```
import pandas as pd
from sklearn import linear_model, metrics
```

```
df = pd.read_csv('temperature.csv')
```

```
df.head()
```

```
   YEAR    JAN    FEB    MAR    APR    MAY    JUN    JUL    AUG    SEP    OCT

0  1901  22.40  24.14  29.07  31.91  33.41  33.18  31.21  30.39  30.47  29.97  2
1  1902  24.93  26.58  29.77  31.78  33.73  32.91  30.92  30.73  29.80  29.12  2
2  1903  23.44  25.03  27.83  31.39  32.91  33.00  31.34  29.98  29.85  29.04  2
3  1904  22.50  24.73  28.21  32.02  32.64  32.07  30.36  30.09  30.04  29.20  2
4  1905  22.00  22.83  26.68  30.01  33.32  33.25  31.44  30.68  30.12  30.67  2
```

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
df.isnull().sum()
```

```
          0

YEAR      0
JAN       0
FEB       0
MAR       0
APR       0
MAY       0
JUN       0
JUL       0
AUG       0
SEP       0
OCT       0
NOV       0
DEC       0
ANNUAL    0
JAN-FEB   0
```

```
MAR-MAY  0
JUN-SEP  0
OCT-DEC  0

dtype: int64
```

```
df.columns
```

```
Index(['YEAR', 'JAN', 'FEB', 'MAR', 'APR', 'MAY', 'JUN', 'JUL', 'AUG',
'SEP',
       'OCT', 'NOV', 'DEC', 'ANNUAL', 'JAN-FEB', 'MAR-MAY', 'JUN-SEP',
       'OCT-DEC'],
      dtype='object')
```

```
#split the dataset
x = df.iloc[:,0].values
y = df.iloc[:,1].values

#make it 2 dimensional
x = x.reshape(117,1)
```

```
x
x.shape
```

```
(117, 1)
```

```
y
```

```
array([22.4 , 24.93, 23.44, 22.5 , 22.  , 22.28, 24.46, 23.57, 22.67,
       23.24, 23.22, 23.7 , 23.71, 24.42, 22.6 , 24.13, 23.68, 22.06,
       23.32, 22.87, 23.57, 22.96, 23.25, 22.84, 22.56, 23.54, 23.23,
       23.33, 23.05, 22.6 , 24.57, 24.13, 22.85, 22.76, 22.28, 23.1 ,
       23.34, 22.95, 23.61, 23.79, 23.18, 22.99, 22.97, 23.17, 22.38,
       24.41, 22.61, 22.87, 24.31, 23.56, 24.36, 23.91, 22.96, 22.79,
       23.46, 23.16, 22.98, 24.75, 23.33, 23.78, 24.14, 22.89, 22.9 ,
       23.06, 23.82, 24.11, 23.72, 22.67, 23.78, 24.19, 23.91, 24.46,
       24.02, 23.54, 23.15, 23.91, 23.98, 23.6 , 24.6 , 24.35, 23.73,
       24.23, 23.89, 23.59, 23.84, 23.61, 23.81, 24.12, 23.22, 24.24,
       23.31, 23.84, 23.82, 24.67, 24.44, 25.18, 23.3 , 23.95, 23.57,
       24.44, 24.36, 24.56, 24.27, 23.89, 24.18, 25.66, 25.19, 23.97,
       25.27, 24.89, 24.18, 23.61, 24.56, 23.83, 24.58, 26.94, 26.45])
```

```
#Diving dataset into training and testing ( Data Preparation)
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.4,random_state
```

```
print("size of: \n")
print("x_train: ", x_train.shape)
print("y_train: ", y_train.shape)
print("x_test: ", x_test.shape)
print("y_test: ", y_test.shape)
```

```
    size of:

    x_train:  (70, 1)
    y_train:  (70,)
    x_test:  (47, 1)
    y_test:  (47,)
```

```
#model selection
from sklearn.linear_model import LinearRegression
model = LinearRegression()
model.fit(x_train,y_train)
```

```
    ▾  LinearRegression ⓘ ?

    LinearRegression()
```

```
print("model coefficient (slop = ) :" , model.coef_)
print("model intercept : ", model.intercept_)
```

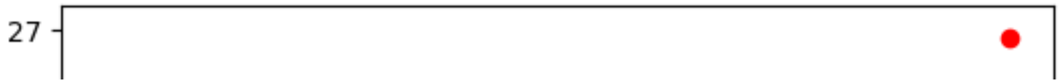
```
    model coefficient (slop = ) : [0.01417831]
    model intercept :  -4.22155455834347
```

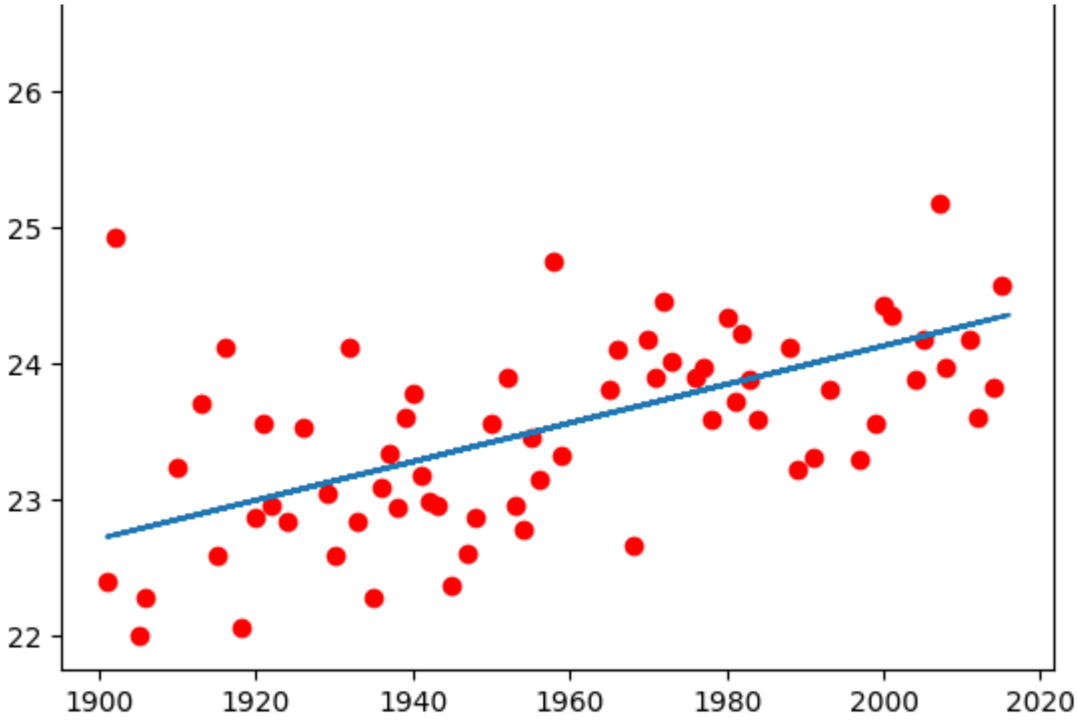
```
#model predictions
predicted = model.predict(x_test)
print("Predicted Temperature :", predicted)
print("Actual Temperature :", y_test)
```

```
    Predicted Temperature : [22.87319172 23.56792881 24.07834789 24.02163466 22
     23.10004465 24.31937912 23.04333142 24.06416958 24.16341774 22.75976526
     23.44032404 23.07168803 24.37609235 23.61046373 24.04999127 23.78060342
     23.59628542 23.76642512 22.95826157 24.26266589 22.91572665 23.34107588
     24.22013097 23.1992928  23.15675788 23.52539388 23.41196742 22.84483511
     24.1067045  23.66717696 24.2768442  22.77394357 23.62464204 22.81647849
     23.69553358 23.3694325  23.99327804 24.17759604 23.93656481 23.83731666
     23.9223865  23.11422296 22.98661819 22.88737003 23.58210711]
    Actual Temperature : [23.22 23.78 25.18 23.84 23.57 23.81 23.23 24.56 23.25
     24.36 22.56 26.45 22.9  24.67 23.15 22.89 23.54 23.68 25.27 24.42 23.17
     25.66 22.76 24.57 22.98 24.31 22.67 23.95 23.72 24.89 22.5  23.06 24.46
     23.78 24.41 24.24 24.27 23.61 24.6  23.84 23.33 23.32 23.7  24.14]
```

```
#line plotting
import matplotlib.pyplot as plt
plt.scatter(x_train,y_train,color='red')
plt.plot(x_train,model.predict(x_train))
```

```
    [<matplotlib.lines.Line2D at 0x7c2eaefa5d50>]
```





```
model
```

```
▾ LinearRegression ⓘ ?
LinearRegression()
```

```
#calculate the matrics here
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_sc
```

```
print( f"Mean Squared Error: {mean_squared_error(y_test,predicted)}")
```

```
Mean Squared Error: 0.5491592537871481
```

```
print( f"Mean Absolute Error: {mean_absolute_error(y_test,predicted)}")
```

```
Mean Absolute Error: 0.5814161027244875
```

```
print( f"r2 score: {r2_score(y_test,predicted)}")
```

```
r2 score: 0.19964742869898222
```

Start coding or generate with AI.