

```
In [1]: #Import all libraries
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

In [2]: #Imports Data
menu_data = pd.read_csv(r'D:\Vars\Python\Projects\Project 1 - Restaurant order sales Analysis\menu_items.csv')
order_data = pd.read_csv(r'D:\Vars\Python\Projects\Project 1 - Restaurant order sales Analysis\order_details.csv')

In [3]: menu_data.head()

Out[3]:
   menu_item_id  item_name  category  price
0             101  Hamburger  American  12.95
1             102  Cheeseburger  American  13.95
2             103    Hot Dog  American   9.00
3             104  Veggie Burger  American  10.50
4             105  Mac & Cheese  American   7.00

In [4]: order_data.head()

Out[4]:
   order_details_id  order_id  order_date  order_time  item_id
0                  1         1    1/1/23  11:38:36 AM    109.0
1                  2         2    1/1/23  11:57:40 AM    108.0
2                  3         2    1/1/23  11:57:40 AM    124.0
3                  4         2    1/1/23  11:57:40 AM    117.0
4                  5         2    1/1/23  11:57:40 AM    129.0

In [5]: menu_data.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 32897 entries, 0 to 32896
Data columns (total 4 columns):
 #   Column            Non-Null Count  Dtype
---  --
 0   menu_item_id      32 non-null     int64
 1   item_name         32 non-null     object
 2   category          32 non-null     object
 3   price            32 non-null     float64
dtypes: float64(1), int64(1), object(2)
memory usage: 3.1+ MB

In [6]: order_data.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 12234 entries, 0 to 12233
Data columns (total 5 columns):
 #   Column            Non-Null Count  Dtype
---  --
 0   order_details_id  12234 non-null  int64
 1   order_id          12234 non-null  int64
 2   order_date        12234 non-null  object
 3   order_time        12234 non-null  object
 4   item_id           12234 non-null  float64
dtypes: float64(1), int64(2), object(2)
memory usage: 478.9+ KB

In [7]: #Check Null
menu_data.isnull().sum()

Out[7]:
menu_item_id    0
item_name       0
category        0
price          0
dtype: int64

In [8]: #Check Null
order_data.isnull().sum()

Out[8]:
order_details_id    0
order_id            0
order_date          0
order_time          0
item_id            137
dtype: int64

In [9]: #Check Duplication
menu_data.duplicated().sum()

Out[9]:
0

In [10]: #Check Duplication
order_data.duplicated().sum()

Out[10]:
0

In [11]: order_data["item_id"].unique()

Out[11]:
array([109., 108., 124., 117., 129., 106., 119., 101., 114., 112., 126.,
       110., 122., 138., 127., 105., 102., 131., 104., 107., 125., 128.,
       111., 116., 127., 128., 118., 131., 128., 103., 112., 115.])

In [12]: order_data = order_data.dropna()

In [13]: order_data

Out[13]:
   order_details_id  order_id  order_date  order_time  item_id
0                  1         1    1/1/23  11:38:36 AM    109.0
1                  2         2    1/1/23  11:57:40 AM    108.0
2                  3         2    1/1/23  11:57:40 AM    124.0
3                  4         2    1/1/23  11:57:40 AM    117.0
4                  5         2    1/1/23  11:57:40 AM    129.0
...
12229  12230  6369  3/5/23  10:05:04 PM    109.0
12230  12231  6369  3/5/23  10:05:04 PM    129.0
12231  12232  6369  3/5/23  10:05:04 PM    129.0
12232  12233  6369  3/5/23  10:05:04 PM    122.0
12233  12234  6370  3/5/23  10:15:48 PM    122.0
12297 rows x 5 columns

In [14]: order_data["item_id"].unique()

Out[14]:
array([109., 108., 124., 117., 129., 106., 119., 101., 114., 112., 126.,
       110., 122., 138., 127., 105., 102., 131., 104., 107., 125., 128.,
       111., 116., 127., 128., 118., 131., 128., 103., 112., 115.])

In [15]: #Check Null
order_data.isnull().sum()

Out[15]:
order_details_id    0
order_id            0
order_date          0
order_time          0
item_id            0
dtype: int64

In [16]: order_data.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 12297 entries, 0 to 12296
Data columns (total 5 columns):
 #   Column            Non-Null Count  Dtype
---  --
 0   order_details_id  12297 non-null  int64
 1   order_id          12297 non-null  int64
 2   order_date        12297 non-null  object
 3   order_time        12297 non-null  object
 4   item_id           12297 non-null  float64
dtypes: float64(1), int64(2), object(2)
memory usage: 507.9+ KB

In [17]: #convert item_id datatype float into int
order_data["item_id"] = np.ceil(order_data["item_id"]).astype(int)

In [18]: order_data["item_id"].dtype

Out[18]:
dtype('int32')

In [19]: #convert order_date datatype object into date
order_data["order_date"] = pd.to_datetime(order_data["order_date"], format="%d/%m/%Y")

In [20]: order_data["order_date"].info()
<class 'pandas.core.series.Series'>
Int64Index: 12297 entries, 0 to 12296
Series name: order_date
Non-Null Count: 12297
dtypes: datetime64[ns](1)
memory usage: 309.0 KB

In [21]: order_data.head()

Out[21]:
   order_details_id  order_id  order_date  order_time  item_id
0                  1         1  2023-01-01  11:38:36 AM    109
1                  2         2  2023-01-01  11:57:40 AM    108
2                  3         2  2023-01-01  11:57:40 AM    124
3                  4         2  2023-01-01  11:57:40 AM    117
4                  5         2  2023-01-01  11:57:40 AM    129

In [24]: #Extract Month from date
order_data["order_month"] = order_data["order_date"].dt.strftime('%b')

In [25]: order_data.head()

Out[25]:
   order_details_id  order_id  order_date  order_time  item_id  Order Month
0                  1         1  2023-01-01  11:38:36 AM    109        Jan
1                  2         2  2023-01-01  11:57:40 AM    108        Jan
2                  3         2  2023-01-01  11:57:40 AM    124        Jan
3                  4         2  2023-01-01  11:57:40 AM    117        Jan
4                  5         2  2023-01-01  11:57:40 AM    129        Jan

In [26]: #Merge two DataFrames
data = pd.merge(menu_data, order_data, left_on='menu_item_id', right_on='item_id', how='inner')
data.head()

Out[26]:
   menu_item_id  item_name  category  price  order_details_id  order_id  order_date  order_time  item_id  Order Month
0             101  Hamburger  American  12.95              11         6  2023-01-01  12:29:36 PM    101        Jan
1             101  Hamburger  American  12.95              26         6  2023-01-01  1:02:09 PM    101        Jan
2             101  Hamburger  American  12.95              43        17  2023-01-01  1:53:00 PM    101        Jan
3             101  Hamburger  American  12.95              63        24  2023-01-01  2:23:01 PM    101        Jan
4             101  Hamburger  American  12.95              71        27  2023-01-01  3:11:17 PM    101        Jan

In [27]: #Category in Menu
category_list = pd.DataFrame(menu_data[category].unique(), columns = ["category"])
category_list

Out[27]:
   category
0  American
1    Asian
2  Mexican
3    Italian

In [28]: #Item List in Menu
item_list = pd.DataFrame(menu_data[item_name].unique(), columns = ["items"])
item_list

Out[28]:
   items
0  Hamburger
1  Cheeseburger
2    Hot Dog
3  Veggie Burger
4  Mac & Cheese
5  French Fries
6  Orange Chicken
7  Tofu Pad Thai
8  Korean Beef Bowl
9  Pork Ramen
10 California Roll
11 Salmon Roll
12 Edamame
13 Potstickers
14 Chicken Tacos
15 Steak Tacos
16 Chicken Burrito
17 Steak Burrito
18 Chicken Torta
19 Steak Torta
20 Cheese Quesadillas
21 Chips & Salsa
22 Chips & Guacamole
23 Spaghetti
24 Spaghetti & Meatballs
25 Fettuccine Alfredo
26 Meat Lasagna
27 Cheese Lasagna
28 Mushroom Ravioli
29 Shrimp Scampi
30 Chicken Parmesan
31 Eggplant Parmesan

In [29]: #Most Expensive Item in Menu
most_exp_item = menu_data.sort_values('price', ascending=False)
most_exp_item.head(1)

Out[29]:
   menu_item_id  item_name  category  price
29             130  Shrimp Scampi  Italian  19.95

In [30]: #Most Cheap Item
most_cheap_item = menu_data.sort_values('price', ascending=True)
most_cheap_item.head(1)

Out[30]:
   menu_item_id  item_name  category  price
12             113  Edamame    Asian    5.0

In [31]: #Print Total orders :
order_data["order_id"].unique()

Total orders : 5343

In [32]: #Print Total Sales :
data["price"].sum().round(2)

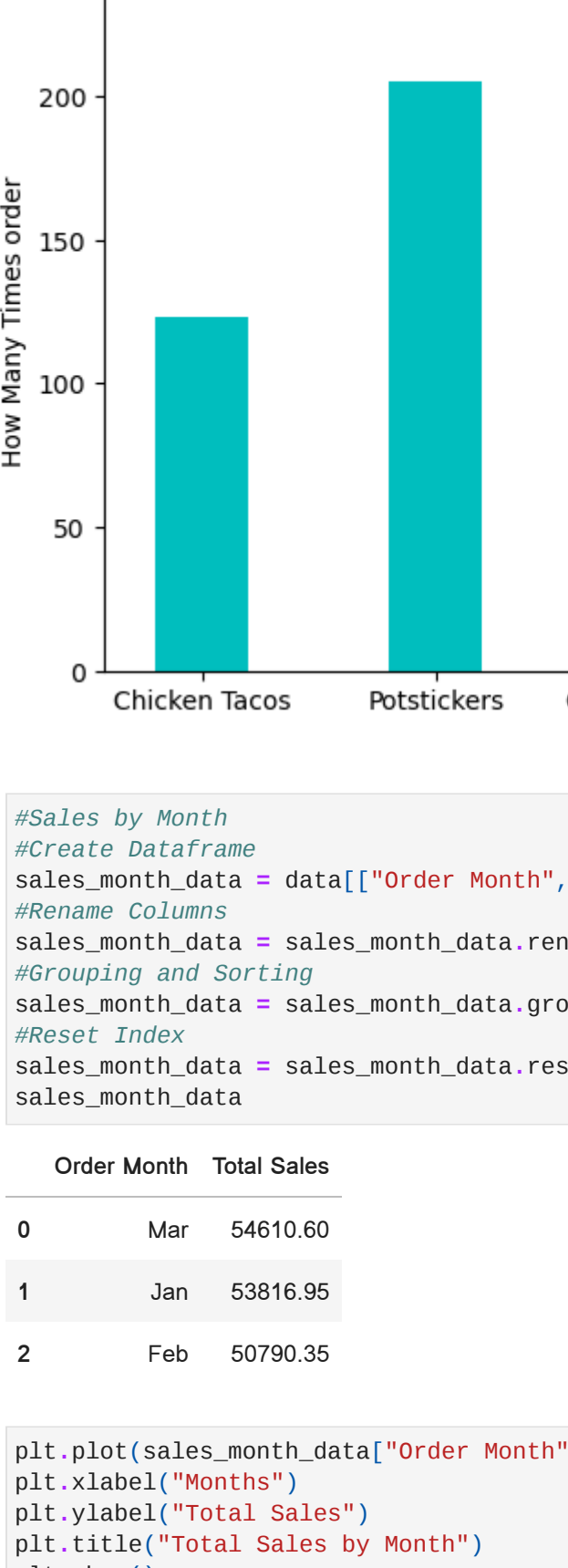
Total Sales : 159217.9

In [43]: #Most Item ordered by which Category
#Create DataFrame
category_data = data[["order_details_id", "category"]].copy()
#Rename columns names
category_data = category_data.rename(columns={"order_details_id": 'order', 'category': 'category'})
#Group by and Sorting
category_data = category_data.groupby('category').count().sort_values('order', ascending=False)
category_data.head()
#Reset Index
category_data = category_data.reset_index()
category_data

Out[43]:
   category  order
0  Asian    3470
1  Italian  2948
2  Mexican  2945
3  American  2734

In [50]: #plt.plot(category_data["order"], labels = category_data["category"], autopct = "%0.2f%%", radius = 0.40)
plt.legend(loc=2)
plt.title("Most Item ordered by which Category")
plt.show()

Most Item ordered by which Category




In [36]: #Top Five most ordered item
#create DataFrame
most_order_data = data[["order_details_id", "item_name"]].copy()
#Rename columns names
most_order_data = most_order_data.rename(columns={"order_details_id": 'How Many Times order', 'item_name': 'Items'})
#Group by and Sorting
most_order_data = most_order_data.groupby('Items').count().sort_values('How Many Times order', ascending=False)
most_order_data.head()
#Reset Index
most_order_data = most_order_data.reset_index()
most_order_data

Out[36]:
   Items  How Many Times order
0  Hamburger              620
1  Edamame                602
2  Korean Beef Bowl       588
3  Cheeseburger           583
4  French Fries           571

In [42]: #plt.figure(figsize=(9,5))
plt.bar(most_order_data["Items"], most_order_data["How Many Times order"], width = 0.4, color = "c")
plt.xlabel("Items")
plt.ylabel("How Many Times order")
plt.title("Top Five Most Ordered Items")
plt.show()

Top Five Most Ordered Items

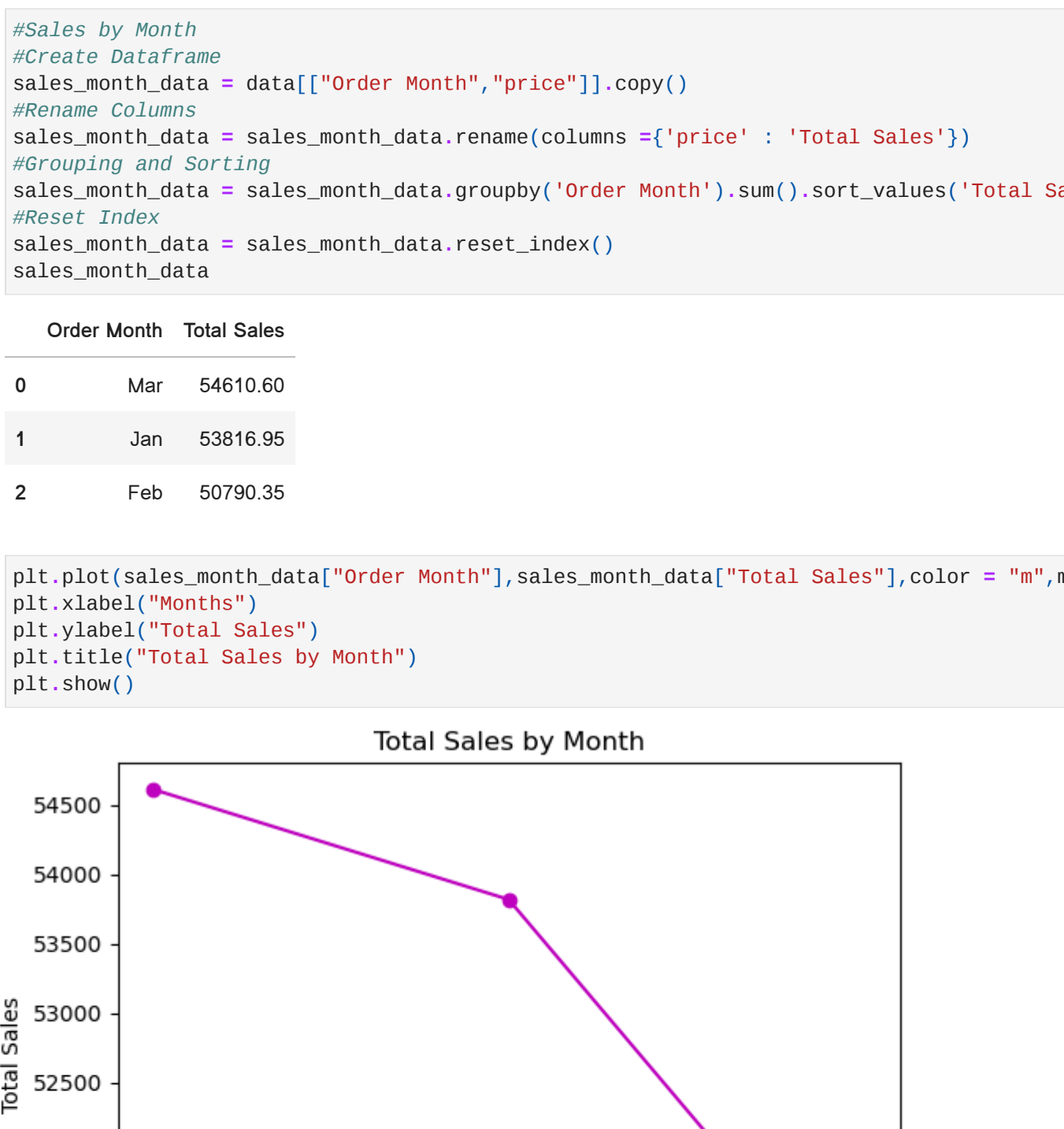


In [45]: #Top Five Less ordered item
#create DataFrame
less_order_data = data[["order_details_id", "item_name"]].copy()
#Rename columns names
less_order_data = less_order_data.rename(columns={"order_details_id": 'How Many Times order', 'item_name': 'Items'})
#Group by and Sorting
less_order_data = less_order_data.groupby('Items').count().sort_values('How Many Times order', ascending=True)
less_order_data.head()
#Reset Index
less_order_data = less_order_data.reset_index()
less_order_data

Out[45]:
   Items  How Many Times order
0  Chicken Tacos              123
1  Potstickers               205
2  Cheese Lasagna           214
3  Steak Torta              217
4  Cheese Quesadillas       233

In [48]: #plt.figure(figsize=(9,5))
plt.bar(less_order_data["Items"], less_order_data["How Many Times order"], width = 0.4, color = "c")
plt.xlabel("Items")
plt.ylabel("How Many Times order")
plt.title("Top Five Less Ordered Items")
plt.show()

Top Five Less Ordered Items

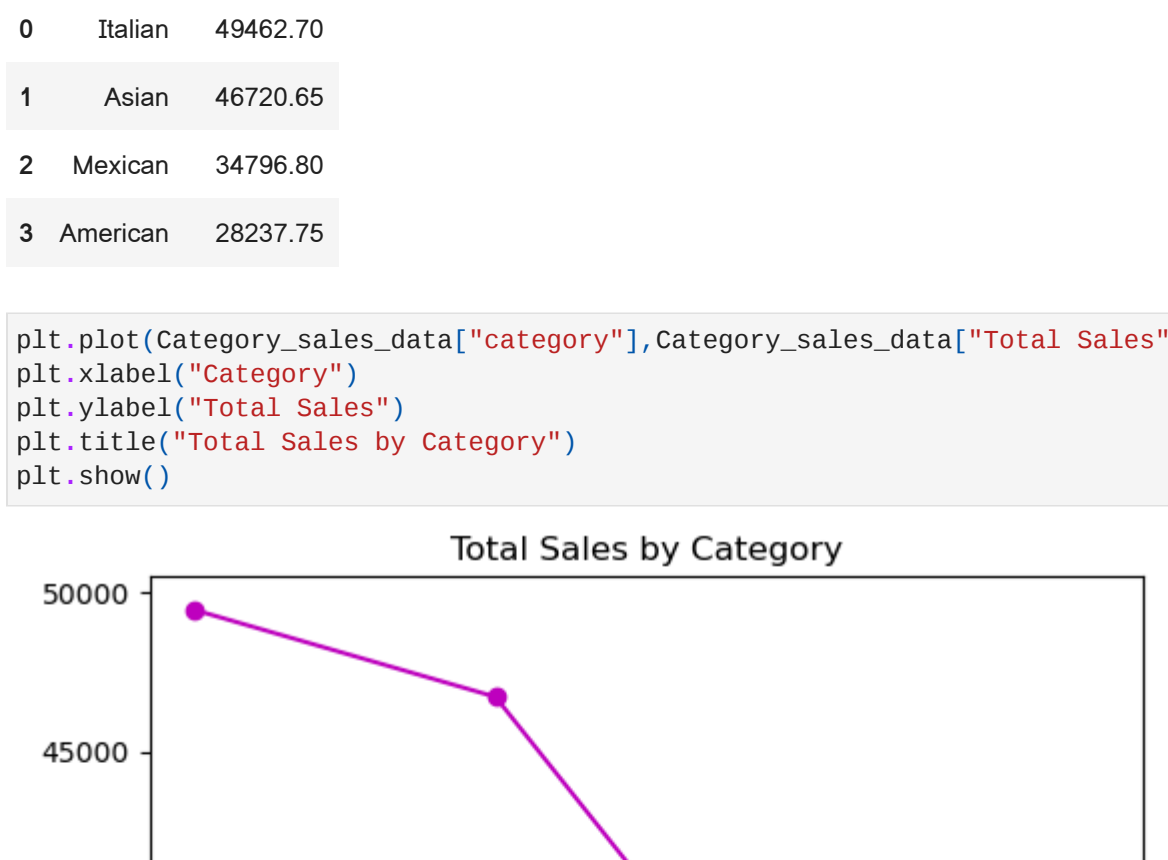


In [56]: #Sales by Month
#create DataFrame
sales_month_data = data[["order_month", "price"]].copy()
#Rename columns
sales_month_data = sales_month_data.rename(columns={"price": 'Total Sales'})
#Grouping and Sorting
sales_month_data = sales_month_data.groupby('order_month').sum().sort_values('Total Sales', ascending=False)
#Reset Index
sales_month_data = sales_month_data.reset_index()
sales_month_data

Out[56]:
   Order Month  Total Sales
0  Mar      54510.00
1  Jan      53816.95
2  Feb      60790.35

In [57]: #plt.plot(sales_month_data["Order Month"], sales_month_data["Total Sales"], color = "m", marker = "o")
plt.xlabel("Months")
plt.ylabel("Total Sales")
plt.title("Total Sales by Month")
plt.show()

Total Sales by Month

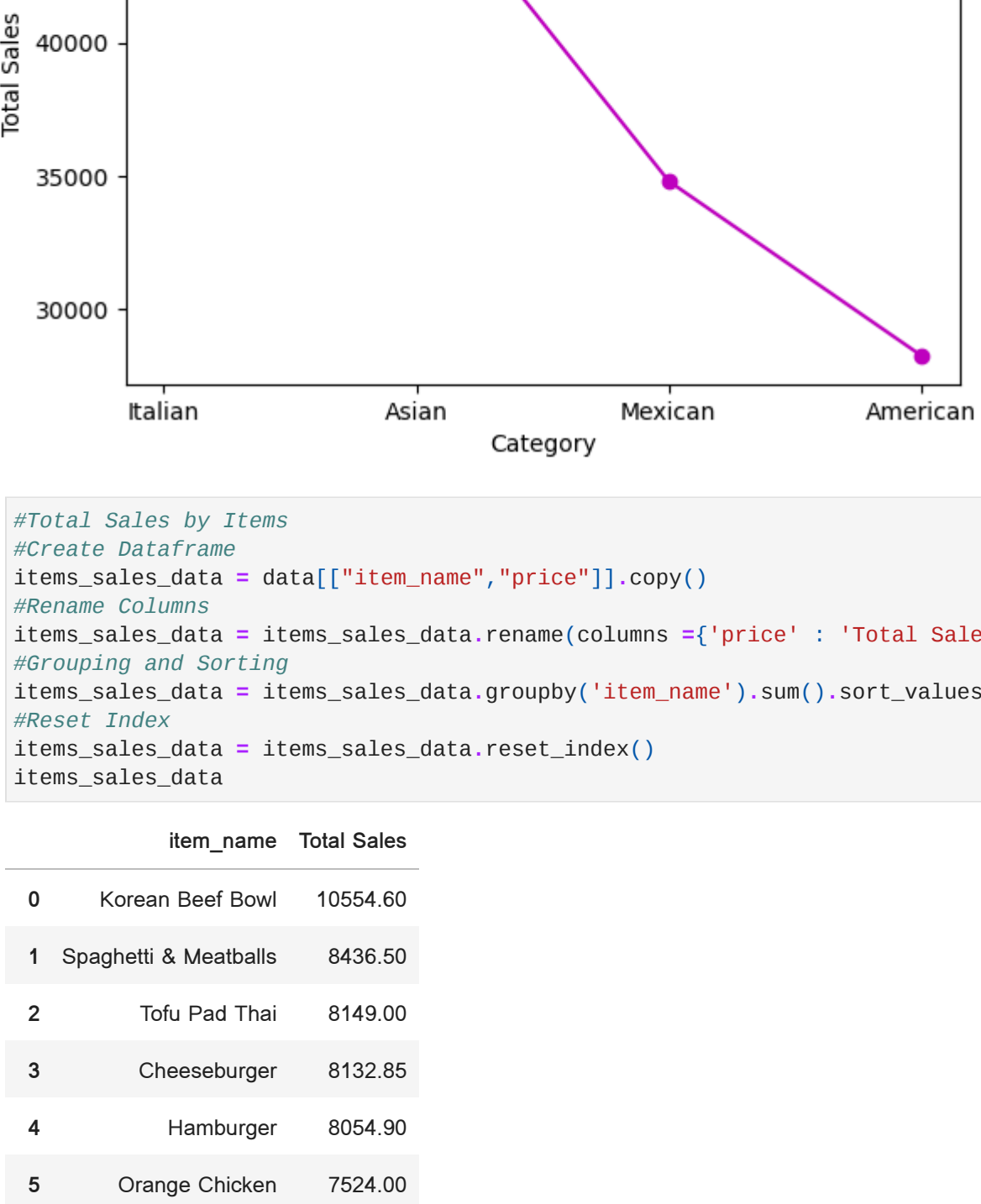


In [58]: #Total Sales by Category
#create DataFrame
Category_sales_data = data[["category", "price"]].copy()
#Rename columns
Category_sales_data = Category_sales_data.rename(columns={"price": 'Total Sales'})
#Grouping and Sorting
Category_sales_data = Category_sales_data.groupby('category').sum().sort_values('Total Sales', ascending=False)
#Reset Index
Category_sales_data = Category_sales_data.reset_index()
Category_sales_data

Out[58]:
   category  Total Sales
0  Italian  49462.70
1  Asian   40720.65
2  Mexican 34796.80
3  American 28237.75

In [59]: #plt.plot(category_sales_data["category"], Category_sales_data["Total Sales"], color = "m", marker = "o")
plt.xlabel("Category")
plt.ylabel("Total Sales")
plt.title("Total Sales by Category")
plt.show()

Total Sales by Category



In [60]: #Total Sales by Items
#create DataFrame
Items_sales_data = data[["item_name", "price"]].copy()
#Rename columns
Items_sales_data = Items_sales_data.rename(columns={"price": 'Total Sales'})
#Grouping and Sorting
Items_sales_data = Items_sales_data.groupby('item_name').sum().sort_values('Total Sales', ascending=False)
#Reset Index
Items_sales_data = Items_sales_data.reset_index()
Items_sales_data

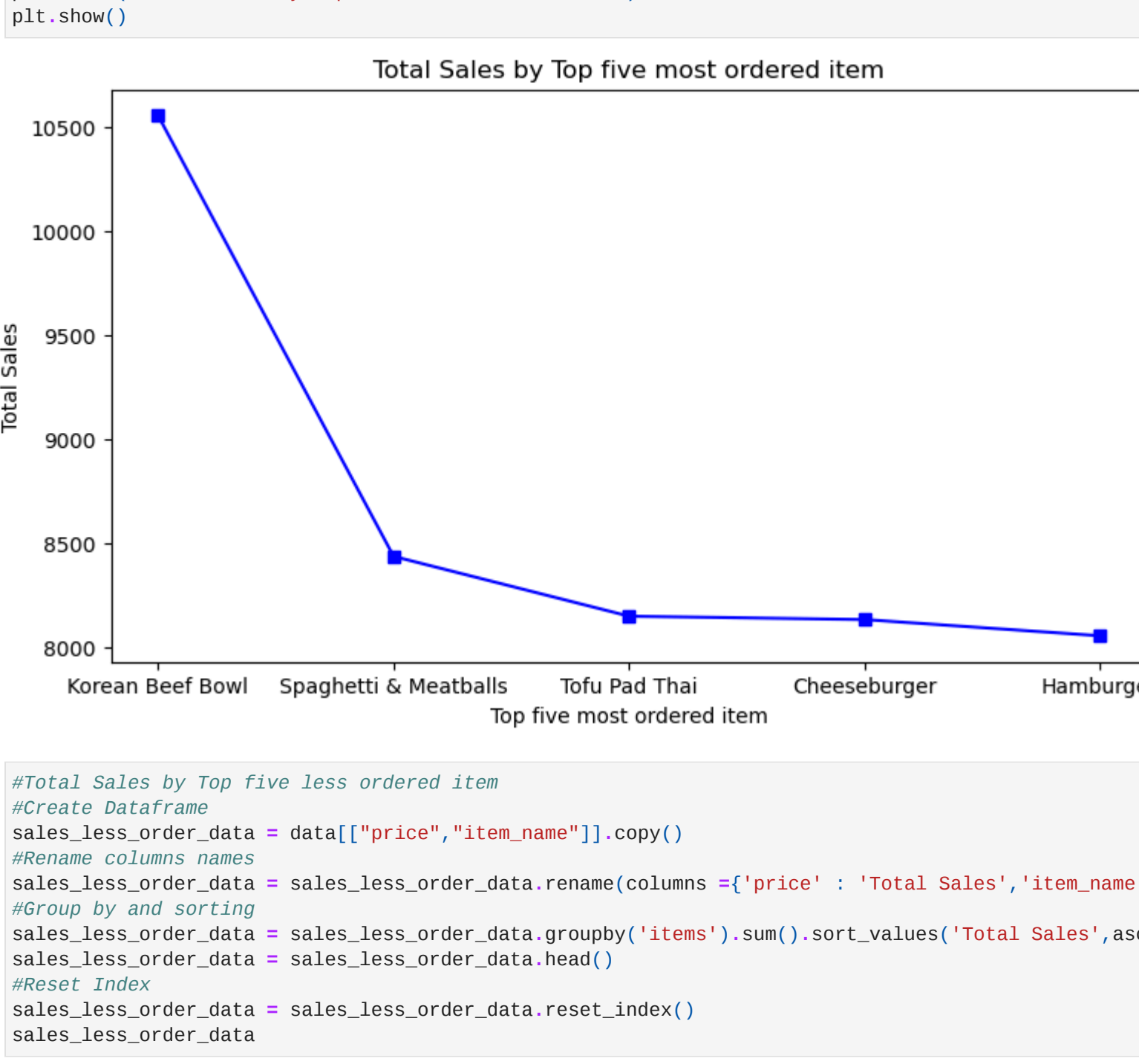
Out[60]:
   item_name  Total Sales
0  Korean Beef Bowl  10554.60
1  Spaghetti & Meatballs  8436.50
2    Tofu Pad Thai    8149.00
3  Cheeseburger     8132.85
4  Hamburger       8054.90
5  Orange Chicken   7524.00
6  Eggplant Parmesan  7119.00
7  Steak Torta      6821.55
8  Chicken Parmesan  6533.80
9  Pork Ramen       6462.00
10 Chicken Burrito  5892.25
11  Fettuccine Alfredo  5664.50
12  Spaghetti       5321.50
13  Steak Burrito   5292.30
14  Meat Lasagna    4800.35
15  Shrimp Scampi   4768.05
17  Chicken Torta   4529.05
18 California Roll  4242.25
19  French Fries    3997.00
20  Fettuccine Alfredo  3610.50
21  Mac & Cheese    3241.00
22  Chips & Salsa   3227.00
23  Cheese Lasagna  3208.50
24  Edamame         3100.00
25  Steak Tacos     2965.30
26  Veggie Burger   2499.00
27  Cheese Quesadillas  2446.50
28    Hot Dog       2313.00
29  Chips & Guacamole  2133.00
30  Potstickers     1845.00
31  Chicken Tacos   1469.85

In [62]: #Total Sales by Top Five most ordered item
#create DataFrame
sales_most_order_data = data[["price", "item_name"]].copy()
#Rename columns names
sales_most_order_data = sales_most_order_data.rename(columns={"price": 'Total Sales', 'item_name': 'Items'})
#Group by and Sorting
sales_most_order_data = sales_most_order_data.groupby('Items').sum().sort_values('Total Sales', ascending=False)
#Reset Index
sales_most_order_data = sales_most_order_data.reset_index()
sales_most_order_data

Out[62]:
   Items  Total Sales
0  Korean Beef Bowl  10554.60
1  Spaghetti & Meatballs  8436.50
2    Tofu Pad Thai    8149.00
3  Cheeseburger     8132.85
4  Hamburger       8054.90

In [65]: #plt.figure(figsize=(9,5))
plt.plot(sales_most_order_data["Items"], sales_most_order_data["Total Sales"], color = "b", marker = "s")
plt.xlabel("Top Five most ordered item")
plt.ylabel("Total Sales")
plt.title("Total Sales by Top Five most ordered item")
plt.show()

Total Sales by Top Five most ordered item



In [66]: #Total Sales by Top Five less ordered item
#create DataFrame
sales_less_order_data = data[["price", "item_name"]].copy()
#Rename columns names
sales_less_order_data = sales_less_order_data.rename(columns={"price": 'Total Sales', 'item_name': 'Items'})
#Group by and Sorting
sales_less_order_data = sales_less_order_data.groupby('Items').sum().sort_values('Total Sales', ascending=True)
#Reset Index
sales_less_order_data = sales_less_order_data.reset_index()
sales_less_order_data

Out[66]:
   Items  Total Sales
0  Chicken Tacos    1469.85
1  Potstickers     1845.00
2  Chips & Guacamole  2133.00
3    Hot Dog       2313.00
4  Cheese Quesadillas  2446.50

In [67]: #plt.figure(figsize=(9,5))
plt.plot(sales_less_order_data["Items"], sales_less_order_data["Total Sales"], color = "b", marker = "s")
plt.xlabel("Top Five Less ordered item")
plt.ylabel("Total Sales")
plt.title("Total Sales by Top Five Less ordered item")
plt.show()
```



