

SALES PREDICTION

▼ IMPORTING LIBRARIES

```
import warnings
warnings.filterwarnings("ignore")
```

```
import numpy as np
import pandas as pd
```

▼ IMPORT DATASET

```
data=pd.read_csv("advertising.csv")
data.head()
```

	Unnamed: 0	TV	Radio	Newspaper	Sales
0	1	230.1	37.8	69.2	22.1
1	2	44.5	39.3	45.1	10.4
2	3	17.2	45.9	69.3	9.3
3	4	151.5	41.3	58.5	18.5
4	5	180.8	10.8	58.4	12.9

▼ REMOVE UNWANTED COLUMNS

```
data = data.drop(columns=['Unnamed: 0'])
```

▼ DATA ANALYSIS

```
data.shape
```

```
(200, 4)
```

```
data.describe()
```



	TV	Radio	Newspaper	Sales
count	200.000000	200.000000	200.000000	200.000000
mean	147.042500	23.264000	30.554000	14.022500
std	85.854236	14.846809	21.778621	5.217457
min	0.700000	0.000000	0.300000	1.600000
25%	74.375000	9.975000	12.750000	10.375000
50%	149.750000	22.900000	25.750000	12.900000
75%	218.825000	36.525000	45.100000	17.400000
max	296.400000	49.600000	114.000000	27.000000

```
data.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
```

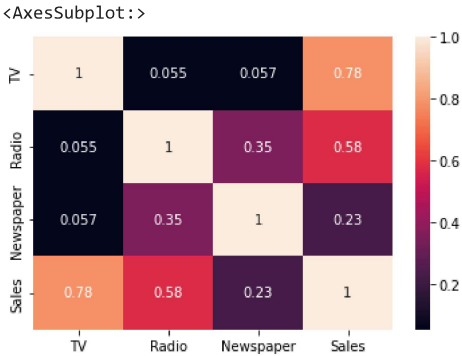
```
import seaborn as sns
```

```
correlation = data.corr()
```

```
correlation
```

	TV	Radio	Newspaper	Sales
TV	1.000000	0.054809	0.056648	0.782224
Radio	0.054809	1.000000	0.354104	0.576223
Newspaper	0.056648	0.354104	1.000000	0.228299
Sales	0.782224	0.576223	0.228299	1.000000

```
sns.heatmap(correlation,annot=True)
```



▼ SPLITTING DATA INTO TWO DATASETS

```
x=data.iloc[:, 0:-1]
```

```
x
```

	TV	Radio	Newspaper
0	230.1	37.8	69.2
1	44.5	39.3	45.1
2	17.2	45.9	69.3
3	151.5	41.3	58.5
4	180.8	10.8	58.4
...
195	38.2	3.7	13.8
196	94.2	4.9	8.1
197	177.0	9.3	6.4
198	283.6	42.0	66.2
199	232.1	8.6	8.7

200 rows × 3 columns

```
y=data.iloc[:, -1]
```

```
y
```

0	22.1
1	10.4
2	9.3
3	18.5
4	12.9
...	...
195	7.6

```
196    9.7
197   12.8
198   25.5
199   13.4
Name: Sales, Length: 200, dtype: float64
```

▼ SPLITTING INTO TRAIN AND TEST DATA

```
from sklearn.model_selection import train_test_split
xtrain,xtest,ytrain,ytest=train_test_split(x,y,test_size=0.3)
```

▼ CONVERTING DATA INTO INTEGERS

```
xtrain=xtrain.astype(int)
ytrain=ytrain.astype(int)
xtest=xtest.astype(int)
ytest=ytest.astype(int)
```

```
xtrain,ytrain
```

```
(      TV  Radio  Newspaper
137  273    28         59
141  193    35         75
66    31    24          2
135   48    47          8
6     57    32         23
..    ...    ...        ...
1     44    39         45
15   195    47         52
184  253    21         30
197  177     9          6
98   289    42         51
```

```
[140 rows x 3 columns],
```

```
137    20
141    19
66     9
135    11
6     11
```

```
..
1     10
15    22
184   17
197   12
98    25
```

```
Name: Sales, Length: 140, dtype: int32)
```

```
xtest,ytest
```

```
156 15
170 8
148 10
51 10
108 5
69 22
105 19
153 19
57 13
114 14
44 8
65 9
107 8
160 14
54 20
162 14
2 9
121 7
27 15
95 16
18 11
14 19
24 9
73 11
142 20
90 11
```

```
Name: Sales, dtype: int32)
```

```
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
```

```
scaled_x_train=sc.fit_transform(xtrain)
```

```
scaled_x_test=sc.fit_transform(xtest)
```

▼ TRAINING LINEAR REGRESSION MODEL

```
from sklearn.linear_model import LinearRegression
```

```
lr=LinearRegression()
```

```
lr.fit(scaled_x_train,ytrain)
```

```
LinearRegression()
```

```
print("accuracy:",lr.score(scaled_x_test,ytest))
```

```
accuracy: 0.883714891274296
```

