EX.NO: 3
2/8/25

Model Planning and Building

AIM

To implement and compare simple ML models (eg linear regression or k-means clustering)

CODE

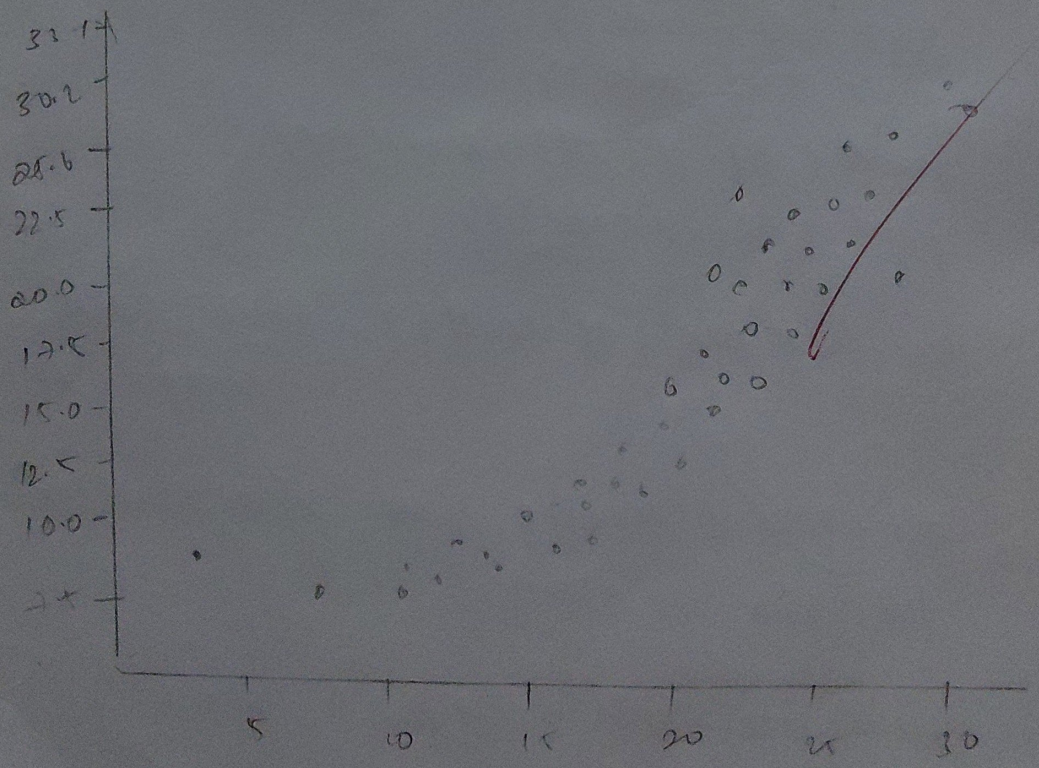
```
import pandas as pd
import seaborn as sns
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
from sklearn.preprocessing import StandardScaler

df = pd.read_csv('advertising.csv')
print(df.head())
print(df.describe())

X = df[['TV', 'Radio', 'Newspaper']]
y = df['Sales']

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

model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)

mse = mean_squared_error(y_test, y_pred)

plt.figure(figsize=(8,6))
```

Output

	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

Linear Regression MSE : 4.522515204129

Linear Regression: Actual vs Predicted Sales



```
sns.scatterplot(x=y_test, y=y_pred)
plt.xlabel("Actual Sales")
plt.ylabel("Predicted sales")
plt.title("Linear Regression: Actual vs Predicted Sales")
plt.show()

scaler = StandardScaler()
scaled = scaler.fit_transform(df[['TV', 'Radio',
                                  'Newspaper']])

kmeans = KMeans(n_clusters=3, random_state=0)
df['Cluster'] = kmeans.fit_predict(scaled)
plt.figure(figsize=(8,6))
sns.scatterplot(data=df, x='TV', y='Sales')
plt.show()
```

Results:

Thus the model building and planning has been implemented successfully