EXERCISE

```
import pandas as pd
```

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

```
df
```

	Product_Code	W0	W1	W2	W3	W4	W5	W6	W7	W8	...	Normalized 42	Normalized 43	Normalized 44	Normalized 45	Normalized 46	Normalized 47	Normalized 48	Norm
0	P1	11	12	10	8	13	12	14	21	6	...	0.06	0.22	0.28	0.39	0.50	0.00	0.22	
1	P2	7	6	3	2	7	1	6	3	3	...	0.20	0.40	0.50	0.10	0.10	0.40	0.50	
2	P3	7	11	8	9	10	8	7	13	12	...	0.27	1.00	0.18	0.18	0.36	0.45	1.00	
3	P4	12	8	13	5	9	6	9	13	13	...	0.41	0.47	0.06	0.12	0.24	0.35	0.71	
4	P5	8	5	13	11	6	7	9	14	9	...	0.27	0.53	0.27	0.60	0.20	0.20	0.13	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
806	P815	0	0	1	0	0	2	1	0	0	...	0.00	0.33	0.33	0.00	0.00	0.33	0.00	
807	P816	0	1	0	0	1	2	2	6	0	...	0.43	0.43	0.57	0.29	0.57	0.71	0.71	
808	P817	1	0	0	0	1	1	2	1	1	...	0.50	0.00	0.00	0.50	0.50	0.00	0.00	
809	P818	0	0	0	1	0	0	0	0	1	...	0.00	0.00	0.00	0.50	0.50	0.00	0.00	
810	P819	0	1	0	0	0	0	0	0	0	...	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

811 rows × 107 columns

```
df.head()
```

	Product_Code	W0	W1	W2	W3	W4	W5	W6	W7	W8	...	Normalized 42	Normalized 43	Normalized 44	Normalized 45	Normalized 46	Normalized 47	Normalized 48	Normali
0	P1	11	12	10	8	13	12	14	21	6	...	0.06	0.22	0.28	0.39	0.50	0.00	0.22	
1	P2	7	6	3	2	7	1	6	3	3	...	0.20	0.40	0.50	0.10	0.10	0.40	0.50	
2	P3	7	11	8	9	10	8	7	13	12	...	0.27	1.00	0.18	0.18	0.36	0.45	1.00	
3	P4	12	8	13	5	9	6	9	13	13	...	0.41	0.47	0.06	0.12	0.24	0.35	0.71	
4	P5	8	5	13	11	6	7	9	14	9	...	0.27	0.53	0.27	0.60	0.20	0.20	0.13	

5 rows × 107 columns

```
df.tail()
```

	Product_Code	W0	W1	W2	W3	W4	W5	W6	W7	W8	...	Normalized 42	Normalized 43	Normalized 44	Normalized 45	Normalized 46	Normalized 47	Normalized 48	Norm
806	P815	0	0	1	0	0	2	1	0	0	...	0.00	0.33	0.33	0.00	0.00	0.33	0.00	
807	P816	0	1	0	0	1	2	2	6	0	...	0.43	0.43	0.57	0.29	0.57	0.71	0.71	
808	P817	1	0	0	0	1	1	2	1	1	...	0.50	0.00	0.00	0.50	0.50	0.00	0.00	
809	P818	0	0	0	1	0	0	0	0	1	...	0.00	0.00	0.00	0.50	0.50	0.00	0.00	
810	P819	0	1	0	0	0	0	0	0	0	...	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

5 rows × 107 columns

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 811 entries, 0 to 810
Columns: 107 entries, Product_Code to Normalized 51
dtypes: float64(52), int64(54), object(1)
memory usage: 678.1+ KB
```

```
print(df.info())
print(df.describe())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 811 entries, 0 to 810
Columns: 107 entries, Product_Code to Normalized 51
dtypes: float64(52), int64(54), object(1)
memory usage: 678.1+ KB
None
                W0          W1          W2          W3          W4          W5  \
count  811.000000  811.000000  811.000000  811.000000  811.000000  811.000000
mean     8.902589    9.129470    9.389642    9.717633    9.574599    9.466091
std     12.067163   12.564766   13.045073   13.553294   13.095765   12.823195
min      0.000000    0.000000    0.000000    0.000000    0.000000    0.000000
25%      0.000000    0.000000    0.000000    0.000000    0.000000    0.000000
50%      3.000000    3.000000    3.000000    4.000000    4.000000    3.000000
75%     12.000000   12.000000   12.000000   13.000000   13.000000   12.500000
max     54.000000   53.000000   56.000000   59.000000   61.000000   52.000000

                W6          W7          W8          W9  ...  Normalized 42  \
count  811.000000  811.000000  811.000000  811.000000  ...     811.000000
mean     9.720099    9.585697    9.784217    9.681874  ...       0.299149
std     13.347375   13.049138   13.550237   13.137916  ...       0.266993
min      0.000000    0.000000    0.000000    0.000000  ...       0.000000
25%      0.000000    0.000000    0.000000    0.000000  ...       0.000000
50%      4.000000    4.000000    4.000000    4.000000  ...       0.280000
75%     13.000000   12.500000   13.000000   13.000000  ...       0.490000
max     56.000000   62.000000   63.000000   52.000000  ...       1.000000

       Normalized 43  Normalized 44  Normalized 45  Normalized 46  \
count     811.000000     811.000000     811.000000     811.000000
mean        0.287571       0.304846       0.316017       0.334760
std         0.256630       0.263396       0.262226       0.275203
min         0.000000       0.000000       0.000000       0.000000
25%         0.000000       0.000000       0.020000       0.085000
50%         0.270000       0.300000       0.310000       0.330000
75%         0.450000       0.500000       0.500000       0.500000
max         1.000000       1.000000       1.000000       1.000000
```

```
       Normalized 47  Normalized 48  Normalized 49  Normalized 50  \
count     811.000000      811.00000     811.000000     811.000000
mean        0.314636        0.33815       0.358903       0.373009
std         0.266029        0.27569       0.286665       0.295197
min         0.000000        0.00000       0.000000       0.000000
25%         0.000000        0.10500       0.100000       0.110000
50%         0.310000        0.33000       0.330000       0.350000
75%         0.500000        0.50000       0.550000       0.560000
max         1.000000        1.00000       1.000000       1.000000

       Normalized 51
count     811.000000
mean        0.427941
std         0.342360
min         0.000000
25%         0.090000
50%         0.430000
75%         0.670000
max         1.000000

[8 rows x 106 columns]
```

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

```
Product_Code     0
W0               0
W1               0
W2               0
W3               0
                ..
Normalized 47    0
Normalized 48    0
Normalized 49    0
Normalized 50    0
Normalized 51    0
Length: 107, dtype: int64
```

```
df.describe()
```

	W0	W1	W2	W3	W4	W5	W6	W7	W8	W9	...	Normalized 42	Normalized 43	Nor
count	811.000000	811.000000	811.000000	811.000000	811.000000	811.000000	811.000000	811.000000	811.000000	811.000000	...	811.000000	811.000000	811
mean	8.902589	9.129470	9.389642	9.717633	9.574599	9.466091	9.720099	9.585697	9.784217	9.681874	...	0.299149	0.287571	0
std	12.067163	12.564766	13.045073	13.553294	13.095765	12.823195	13.347375	13.049138	13.550237	13.137916	...	0.266993	0.256630	0
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000	0.000000	0
25%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000	0.000000	0
50%	3.000000	3.000000	3.000000	4.000000	4.000000	3.000000	4.000000	4.000000	4.000000	4.000000	...	0.280000	0.270000	0
75%	12.000000	12.000000	12.000000	13.000000	13.000000	12.500000	13.000000	12.500000	13.000000	13.000000	...	0.490000	0.450000	0
max	54.000000	53.000000	56.000000	59.000000	61.000000	52.000000	56.000000	62.000000	63.000000	52.000000	...	1.000000	1.000000	1

8 rows × 106 columns

```python
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv('Sales_Transactions_Dataset_Weekly.csv')
df=df.head(10)

plt.figure(figsize=(10, 6))
plt.bar(df['Product_Code'],df['W1'])
plt.xlabel('Product_Code')
plt.ylabel('W1')
plt.title('Bar Graph of the Product vs their weekly sales')
plt.xticks(rotation=45)
plt.show()
```