To – Do – 1 :

#2

```
First 5 rows:
   Math  Reading  Writing
0    48       68       63
1    62       81       72
2    79       80       78
3    76       83       79
4    59       64       62

Last 5 rows:
     Math  Reading  Writing
995    72       74       70
996    73       86       90
997    89       87       94
998    83       82       78
999    66       66       72
```

#3

```
Info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 3 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   Math     1000 non-null   int64
 1   Reading  1000 non-null   int64
 2   Writing  1000 non-null   int64
dtypes: int64(3)
memory usage: 23.6 KB
```

#4

[7]:

Descriptive info:

	Math	Reading	Writing
count	1000.000000	1000.000000	1000.000000
mean	67.290000	69.872000	68.616000
std	15.085008	14.657027	15.241287
min	13.000000	19.000000	14.000000
25%	58.000000	60.750000	58.000000
50%	68.000000	70.000000	69.500000
75%	78.000000	81.000000	79.000000
max	100.000000	100.000000	100.000000

To – Do – 2 :

```
X shape: (2, 1000)
W shape: (2, 1)
Y shape: (1000,)
Y_pred shape: (1000, 1)
```

To – Do – 5 :

```
Proceed Further
Cost function output: 0.0
```

To – Do – 7 :

```
Final Parameters: [[0.20551667]
 [0.54295081]
 [0.10388027]]
Cost History: [np.float64(0.10711197094660153), np.float64(0.10634880599939901), np.float64(0.10559826315680618), np.float64(0.10486012948320558), n
p.float64(0.1041341956428534), np.float64(0.10342025583900626), np.float64(0.1027181077540776), np.float64(0.1020275524908062), np.float64(0.10134839
451441931), np.float64(0.1006804415957737), np.float64(0.1000235047554587), np.float64(0.09937739820884377), np.float64(0.09874193931205609), np.floa
t64(0.09811694850887098), np.float64(0.09750224927850094), np.float64(0.0968976680842672), np.float64(0.09630303432313951), np.float64(0.095718180276
12913), np.float64(0.09514294105952065), np.float64(0.09457715457692842), np.float64(0.09402066147216397), np.float64(0.09347330508290017), np.float6
4(0.09293493139511913), np.float64(0.09240538899833017), np.float64(0.09188452904154543), np.float64(0.0913722051899995), np.float64(0.09086827358260
123), np.float64(0.09037259279010502), np.float64(0.08988502377398919), np.float64(0.08940542984603007), np.float64(0.08893367662855953), np.float64
(0.08846963201539432), np.float64(0.08801316613342668), np.float64(0.08756415130486386), np.float64(0.08712246201010665), np.float64(0.08668797485125
508), np.float64(0.08626056851623207), np.float64(0.08584012374351278), np.float64(0.08542652328745133), np.float64(0.08501965188419301), np.float64
(0.0846193962181636), np.float64(0.08422564488912489), np.float64(0.08383828837978763), np.float64(0.08345721902397185), np.float64(0.083082330975305
82), np.float64(0.08271352017645425), np.float64(0.08235068432886682), np.float64(0.08199372286303817), np.float64(0.08164253690927113), np.float64
(0.08129702926893387), np.float64(0.08095710438620353), np.float64(0.08062266832028739), np.float64(0.08029362871811391), np.float64(0.07996989478748
553), np.float64(0.0796513772706855), np.float64(0.07933798841853089), np.float64(0.07902964196486459), np.float64(0.07872625310147845), np.float64
(0.07842773845346054), np.float64(0.07813401605495938), np.float64(0.0778450053253578), np.float64(0.0775606270458499), np.float64(0.07728080333641404), np.float64(0.07700545763317514), np.float64(0.07673451466614989), np.float64(0.07646790043736812), np.float64(0.07620554219936448), np.float64(0.
```

```
4), np.float64(0.07700545763317514), np.float64(0.07673451466614989), np.float64(0.07646790043736812), np.float64(0.07620554219936448), np.float64(0.
07594736843403344), np.float64(0.07569330883184205), np.float64(0.07544329427139428), np.float64(0.07519725679934074), np.float64(0.0749551296106282
1), np.float64(0.07471684702908327), np.float64(0.07448234448832412), np.float64(0.0742515585129952), np.float64(0.07402442670031911), np.float64(0.0
7380088770196077), np.float64(0.07358088120619749), np.float64(0.0733643479203919), np.float64(0.07315122955375959), np.float64(0.07294146880042966),
np.float64(0.07273500932279067), np.float64(0.07253179573511871), np.float64(0.07233177358748233), np.float64(0.0721348893499193), np.float64(0.07194
109039688139), np.float64(0.07175032499194182), np.float64(0.07156254227276149), np.float64(0.07137769223630935), np.float64(0.07119572572433286), n
p.float64(0.07101659440907385), np.float64(0.07084025077922623), np.float64(0.0706666648126131), np.float64(0.07049574053020462), np.float64(0.0703274
8284759716), np.float64(0.07016183069707572), np.float64(0.06999874044712999), np.float64(0.06983816920329523), np.float64(0.06968007479569092), np.fl
oat64(0.06952441576676843), np.float64(0.06937115135926715), np.float64(0.06922024150437375), np.float64(0.06907164681008185), np.float64(0.068925328
54974835), np.float64(0.0687812486508435), np.float64(0.06863936968389095), np.float64(0.06849965485159508), np.float64(0.06836206797815195), np.floa
t64(0.06822657349874123), np.float64(0.06809313644919561), np.float64(0.067961722455845), np.float64(0.06783229772553254), np.float64(0.0677048290357
9932), np.float64(0.06757928372523506), np.float64(0.06745562968399212), np.float64(0.06733383534445969), np.float64(0.06721386967209597), np.float64
(0.06709570215641501), np.float64(0.06697930280212627), np.float64(0.06686464212042395), np.float64(0.06675169112042348), np.float64(0.06664042130074
29), np.float64(0.06653080464122665), np.float64(0.06642281359480932), np.float64(0.0663164210795167), np.float64(0.06621160047060279), np.float64
(0.06610832559281864), np.float64(0.06600657071281309), np.float64(0.0659063105316614), np.float64(0.06580752017752023), np.float64(0.065710175198406
98), np.float64(0.06561425155510119), np.float64(0.06551972561416586), np.float64(0.06542657414108709), np.float64(0.06533477429352925), np.float64
(0.06524430361470467), np.float64(0.06515514002685512), np.float64(0.06506726182484374), np.float64(0.06498064766985515), np.float64(0.06489527658320
228), np.float64(0.06481112794023773), np.float64(0.06472818146436811), np.float64(0.0646464172211699), np.float64(0.06456581561260431), np.float64
```

```
46), np.float64(0.06410556476968135), np.float64(0.06403257994440141), np.float64(0.0639606124166433), np.float64(0.06388964537111992), np.float64(0.
06381966227619645), np.float64(0.06375064687909507), np.float64(0.06368258320118077), np.float64(0.06361545553332655), np.float64(0.0635492484313575
5), np.float64(0.06348394671157162), np.float64(0.06341953544633615), np.float64(0.06335599995975896), np.float64(0.06329332582343267), np.float64(0.
06323149885225086), np.float64(0.06317050510029515), np.float64(0.06311033085679153), np.float64(0.06305096264213547), np.float64(0.0629923872039838
4), np.float64(0.0629345915134133), np.float64(0.06287756276114324), np.float64(0.06282128835382297), np.float64(0.0627675559103815), np.float64(0.06
271095325843898), np.float64(0.06265686843077901), np.float64(0.06260348966188053), np.float64(0.06255080538450809), np.float64(0.06249880422636036),
np.float64(0.06244747500677472), np.float64(0.06239680673348793), np.float64(0.06234678859945137), np.float64(0.06229740997970036), np.float64(0.0622
48660428276 2), np.float64(0.06220052967520031), np.float64(0.062153007623499706), np.float64(0.062106084346282515), np.float64(0.062059750083863094),
np.float64(0.06201399524093575), np.float64(0.06196881038379625), np.float64(0.061924186237610215), np.float64(0.06188011368372787), np.float64(0.061
8365837570441), np.float64(0.06179358764340313), np.float64(0.061751116677047156), np.float64(0.06170916233810801), np.float64(0.0616677162501414), n
p.float64(0.06162677017770278), np.float64(0.061586316023964055), np.float64(0.0615463458283708), np.float64(0.06150685176433905), np.float64(0.06146
782613699094), np.float64(0.0614292613809287), np.float64(0.061391150058046254), np.float64(0.06135348485537795), np.float64(0.061311625858298352), n
p.float64(0.06127946417186706), np.float64(0.06124309467194143), np.float64(0.061207143250002184), np.float64(0.06117160318776841), np.float64(0.061
13646787993252), np.float64(0.061101730832252524), np.float64(0.06106738565967507), np.float64(0.06103342608449018), np.float64(0.06099984593451716),
np.float64(0.06096663914132128), np.float64(0.0609333797384604), np.float64(0.0609013218597616), np.float64(0.06086919973762659), np.float64(0.060837
42770136588), np.float64(0.06080600017556133), np.float64(0.06077491167845612), np.float64(0.06074415682037193), np.float64(0.06071373030215326), np.
float64(0.060683626913637524), np.float64(0.06065384153215141), np.float64(0.06062436912103256), np.float64(0.0605952047281761), np.float64(0.0605663
4348460599), np.float64(0.060537780603070336), np.float64(0.06050511376660545), np.float64(0.0604815311774538), np.float64(0.060453835455178496), n
p.float64(0.060426419735902281), np.float64(0.060399279620742161), np.float64(0.060372410784596583), np.float64(0.060345809074898815), np.float64(0.060
```

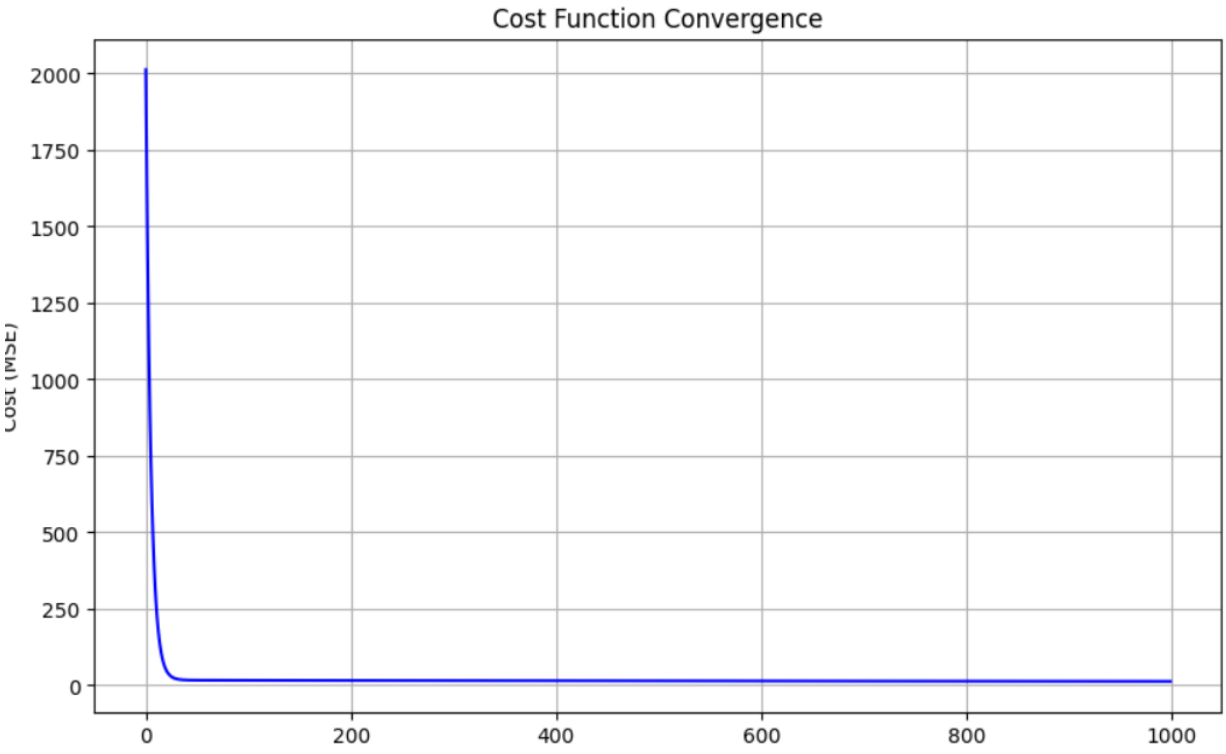
```
p.float64(0.059262785383511338), np.float64(0.059245653124979226), np.float64(0.059228640154671547), np.float64(0.059211744759755505), np.float64(0.0591
94965255076046), np.float64(0.05917829998361292), np.float64(0.05916174731582212), np.float64(0.05914530564917231 5), np.float64(0.05912897340768892
6), np.float64(0.059112749041506096), np.float64(0.05909663102642617), np.float64(0.05908061786348662), np.float64(0.059064708078534194), np.float64
(0.05904890022180654), np.float64(0.05903319286752055), np.float64(0.05901758461346795), np.float64(0.05900207408061755), np.float64(0.05898665991272
4324), np.float64(0.05897134077594505), np.float64(0.058956115358460404), np.float64(0.05894098237010357), np.float64(0.05892594054199501), np.float64
(0.05891098862618344), np.float64(0.05889612539529293), np.float64(0.05888134964217589), np.float64(0.05886666017957195), np.float64(0.05885205583997
72675), np.float64(0.05883753547429179), np.float64(0.058823097953541174), np.float64(0.058808742166512155), np.float64(0.05879446702046235), np.floa
t64(0.058780271440607684), np.float64(0.05876615436981961), np.float64(0.05875211476832761), np.float64(0.05873815161342641), np.float64(0.0587242638
9918856), np.float64(0.058710450636181515), np.float64(0.05869671085118971), np.float64(0.058683043586941104), np.float64(0.058669447901838714), np.f
loat64(0.05865592286969638), np.float64(0.05864246757947903), np.float64(0.05862908113504752), np.float64(0.05861576265490756), np.float64(0.05860251
1271963004), np.float64(0.05858932613327336), np.float64(0.058576206399815284), np.float64(0.05856315124624814), np.float64(0.05855015986068357), np.
float64(0.05853723144445888), np.float64(0.05852436521191438), np.float64(0.05851156039017436), np.float64(0.0584988162189318), np.float64(0.05848613
195023677), np.float64(0.05847350684828838), np.float64(0.05846094018923017 6), np.float64(0.05844843126094919), np.float64(0.05843597936287807), np.f
loat64(0.05842358380580092), np.float64(0.05841124391166213), np.float64(0.05839895901337858), np.float64(0.05838672845465502), np.float64(0.05837455
158980245), np.float64(0.05836242778355972), np.float64(0.05835035641091811), np.float64(0.05833833685694878), np.float64(0.05832636851663321), np.fl
oat64(0.05831445079469655), np.float64(0.05830258310544367), np.float64(0.05829076487259809), np.float64(0.05827899552914358), np.float64(0.058267274
51716845), np.float64(0.058255601287712386), np.float64(0.0582439753006161), np.float64(0.058232396024373266), np.float64(0.05822208629359851 1), np.fl
oat64(0.05820937552081735 5), np.float64(0.05819793372459756), np.float64(0.05818653569258779), np.float64(0.05817518229082684), np.float64(0.0581638
```

To – Do – 10 :

```
Final Weights: [[0.34811659]
 [0.64614558]]
Cost History (First 10 iterations): [np.float64(2013.165570783755), np.float64(1640.286832599692), np.float64(1337.0619994901588), np.float64(1090.4794
892850578), np.float64(889.9583270083234), np.float64(726.8940993009545), np.float64(594.2897260808594), np.float64(486.4552052951635), np.float64(398.
7634463599484), np.float64(327.4517147324688)]
RMSE on Test Set: 5.2798239764188635
R-Squared on Test Set: 0.8886354462786421
```

To – Do – 11 :

1.



The model's performance is acceptable and the cost function decreases smoothly indicating proper convergence. The RMSE value being around 5.28 shows that the prediction error is low and R-squared value of approx 0.89 indicates that the model explains most of the variance in writing marks. Therefore, the model neither underfits nor overfits the data.

2.

```
Learning rate: 1e-06
  Final Cost: 16.535602355147176
  RMSE: 5.856694748793876
  R²: 0.8629707528684534

Learning rate: 1e-05
  Final Cost: 13.150619992105618
  RMSE: 5.2798239764188635
  R²: 0.8886354462786421

Learning rate: 0.0001
  Final Cost: 10.26076310841341
  RMSE: 4.792607360540954
  R²: 0.908240340333986
```

According to the data above, as the learning rate increases from 0.000001 to 0.0001, the model shows improved convergence with decrease in final cost and RMSE and increase in R-Squared. Therefore, smaller learning rate results in slower convergence while moderate learning rate results in quicker covergence and is more effective but with bigger learning rate it might start to diverge.