Task 2

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

Step 1: Import Required Libraries

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

Step 2: Load the Pizza Dataset

```
data=pd.read_csv("/content/pizza - pizza.csv")
data.head(5)
```

	order_details_id	order_id	pizza_id	quantity	order_date	order_time	unit_price	total_price	pizza_size	pizza_category	pizza_ingredients	pizza_name
0	1	1	hawaiian_m	1	1/1/2015	11:38:36	13.25	13.25	M	Classic	Sliced Ham, Pineapple, Mozzarella Cheese	The Hawaiian Pizza
1	2	2	classic_dlx_m	1	1/1/2015	11:57:40	16.00	16.00	M	Classic	Pepperoni, Mushrooms, Red Onions, Red Peppers,...	The Classic Deluxe Pizza
2	3	2	five_cheese_l	1	1/1/2015	11:57:40	18.50	18.50	L	Veggie	Mozzarella Cheese, Provolone Cheese, Smoked Go...	The Five Cheese Pizza
3	4	2	ital_supr_l	1	1/1/2015	11:57:40	20.75	20.75	L	Supreme	Calabrese Salami, Capocollo, Tomatoes, Red Oni...	The Italian Supreme Pizza
4	5	2	mexicana_m	1	1/1/2015	11:57:40	16.00	16.00	M	Veggie	Tomatoes, Red Peppers, Jalapeno Peppers, Red O...	The Mexicana Pizza

Step 3: Basic Data Exploration

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48620 entries, 0 to 48619
Data columns (total 12 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   order_details_id  48620 non-null  int64
 1   order_id          48620 non-null  int64
 2   pizza_id          48620 non-null  object
 3   quantity          48620 non-null  int64
 4   order_date        48620 non-null  object
 5   order_time        48620 non-null  object
 6   unit_price        48620 non-null  float64
 7   total_price       48620 non-null  float64
 8   pizza_size        48620 non-null  object
 9   pizza_category    48620 non-null  object
 10  pizza_ingredients 48620 non-null  object
 11  pizza_name        48620 non-null  object
dtypes: float64(2), int64(3), object(7)
memory usage: 4.5+ MB
```

```
data.value_counts()
```

order_details_id	order_id	pizza_id	quantity	order_date	order_time	unit_price	total_price	pizza_size	pizza_category
48620	21350	bbq_ckn_s	1	12/31/2015	23:02:05	12.75	12.75	S	Chicken
1	1	hawaiian_m	1	1/1/2015	11:38:36	13.25	13.25	M	Classic
2	2	classic_dlx_m	1	1/1/2015	11:57:40	16.00	16.00	M	Classic
3	2	five_cheese_l	1	1/1/2015	11:57:40	18.50	18.50	L	Veggie

```
data.duplicated().sum()
```

```
np.int64(0)
```

```
data.describe()
```

	order_details_id	order_id	quantity	unit_price	total_price
count	48620.000000	48620.000000	48620.000000	48620.000000	48620.000000
mean	24310.500000	10701.479761	1.019622	16.494132	16.821474
std	14035.529381	6180.119770	0.143077	3.621789	4.437398
min	1.000000	1.000000	1.000000	9.750000	9.750000
25%	12155.750000	5337.000000	1.000000	12.750000	12.750000
50%	24310.500000	10682.500000	1.000000	16.500000	16.500000
75%	36465.250000	16100.000000	1.000000	20.250000	20.500000
max	48620.000000	21350.000000	4.000000	35.950000	83.000000

```
data.columns
```

```
Index(['order_details_id', 'order_id', 'pizza_id', 'quantity', 'order_date',
       'order_time', 'unit_price', 'total_price', 'pizza_size',
       'pizza_category', 'pizza_ingredients', 'pizza_name'],
      dtype='object')
```

```
data.tail(5)
```

	order_details_id	order_id	pizza_id	quantity	order_date	order_time	unit_price	total_price	pizza_size	pizza_category	pizza_ingredients	pizza_name
48615	48616	21348	ckn_alfredo_m	1	12/31/2015	21:23:10	16.75	16.75	M	Chicken	Chicken, Red Onions, Red Peppers, Mushrooms, A...	The Chicken Alfredo Pizza
48616	48617	21348	four_cheese_l	1	12/31/2015	21:23:10	17.95	17.95	L	Veggie	Ricotta Cheese, Gorgonzola Piccante Cheese, Mo...	The Four Cheese Pizza
48617	48618	21348	napolitana_s	1	12/31/2015	21:23:10	12.00	12.00	S	Classic	Tomatoes, Anchovies, Green Olives, Red Onions,...	The Napolitana Pizza
48618	48619	21349	mexicana_l	1	12/31/2015	22:09:54	20.25	20.25	L	Veggie	Tomatoes, Red Peppers, Jalapeno Peppers, Red O...	The Mexicana Pizza
48619	48620	21350	bbq_ckn_s	1	12/31/2015	23:02:05	12.75	12.75	S	Chicken	Barbecued Chicken, Red Peppers, Green Peppers,...	The Barbecue Chicken Pizza

Step 4: Data Cleaning

```
data.isnull().sum()
```

	0
order_details_id	0
order_id	0
pizza_id	0
quantity	0
order_date	0
order_time	0
unit_price	0
total_price	0
pizza_size	0
pizza_category	0
pizza_ingredients	0
pizza_name	0

dtype: int64

```
data.dropna(inplace=True)
```

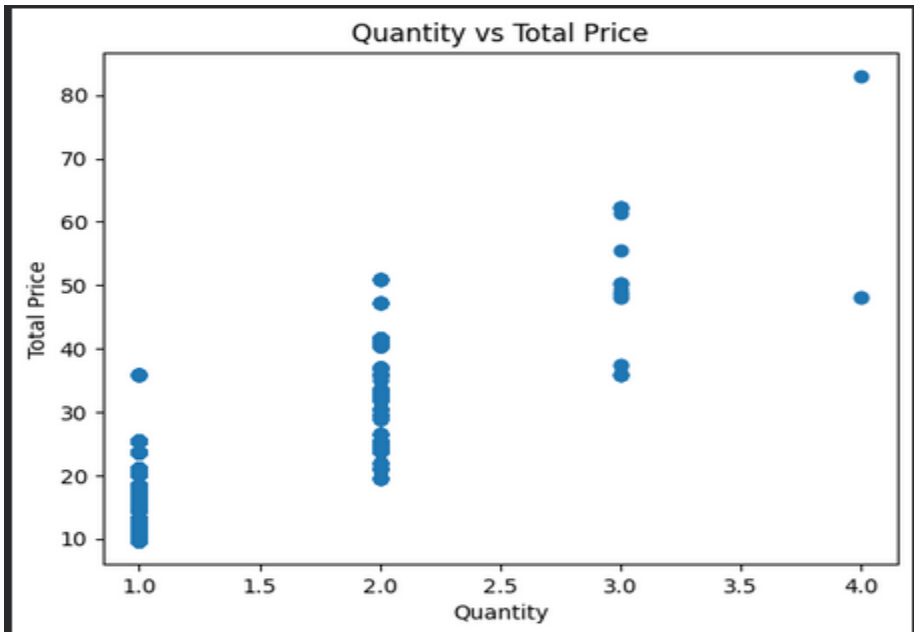
```
data.drop_duplicates(inplace=True)
```

```
data['order_date'] = pd.to_datetime(data['order_date'])
data['order_time'] = pd.to_datetime(data['order_time'], format='%H:%M:%S')
data.head(5)
```

	order_details_id	order_id	pizza_id	quantity	order_date	order_time	unit_price	total_price	pizza_size	pizza_category	pizza_ingredients	pizza_name
0	1	1	hawaiian_m	1	2015-01-01	1900-01-01 11:38:36	13.25	13.25	M	Classic	Sliced Ham, Pineapple, Mozzarella Cheese	The Hawaiian Pizza
1	2	2	classic_dlx_m	1	2015-01-01	1900-01-01 11:57:40	16.00	16.00	M	Classic	Pepperoni, Mushrooms, Red Onions, Red Peppers,...	The Classic Deluxe Pizza
2	3	2	five_cheese_l	1	2015-01-01	1900-01-01 11:57:40	18.50	18.50	L	Veggie	Mozzarella Cheese, Provolone Cheese, Smoked Go...	The Five Cheese Pizza

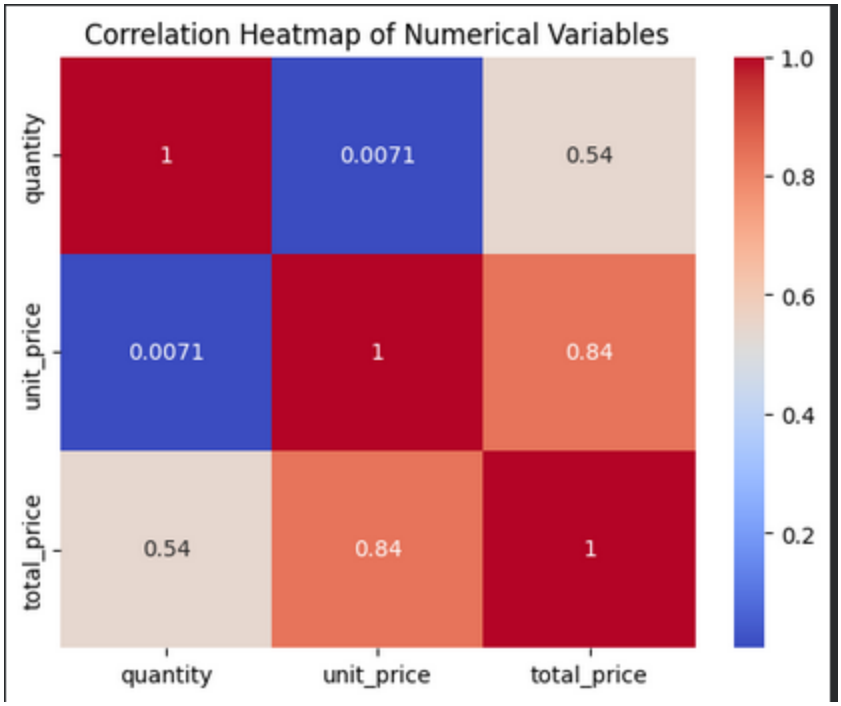
Step 5: Exploratory Data Analysis (EDA)

```
plt.figure()
plt.scatter(data['quantity'], data['total_price'])
plt.xlabel("Quantity")
plt.ylabel("Total Price")
plt.title("Quantity vs Total Price")
plt.show()
```

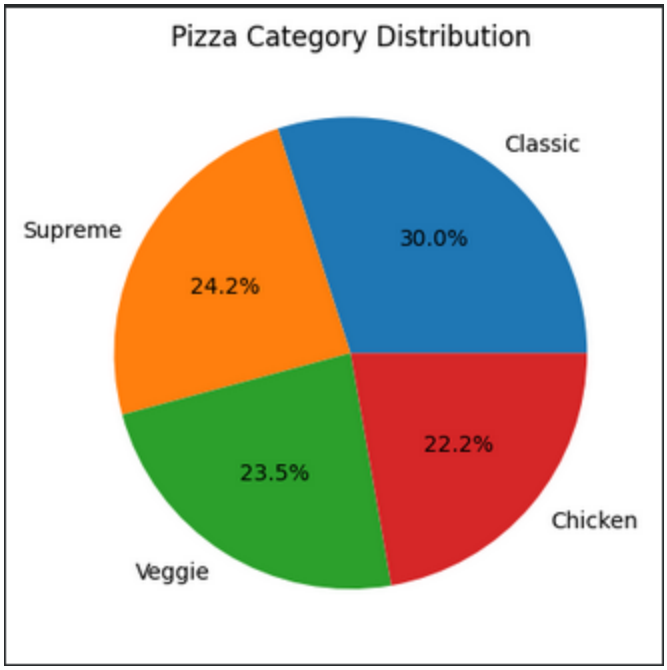



```
data['hour'] = data['order_time'].dt.hour

plt.figure()
data['hour'].value_counts().sort_index().plot()
plt.xlabel("Hour of Day")
plt.ylabel("Number of Orders")
plt.title("Orders by Time of Day")
plt.show()
```

```python
numeric_cols = data[['quantity', 'unit_price', 'total_price']]

corr = numeric_cols.corr()

plt.figure()
sns.heatmap(corr, annot=True, cmap='coolwarm')

plt.title("Correlation Heatmap of Numerical Variables")
plt.show()
```




```python
category_counts = data['pizza_category'].value_counts()

plt.figure()
plt.pie(category_counts, labels=category_counts.index, autopct='%1.1f%%')
plt.title("Pizza Category Distribution")
plt.show()
```




```
grouped_data = [
    group['total_price'].values
    for name, group in data.groupby('pizza_size')
]

# Get pizza size labels
labels = data.groupby('pizza_size').groups.keys()

# Create box plot
plt.figure()
plt.boxplot(grouped_data, labels=labels)

plt.xlabel("Pizza Size")
plt.ylabel("Total Price")
plt.title("Box Plot of Total Price by Pizza Size (Using GroupBy)")
plt.show()
```

```
/tmp/ipython-input-899611980.py:11: MatplotlibDeprecationWarning: The 'labels' parameter of boxplot()
  plt.boxplot(grouped_data, labels=labels)
```