*#Anuprita Girme #COBA56*

import numpy as np

import matplotlib.pyplot as plt

def gradient\_descent(learning\_rate, max\_iterations, initial\_x): x = initial\_x

x\_history = []

for \_ in range(max\_iterations): gradient = 2 \* (x + 3)

x = x - learning\_rate \* gradient x\_history.append(x)

return x, x\_history learning\_rate = 0.1

max\_iterations = 1000

initial\_x = 2

local\_minimum, x\_history = gradient\_descent(learning\_rate, max\_iterations, initial\_x)

print(f"Local Minimum at x = {local\_minimum}") x\_values = np.linspace(-10, 10, 400)

y\_values = (x\_values + 3)\*\*2

plt.plot(x\_values, y\_values, label='y = (x + 3)^2', color='blue') plt.scatter(x\_history, [(x + 3)\*\*2 for x in x\_history], label='Gradient Descent Path', color='red', marker='x') plt.xlabel('x')

plt.ylabel('y') plt.legend()

plt.title('Gradient Descent Convergence') plt.grid(True)

plt.show()

Local Minimum at x = -2.999999999999999

