

```
In [28]: import pandas as pd
df=pd.read_csv("company.csv")
```

```
In [2]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 4 columns):
#   Column      Non-Null Count  Dtype
---  -
0   TV           200 non-null    float64
1   Radio        200 non-null    float64
2   Newspaper    200 non-null    float64
3   Sales        200 non-null    float64
dtypes: float64(4)
memory usage: 6.4 KB
```

```
In [3]: from sklearn.linear_model import LinearRegression
regressor=LinearRegression()
```

```
In [4]: x=df.iloc[:, :-1]
```

```
In [5]: y=df.iloc[:, -1]
```

```
In [6]: x.shape
```

```
Out[6]: (200, 3)
```

```
In [7]: y.shape
```

```
Out[7]: (200,)
```

```
In [8]: from sklearn.model_selection import train_test_split
```

```
In [12]: xtrain,xtest,ytrain,ytest=train_test_split(x,y,test_size=0.25,random_state=1)
```

```
In [13]: xtrain.shape
```

```
Out[13]: (150, 3)
```

```
In [14]: xtest.shape
```

```
Out[14]: (50, 3)
```

```
In [15]: ytrain.shape
```

```
Out[15]: (150,)
```

```
In [16]: ytest.shape
```

```
Out[16]: (50,)
```

```
In [19]: regressor.fit(xtrain,ytrain)
```

```
Out[19]: LinearRegression()
```

```
In [21]: predictions=regressor.predict(xtest)
         predictions
```

```
Out[21]: array([21.29142142, 18.04194205, 10.03065109, 21.04818991, 20.75167113,
                24.52948332, 16.85936471, 15.69800653, 10.17237239, 18.90266535,
                15.82750789, 10.53768469, 18.86292327, 15.57969067, 17.89853384,
                15.35974564, 13.76329903, 21.00074026, 10.03000895, 19.20973979,
                11.12327234, 12.18140998,  8.67037889, 11.96248005, 12.63332284,
                16.83484748,  9.75317608, 21.06453498, 18.075131  , 19.52477289,
                22.04444095, 17.90745151, 16.48534479, 14.79190923, 21.3573104  ,
                16.9364024  , 17.18742868, 12.33688324, 21.02974736,  7.77584247,
                 5.413222  ,  9.64006117,  6.91408859, 19.22413789,  7.93800502,
                15.16854596, 13.73132724, 21.01419263, 20.49929224, 20.57792216])
```

```
In [25]: results=pd.DataFrame({"Actual":ytest,"Predictions":predictions})  
results
```

Out[25]:

	Actual	Predictions
58	23.8	21.291421
40	16.6	18.041942
34	11.9	10.030651
102	19.8	21.048190
184	17.6	20.751671
198	25.5	24.529483
95	16.9	16.859365
4	17.9	15.698007
29	10.5	10.172372
168	17.1	18.902665
171	17.5	15.827508
18	11.3	10.537685
11	17.4	18.862923
89	16.7	15.579691
110	18.4	17.898534
118	15.9	15.359746
159	12.9	13.763299
35	17.8	21.000740
136	9.5	10.030009
59	18.4	19.209740
51	10.7	11.123272
16	12.5	12.181410
44	8.5	8.670379
94	11.5	11.962480
31	11.9	12.633323
162	19.9	16.834847
38	10.1	9.753176
28	18.9	21.064535
193	19.6	18.075131
27	20.9	19.524773
47	23.2	22.044410
165	16.9	17.907452
194	17.3	16.485345
177	16.7	14.791909
176	20.2	21.357310
97	20.5	16.936402
174	16.5	17.187429
73	11.0	12.336883

	Actual	Predictions
69	22.3	21.029747
172	7.6	7.775842
108	5.3	5.413222
107	12.0	9.640061
189	6.7	6.914089
14	19.0	19.224138
56	5.5	7.938005
19	14.6	15.168546
114	14.6	13.731327
39	21.5	21.014193
185	22.6	20.499292
124	19.7	20.577922

In [22]: `from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score`

In [26]: `mean_absolute_error(ytest, predictions)`

Out[26]: 1.2187904107011895

In [27]: `r2_score(ytest, predictions)`

Out[27]: 0.9048917241361681

In [29]: `df.head(5)`

Out[29]:

	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	12.0
3	151.5	41.3	58.5	16.5
4	180.8	10.8	58.4	17.9

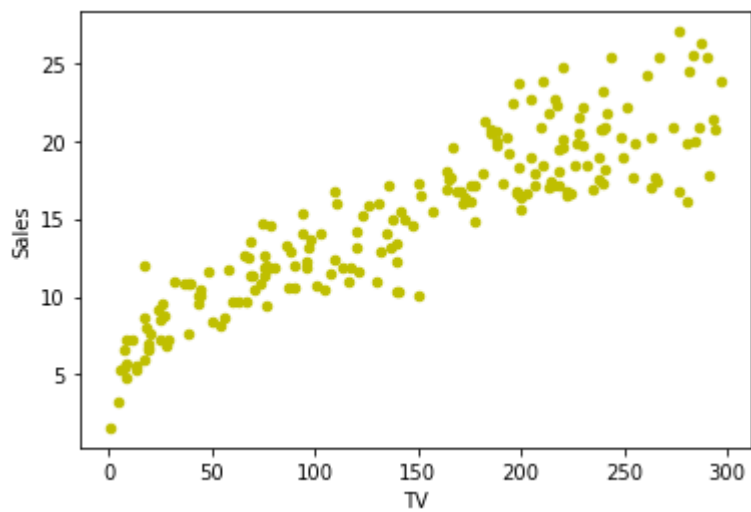
In [30]: `df.tail(5)`

Out[30]:

	TV	Radio	Newspaper	Sales
195	38.2	3.7	13.8	7.6
196	94.2	4.9	8.1	14.0
197	177.0	9.3	6.4	14.8
198	283.6	42.0	66.2	25.5
199	232.1	8.6	8.7	18.4

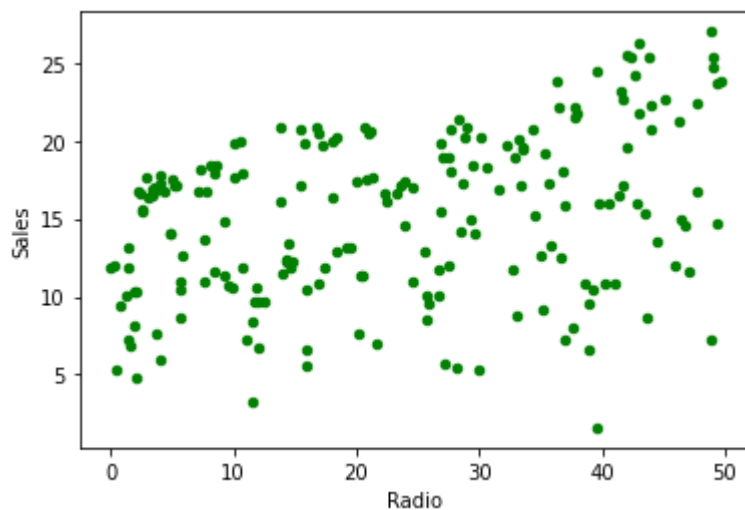
```
In [39]: import matplotlib.pyplot as plt  
df.plot(x="TV",y="Sales",c="y",kind="scatter")
```

Out[39]: <AxesSubplot:xlabel='TV', ylabel='Sales'>



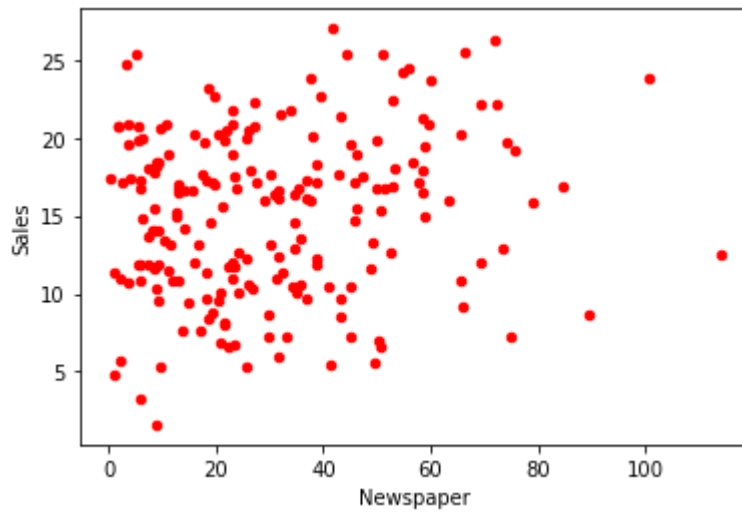
```
In [40]: df.plot(x="Radio",y="Sales",c="g",kind="scatter")
```

Out[40]: <AxesSubplot:xlabel='Radio', ylabel='Sales'>



```
In [41]: df.plot(x="Newspaper",y="Sales",c="r",kind="scatter")
```

```
Out[41]: <AxesSubplot:xlabel='Newspaper', ylabel='Sales'>
```



```
In [ ]:
```