ML GRADIENT DESCENT 2

```
# //Gradient descent in Python :
# Step 1 : Initialize parameters
cur_x = 3 # 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+5) #Gradient of our function

# Step 2 : Run a loop to perform gradient descent :
# i. Stop loop when difference between x values from 2 consecutive iterations is less than 0.000001
# or when number of iterations exceeds 10,000
while previous_step_size > precision and iters < max_iters:
 prev_x = cur_x #Store current x value in prev_x
 cur_x = cur_x - rate * df(prev_x) #Grad descent
 previous_step_size = abs(cur_x - prev_x) #Change in x
 iters = iters+1 #iteration count
 print("Iteration",iters,"\nX value is",cur_x) #Print iterations
 print("The local minimum occurs at", cur_x)
```