_samples),
    'parch': np.random.randint(0, 6, size=n_samples),
    'fare': np.random.uniform(10, 300, size=n_samples),
    'embarked': np.random.choice(['C', 'Q', 'S'], size=n_samples)
})

# Data Cleaning

# Check for missing values
missing_values = synthetic_data.isnull().sum()
print("Missing values:\n", missing_values)

# Check for duplicates
duplicates = synthetic_data.duplicated().sum()
print("Number of duplicates:", duplicates)

# Check data types
data_types = synthetic_data.dtypes
print("Data types:\n", data_types)
```

```
# Exploratory Data Analysis (EDA)

# Descriptive statistics for numerical columns
desc_stats = synthetic_data.describe()
print("Descriptive statistics:\n", desc_stats)

# Distribution of categorical variables
plt.figure(figsize=(16, 4))

# Distribution of sex
plt.subplot(1, 3, 1)
sns.countplot(data=synthetic_data, x='sex')
plt.title('Distribution of Sex')

# Distribution of pclass
plt.subplot(1, 3, 2)
sns.countplot(data=synthetic_data, x='pclass')
plt.title('Distribution of Pclass')

# Distribution of embarked
plt.subplot(1, 3, 3)
sns.countplot(data=synthetic_data, x='embarked')
plt.title('Distribution of Embarked')

plt.tight_layout()
plt.show()

# Correlation analysis
numeric_data = synthetic_data.select_dtypes(include=[np.number])
corr_matrix = numeric_data.corr()

plt.figure(figsize=(10, 8))
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', vmin=-1, vmax=1)
plt.title('Correlation Matrix')
plt.show()

# Survival analysis
survival_sex = synthetic_data.groupby('sex')['survived'].mean()
```

```
print("Survival rate by sex:\n", survival_sex)

survival_pclass = synthetic_data.groupby('pclass')['survived'].mean()
print("Survival rate by pclass:\n", survival_pclass)

survival_embarked = synthetic_data.groupby('embarked')['survived'].mean()
print("Survival rate by embarked:\n", survival_embarked)
```