

sales_database

April 20, 2025

0.0.1 TASK 7: Get Basic Sales Summary from a MySQL Database using Python

```
[1]: # import required libraries

import mysql.connector
import pandas as pd
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
```

```
[2]: # Connect to Mysql
mydb = mysql.connector.connect(
    host = "localhost",
    user = "root",
    password = "12345678",
    database = "online_sales",
    use_pure = True,
)
# create cursor
mycursor = mydb.cursor()
print(mydb)
```

<mysql.connector.connection.MySQLConnection object at 0x000002915A69E240>

```
[3]: # run query
mycursor.execute("SELECT * FROM adidas_sales")

output = mycursor.fetchall()
```

```
[4]: query = "SELECT * FROM adidas_sales"

sales_data = pd.read_sql(query, mydb)
```

```
[5]: # view the dataset
sales_data.head()
```

```
[5]:      Retailer  Retailer_ID Invoice_Date   Region   State   City \
0  Foot Locker    1185732    1/1/2020  Northeast  New York  New York
1  Foot Locker    1185732    1/2/2020  Northeast  New York  New York
```

2	Foot Locker	1185732	1/3/2020	Northeast	New York	New York
3	Foot Locker	1185732	1/4/2020	Northeast	New York	New York
4	Foot Locker	1185732	1/5/2020	Northeast	New York	New York

	Product	Price_per_Unit	Units_Sold	Total_Sales	\
0	Men's Street Footwear	50	1200	600000	
1	Men's Athletic Footwear	50	1000	500000	
2	Women's Street Footwear	40	1000	400000	
3	Women's Athletic Footwear	45	850	382500	
4	Men's Apparel	60	900	540000	

	Operating_Profit	Operating_Margin	Sales_Method
0	300000	50%	In-store
1	150000	30%	In-store
2	140000	35%	In-store
3	133875	35%	In-store
4	162000	30%	In-store

1 Exploratory Data Analysis

```
[9]: sales_data.shape
```

```
[9]: (9648, 13)
```

```
[10]: sales_data.columns
```

```
[10]: Index(['Retailer', 'Retailer_ID', 'Invoice_Date', 'Region', 'State', 'City',
          'Product', 'Price_per_Unit', 'Units_Sold', 'Total_Sales',
          'Operating_Profit', 'Operating_Margin', 'Sales_Method'],
          dtype='object')
```

```
[11]: sales_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9648 entries, 0 to 9647
Data columns (total 13 columns):
#   Column          Non-Null Count  Dtype
---  -
0   Retailer        9648 non-null   object
1   Retailer_ID     9648 non-null   int64
2   Invoice_Date     9648 non-null   object
3   Region          9648 non-null   object
4   State           9648 non-null   object
5   City            9648 non-null   object
6   Product         9648 non-null   object
7   Price_per_Unit  9648 non-null   int64
8   Units_Sold      9648 non-null   int64
```

```

9   Total_Sales      9648 non-null   int64
10  Operating_Profit  9648 non-null   int64
11  Operating_Margin  9648 non-null   object
12  Sales_Method      9648 non-null   object
dtypes: int64(5), object(8)
memory usage: 980.0+ KB

```

```
[12]: sales_data.describe()
```

```

[12]:      Retailer_ID  Price_per_Unit  Units_Sold  Total_Sales \
count  9.648000e+03      9648.000000  9648.000000    9648.000000
mean    1.173850e+06      45.216625   256.930037   93273.437500
std     2.636038e+04     14.705397   214.252030  141916.016727
min     1.128299e+06      7.000000    0.000000    0.000000
25%     1.185732e+06     35.000000   106.000000   4254.500000
50%     1.185732e+06     45.000000   176.000000   9576.000000
75%     1.185732e+06     55.000000   350.000000  150000.000000
max     1.197831e+06    110.000000  1275.000000  825000.000000

      Operating_Profit
count      9648.000000
mean      34425.282131
std       54193.124141
min         0.000000
25%       1922.000000
50%       4371.500000
75%       52063.000000
max      390000.000000

```

```

[13]: # Run sql to summarize sales data

# Find revenue for each product
query = """
SELECT
    Product AS product,
    SUM(Units_Sold) AS total_qty,
    SUM(Units_Sold * Price_per_Unit) AS revenue
FROM adidas_sales
GROUP BY Product
"""

df_revenue = pd.read_sql_query(query, mydb)

# print the result
print("\nSales Summary:")
print(df_revenue)

```

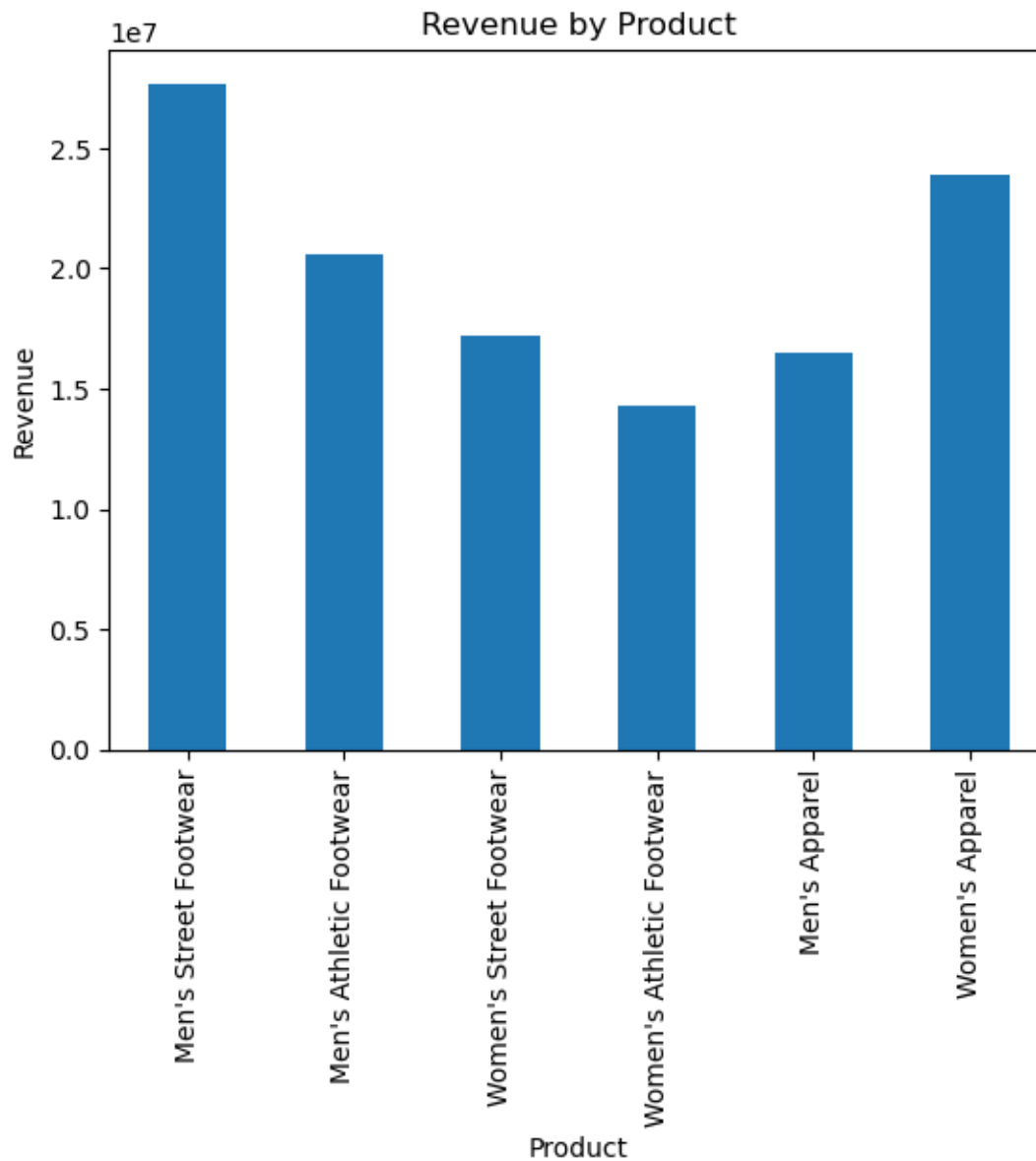
Sales Summary:

	product	total_qty	revenue
0	Men's Street Footwear	593320.0	27680769.0
1	Men's Athletic Footwear	435526.0	20577180.0
2	Women's Street Footwear	392269.0	17201563.0
3	Women's Athletic Footwear	317236.0	14315521.0
4	Men's Apparel	306683.0	16520632.0
5	Women's Apparel	433827.0	23870985.0

Bar Chart Revenue by Product

```
[14]: # Plot Revenue by Product

df_revenue.plot(kind='bar', x = 'product', y='revenue', legend=False)
plt.title("Revenue by Product")
plt.xlabel("Product")
plt.ylabel("Revenue")
plt.savefig("Sales_chart.png")
plt.show()
```



```
[15]: # Revenue by Region

query_region = """
SELECT
    Region,
    SUM(Units_Sold * Price_per_Unit) AS Total_Revenue
FROM adidas_sales
GROUP BY Region
ORDER BY Total_Revenue DESC;
"""
```

```
df_region = pd.read_sql_query(query_region, mydb)
# print the result
print("\nSales Summary:")
print(df_region)
```

Sales Summary:

	Region	Total_Revenue
0	West	36436157.0
1	Northeast	25078267.0
2	Southeast	21374436.0
3	South	20603356.0
4	Midwest	16674434.0