```
import pandas as pd
```

```
ds=pd.read_csv("/content/temperatures.csv")
```

```
print(ds)
```

```
      YEAR    JAN    FEB    MAR    APR    MAY    JUN    JUL    AUG    SEP  \
0    1901  22.40  24.14  29.07  31.91  33.41  33.18  31.21  30.39  30.47
1    1902  24.93  26.58  29.77  31.78  33.73  32.91  30.92  30.73  29.80
2    1903  23.44  25.03  27.83  31.39  32.91  33.00  31.34  29.98  29.85
3    1904  22.50  24.73  28.21  32.02  32.64  32.07  30.36  30.09  30.04
4    1905  22.00  22.83  26.68  30.01  33.32  33.25  31.44  30.68  30.12
..    ...    ...    ...    ...    ...    ...    ...    ...    ...    ...
112  2013  24.56  26.59  30.62  32.66  34.46  32.44  31.07  30.76  31.04
113  2014  23.83  25.97  28.95  32.74  33.77  34.15  31.85  31.32  30.68
114  2015  24.58  26.89  29.07  31.87  34.09  32.48  31.88  31.52  31.55
115  2016  26.94  29.72  32.62  35.38  35.72  34.03  31.64  31.79  31.66
116  2017  26.45  29.46  31.60  34.95  35.84  33.82  31.88  31.72  32.22

       OCT    NOV    DEC  ANNUAL  JAN-FEB  MAR-MAY  JUN-SEP  OCT-DEC
0    29.97  27.31  24.49   28.96    23.27    31.46    31.27    27.25
1    29.12  26.31  24.04   29.22    25.75    31.76    31.09    26.49
2    29.04  26.08  23.65   28.47    24.24    30.71    30.92    26.26
3    29.20  26.36  23.63   28.49    23.62    30.95    30.66    26.40
4    30.67  27.52  23.82   28.30    22.25    30.00    31.33    26.57
..     ...    ...    ...     ...      ...      ...      ...      ...
112  30.27  27.83  25.37   29.81    25.58    32.58    31.33    27.83
113  30.29  28.05  25.08   29.72    24.90    31.82    32.00    27.81
114  31.04  28.10  25.67   29.90    25.74    31.68    31.87    28.27
115  31.98  30.11  28.01   31.63    28.33    34.57    32.28    30.03
116  32.29  29.60  27.18   31.42    27.95    34.13    32.41    29.69

[117 rows x 18 columns]
```

```
ds.head()
```

	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
0	1901	22.40	24.14	29.07	31.91	33.41	33.18	31.21	30.39	30.47	29.97	27.31	24.49
1	1902	24.93	26.58	29.77	31.78	33.73	32.91	30.92	30.73	29.80	29.12	26.31	24.04
2	1903	23.44	25.03	27.83	31.39	32.91	33.00	31.34	29.98	29.85	29.04	26.08	23.65
3	1904	22.50	24.73	28.21	32.02	32.64	32.07	30.36	30.09	30.04	29.20	26.36	23.63

```
ds.describe()
```

	YEAR	JAN	FEB	MAR	APR	MAY	JU
count	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.0000
mean	1959.000000	23.687436	25.597863	29.085983	31.975812	33.565299	32.7742
std	33.919021	0.834588	1.150757	1.068451	0.889478	0.724905	0.6331
min	1901.000000	22.000000	22.830000	26.680000	30.010000	31.930000	31.1000
25%	1930.000000	23.100000	24.780000	28.370000	31.460000	33.110000	32.3400
50%	1959.000000	23.680000	25.480000	29.040000	31.950000	33.510000	32.7300
75%	1988.000000	24.180000	26.310000	29.610000	32.420000	34.030000	33.1800
max	2017.000000	26.940000	29.720000	32.620000	35.380000	35.840000	34.4800

```
ds.shape
```

```
(117, 18)
```

handing duplicates

```
ds=ds.drop_duplicates()
```

```
ds.shape
```

```
(117, 18)
```

```
print(ds)
```

```
     YEAR    JAN    FEB    MAR    APR    MAY    JUN    JUL    AUG    SEP  \
0    1901  22.40  24.14  29.07  31.91  33.41  33.18  31.21  30.39  30.47
1    1902  24.93  26.58  29.77  31.78  33.73  32.91  30.92  30.73  29.80
2    1903  23.44  25.03  27.83  31.39  32.91  33.00  31.34  29.98  29.85
3    1904  22.50  24.73  28.21  32.02  32.64  32.07  30.36  30.09  30.04
4    1905  22.00  22.83  26.68  30.01  33.32  33.25  31.44  30.68  30.12
..    ...    ...    ...    ...    ...    ...    ...    ...    ...    ...
112  2013  24.56  26.59  30.62  32.66  34.46  32.44  31.07  30.76  31.04
113  2014  23.83  25.97  28.95  32.74  33.77  34.15  31.85  31.32  30.68
114  2015  24.58  26.89  29.07  31.87  34.09  32.48  31.88  31.52  31.55
115  2016  26.94  29.72  32.62  35.38  35.72  34.03  31.64  31.79  31.66
116  2017  26.45  29.46  31.60  34.95  35.84  33.82  31.88  31.72  32.22

       OCT    NOV    DEC  ANNUAL  JAN-FEB  MAR-MAY  JUN-SEP  OCT-DEC
0    29.97  27.31  24.49   28.96    23.27    31.46    31.27    27.25
1    29.12  26.31  24.04   29.22    25.75    31.76    31.09    26.49
2    29.04  26.08  23.65   28.47    24.24    30.71    30.92    26.26
3    29.20  26.36  23.63   28.49    23.62    30.95    30.66    26.40
4    30.67  27.52  23.82   28.30    22.25    30.00    31.33    26.57
..     ...    ...    ...     ...      ...      ...      ...      ...
112  30.27  27.83  25.37   29.81    25.58    32.58    31.33    27.83
113  30.29  28.05  25.08   29.72    24.90    31.82    32.00    27.81
114  31.04  28.10  25.67   29.90    25.74    31.68    31.87    28.27
115  31.98  30.11  28.01   31.63    28.33    34.57    32.28    30.03
116  32.29  29.60  27.18   31.42    27.95    34.13    32.41    29.69

[117 rows x 18 columns]
```

handling missing data

```
ds.isnull()
```

	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
0	False	False	False	False	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...
112	False	False	False	False	False	False	False	False	False	False	False	False	False
113	False	False	False	False	False	False	False	False	False	False	False	False	False
114	False	False	False	False	False	False	False	False	False	False	False	False	False
115	False	False	False	False	False	False	False	False	False	False	False	False	False
116	False	False	False	False	False	False	False	False	False	False	False	False	False

```
ds.isnull().sum()
```

```
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
```

```
ds.head()
```

	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
0	1901	22.40	24.14	29.07	31.91	33.41	33.18	31.21	30.39	30.47	29.97	27.31	24.49
1	1902	24.93	26.58	29.77	31.78	33.73	32.91	30.92	30.73	29.80	29.12	26.31	24.04
2	1903	23.44	25.03	27.83	31.39	32.91	33.00	31.34	29.98	29.85	29.04	26.08	23.65
3	1904	22.50	24.73	28.21	32.02	32.64	32.07	30.36	30.09	30.04	29.20	26.36	23.63

```
y = ds['ANNUAL'].values.reshape(-1, 1)
X = ds['YEAR'].values.reshape(-1, 1)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


model = LinearRegression()
model.fit(X_train, y_train)
```

```
LinearRegression()
```

```
print(model.intercept_)
print(model.coef_)
```

```
[4.61705135]
[[0.0125253]]
```

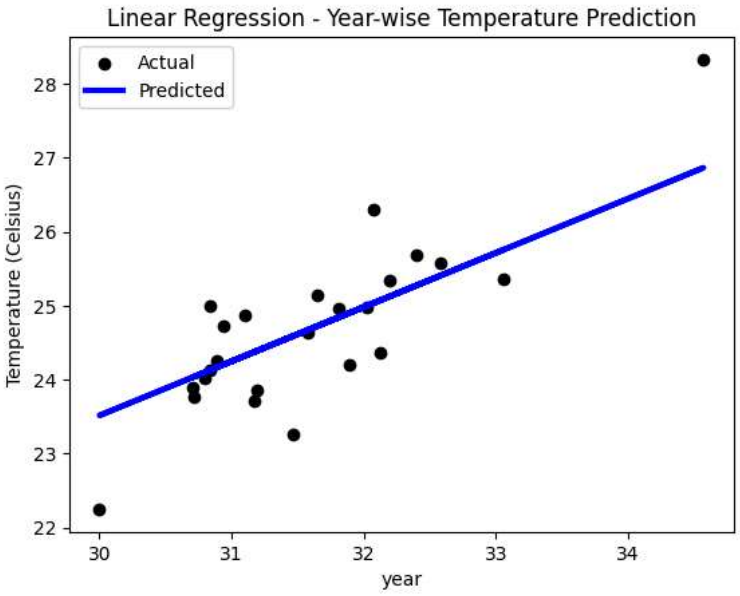
```
y_pred = model.predict(X_test)


from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
import matplotlib.pyplot as plt
mse = mean_squared_error(y_test, y_pred)
mae = mean_absolute_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print("Mean Squared Error (MSE):", mse)
print("Mean Absolute Error (MAE):", mae)
print("R-Square (R2):", r2)
```

```
Mean Squared Error (MSE): 0.2062573233109893
Mean Absolute Error (MAE): 0.28045479124619366
R-Square (R2): 0.5733931056341353
```

```
plt.scatter(X_test, y_test, color='black', label='Actual')
plt.plot(X_test, y_pred, color='blue', linewidth=3, label='Predicted')
plt.xlabel('year')
plt.ylabel('Temperature (Celsius)')
plt.title('Linear Regression - Year-wise Temperature Prediction')
plt.legend()
plt.show()
```




```
y = ds['JAN-FEB'].values.reshape(-1, 1)
X = ds['ANNUAL'].values.reshape(-1, 1)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


model = LinearRegression()
model.fit(X_train, y_train)
```

```
▾ LinearRegression
LinearRegression()
```

```
print(model.intercept_)
print(model.coef_)
```

```
[-15.02614956]
[[1.35948323]]
```

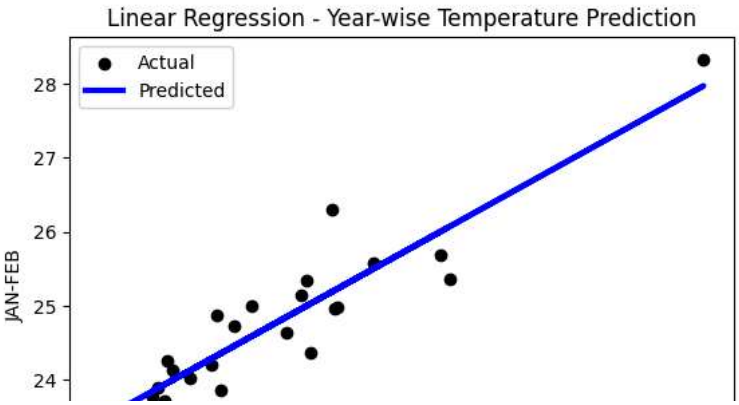
```
y_pred = model.predict(X_test)


from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
import matplotlib.pyplot as plt
mse = mean_squared_error(y_test, y_pred)
mae = mean_absolute_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print("Mean Squared Error (MSE):", mse)
print("Mean Absolute Error (MAE):", mae)
print("R-Square (R2):", r2)
```

```
Mean Squared Error (MSE): 0.2620508077116476
Mean Absolute Error (MAE): 0.39298505063880435
R-Square (R2): 0.7998058219730977
```

```
plt.scatter(X_test, y_test, color='black', label='Actual')
plt.plot(X_test, y_pred, color='blue', linewidth=3, label='Predicted')
plt.xlabel('year')
plt.ylabel('JAN-FEB')
plt.title('Linear Regression - Year-wise Temperature Prediction')
plt.legend()
plt.show()
```


Linear Regression - Year-wise Temperature Prediction
Actual
Predicted
JAN-FEB
28
27
26
25
24