```
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from matplotlib import pyplot as plt
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, confusion_matrix


from google.colab import files
uploaded = files.upload()
```

Choose Files drug.csv

- **drug.csv**(text/csv) - 6029 bytes, last modified: 3/24/2025 - 100% done

```
Saving drug.csv to drug.csv
```

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


df
```

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	HIGH	HIGH	25.355	drugY
1	47	M	LOW	HIGH	13.093	drugC
2	47	M	LOW	HIGH	10.114	drugC
3	28	F	NORMAL	HIGH	7.798	drugX
4	61	F	LOW	HIGH	18.043	drugY
...	...	...	...	...	...	...
195	56	F	LOW	HIGH	11.567	drugC
196	16	M	LOW	HIGH	12.006	drugC
197	52	M	NORMAL	HIGH	9.894	drugX
198	23	M	NORMAL	NORMAL	14.020	drugX
199	40	F	LOW	NORMAL	11.349	drugX

200 rows × 6 columns

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
df['Sex'] = df['Sex'].map({'F': 0, 'M': 1})
df['BP'] = df['BP'].map({'LOW': 0, 'NORMAL': 1, 'HIGH': 2})
df['Cholesterol'] = df['Cholesterol'].map({'NORMAL': 0, 'HIGH': 1})


df
```

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	0	2	1	25.355	drugY
1	47	1	0	1	13.093	drugC
2	47	1	0	1	10.114	drugC
3	28	0	1	1	7.798	drugX
4	61	0	0	1	18.043	drugY
...	...	...	...	...	...	...
195	56	0	0	1	11.567	drugC
196	16	1	0	1	12.006	drugC
197	52	1	1	1	9.894	drugX
198	23	1	1	0	14.020	drugX
199	40	0	0	0	11.349	drugX

200 rows × 6 columns

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
X = df.drop(columns=['Drug'])  # Assuming 'Drug' is the target column
y = df['Drug']


df
```

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	0	2	1	25.355	drugY
1	47	1	0	1	13.093	drugC

Next steps: Generate code with df View recommended plots New interactive sheet

```
from sklearn.tree import DecisionTreeClassifier
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Create and train the Decision Tree model
model = DecisionTreeClassifier()
model.fit(X_train, y_train)

# Predict on test data
y_pred = model.predict(X_test)
```