DEENA 20104016

importing libraries

LINEAR REGRESSION

In [1]:
```
import pandas as pd
import numpy as np
```

In [2]:
```
data = pd.read_csv("22_countries.csv")
```

Out[2]:

	id	name	iso3	iso2	numeric_code	phone_code	capital	currency	currency_na
0	1	Afghanistan	AFG	AF	4	93	Kabul	AFN	Afghan afgh
1	2	Aland Islands	ALA	AX	248	+358-18	Mariehamn	EUR	E
2	3	Albania	ALB	AL	8	355	Tirana	ALL	Albanian
3	4	Algeria	DZA	DZ	12	213	Algiers	DZD	Algerian di
4	5	American Samoa	ASM	AS	16	+1-684	Pago Pago	USD	US Dc
...	...	...	...	...	...	...	...	...	
245	243	Wallis And Futuna Islands	WLF	WF	876	681	Mata Utu	XPF	CFP fr
246	244	Western Sahara	ESH	EH	732	212	El-Aaiun	MAD	Moroc Dirh
247	245	Yemen	YEM	YE	887	967	Sanaa	YER	Yemeni
248	246	Zambia	ZMB	ZM	894	260	Lusaka	ZMW	Zamb kwa
249	247	Zimbabwe	ZWE	ZW	716	263	Harare	ZWL	Zimbak Dc

250 rows × 19 columns

In [3]:

```
[illegible]
```

Out[3]:

	id	name	iso3	iso2	numeric_code	phone_code	capital	currency	currency_name
0	1	Afghanistan	AFG	AF	4	93	Kabul	AFN	Afghan afghani
1	2	Aland Islands	ALA	AX	248	+358-18	Mariehamn	EUR	Euro
2	3	Albania	ALB	AL	8	355	Tirana	ALL	Albanian lek
3	4	Algeria	DZA	DZ	12	213	Algiers	DZD	Algerian dinar
4	5	American Samoa	ASM	AS	16	+1-684	Pago Pago	USD	US Dollar

In [4]:

Out[4]:

```
<bound method DataFrame.info of       id                         name iso3 iso2  numeric_code phone_code  \
0      1                  Afghanistan  AFG   AF             4         93
1      2                Aland Islands  ALA   AX           248    +358-18
2      3                      Albania  ALB   AL             8        355
3      4                      Algeria  DZA   DZ            12        213
4      5               American Samoa  ASM   AS            16     +1-684
..   ...                          ...  ...  ...           ...        ...
245  243  Wallis And Futuna Islands  WLF   WF           876        681
246  244               Western Sahara  ESH   EH           732        212
247  245                        Yemen  YEM   YE           887        967
248  246                       Zambia  ZMB   ZM           894        260
249  247                     Zimbabwe  ZWE   ZW           716        263

       capital currency    currency_name currency_symbol  tld  \
0        Kabul      AFN   Afghan afghani               ؋  .af
1    Mariehamn      EUR             Euro               €  .ax
2       Tirana      ALL      Albanian lek             Lek  .al
3      Algiers      DZD   Algerian dinar              دج  .dz
4    Pago Pago      USD        US Dollar               $  .as
..         ...      ...              ...             ...  ...
245   Mata Utu      XPF        CFP franc               ₣  .wf
246   El-Aaiun      MAD  Moroccan Dirham             MAD  .eh
247      Sanaa      YER      Yemeni rial             ريال  .ye
248     Lusaka      ZMW    Zambian kwacha              ZK  .zm
249     Harare      ZWL  Zimbabwe Dollar               $  .zw

               native   region        subregion  \
0           افغانستان     Asia     Southern Asia
1               Åland   Europe  Northern Europe
2            Shqipëria   Europe  Southern Europe
3              الجزائر   Africa  Northern Africa
4       American Samoa  Oceania        Polynesia
..                 ...      ...              ...
245  Wallis et Futuna  Oceania        Polynesia
246    الصحراء الغربية   Africa  Northern Africa
247              اليَمَن     Asia     Western Asia
248             Zambia   Africa   Eastern Africa
249           Zimbabwe   Africa   Eastern Africa

                                             timezones   latitude  longitude  
\
0    [{zoneName:'Asia\/Kabul',gmtOffset:16200,gmtOf...  33.000000       65.0
1    [{zoneName:'Europe\/Mariehamn',gmtOffset:7200,...  60.116667       19.9
2    [{zoneName:'Europe\/Tirane',gmtOffset:3600,gmt...  41.000000       20.0
3    [{zoneName:'Africa\/Algiers',gmtOffset:3600,gm...  28.000000        3.0
4    [{zoneName:'Pacific\/Pago_Pago',gmtOffset:-396... -14.333333     -170.0
..                                                 ...        ...        ...
245  [{zoneName:'Pacific\/Wallis',gmtOffset:43200,g... -13.300000     -176.2
246  [{zoneName:'Africa\/El_Aaiun',gmtOffset:3600,g...  24.500000      -13.0
247  [{zoneName:'Asia\/Aden',gmtOffset:10800,gmtOff...  15.000000       48.0
248  [{zoneName:'Africa\/Lusaka',gmtOffset:7200,gmt... -15.000000       30.0
249  [{zoneName:'Africa\/Harare',gmtOffset:7200,gmt... -20.000000       30.0
```

```
    emoji           emojiU
0      🇦🇫  U+1F1E6 U+1F1EB
1      🇦🇽  U+1F1E6 U+1F1FD
2      🇦🇱  U+1F1E6 U+1F1F1
3      🇩🇿  U+1F1E9 U+1F1FF
4      🇦🇸  U+1F1E6 U+1F1F8
..    ...              ...
245    🇼🇫  U+1F1FC U+1F1EB
246    🇪🇭  U+1F1EA U+1F1ED
247    🇾🇪  U+1F1FE U+1F1EA
248    🇿🇲  U+1F1FF U+1F1F2
249    🇿🇼  U+1F1FF U+1F1FC

[250 rows x 19 columns]>
```

In [5]:

Out[5]:

	id	numeric_code	latitude	longitude
count	250.000000	250.00000	250.000000	250.00000
mean	125.500000	435.80400	16.402597	13.52387
std	72.312977	254.38354	26.757204	73.45152
min	1.000000	4.00000	-74.650000	-176.20000
25%	63.250000	219.00000	1.000000	-49.75000
50%	125.500000	436.00000	16.083333	17.00000
75%	187.750000	653.50000	39.000000	48.75000
max	250.000000	926.00000	78.000000	178.00000

Train the model

In [6]:

```python
x = data[['id']]
```

In [7]:

```python
# to split my dataset into training and test data
from sklearn.model_selection import train_test_split
```

In [8]:

```python
from sklearn.linear_model import LinearRegression
lr = LinearRegression()
```

Out[8]: LinearRegression()

In [9]:

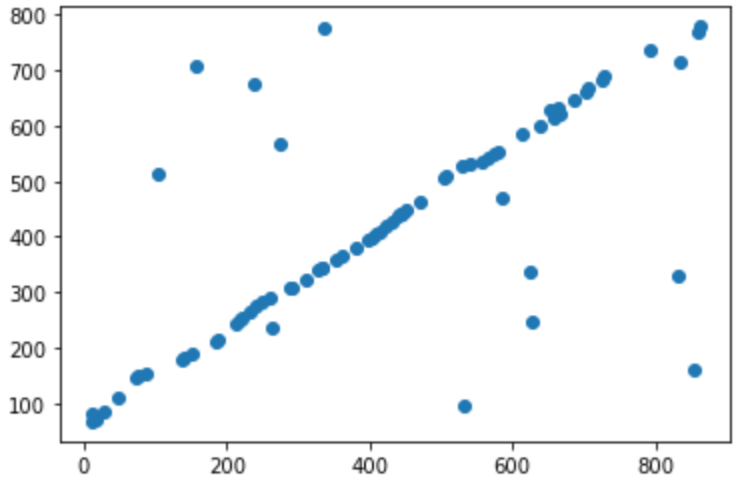
```python
coeff = pd.DataFrame(lr.coef_,x.columns,columns=['Co-efficient'])
```

Out[9]:

	Co-efficient
id	3.027671

In [10]:
```
prediction= lr.predict(x_test)
```

Out[10]: <matplotlib.collections.PathCollection at 0x2d1561828b0>



In [11]:

Out[11]: 0.46362632452912544

LASSO AND RIDGE

In [12]:
```
from sklearn.linear_model import Ridge,Lasso
rr=Ridge(alpha=10)
```

Out[12]: Ridge(alpha=10)

In [13]:

Out[13]: 0.4636288650423135

In [14]:
```
la=Lasso(alpha=10)
```

Out[14]: Lasso(alpha=10)

In [15]:

Out[15]: 0.46377291785429653

ELASTICNET

In [16]:
```
from sklearn.linear_model import ElasticNet
a=ElasticNet()
```

Out[16]: ElasticNet()

In [17]:

```
print(a.coef_)
print(a.intercept_)
print(a.score(x_test,y_test))
```

```
[3.02731073]
54.94907688422035
0.46365588434303895
[509.04568613 506.01837541 439.41753938 708.8481942  645.27466891
  67.0583198  681.60239765  70.08563053 512.07299686  82.19487344
 245.66965277 148.79570946 394.00787846 736.09399076 345.5709068
 248.6969635  421.25367501 542.34610415 179.06881675 188.15074893
 687.6570191  215.39654549 669.49315473 339.51628535 182.09612747
 109.44066999 321.35242098 254.75158495 275.94276005 527.2095505
 536.29148269 769.39440877 357.68014972 151.82302019 778.47634095
 330.43435316 342.54359608 463.63602521 291.07931369 714.90281566
 599.86500798 551.42803633 660.41122255 263.83351714 409.1444321
 436.39022865 309.24317806 406.11712137  85.22218417 236.58772058
 378.87132482 397.03518919 212.36923476 584.72845434 336.48897462
 548.4007256  427.30829647 281.99738151 630.13811527 675.54777619
 160.90495238 621.05618308 566.56458997 306.21586734 530.23686123
 242.64234204 145.76839873 627.11080454 363.73477117  94.30411635
 615.00156162 448.49947157 469.69064667 442.44485011 775.44903023]
```

In [18]:

```
from sklearn import metrics
print(" Mean Absolute Error :",metrics.mean_absolute_error(y_test,prediction))
print(" Mean Squared Error :",metrics.mean_squared_error(y_test,prediction))
```

```
 Mean Absolute Error : 86.40826403053133
 Mean Squared Error : 29362.85964082914
 Root Mean Absolute Error : 9.295604554332726
```

PREDICTION

In [19]:

```
import pickle
fn="prediction"
```

In [20]:

```
import pandas as pd
import pickle
fn="prediction"
```

In [21]:

```
r=[[10],[20]]
result=m.predict(r)
```

Out[21]:

```
array([ 85.17935945, 115.45606942])
```