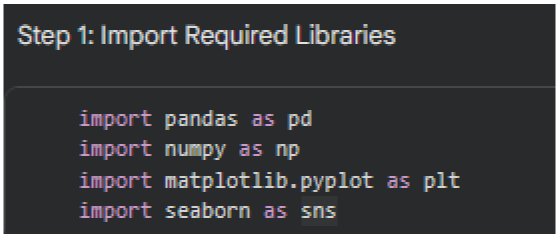
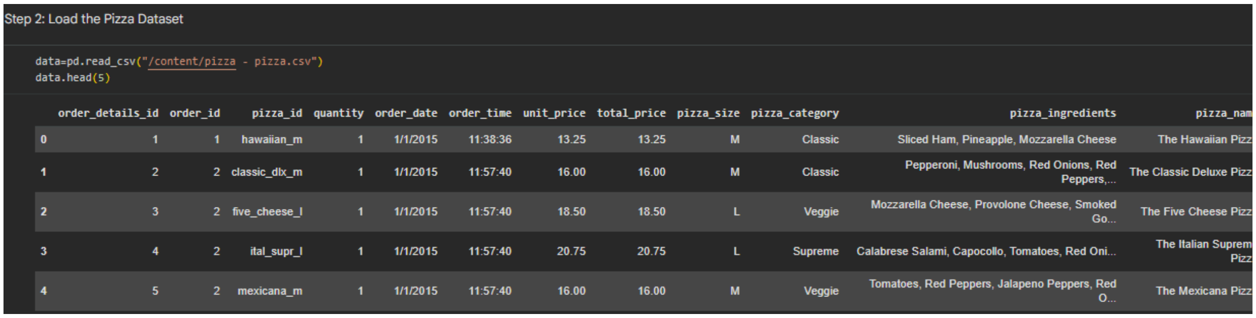
Task 5

Analyze traffic accident data to identify patterns related to road conditions, weather, and time of day. Visualize accident hotspots and contributing factors.



Step 1: Import Required Libraries

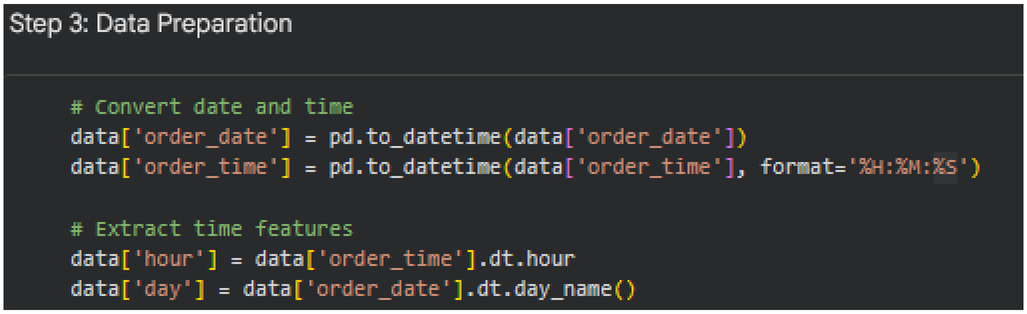
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```



Step 2: Load the Pizza Dataset

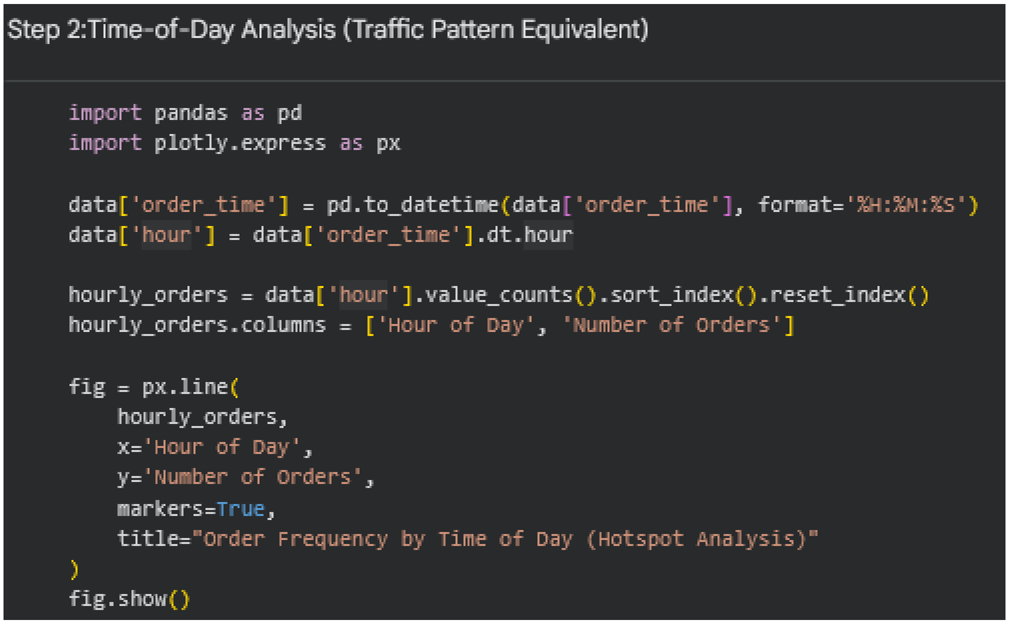
```
data=pd.read_csv("/content/pizza - pizza.csv")
data.head(5)
```

	order_details_id	order_id	pizza_id	quantity	order_date	order_time	unit_price	total_price	pizza_size	pizza_category	pizza_ingredients	pizza_nam
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 Pizz
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 Pizz
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 Pizz
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 Suprem Pizz
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 Pizz



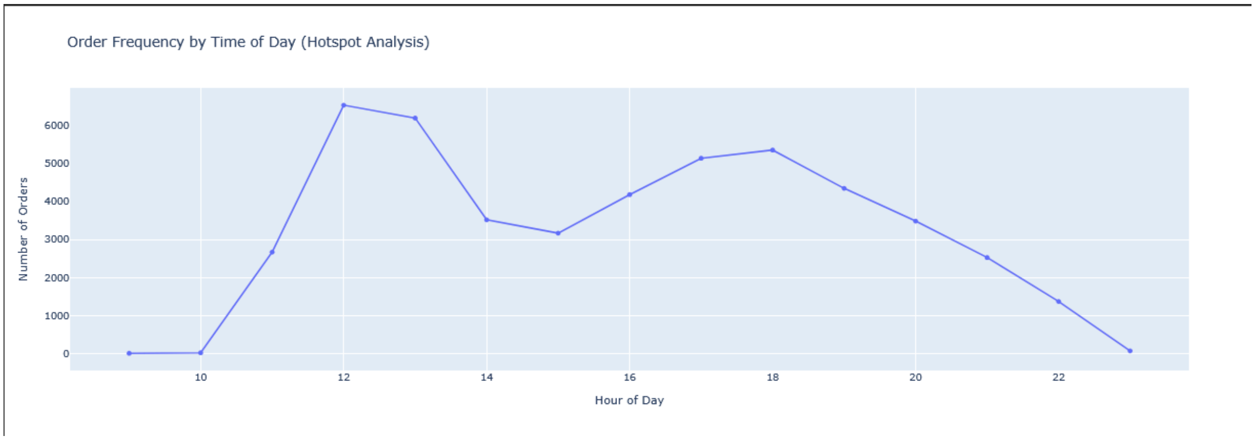
Step 3: Data Preparation

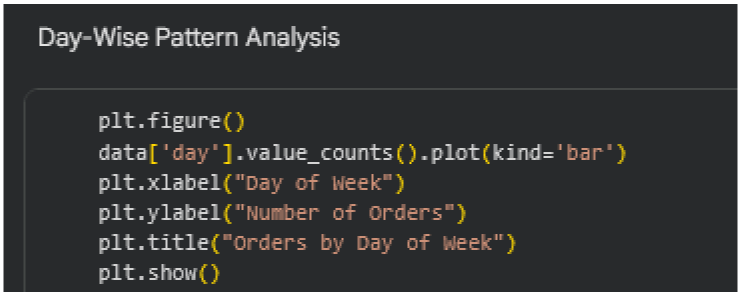
```
# Convert date and time
data['order_date'] = pd.to_datetime(data['order_date'])
data['order_time'] = pd.to_datetime(data['order_time'], format='%H:%M:%S')

# Extract time features
data['hour'] = data['order_time'].dt.hour
data['day'] = data['order_date'].dt.day_name()
```



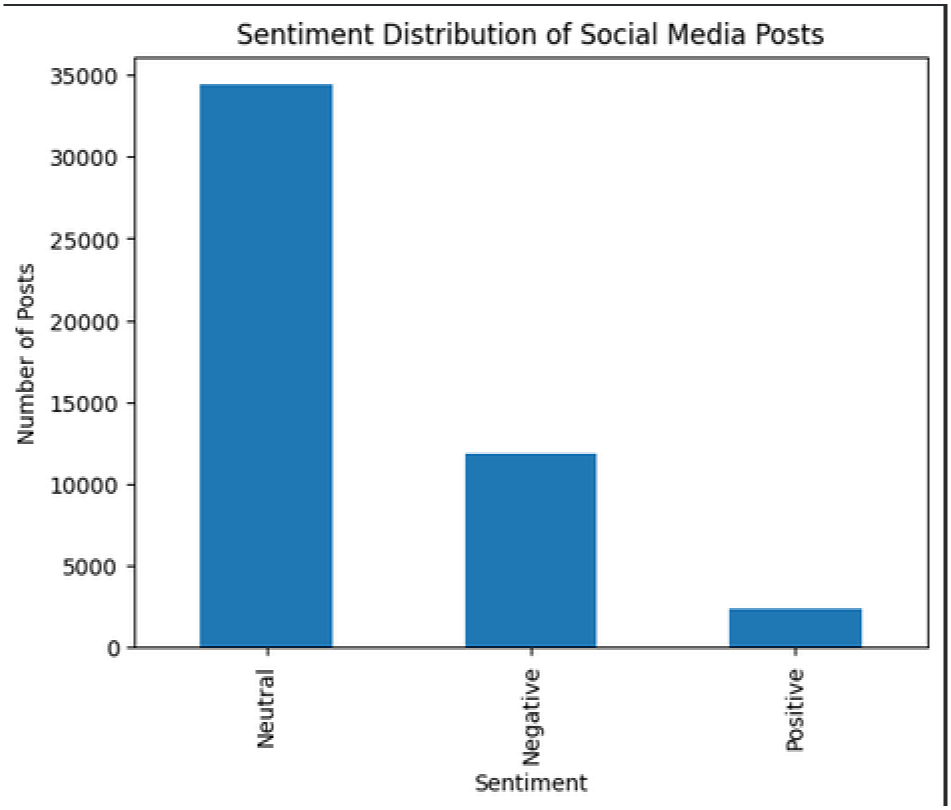
Step 2:Time-of-Day Analysis (Traffic Pattern Equivalent)

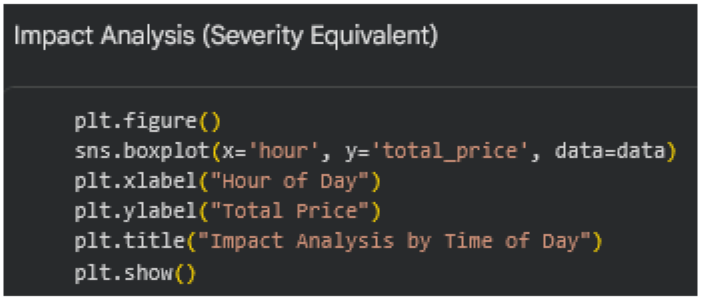
```
import pandas as pd
import plotly.express as px

data['order_time'] = pd.to_datetime(data['order_time'], format='%H:%M:%S')
data['hour'] = data['order_time'].dt.hour

hourly_orders = data['hour'].value_counts().sort_index().reset_index()
hourly_orders.columns = ['Hour of Day', 'Number of Orders']

fig = px.line(
    hourly_orders,
    x='Hour of Day',
    y='Number of Orders',
    markers=True,
    title="Order Frequency by Time of Day (Hotspot Analysis)"
)
fig.show()
```


Order Frequency by Time of Day (Hotspot Analysis)


Day-Wise Pattern Analysis

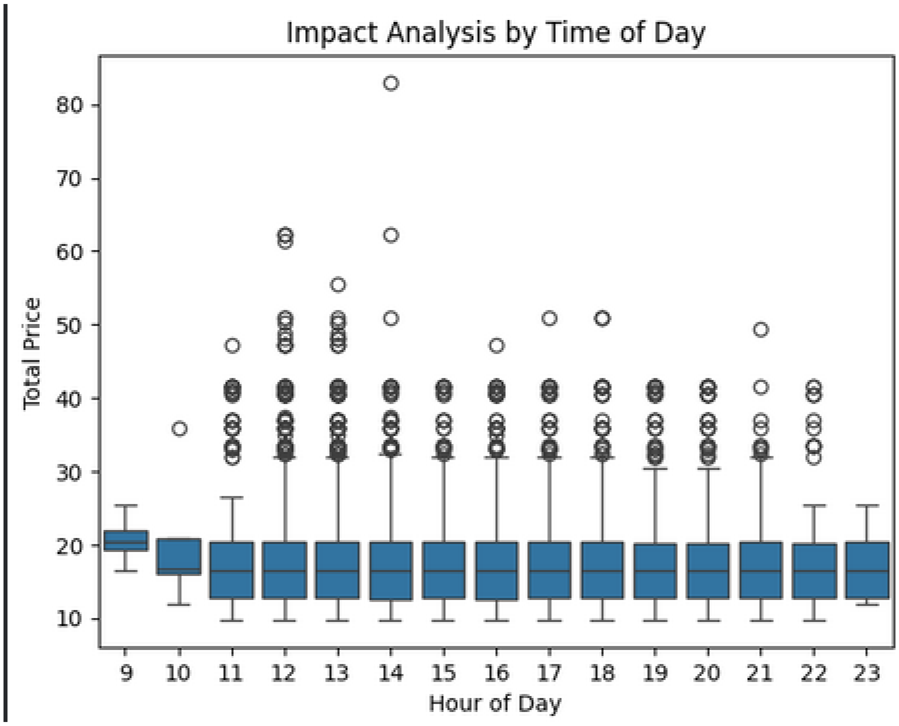
```
plt.figure()
data['day'].value_counts().plot(kind='bar')
plt.xlabel("Day of Week")
plt.ylabel("Number of Orders")
plt.title("Orders by Day of Week")
plt.show()
```

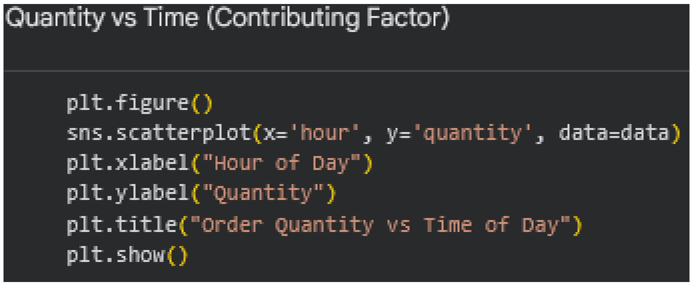

Sentiment Distribution of Social Media Posts



Impact Analysis (Severity Equivalent)

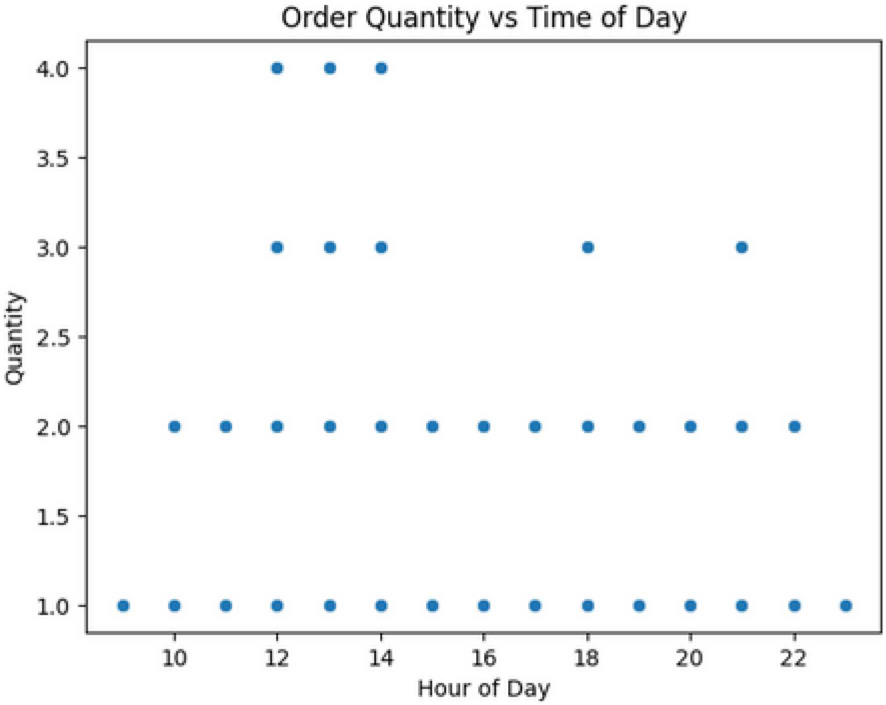
```
plt.figure()
sns.boxplot(x='hour', y='total_price', data=data)
plt.xlabel("Hour of Day")
plt.ylabel("Total Price")
plt.title("Impact Analysis by Time of Day")
plt.show()
```





Quantity vs Time (Contributing Factor)

```
plt.figure()
sns.scatterplot(x='hour', y='quantity', data=data)
plt.xlabel("Hour of Day")
plt.ylabel("Quantity")
plt.title("Order Quantity vs Time of Day")
plt.show()
```


Order Quantity vs Time of Day
Quantity
4.0
3.5
3.0
2.5
2.0
1.5
1.0
10
12
14
16
18
20
22
Hour of Day