import pandas as pd

import matplotlib.pyplot as plt

import numpy as np

import statsmodels.api as sm

# Load the car data into a DataFrame

car\_data = pd.read\_csv("C:/Users/Sumit Prajapat/PycharmProjects/pythonBANKProject1/Car\_data.csv")

# Task 1.A:

market\_popularity\_pivot = pd.pivot\_table(car\_data, index='Market Category', values=['Popularity'], aggfunc=['count', 'mean'])

print("Task 1.A: Pivot Table - Number of Car Models and Popularity by Market Category")

print(market\_popularity\_pivot)

# Task 1.B: Create a combo chart

fig, ax1 = plt.subplots()

bar\_chart = market\_popularity\_pivot[('count', 'Popularity')].plot(kind='bar', ax=ax1, color='blue')

ax1.set\_xlabel('Market Category')

ax1.set\_ylabel('Number of Car Models')

ax1.set\_title('Number of Car Models in Each Market Category')

ax2 = ax1.twinx()

line\_chart = market\_popularity\_pivot[('mean', 'Popularity')].plot(kind='line', ax=ax2, color='red', marker='o')

ax2.set\_ylabel('Popularity Score')

plt.tight\_layout()

plt.show()

# Task 2:

engine\_price\_data = car\_data[['Engine HP', 'MSRP']].dropna()

x = engine\_price\_data['Engine HP']

y = engine\_price\_data['MSRP']

plt.scatter(x, y)

plt.xlabel('Engine Power (HP)')

plt.ylabel('Price')

plt.title('Relationship between Engine Power and Price')

z = np.polyfit(x, y, 1)

p = np.poly1d(z)

plt.plot(x, p(x), color='red')

plt.tight\_layout()

plt.show()

# Task 3: Use regression analysis to identify variables

# Load the car data into a DataFrame

file\_path = "C:/Users/Sumit Prajapat/PycharmProjects/pythonBANKProject1/Car\_data.csv"

data = pd.read\_csv(file\_path)

# Select the relevant columns for the analysis

columns = ["Engine HP", "Engine Cylinders", "highway MPG", "city mpg", "Popularity", "MSRP"]

data = data[columns].dropna()

# Set the dependent variable (y) and independent variables (X)

y = data["MSRP"]

X = data.drop("MSRP", axis=1)

# Add a constant term to the independent variables

X = sm.add\_constant(X)

# Perform multiple linear regression

model = sm.OLS(y, X)

results = model.fit()

# Print the regression summary

print(results.summary())

# Plot the coefficient values as a bar chart

coefficient\_values = results.params[1:]

coefficient\_values.plot(kind="bar")

plt.xlabel("Variable")

plt.ylabel("Coefficient Value")

plt.title("Variable Coefficients in Regression Analysis")

plt.show()

# Task 4.A: Create a pivot table

manufacturer\_avg\_price = pd.pivot\_table(car\_data, index='Make', values=['MSRP'], aggfunc='mean')

print("Task 4.A: Pivot Table - Average Price of Cars by Manufacturer")

print(manufacturer\_avg\_price)

# Task 4.B: Create a bar chart that visualizes the relationship between manufacturer and average price

manufacturer\_avg\_price.plot(kind='bar', legend=False)

plt.xlabel('Manufacturer')

plt.ylabel('Average Price')

plt.title('Average Price of Cars by Manufacturer')

plt.tight\_layout()

plt.show()

# Task 5.A: Create a scatter plot

cylinder\_data = car\_data[['Engine Cylinders', 'highway MPG']].dropna()

x = cylinder\_data['Engine Cylinders']

y = cylinder\_data['highway MPG']

plt.scatter(x, y)

plt.xlabel('Number of Cylinders')

plt.ylabel('Highway MPG')

plt.title('Relationship between Number of Cylinders and Highway MPG')

z = np.polyfit(x, y, 1)

p = np.poly1d(z)

plt.plot(x, p(x), color='red')

plt.tight\_layout()

plt.show()

# Task 5.B: Calculate the correlation coefficient

correlation\_coef = cylinder\_data['Engine Cylinders'].corr(cylinder\_data['highway MPG'])

print("Task 5.B: Correlation Coefficient:", correlation\_coef)