coffee-sales-analysis

September 3, 2024

[50]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import mysql.connector
import numpy as np

db = mysql.connector.connect(host = "localhost",
                             username = "root",
                             password = "1012",
                             database = "coffee")
cur = db.cursor()
```

[5]:
```
data = pd.read_csv("C:/Users/ASUS/Desktop/Power BI Practice/Coffee.csv")
data.head(10)
```

[5]:

	date	datetime	cash_type	card	money	coffee_name
0	2024-03-01	2024-03-01 10:15:50.520	card	ANON-0000-0000-0001	38.7	Latte
1	2024-03-01	2024-03-01 12:19:22.539	card	ANON-0000-0000-0002	38.7	Hot Chocolate
2	2024-03-01	2024-03-01 12:20:18.089	card	ANON-0000-0000-0002	38.7	Hot Chocolate
3	2024-03-01	2024-03-01 13:46:33.006	card	ANON-0000-0000-0003	28.9	Americano
4	2024-03-01	2024-03-01 13:48:14.626	card	ANON-0000-0000-0004	38.7	Latte
5	2024-03-01	2024-03-01 15:39:47.726	card	ANON-0000-0000-0005	33.8	Americano with Milk
6	2024-03-01	2024-03-01 16:19:02.756	card	ANON-0000-0000-0006	38.7	Hot Chocolate
7	2024-03-01	2024-03-01 18:39:03.580	card	ANON-0000-0000-0007	33.8	Americano with Milk
8	2024-03-01	2024-03-01 19:22:01.762	card	ANON-0000-0000-0008	38.7	Cocoa
9	2024-03-01	2024-03-01 19:23:15.887	card	ANON-0000-0000-0008	33.8	

```
9  Americano with Milk
```

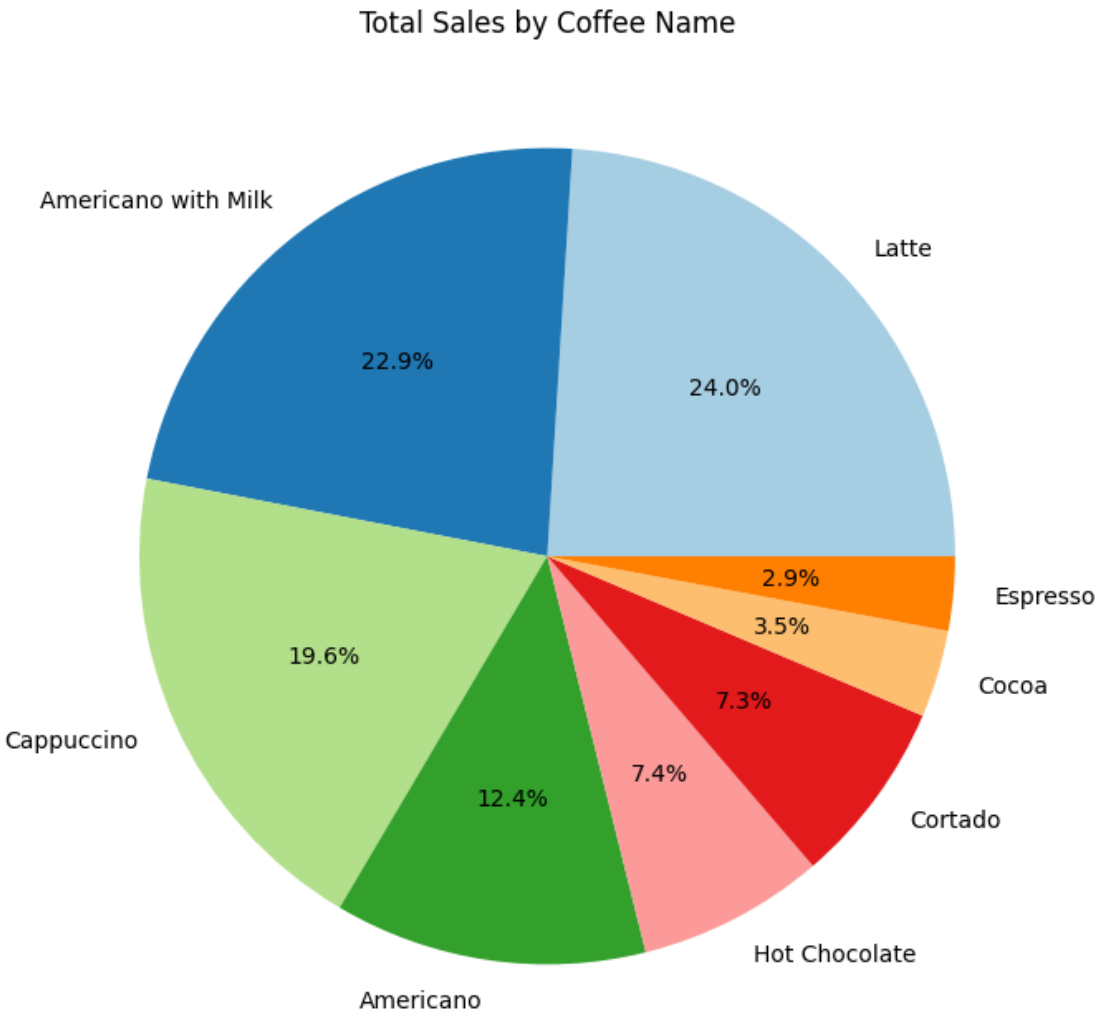
1 1:- Total Sales Per Coffee Type.

[6]:
```
query = """SELECT
    coffee_name, ROUND(SUM(money), 2) AS total_sales
FROM
    coffee.coffee
GROUP BY coffee_name
ORDER BY total_sales DESC"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Coffee Name","Total Sales"])
df
```

[6]:

	Coffee Name	Total Sales
0	Latte	9009.14
1	Americano with Milk	8601.94
2	Cappuccino	7333.14
3	Americano	4644.54
4	Hot Chocolate	2778.48
5	Cortado	2745.08
6	Cocoa	1295.94
7	Espresso	1100.62

[8]:
```
plt.figure(figsize=(8, 8))
plt.pie(df['Total Sales'], labels=df['Coffee Name'], autopct='%1.1f%%',␣
↪colors=plt.cm.Paired.colors)
plt.title('Total Sales by Coffee Name')
plt.show()
```



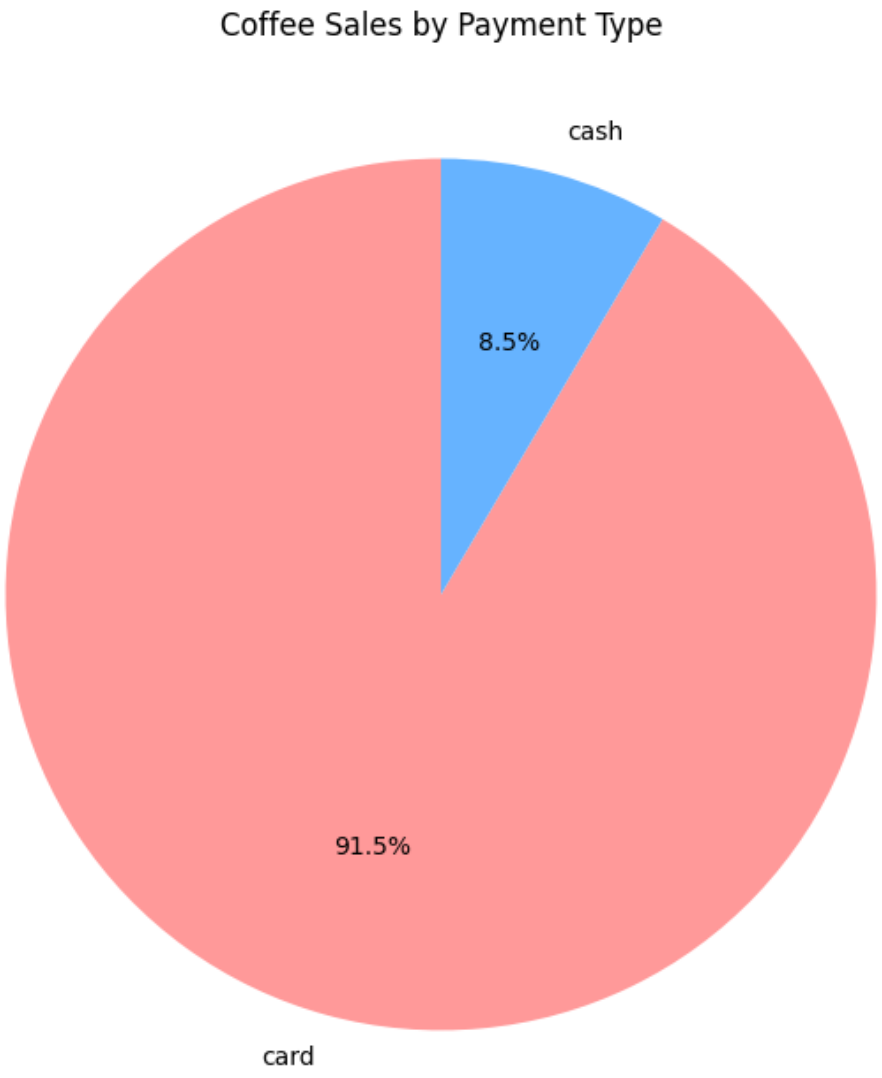
2 2:- Total Sales by Payment Method (Cash vs. Card).

[9]:
```
query = """SELECT
    cash_type, ROUND(SUM(money), 2) AS total_sales
FROM
    coffee.coffee
GROUP BY cash_type
ORDER BY total_sales DESC"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Payment Type","Total Sales"])
df
```

[9]:

	Payment Type	Total Sales
0	card	34322.88
1	cash	3186.00

[11]:
```
colors = ['#ff9999','#66b3ff','#99ff99','#ffcc99']
plt.figure(figsize=(8, 8))
plt.pie(df['Total Sales'], labels=df['Payment Type'], autopct='%1.1f%%',␣
 ↪colors=colors, startangle=90)
plt.title('Coffee Sales by Payment Type')
plt.show()
```



3 3:- Daily Sales Trend.

[12]:
```
query = """SELECT
    date, ROUND(SUM(money), 2) AS total_sales
FROM
    coffee.coffee
GROUP BY date
ORDER BY date ASC"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Date","Total Sales"])
df
```

[12]:

	Date	Total Sales
0	2024-03-01	396.30
1	2024-03-02	228.10
2	2024-03-03	349.10
3	2024-03-04	135.20
4	2024-03-05	338.50
..	…	…
145	2024-07-27	372.76
146	2024-07-28	78.86
147	2024-07-29	321.82
148	2024-07-30	650.48
149	2024-07-31	633.84

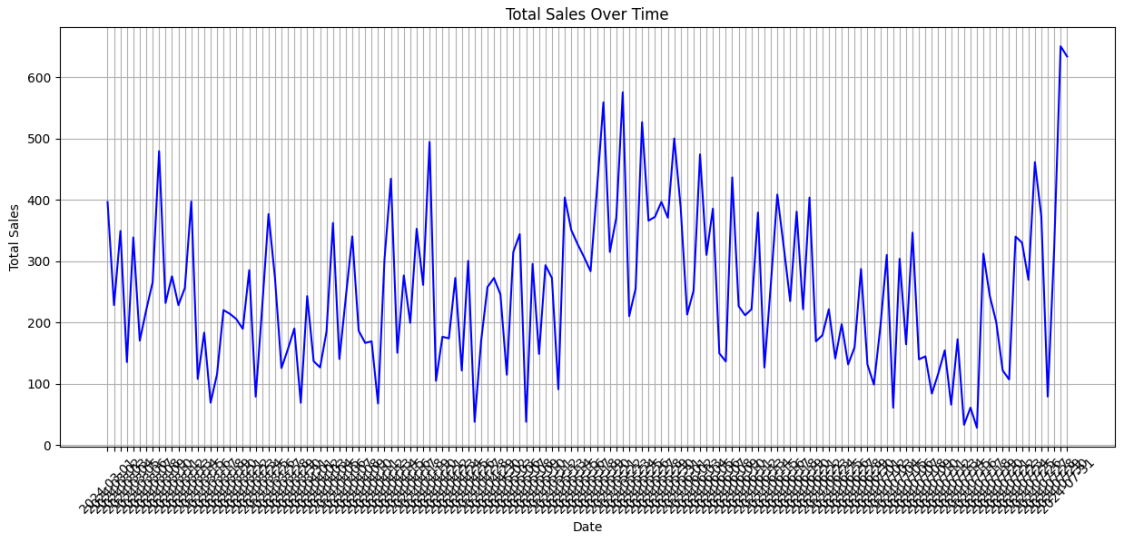
[150 rows x 2 columns]

[23]:
```
plt.figure(figsize=(15, 6))

# Create the line plot with a custom color
sns.lineplot(x='Date', y='Total Sales', data=df, color="blue")

# Customize the plot
plt.title('Total Sales Over Time')
plt.xlabel('Date')
plt.ylabel('Total Sales')
plt.xticks(rotation=45)
plt.grid(True)

# Show the plot
plt.show()
```



4 4:- Selling Coffee Types.

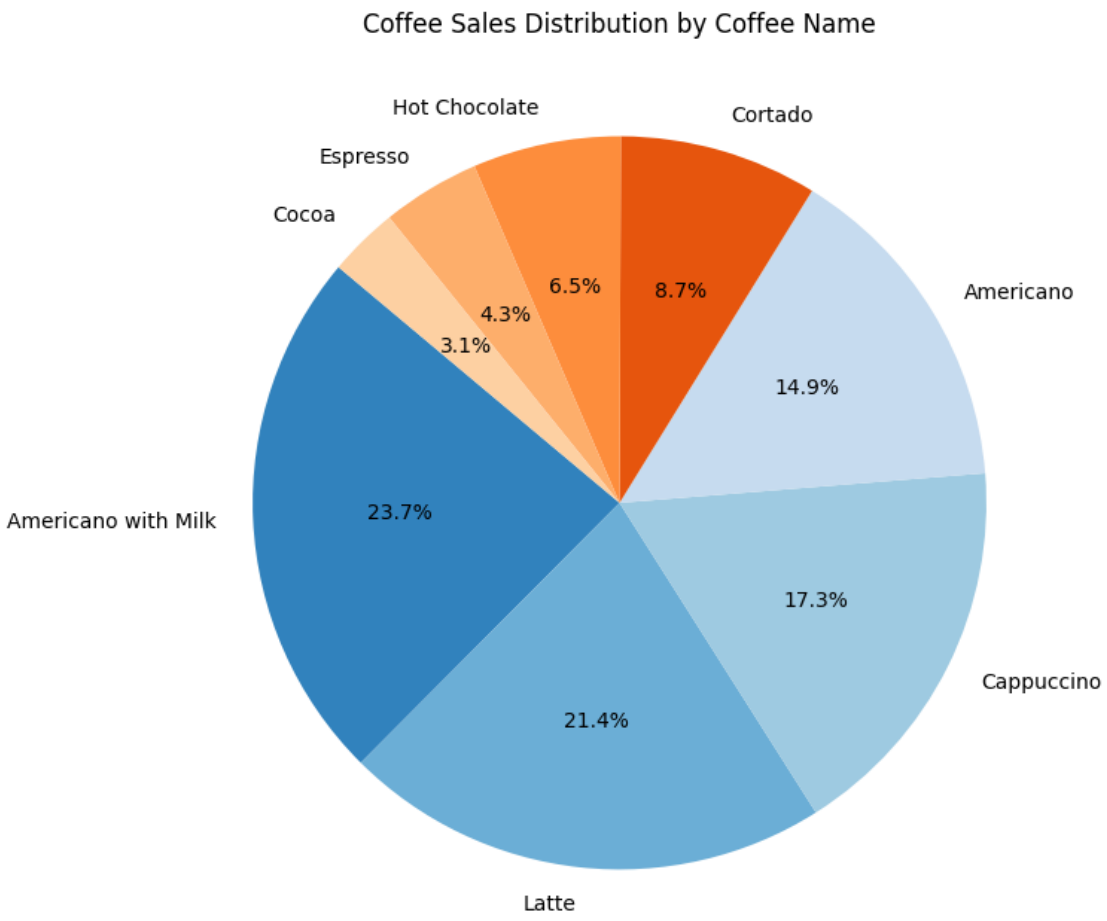
[24]:
```
query = """SELECT
    coffee_name, COUNT(*) AS sales_count
FROM
    coffee.coffee
GROUP BY coffee_name
ORDER BY sales_count DESC"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Coffee Name","Sales Count"])
df
```

[24]:

	Coffee Name	Sales Count
0	Americano with Milk	268
1	Latte	243
2	Cappuccino	196
3	Americano	169
4	Cortado	99
5	Hot Chocolate	74
6	Espresso	49
7	Cocoa	35

[26]:
```
colors = plt.get_cmap('tab20c').colors
plt.figure(figsize=(8, 8))
plt.pie(df['Sales Count'], labels=df['Coffee Name'], autopct='%1.1f%%',␣
 ↪colors=colors, startangle=140)
plt.title('Coffee Sales Distribution by Coffee Name')
```

```
plt.show()
```



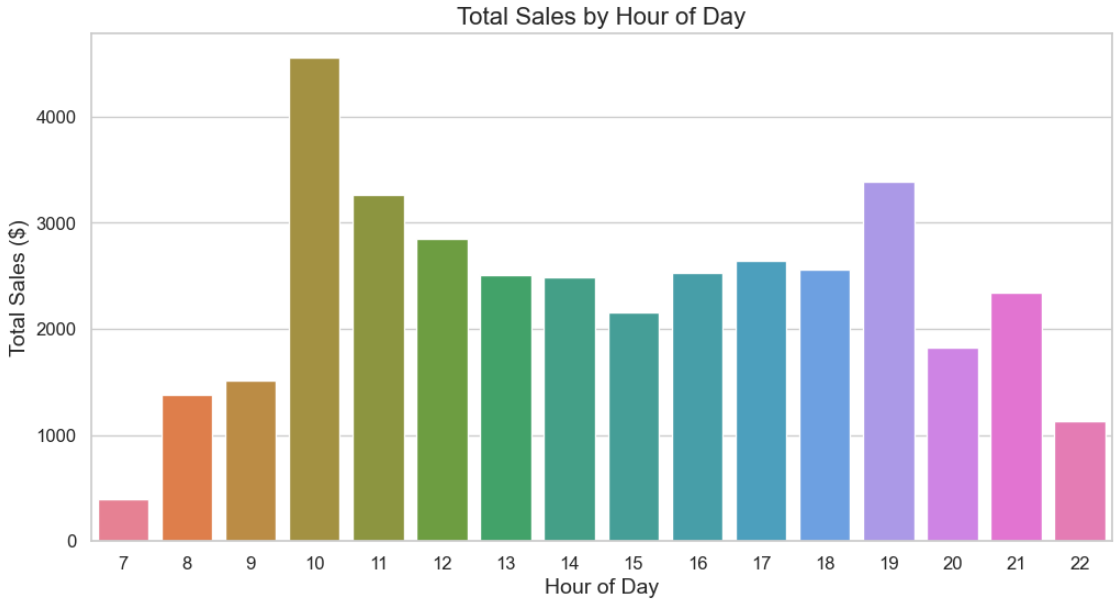
5 5:- Sales Distribution by Hour of the Day.

[28]:
```
query = """SELECT
    EXTRACT(HOUR FROM datetime) AS hour_of_day,
    ROUND(SUM(money), 2) AS total_sales
FROM
    coffee.coffee
GROUP BY EXTRACT(HOUR FROM datetime)
ORDER BY hour_of_day ASC"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Hour of Day","Total Sales"])
df
```

[28]:

	Hour of Day	Total Sales
0	7	392.80
1	8	1380.38
2	9	1515.48
3	10	4553.18
4	11	3258.64
5	12	2850.60
6	13	2511.60
7	14	2484.92
8	15	2158.76
9	16	2525.36
10	17	2639.08
11	18	2558.04
12	19	3388.32
13	20	1819.92
14	21	2343.86
15	22	1127.94

[33]:
```
sns.set(style="whitegrid")
palette = sns.color_palette("husl", len(df))
plt.figure(figsize=(12, 6))
sns.barplot(x="Hour of Day", y="Total Sales", data=df, hue="Hour of Day",␣
 ↪palette=palette, dodge=False, legend=False)
plt.title("Total Sales by Hour of Day", fontsize=16)
plt.xlabel("Hour of Day", fontsize=14)
plt.ylabel("Total Sales ($)", fontsize=14)
plt.xticks(fontsize=12)
plt.yticks(fontsize=12)
plt.show()
```



6 6:- Average Sale Amount Per Transaction.

[34]:
```
query = """SELECT round(AVG(money),4) AS average_sale
FROM coffee.coffee;"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Average Sales"])
df
```

[34]:

	Average Sales
0	33.1058

7 7:- Sales by Weekday.

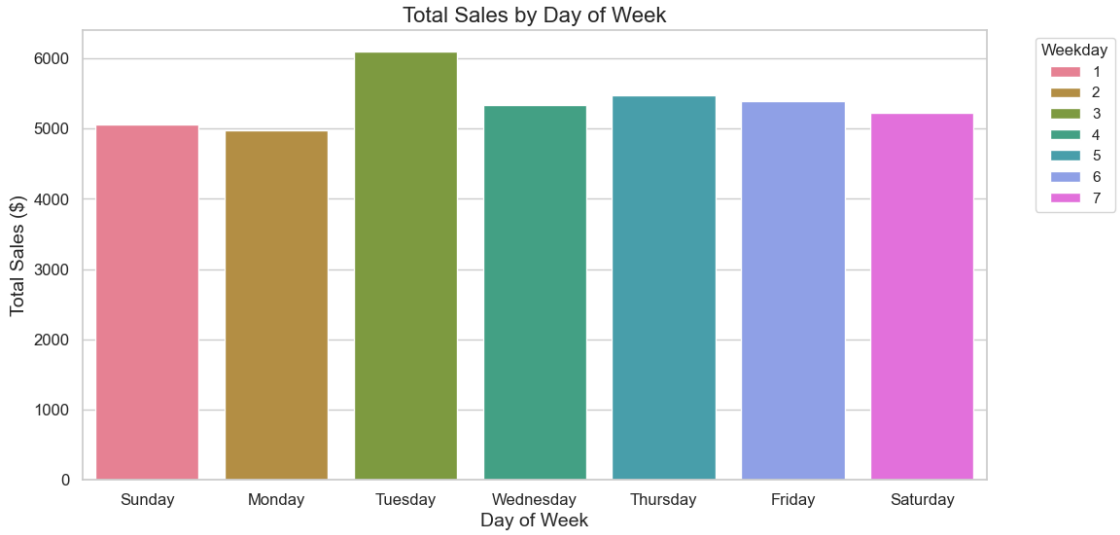
[35]:
```
query = """SELECT
    DAYOFWEEK(date) AS weekday,
    ROUND(SUM(money), 2) AS total_sales
FROM
    coffee.coffee
GROUP BY DAYOFWEEK(date)
ORDER BY DAYOFWEEK(date) ASC"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Weekday","Total Sales"])
df
```

[35]:

	Weekday	Total Sales
0	1	5050.20
1	2	4969.68
2	3	6092.48
3	4	5327.20
4	5	5466.74
5	6	5386.32
6	7	5216.26

[38]:
```
df['Weekday'] = pd.Categorical(df['Weekday'], categories=[1, 2, 3, 4, 5, 6, 7],␣
 ↪ordered=True)
df = df.sort_values('Weekday')
sns.set(style="whitegrid")
plt.figure(figsize=(12, 6))
sns.barplot(x="Weekday", y="Total Sales", data=df, hue="Weekday", palette=sns.
 ↪color_palette("husl", len(df)), dodge=False)
```

```
plt.title("Total Sales by Day of Week", fontsize=16)
plt.xlabel("Day of Week", fontsize=14)
plt.ylabel("Total Sales ($)", fontsize=14)
plt.xticks(ticks=range(7), labels=['Sunday', 'Monday', 'Tuesday', 'Wednesday',␣
↪'Thursday', 'Friday', 'Saturday'], fontsize=12)
plt.yticks(fontsize=12)
plt.legend(title='Weekday', loc='upper right', bbox_to_anchor=(1.15, 1))
plt.show()
```



8 8:- Peak Sales Day.

[39]:
```
query = """SELECT
    date, ROUND(SUM(money), 2) AS total_sales
FROM
    coffee.coffee
GROUP BY date
ORDER BY total_sales DESC
LIMIT 1"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Date","Total Sales"])
df
```

[39]:

	Date	Total Sales
0	2024-07-30	650.48

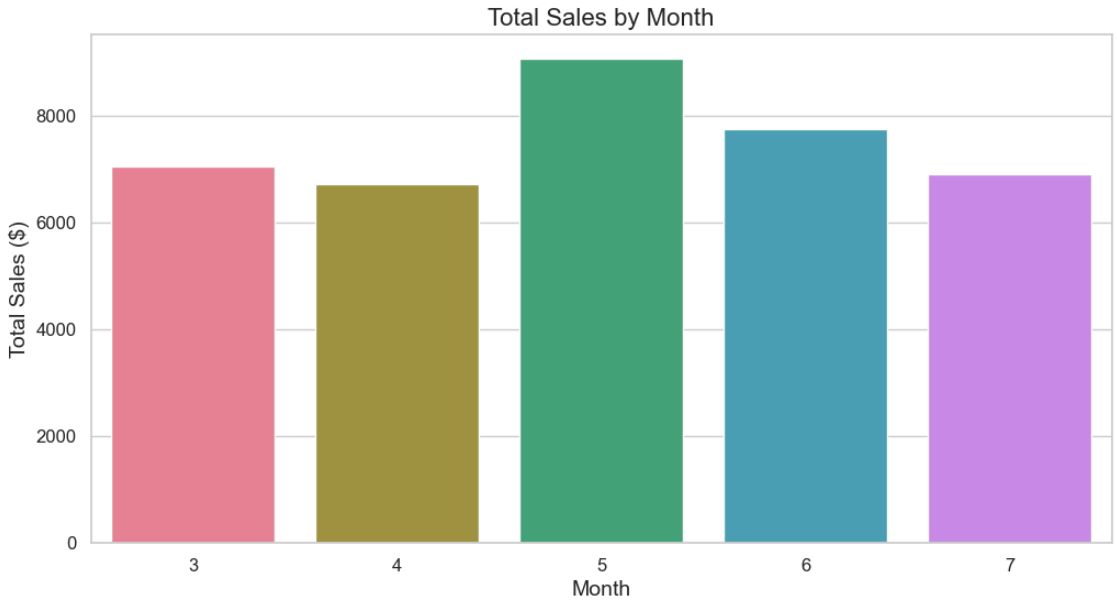
9 9:- Monthly Sales Summary.

[40]:
```
query = """SELECT
    EXTRACT(MONTH FROM date) AS month,
    ROUND(SUM(money), 2) AS total_sales
FROM
    coffee.coffee
GROUP BY month
ORDER BY month ASC"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Date","Total Sales"])
df
```

[40]:

	Date	Total Sales
0	3	7050.20
1	4	6720.56
2	5	9063.42
3	6	7758.76
4	7	6915.94

[43]:
```
sns.set(style="whitegrid")
palette = sns.color_palette("husl", len(df))
df['Color'] = df['Month']
plt.figure(figsize=(12, 6))
sns.barplot(x="Month", y="Total Sales", data=df, hue="Color", palette=palette,␣
 ↪dodge=False, legend=False)
plt.title("Total Sales by Month", fontsize=16)
plt.xlabel("Month", fontsize=14)
plt.ylabel("Total Sales ($)", fontsize=14)
plt.xticks(fontsize=12)
plt.yticks(fontsize=12)
plt.show()
```



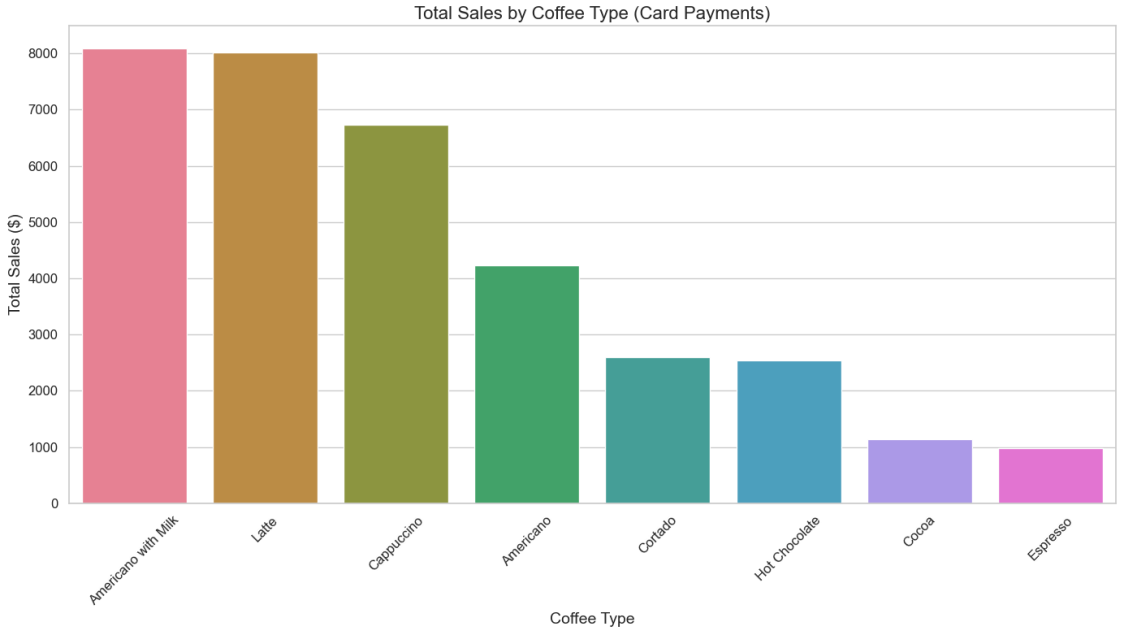
10 10:- Sales by Payment Method (Card) for Each Coffee Type.

[44]:
```
query = """SELECT
    coffee_name, ROUND(SUM(money), 2) AS total_sales
FROM
    coffee.coffee
WHERE
    cash_type = 'card'
GROUP BY coffee_name
ORDER BY total_sales DESC"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Coffee Name","Total Sales"])
df
```

[44]:

	Coffee Name	Total Sales
0	Americano with Milk	8083.94
1	Latte	8018.14
2	Cappuccino	6738.14
3	Americano	4232.54
4	Cortado	2595.08
5	Hot Chocolate	2539.48
6	Cocoa	1138.94
7	Espresso	976.62

[47]:
```
df.columns = ["Coffee Name", "Total Sales"]
sns.set(style="whitegrid")
palette = sns.color_palette("husl", len(df))
plt.figure(figsize=(14, 8))
sns.barplot(x="Coffee Name", y="Total Sales", data=df, palette=palette,␣
 ↪hue="Coffee Name", dodge=False, legend=False)
plt.title("Total Sales by Coffee Type (Card Payments)", fontsize=16)
plt.xlabel("Coffee Type", fontsize=14)
plt.ylabel("Total Sales ($)", fontsize=14)
plt.xticks(rotation=45, fontsize=12)
plt.yticks(fontsize=12)
plt.tight_layout()
plt.show()
```



11 11:- Sales by Payment Method (Cash) for Each Coffee Type.

[48]:
```
query = """SELECT
    coffee_name, ROUND(SUM(money), 2) AS total_sales
FROM
    coffee.coffee
WHERE
    cash_type = 'cash'
GROUP BY coffee_name
ORDER BY total_sales DESC"""
cur.execute(query)
```

```
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Coffee Name","Total Sales"])
df
```

[48]:

	Coffee Name	Total Sales
0	Latte	991.0
1	Cappuccino	595.0
2	Americano with Milk	518.0
3	Americano	412.0
4	Hot Chocolate	239.0
5	Cocoa	157.0
6	Cortado	150.0
7	Espresso	124.0

[49]:
```
df.columns = ["Coffee Name", "Total Sales"]
sns.set(style="whitegrid")
palette = sns.color_palette("husl", len(df))
plt.figure(figsize=(14, 8))
sns.barplot(x="Coffee Name", y="Total Sales", data=df, palette=palette,␣
 ↪hue="Coffee Name", dodge=False, legend=False)
plt.title("Total Sales by Coffee Type (Cash Payments)", fontsize=16)
plt.xlabel("Coffee Type", fontsize=14)
plt.ylabel("Total Sales ($)", fontsize=14)
plt.xticks(rotation=45, fontsize=12)
plt.yticks(fontsize=12)
plt.tight_layout()
plt.show()
```