

```
In [601]: List_No_of_rooms=[1,2,3,4,5,6,7]
          #List_Price_to_pay=[150,200,250,300,350,400,450]
```

```
In [602]: List_Price_to_pay=[150,190,260,320,370,385,490]
```

```
In [603]: print(List_No_of_rooms,",","Length of List_No_of_rooms=",len(List_No_of_rooms))
          [1, 2, 3, 4, 5, 6, 7] , Length of List_No_of_rooms= 7
```

```
In [604]: print(List_Price_to_pay,",","Length of List_Price_to_pay=",len(List_Price_to_pay))
          [150, 190, 260, 320, 370, 385, 490] , Length of List_Price_to_pay= 7
```

```
In [605]: diff_X=[]
          for i in range(0,len(List_No_of_rooms)):
              if(i<len(List_No_of_rooms)-1):
                  temp=List_No_of_rooms[i+1]-List_No_of_rooms[i]
                  diff_X.append(temp)
          print(diff_X,",","Length of diff_X=",len(diff_X))
          [1, 1, 1, 1, 1, 1] , Length of diff_X= 6
```

```
In [606]: diff_Y=[]
          for i in range(0,len(List_Price_to_pay)):
              if(i<len(List_Price_to_pay)-1):
                  temp=List_Price_to_pay[i+1]-List_Price_to_pay[i]
                  diff_Y.append(temp)
          print(diff_Y,",","Length of diff_Y=",len(diff_X))
          [40, 70, 60, 50, 15, 105] , Length of diff_Y= 6
```

```
In [607]: #calculate m(gradient)
          gard=0
          grad=[]

          for i in range(0,len(diff_X)):
              for j in range(i,len(diff_Y)):
                  temp=diff_Y[j]/diff_X[i]
                  grad.append(temp)
                  break
          print(grad)
          [40.0, 70.0, 60.0, 50.0, 15.0, 105.0]
```

```
In [608]: # using naive method to
# get most frequent element
#grad=[5,15,20,10,7,10]
max = 0
most_frequent = grad[0]
print(most_frequent)
for i in grad:
    freq = grad.count(i)
    if freq > max:
        max = freq
        most_frequent = i
        print(most_frequent)
# printing result
print ("Most frequent number is : " + str(most_frequent))
```

40.0

40.0

Most frequent number is : 40.0

```
In [609]: print(len(List_No_of_rooms))
print(len(bias))
print(len(grad))
sum=0
for i in range(0,len(grad)):
    sum=sum+grad[i]
avg=sum/len(grad)
print(avg)
```

7

7

6

56.666666666666664

```
In [610]: #grad.append(most_frequent)
grad.append(avg)
print(grad)
print(len(grad))
```

[40.0, 70.0, 60.0, 50.0, 15.0, 105.0, 56.666666666666664]

7

```
In [611]: bias=[]
          for i in range (0,len(List_Price_to_pay)):
              for j in range(i,len(List_No_of_rooms)):
                  temp=List_Price_to_pay[i]-(List_No_of_rooms[j]*grad[j])
                  bias.append(temp)
                  break
          print(bias)
          print(len(bias))

[110.0, 50.0, 80.0, 120.0, 295.0, -245.0, 93.33333333333337]
7
```

```
In [612]: #cross verify y=mx+c
          # Price = Rate of change Y over X +bias
```

```
In [613]: Y_pred=[]
          for i in range (0,len(List_No_of_rooms)-1):
              for j in range(i,len(bias)-1):
                  guess_formulla=grad[j]*List_No_of_rooms[i]+bias[j]
                  Y_pred.append(guess_formulla)
                  break

          print(Y_pred)
          print(len(Y_pred))

[150.0, 190.0, 260.0, 320.0, 370.0, 385.0]
6
```

```
In [614]: Y_pred=[]
for i in range (0,len(List_No_of_rooms)):
    #guess_formulla=grad[i]*List_No_of_rooms[i]+bias[i]
    guess_formulla=grad[i]*List_No_of_rooms[i]+100
    print("gardiant=",grad[i])
    print("bias=",bias[i])
    print("No of rooms=",List_No_of_rooms[i])
    Y_pred.append(guess_formulla)
    print("Predicted Price=",Y_pred,"\n")

print(Y_pred)
print(len(Y_pred))
```

```
gardiant= 40.0
bias= 110.0
No of rooms= 1
Predicted Price= [140.0]
```

```
gardiant= 70.0
bias= 50.0
No of rooms= 2
Predicted Price= [140.0, 240.0]
```

```
gardiant= 60.0
bias= 80.0
No of rooms= 3
Predicted Price= [140.0, 240.0, 280.0]
```

```
gardiant= 50.0
bias= 120.0
No of rooms= 4
Predicted Price= [140.0, 240.0, 280.0, 300.0]
```

```
gardiant= 15.0
bias= 295.0
No of rooms= 5
Predicted Price= [140.0, 240.0, 280.0, 300.0, 175.0]
```

```
gardiant= 105.0
bias= -245.0
No of rooms= 6
Predicted Price= [140.0, 240.0, 280.0, 300.0, 175.0, 730.0]
```

```
gardiant= 56.666666666666664
bias= 93.33333333333337
No of rooms= 7
Predicted Price= [140.0, 240.0, 280.0, 300.0, 175.0, 730.0, 496.66666666666663]
```

```
[140.0, 240.0, 280.0, 300.0, 175.0, 730.0, 496.66666666666663]
```

```
7
```

```
In [615]: print("Actual Price",List_Price_to_pay)
          print("Predicted Price",Y_pred)
```

```
Actual Price [150, 190, 260, 320, 370, 385, 490]
```

```
Predicted Price [140.0, 240.0, 280.0, 300.0, 175.0, 730.0, 496.66666666666663]
```

```
In [616]: for i in range(0,len(List_Price_to_pay)):
          if(List_Price_to_pay[i]==Y_pred[i]):
              print("Formuall Found")
```

```
In [617]: print(grad)
          print(bias)
```

```
[40.0, 70.0, 60.0, 50.0, 15.0, 105.0, 56.666666666666664]
```

```
[110.0, 50.0, 80.0, 120.0, 295.0, -245.0, 93.33333333333337]
```

```
In [618]: X=(int(input("Enter the of roons you want to book=")))
```

```
Y_pred_unknown=grad[0]*X +bias[0]
print(Y_pred_unknown)
```

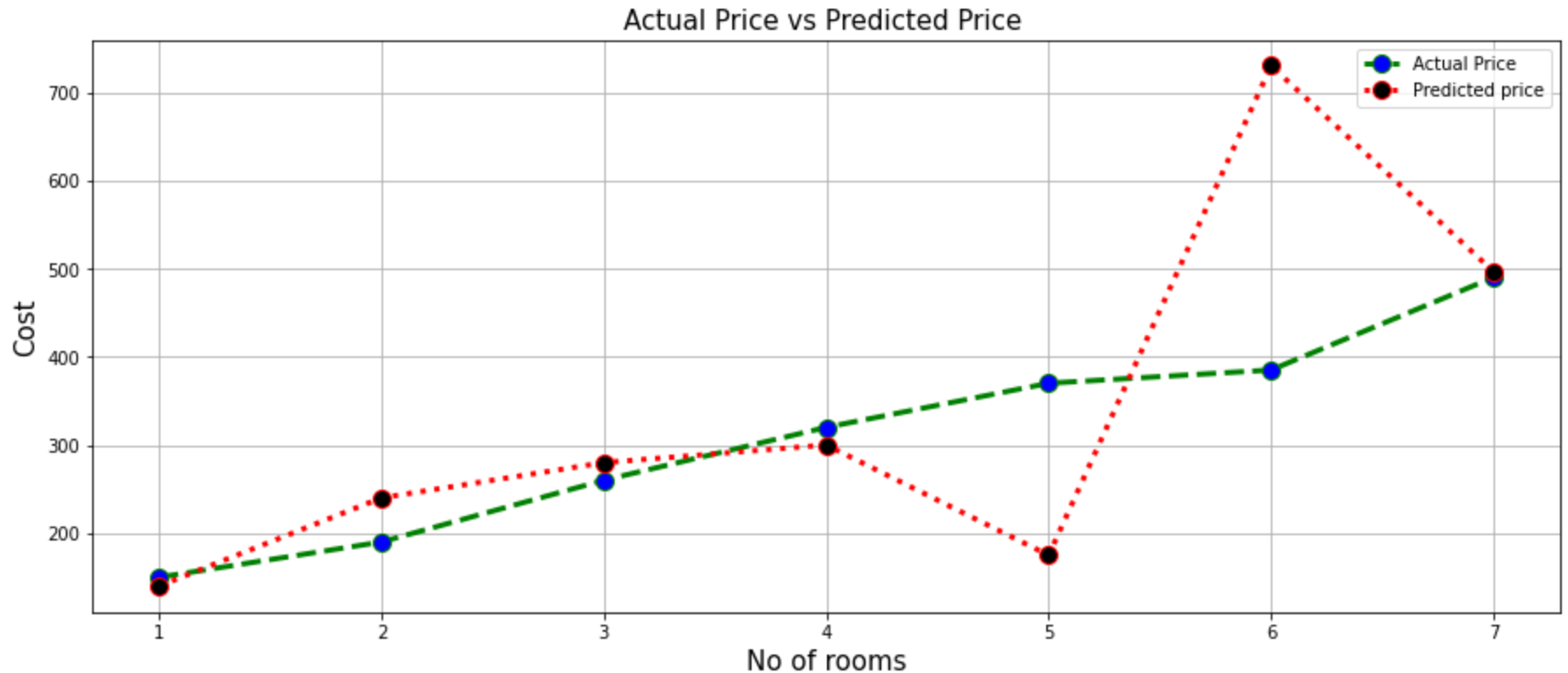
```
Enter the of roons you want to book=500
20110.0
```

```
In [619]: import matplotlib.pyplot as plt
plt.figure(figsize=(15,6))
plt.plot(List_No_of_rooms,List_Price_to_pay, label = "Actual Price",color='green', linestyle='dashed', linewidth=3,
         marker='o', markerfacecolor='blue', markersize=10)

plt.plot(List_No_of_rooms, Y_pred, label = "Predicted price",color='red', linestyle='dotted', linewidth = 3,
         marker='o', markerfacecolor='black', markersize=10)

# naming the x axis
plt.xlabel('No of rooms',fontsize = 15)
# naming the y axis
plt.ylabel('Cost',fontsize = 15)
# giving a title to my graph
plt.title('Actual Price vs Predicted Price ',fontsize = 15)

# show a legend on the plot
plt.legend()
plt.grid(True)
# function to show the plot
plt.show()
```



In [ ]: