import pandas as pd

import matplotlib.pyplot as plt

import mysql.connector

# Establish MySQL connection

conn = mysql.connector.connect(

host="your\_host",

user="your\_username",

password="your\_password",

database="outland\_adventures\_db"

)

# Revised SQL Query: Assuming `sales\_transactions` and `equipment` tables have different column names and structure

query = """

SELECT

e.product\_code,

e.description,

e.condition,

-- Assuming we need to join the sales and rental quantities in separate columns

SUM(CASE WHEN st.transaction\_type = 'sale' THEN st.quantity ELSE 0 END) AS total\_sales,

SUM(CASE WHEN st.transaction\_type = 'rental' THEN st.quantity ELSE 0 END) AS total\_rentals

FROM

sales\_transactions st

JOIN

equipment e ON st.product\_code = e.product\_code

WHERE

e.equipment\_type = 'for-sale' -- Filter to only 'for-sale' equipment

GROUP BY

e.product\_code, e.description, e.condition

ORDER BY

total\_sales DESC;

"""

# Fetch the data into a DataFrame

df = pd.read\_sql(query, conn)

# Close the MySQL connection

conn.close()

# Set 'description' as the index for the bar chart

df.set\_index('description', inplace=True)

# Plotting the sales vs rentals for each product

fig, ax = plt.subplots(figsize=(12, 8)) # Adjusted figure size for better visibility

# Bar plot for sales and rentals

df[['total\_sales', 'total\_rentals']].plot(kind='bar', ax=ax, color=['#1f77b4', '#ff7f0e'], width=0.8)

# Title and labels

plt.title('Sales vs Rentals for Equipment', fontsize=16)

plt.xlabel('Equipment Description', fontsize=12)

plt.ylabel('Quantity', fontsize=12)

# Rotate x-axis labels for better readability

plt.xticks(rotation=60, ha='right', fontsize=10)

# Add gridlines for better readability

plt.grid(axis='y', linestyle='--', alpha=0.7)

# Adjust layout to prevent label overlap

plt.tight\_layout()

# Show the plot

plt.show()