In [1]:
```python
import pandas as pd
```

In [4]:
```python
A=pd.read_csv(r"C:\Users\stud\Downloads\temperatures.csv")
```

In [5]:
```python
#Importing required libraries
```

In [6]:
```python
import numpy as np
```

In [7]:
```python
import seaborn as sb
```

In [8]:
```python
import matplotlib.pyplot as plt
```

In [9]:
```python
A.describe()
```

Out[9]:

	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL	JAN-FEB	MAR-MA
count	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.00000
mean	1959.000000	23.687436	25.597863	29.085983	31.975812	33.565299	32.774274	31.035897	30.507692	30.486752	29.766581	27.285470	24.608291	29.181368	24.629573	31.51760
std	33.919021	0.834588	1.150757	1.068451	0.889478	0.724905	0.633132	0.468818	0.476312	0.544295	0.705492	0.714518	0.782644	0.555555	0.911239	0.74058
min	1901.000000	22.000000	22.830000	26.680000	30.010000	31.930000	31.100000	29.760000	29.310000	29.070000	27.900000	25.700000	23.020000	28.110000	22.250000	29.92000
25%	1930.000000	23.100000	24.780000	28.370000	31.460000	33.110000	32.340000	30.740000	30.180000	30.120000	29.380000	26.790000	24.040000	28.760000	24.110000	31.04000
50%	1959.000000	23.680000	25.480000	29.040000	31.950000	33.510000	32.730000	31.000000	30.540000	30.520000	29.780000	27.300000	24.660000	29.090000	24.530000	31.47000
75%	1988.000000	24.180000	26.310000	29.610000	32.420000	34.030000	33.180000	31.330000	30.760000	30.810000	30.170000	27.720000	25.110000	29.470000	25.150000	31.89000
max	2017.000000	26.940000	29.720000	32.620000	35.380000	35.840000	34.480000	32.760000	31.840000	32.220000	32.290000	30.110000	28.010000	31.630000	28.330000	34.57000

In [22]:
```python
#creaating model for jan month
```

In [10]:
```python
X=A[["YEAR"]]
```

In [11]:
```python
Y=A[["JAN"]]
```

In [ ]:
```python
#spliting data
```

In [12]:
```python
from sklearn.model_selection import train_test_split
```

In [13]:
```python
Xtrain,Xtest,Ytrain,Ytest=train_test_split(X,Y,test_size=0.2)
```

In [ ]:
```python
#creating model
```

In [15]:
```python
from sklearn.linear_model import LinearRegression
```

In [16]:
```python
reg=LinearRegression()
```

In [17]:
```python
#training our model
```

In [18]:
```python
model=reg.fit(Xtrain,Ytrain)
```

In [19]:
```python
#predicting dataset
```

In [20]:
```python
Ypredict=reg.predict(Xtest)
```

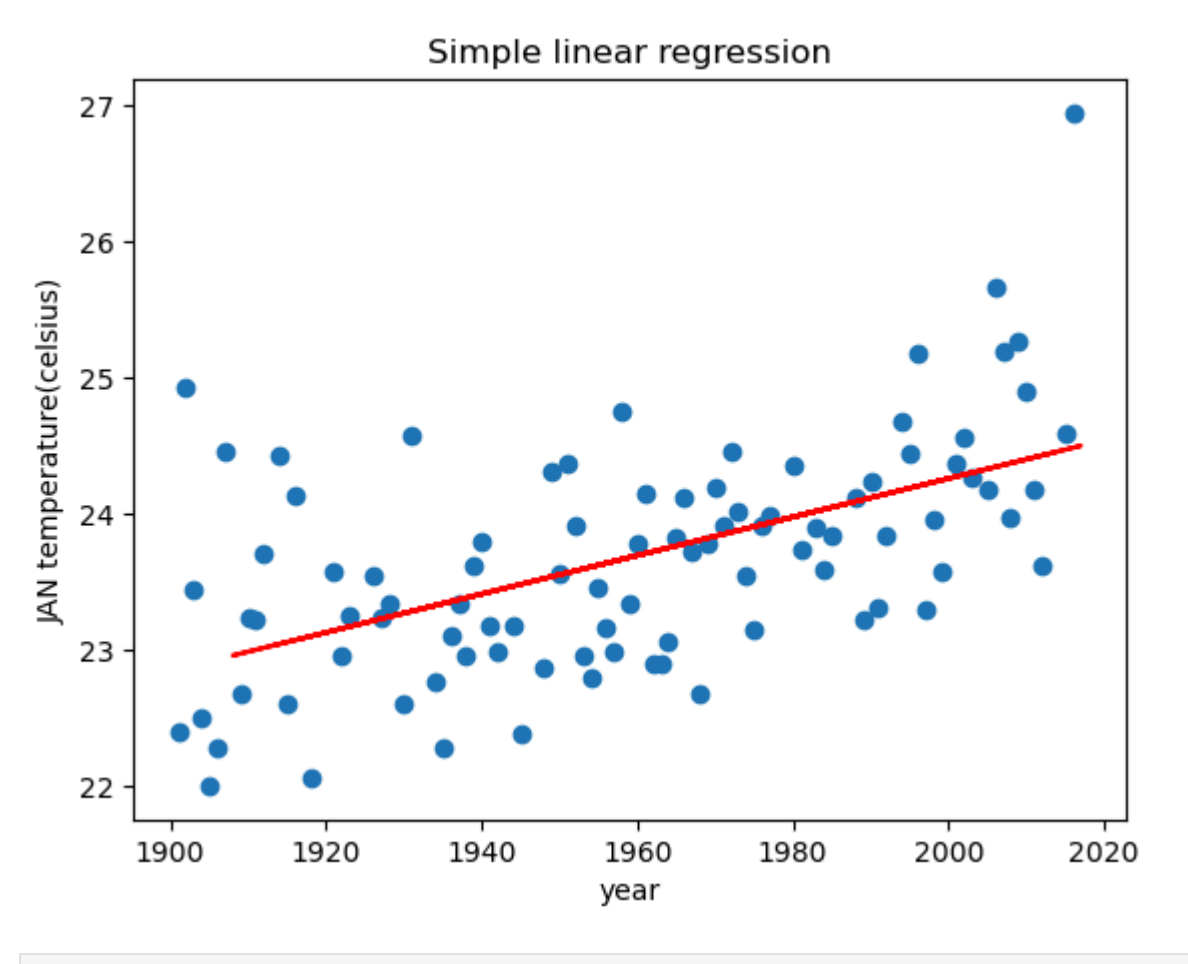
In [21]:
```python
print(Ypredict)
```

```
[[23.1961616 ]
 [24.3124067 ]
 [23.12551317]
 [23.29506939]
 [24.05807238]
 [23.11138349]
 [23.95916458]
 [23.45049593]
 [24.25588796]
 [23.30919908]
 [24.07220206]
 [23.9450349 ]
 [24.00155364]
 [24.15698017]
 [23.08312412]
 [24.49609261]
 [23.25268034]
 [23.02660538]
 [23.18203191]
 [23.50701467]
 [22.95595696]
 [24.43957387]
 [23.49288498]
 [24.45370355]]
```

In [35]:
```python
#ploting graph based on trained model trained model
```

In [29]:
```python
plt.scatter(Xtrain,Ytrain)
plt.plot(Xtest,Ypredict,color="red")
plt.title('Simple linear regression')
plt.xlabel('year')
plt.ylabel('JAN temperature(celsius)')
```

Out[29]: Text(0, 0.5, 'JAN temperature(celsius)')



In [52]:
```python
#evaluating Mean Square Error
```

In [33]:
```python
from sklearn.metrics import mean_squared_error, r2_score
```

In [34]:
```python
mse=mean_squared_error(Xtest,Ypredict)
print("mse is:",mse)
```

```
mse is: 3745652.702467675
```

In [36]:
```python
#creaating model for march month
```

In [37]:
```python
X=A[["YEAR"]]
```

In [38]:
```python
Y=A[["MAR"]]
```

In [39]:
```python
#spliting data
```

In [40]:
```python
from sklearn.model_selection import train_test_split
```

In [41]:
```python
Xtrain,Xtest,Ytrain,Ytest=train_test_split(X,Y,test_size=0.2)
```

In [42]:
```python
#creating model
```

In [43]:
```python
from sklearn.linear_model import LinearRegression
```

In [44]:
```python
reg=LinearRegression()
```

In [45]:
```python
#training our model
```

In [46]:
```python
model=reg.fit(Xtrain,Ytrain)
```

In [47]:
```python
#predicting dataset
```

In [48]:
```python
Ypredict=reg.predict(Xtest)
```

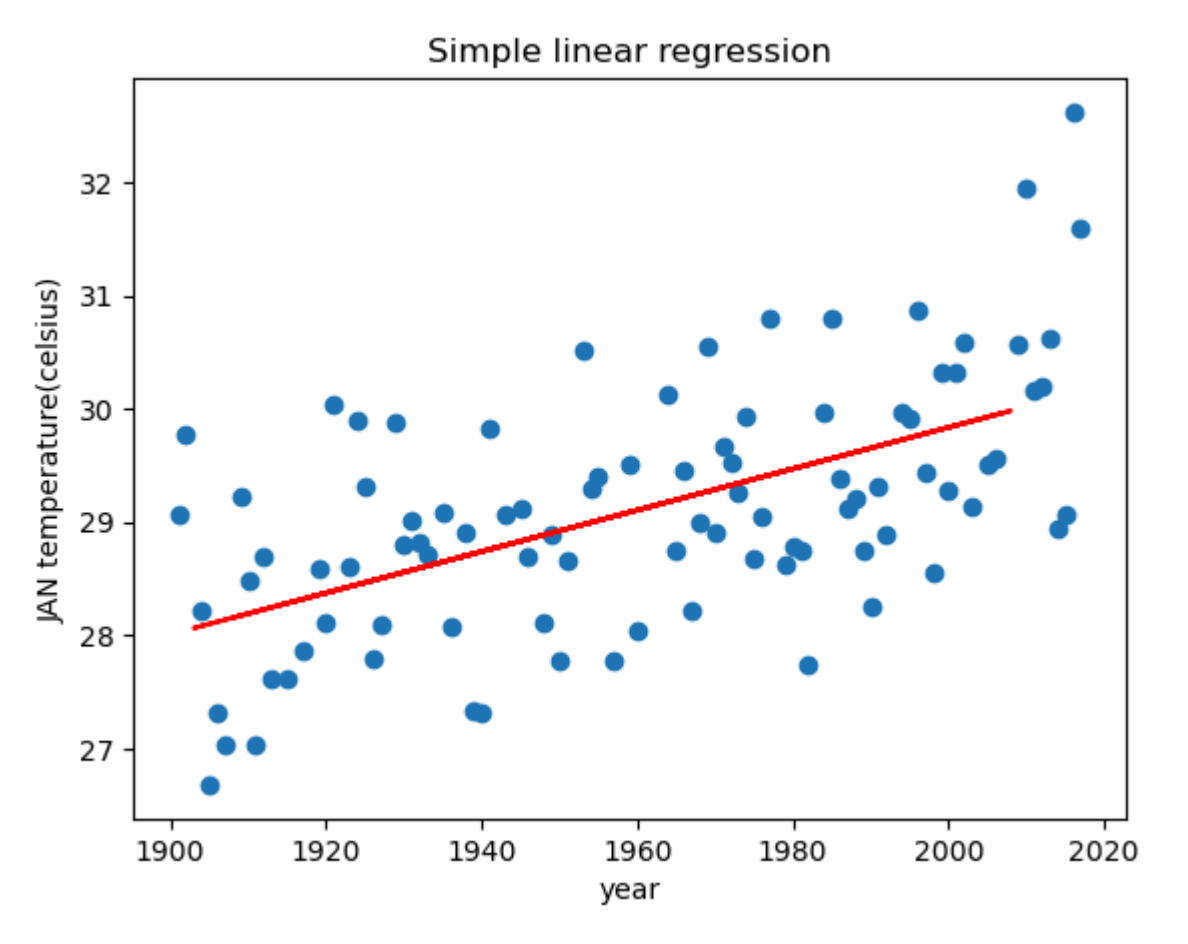
In [49]:
```python
print(Ypredict)
```

```
[[29.90835933]
 [29.14102428]
 [29.06794475]
 [28.06310123]
 [28.86697605]
 [28.62946758]
 [28.8121664 ]
 [28.3006097 ]
 [29.98143886]
 [28.95832546]
 [29.70739063]
 [29.03140499]
 [28.26406994]
 [29.1227544 ]
 [29.15929416]
 [28.68427723]
 [28.77562664]
 [28.33714947]
 [28.51984829]
 [28.15445064]
 [29.96316898]
 [28.41022899]
 [29.52469181]
 [29.43334239]]
```

In [50]:
```python
#ploting graph based on trained model trained model
```

In [51]:
```python
plt.scatter(Xtrain,Ytrain)
plt.plot(Xtest,Ypredict,color="red")
plt.title('Simple linear regression')
plt.xlabel('year')
plt.ylabel('JAN temperature(celsius)')
```

Out[51]: Text(0, 0.5, 'JAN temperature(celsius)')



In [53]:
```python
#evaluating Mean Square Error
```

In [54]:
```python
from sklearn.metrics import mean_squared_error, r2_score
```

In [55]:
```python
mse=mean_squared_error(Xtest,Ypredict)
print("mse is:",mse)
```

```
mse is: 3697439.018823273
```

In [56]:
```python
#creaating model for May month
```

In [57]:
```python
X=A[["YEAR"]]
```

In [58]:
```python
Y=A[["MAY"]]
```

In [59]:
```python
#spliting data
```

In [60]:
```python
from sklearn.model_selection import train_test_split
```

In [61]:
```python
Xtrain,Xtest,Ytrain,Ytest=train_test_split(X,Y,test_size=0.2)
```

In [62]:
```python
#creating model
```

In [63]:
```python
from sklearn.linear_model import LinearRegression
```

In [64]:
```python
reg=LinearRegression()
```

In [65]:
```python
#training our model
```

In [66]:
```python
model=reg.fit(Xtrain,Ytrain)
```

In [67]:
```python
#predicting dataset
```

In [68]:
```python
Ypredict=reg.predict(Xtest)
```

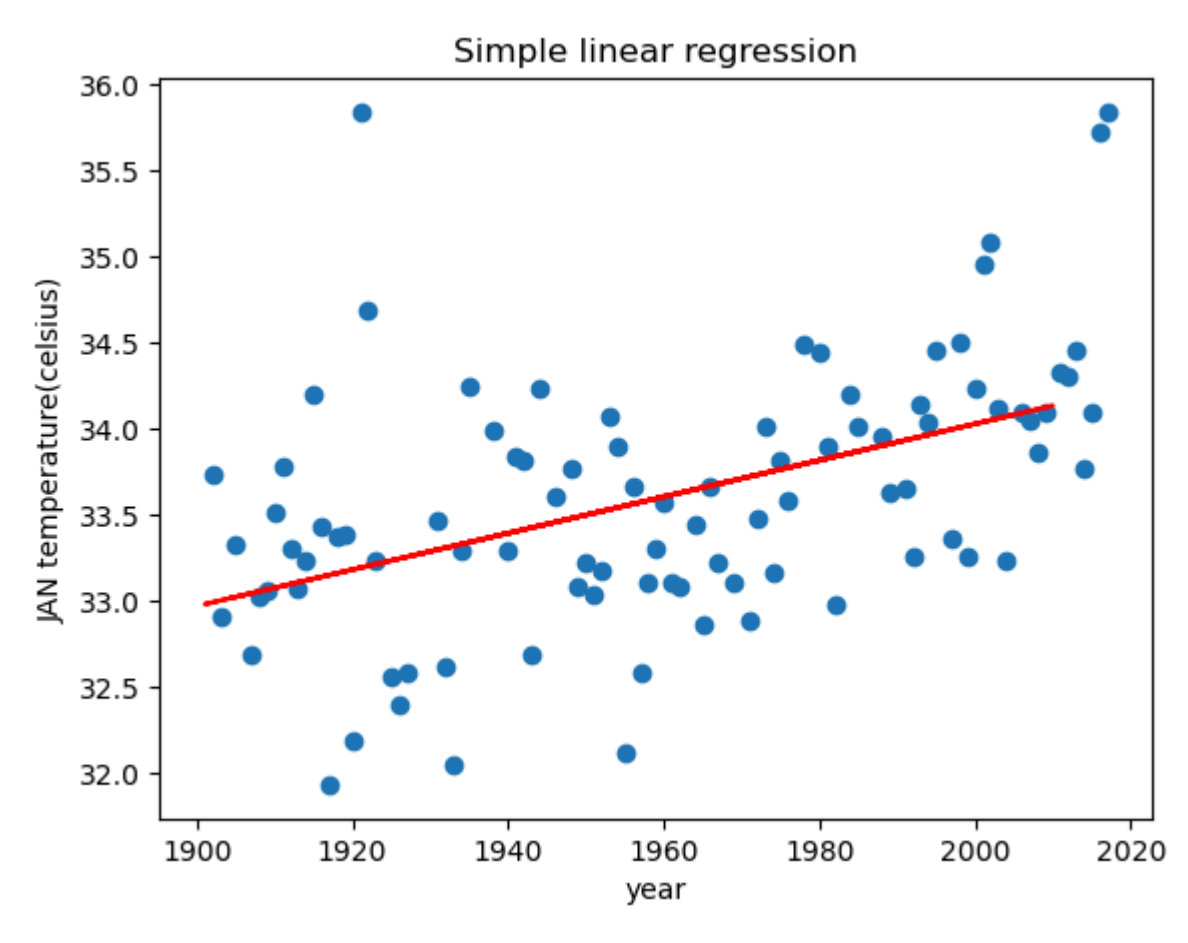
In [69]:
```python
print(Ypredict)
```

```
[[33.27556396]
 [33.36026055]
 [33.98489786]
 [33.22262859]
 [33.03206128]
 [33.01088713]
 [33.68845981]
 [32.97912591]
 [33.70963396]
 [33.8896142 ]
 [33.80491762]
 [34.08018152]
 [33.44495713]
 [33.38143469]
 [33.26497689]
 [33.28615103]
 [34.13311689]
 [33.63552445]
 [33.34967347]
 [33.92137542]
 [33.84726591]
 [33.46613128]
 [33.87902713]
 [33.78374347]]
```

In [70]:
```python
#ploting graph based on trained model trained model
```

In [71]:
```python
plt.scatter(Xtrain,Ytrain)
plt.plot(Xtest,Ypredict,color="red")
plt.title('Simple linear regression')
plt.xlabel('year')
plt.ylabel('JAN temperature(celsius)')
```

Out[71]: Text(0, 0.5, 'JAN temperature(celsius)')



In [72]:
```python
#evaluating Mean Square Error
```

In [73]:
```python
from sklearn.metrics import mean_squared_error, r2_score
```

In [74]:
```python
mse=mean_squared_error(Xtest,Ypredict)
print("mse is:",mse)
```

```
mse is: 3696144.2277363464
```

In [75]:
```python
#creaating model for Jun month
```

In [76]:
```python
X=A[["YEAR"]]
```

In [77]:
```python
Y=A[["JUN"]]
```

In [78]:
```python
#spliting data
```

In [79]:
```python
from sklearn.model_selection import train_test_split
```

In [80]:
```python
Xtrain,Xtest,Ytrain,Ytest=train_test_split(X,Y,test_size=0.2)
```

In [81]:
```python
#creating model
```

In [82]:
```python
from sklearn.linear_model import LinearRegression
```

In [83]:
```python
reg=LinearRegression()
```

In [84]:
```python
#training our model
```

In [85]:
```python
model=reg.fit(Xtrain,Ytrain)
```

In [86]:
```python
#predicting dataset
```

In [87]:
```python
Ypredict=reg.predict(Xtest)
```

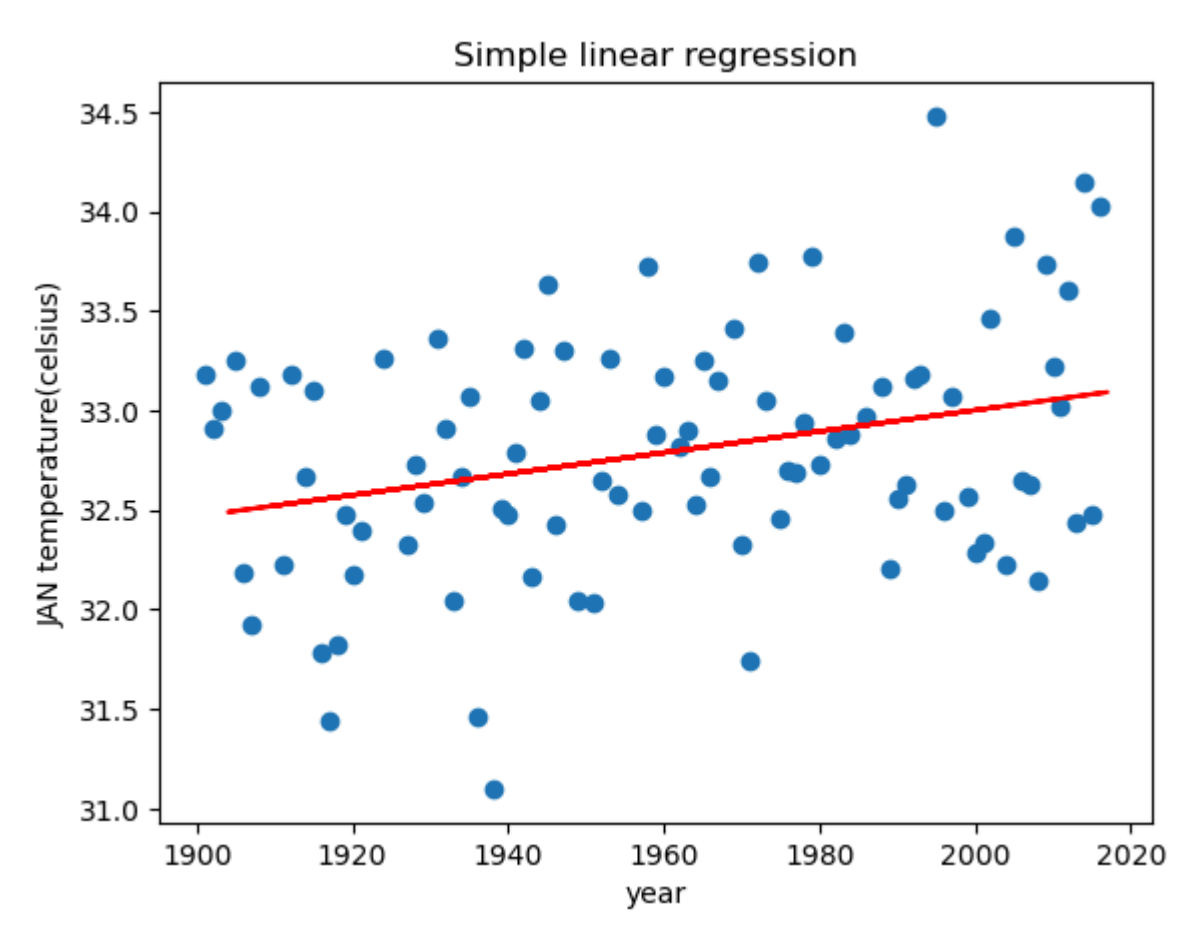
In [88]:
```python
print(Ypredict)
```

```
[[32.58605781]
 [32.59138778]
 [32.49011834]
 [32.60204772]
 [32.62869757]
 [32.83123644]
 [32.51676819]
 [32.96981567]
 [32.99113556]
 [32.86321627]
 [32.72463704]
 [32.52209816]
 [32.90052606]
 [32.7672768 ]
 [32.73529698]
 [32.93250588]
 [32.66600736]
 [32.76194683]
 [32.53808807]
 [33.09240499]
 [32.60737769]
 [33.01778541]
 [32.92184594]
 [32.79392665]]
```

In [89]:
```python
#ploting graph based on trained model trained model
```

In [90]:
```python
plt.scatter(Xtrain,Ytrain)
plt.plot(Xtest,Ypredict,color="red")
plt.title('Simple linear regression')
plt.xlabel('year')
plt.ylabel('JAN temperature(celsius)')
```

Out[90]: Text(0, 0.5, 'JAN temperature(celsius)')



In [91]:
```python
#evaluating Mean Square Error
```

In [92]:
```python
from sklearn.metrics import mean_squared_error, r2_score
```

In [93]:
```python
mse=mean_squared_error(Xtest,Ypredict)
print("mse is:",mse)
```

```
mse is: 3689063.112641784
```

In [94]:
```python
#creaating model for Aug month
```

In [95]:
```python
X=A[["YEAR"]]
```

In [96]:
```python
Y=A[["AUG"]]
```

In [97]:
```python
#spliting data
```

In [98]:
```python
from sklearn.model_selection import train_test_split
```

In [99]:
```python
Xtrain,Xtest,Ytrain,Ytest=train_test_split(X,Y,test_size=0.2)
```

In [100…
```python
#creating model
```

In [101…
```python
from sklearn.linear_model import LinearRegression
```

In [102…
```python
reg=LinearRegression()
```

In [103…
```python
#training our model
```

In [104…
```python
model=reg.fit(Xtrain,Ytrain)
```

In [105…
```python
#predicting dataset
```

In [106…
```python
Ypredict=reg.predict(Xtest)
```

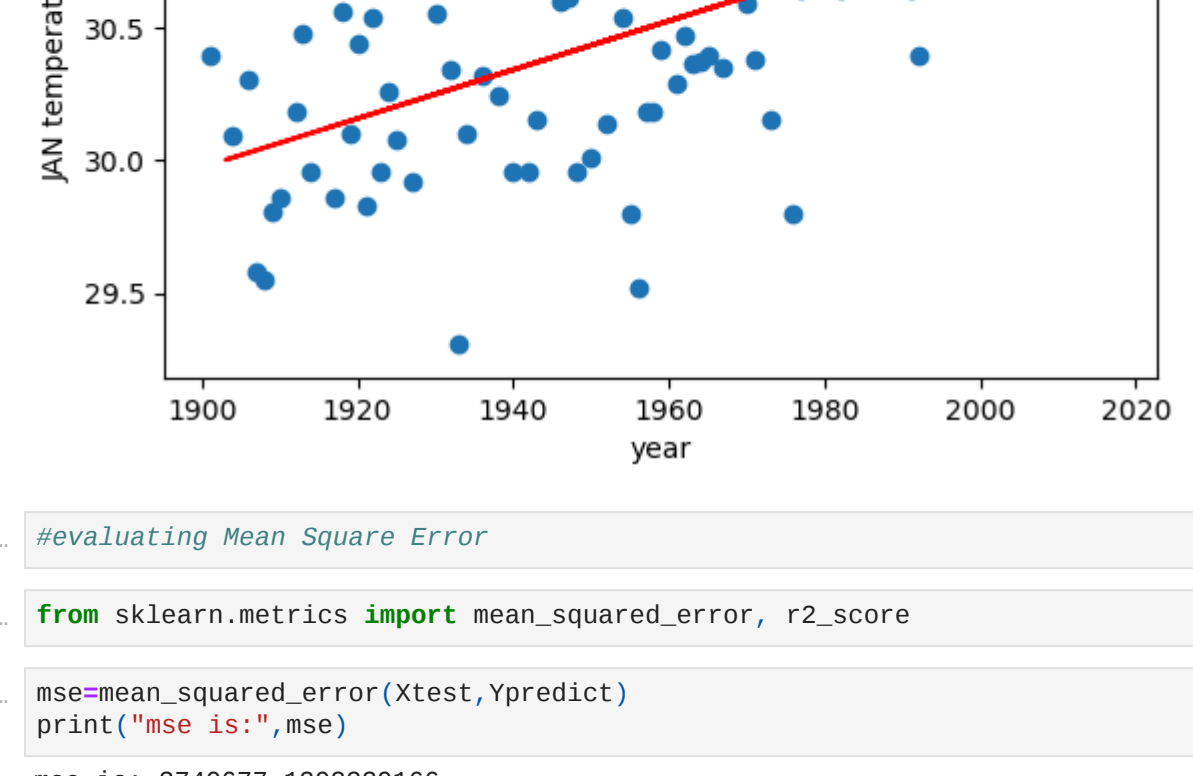
In [107…
```python
print(Ypredict)
```

```
[[30.60649389]
 [30.21203251]
 [30.63401445]
 [30.00104154]
 [30.45971756]
 [30.66153501]
 [30.69822909]
 [30.77161726]
 [30.79913782]
 [30.71657614]
 [30.1202973 ]
 [30.23037955]
 [30.33128827]
 [30.90922007]
 [30.94591415]
 [30.44137052]
 [30.85417894]
 [30.68905557]
 [30.23955307]
 [30.29459419]
 [30.65236149]
 [30.8358319 ]
 [31.01930231]
 [30.42302348]]
```

In [108…
```python
#ploting graph based on trained model trained model
```

In [109…
```python
plt.scatter(Xtrain,Ytrain)
plt.plot(Xtest,Ypredict,color="red")
plt.title('Simple linear regression')
plt.xlabel('year')
plt.ylabel('JAN temperature(celsius)')
```

Out[109]: Text(0, 0.5, 'JAN temperature(celsius)')



In [110…
```python
#evaluating Mean Square Error
```

In [111…
```python
from sklearn.metrics import mean_squared_error, r2_score
```

In [112…
```python
mse=mean_squared_error(Xtest,Ypredict)
print("mse is:",mse)
```

```
mse is: 3740677.1208339166
```

In [ ]: