

```
In [1]: import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, mean_absolute_error
from sklearn.model_selection import train_test_split
```

```
In [2]: dfgpa = pd.read_csv("D:\\24 - Machine_Learning\\download files\\Weather.csv")
dfgpa

C:\Users\studentadmin\AppData\Local\Temp\ipykernel_20540\3805364300.py:1: DtypeWarning: Columns (7,8,18,25) have mixed t
ypes. Specify dtype option on import or set low_memory=False.
dfgpa = pd.read_csv("D:\\24 - Machine_Learning\\download files\\Weather.csv")
```

```
Out[2]:
```

	STA	Date	Precip	WindGustSpd	MaxTemp	MinTemp	MeanTemp	Snowfall	PoorWeather	YR	...	FB	FTI	ITH	PGT	TSHDSBRS	SGF
	0	10001	1942-7-1	1.016		NaN	25.555556	22.222222	23.888889	0.0		NaN	42	...	NaN	NaN	NaN
	1	10001	1942-7-2	0		NaN	28.888889	21.666667	25.555556	0.0		NaN	42	...	NaN	NaN	NaN
	2	10001	1942-7-3	2.54		NaN	26.111111	22.222222	24.444444	0.0		NaN	42	...	NaN	NaN	NaN
	3	10001	1942-7-4	2.54		NaN	26.666667	22.222222	24.444444	0.0		NaN	42	...	NaN	NaN	NaN
	4	10001	1942-7-5	0		NaN	26.666667	21.666667	24.444444	0.0		NaN	42	...	NaN	NaN	NaN

	119035	82506	1945-12-27	0		NaN	28.333333	18.333333	23.333333	0.0		NaN	45	...	NaN	NaN	NaN
	119036	82506	1945-12-28	9.906		NaN	29.444444	18.333333	23.888889	0.0		1.0	45	...	NaN	NaN	NaN
	119037	82506	1945-12-29	0		NaN	28.333333	18.333333	23.333333	0.0		1.0	45	...	NaN	NaN	NaN
	119038	82506	1945-12-30	0		NaN	28.333333	18.333333	23.333333	0.0		NaN	45	...	NaN	NaN	NaN
	119039	82506	1945-12-31	0		NaN	29.444444	17.222222	23.333333	0.0		NaN	45	...	NaN	NaN	NaN

119040 rows × 31 columns

```
In [3]: dfgpa.head(10)
```

```
Out[3]:
```

	STA	Date	Precip	WindGustSpd	MaxTemp	MinTemp	MeanTemp	Snowfall	PoorWeather	YR	...	FB	FTI	ITH	PGT	TSHDSBRS	SGF	SD3
0	10001	1942-7-1	1.016		NaN	25.555556	22.222222	23.888889	0.0	NaN	42	...	NaN	NaN	NaN	NaN	NaN	NaN
1	10001	1942-7-2	0		NaN	28.888889	21.666667	25.555556	0.0	NaN	42	...	NaN	NaN	NaN	NaN	NaN	NaN
2	10001	1942-7-3	2.54		NaN	26.111111	22.222222	24.444444	0.0	NaN	42	...	NaN	NaN	NaN	NaN	NaN	NaN
3	10001	1942-7-4	2.54		NaN	26.666667	22.222222	24.444444	0.0	NaN	42	...	NaN	NaN	NaN	NaN	NaN	NaN
4	10001	1942-7-5	0		NaN	26.666667	21.666667	24.444444	0.0	NaN	42	...	NaN	NaN	NaN	NaN	NaN	NaN
5	10001	1942-7-6	0		NaN	26.666667	21.666667	24.444444	0.0	NaN	42	...	NaN	NaN	NaN	NaN	NaN	NaN
6	10001	1942-7-7	T		NaN	28.333333	22.777778	25.555556	0.0	NaN	42	...	NaN	NaN	NaN	NaN	NaN	NaN
7	10001	1942-7-8	3.556		NaN	26.666667	22.222222	24.444444	0.0	NaN	42	...	NaN	NaN	NaN	NaN	NaN	NaN
8	10001	1942-7-9	T		NaN	27.222222	22.777778	25.000000	0.0	NaN	42	...	NaN	NaN	NaN	NaN	NaN	NaN
9	10001	1942-7-10	3.556		NaN	25.555556	21.666667	23.333333	0.0	NaN	42	...	NaN	NaN	NaN	NaN	NaN	NaN

10 rows × 31 columns

```
In [4]: dfgpa.describe()
```

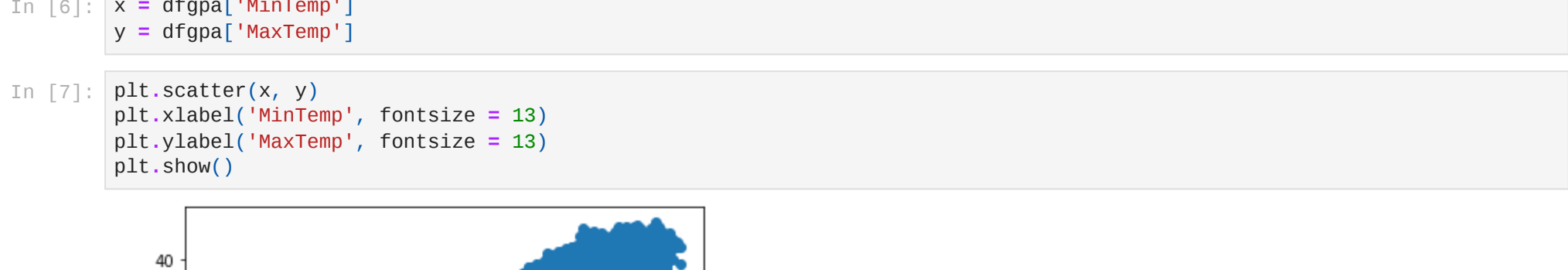
```
Out[4]:
```

	STA	WindGustSpd	MaxTemp	MinTemp	MeanTemp	YR	MO	DA	DR	SP
count	119040.000000	532.000000	119040.000000	119040.000000	119040.000000	119040.000000	119040.000000	119040.000000	533.000000	532.00000
mean	29659.435795	37.774534	27.045111	17.789511	22.411631	43.805284	6.726016	15.797530	26.998124	20.39661
std	20953.209402	10.297808	8.717817	8.334572	8.297982	1.136718	3.425561	8.794541	15.221732	5.56037
min	10001.000000	18.520000	-33.333333	-38.333333	-35.555556	40.000000	1.000000	1.000000	2.000000	10.00000
25%	11801.000000	29.632000	25.555556	15.000000	20.555556	43.000000	4.000000	8.000000	11.000000	16.00000
50%	22508.000000	37.040000	29.444444	21.111111	25.555556	44.000000	7.000000	16.000000	32.000000	20.00000
75%	33501.000000	43.059000	31.666667	23.333333	27.222222	45.000000	10.000000	23.000000	34.000000	23.25000
max	82506.000000	75.932000	50.000000	34.444444	40.000000	45.000000	12.000000	31.000000	78.000000	41.00000

8 rows × 24 columns

```
In [5]: dfgpa.shape
(119040, 31)
```

```
In [6]: x = dfgpa['MinTemp']
y = dfgpa['MaxTemp']
```



```
In [9]: x.shape
(119040,)
```

```
In [10]: X=x.values.reshape(-1, 1)
```

```
In [11]: X
array([[22.22222222],
       [21.66666667],
       [22.22222222],
       ...,
       [18.33333333],
       [18.33333333],
       [17.22222222]])
```

```
In [12]: X.shape
(119040, 1)
```

```
In [13]: X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.1,random_state=41)
```

```
In [14]: X_train.shape
(107136, 1)
```

```
In [15]: X_test.shape
(11904, 1)
```

```
In [16]: LR = LinearRegression()
LR.fit(X_train, y_train)
```

```
Out[16]: LinearRegression()
```

```
In [17]: y_pred = LR.predict(X_test)
```

```
In [18]: y_test
```

```
Out[18]:
```

36086	28.888889
78517	20.000000
116034	29.444444
100134	29.444444
41219	20.000000
...	
6897	29.444444
80516	33.333333
41582	23.333333
118088	29.444444
65652	28.888889
Name: MaxTemp, Length: 11904, dtype: float64	

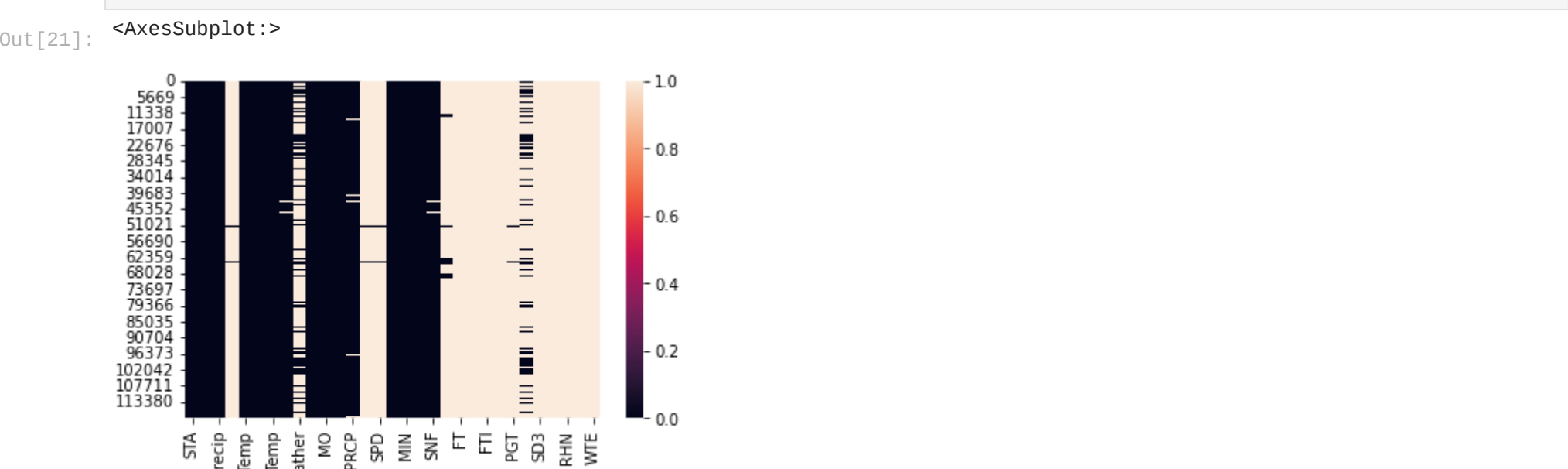
```
In [19]: y_pred
```

```
Out[19]: array([31.12079087, 22.44060078, 31.12079087, ..., 27.54659495,
27.54659495, 22.44060078])
```

```
In [20]: acc = mean_squared_error(y_test, y_pred)
acc
```

```
Out[20]: 16.621201324072025
```

```
In [21]: sns.heatmap(dfgpa.isnull())
```

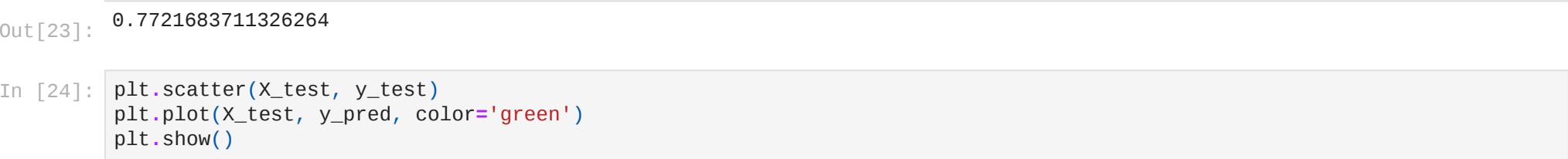


```
In [22]: weights = LR.coef_
intercept = LR.intercept_
print(weights, intercept)
```

```
Out[22]: [0.91907895] 10.696814176955677
```

```
In [23]: LR.score(X_test, y_test)
```

```
Out[23]: 0.7721683711326264
```



```
In [25]: df = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred})
df
```

```
Out[25]:
```

	Actual	Predicted
36086	28.888889	31.120791
78517	20.000000	22.440601
116034	29.444444	31.120791
100134	29.444444	29.078393
41219	20.000000	23.972399
...
6897	29.444444	32.141990
80516	33.333333	26.525396
41582	23.333333	27.546595
118088	29.444444	27.546595
65652	28.888889	22.440601

11904 rows × 2 columns

```
In [26]: df1 = df.head(25)
df1.plot(kind='bar', figsize=(16,10))
plt.grid(which='major', linestyle='-', linewidth='0.5', color='green')
plt.grid(which='minor', linestyle=':', linewidth='0.5', color='black')
plt.show()
```

```
In [28]: LR.predict([[35]])
```

```
Out[28]: array([42.86457748])
```