```python
# -*- coding: utf-8 -*-
"""
Created on Fri Mar  7 11:54:54 2025

@author: PC
"""

"""
Robert Cocker
DATA 3461
Dr. Jawad
3/7/2025
Quiz #9 - Example 1
"""

# Example 1 - run 1 (Model 1)
# Import required libraries
import numpy as np
import matplotlib.pyplot as plt

# Dataset
x = np.array([2, 4, 5])
y = np.array([1.2, 2.8, 5.3])

# Function for Gradient Descent
def gradient_descent(x, y, lr=0.01, iterations=1):
    b0, b1 = 0, 1  # Initialize parameters

    for i in range(iterations):
        partial_wrt_b0 = 0
        partial_wrt_b1 = 0

        for j in range(len(x)):
            y_pred = b0 + b1 * x[j]  # Predict value for given x
            error_cost = y_pred - y[j]  # Calculate error

            partial_wrt_b0 += error_cost  # Partial derivative wrt b0
            partial_wrt_b1 += error_cost * x[j]  # Partial derivative wrt b1

        # Update values
        b0 -= lr * (2/len(x)) * partial_wrt_b0
        b1 -= lr * (2/len(x)) * partial_wrt_b1

    return b0, b1  # Return optimized values

# Different iteration values
iterations_list = [1, 100, 1000]

# Plot original scatter points
plt.scatter(x, y, label="Data Points", color="blue")

# Run gradient descent for different iterations and plot results
for iters in iterations_list:
    b0, b1 = gradient_descent(x, y, iterations=iters)
    y_pred = b0 + b1 * x  # Compute predictions
    plt.plot(x, y_pred, label=f"{iters} Iterations")
```

```
    # Print values for analysis
    print(f"After {iters} iterations: b0 = {b0:.4f}, b1 = {b1:.4f}")

# Final plot adjustments
plt.legend()
plt.xlabel("x")
plt.ylabel("y")
plt.title("Gradient Descent Convergence")
plt.show()

# Questions - Inference
"""
Lecture (class) initial b0, b1 values and equation:
b0 = -0.0113
b1 = 0.9677
y-hat = 0.9677*x_i - 0.0113

a.) Ran code and observed results (output). Our calculations in Python match the handwritten methoc
    b0 and b1 values confirm our handwritten calculations though Python is more precise with 2> dec
    placement in calculations.

b.) Ran 100 iterations

c.) Ran 1000 iterations. Results: After 1000 iterations: b0 = -1.3902, b1 = 1.2306. This suggests t
    parameters compared to our initial calculations during lecture. Initial calculations: After 1 i
"""

"""
References
- lab, lecture, course materials
- ChatGPT
- Google
"""
```