**3.Gradient Descent**

current\_x = 2 # The algorithm starts at x=3

rate = 0.01 # Learning rate

precision = 0.000001 #This tells us when to stop the algorithm

previous\_step\_size = 1

max\_iters = 10000 # maximum number of iterations

iters = 0 #iteration counter

df = lambda x: 2\*(x+3) #Gradient of our function

while previous\_step\_size > precision and iters < max\_iters: #When Previous Step SIze will be less than Precision then we will reach the global maxima

previous\_x = current\_x #Store current x value in prev\_x

current\_x = current\_x - rate \* df(previous\_x) #Grad descent

previous\_step\_size = abs(current\_x - previous\_x) #Change in x

iters = iters+1 #iteration count

print("Iteration",iters,"\nX value is",current\_x) #Print iterations

print("The local minimum occurs at", current\_x)