


```
In [1]: import pandas as pd
```

```
In [2]: import matplotlib.pyplot as plt
```

TASK 3 : Build a decision tree classifier to predict whether a customer will purchase a product or service based on their demographic and behavioral data. Use a dataset such as the Bank Marketing dataset from the UCI Machine Learning Repository.



DATASET : <https://github.com/Prodigy-InfoTech/data-science-datasets/tree/main/Task%203>
(<https://github.com/Prodigy-InfoTech/data-science-datasets/tree/main/Task%203>)

```
In [3]: import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.tree import DecisionTreeClassifier, plot_tree
from sklearn.metrics import accuracy_score, classification_report
```

```
In [4]: df = pd.read_csv(r"C:\PYTHON DATASET\bank+marketing\bank\bank-full.csv", se
print(df.head())
print(df.info())
```

	age	job	marital	education	default	balance	housing	loan	\
0	58	management	married	tertiary	no	2143	yes	no	
1	44	technician	single	secondary	no	29	yes	no	
2	33	entrepreneur	married	secondary	no	2	yes	yes	
3	47	blue-collar	married	unknown	no	1506	yes	no	
4	33	unknown	single	unknown	no	1	no	no	

	contact	day	month	duration	campaign	pdays	previous	poutcome	y
0	unknown	5	may	261	1	-1	0	unknown	no
1	unknown	5	may	151	1	-1	0	unknown	no
2	unknown	5	may	76	1	-1	0	unknown	no
3	unknown	5	may	92	1	-1	0	unknown	no
4	unknown	5	may	198	1	-1	0	unknown	no

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 45211 entries, 0 to 45210

Data columns (total 17 columns):

#	Column	Non-Null	Count	Dtype
0	age	45211	non-null	int64
1	job	45211	non-null	object
2	marital	45211	non-null	object
3	education	45211	non-null	object
4	default	45211	non-null	object
5	balance	45211	non-null	int64
6	housing	45211	non-null	object
7	loan	45211	non-null	object
8	contact	45211	non-null	object
9	day	45211	non-null	int64
10	month	45211	non-null	object
11	duration	45211	non-null	int64
12	campaign	45211	non-null	int64
13	pdays	45211	non-null	int64
14	previous	45211	non-null	int64
15	poutcome	45211	non-null	object
16	y	45211	non-null	object

dtypes: int64(7), object(10)

memory usage: 5.9+ MB

None

```
In [5]: label = LabelEncoder()
for col in df.select_dtypes(include='object').columns:
    df[col] = label.fit_transform(df[col])
```

```
In [6]: X = df.drop('y', axis=1)
y = df['y']
```

```
In [7]: X_train, X_test, y_train, y_test = train_test_split(
        X, y, test_size=0.2, random_state=42
    )
```

```
In [8]: model = DecisionTreeClassifier(max_depth=5, random_state=42)
        model.fit(X_train, y_train)
```

Out[8]: DecisionTreeClassifier(max_depth=5, random_state=42)

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.

On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

```
In [9]: y_pred = model.predict(X_test)

        print("Accuracy:", accuracy_score(y_test, y_pred))
        print(classification_report(y_test, y_pred))
```

Accuracy: 0.8935087913303107

	precision	recall	f1-score	support
0	0.92	0.96	0.94	7952
1	0.59	0.40	0.48	1091
accuracy			0.89	9043
macro avg	0.75	0.68	0.71	9043
weighted avg	0.88	0.89	0.88	9043

```
In [10]: plt.figure(figsize=(20,10))
        plot_tree(model,
                  feature_names=X.columns,
                  class_names=["No", "Yes"],
                  filled=True)
        plt.show()
```

