In [5]:
```python
import pandas as pd
```

In [1]:
```python
from sklearn.tree import DecisionTreeClassifier
```

In [2]:
```python
from sklearn.model_selection import train_test_split
```

In [3]:
```python
from sklearn import metrics
```

In [6]:
```python
pima =  pd.read_csv("diabetes.csv")
```

In [7]:
```python
pima
```

Out[7]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
...	...	...	...	...	...	...	...	...	...
763	10	101	76	48	180	32.9	0.171	63	0
764	2	122	70	27	0	36.8	0.340	27	0
765	5	121	72	23	112	26.2	0.245	30	0
766	1	126	60	0	0	30.1	0.349	47	1
767	1	93	70	31	0	30.4	0.315	23	0

768 rows × 9 columns

In [8]:
```python
feature_cols = ['Pregnancies', 'Insulin', 'BMI', 'Age', 'Glucose', 'BloodPressure', 'DiabetesPedigreeFunction']
```

In [9]:
```python
x = pima[feature_cols]
```

```
In [10]:  y = pima.Outcome
```

```
In [11]:  y
```

```
Out[11]:  0      1
          1      0
          2      1
          3      0
          4      1
                ..
          763    0
          764    0
          765    0
          766    1
          767    0
          Name: Outcome, Length: 768, dtype: int64
```

```
In [12]:  X_train, X_test, Y_train, Y_test = train_test_split(x, y, test_size = 0.3, random_state=1)
```

```
In [13]:  classifier = DecisionTreeClassifier()
```

```
In [14]:  classifier = classifier.fit(X_train, Y_train)
```

```
In [15]:  y_pred = classifier.predict(X_test)
```

```
In [16]:  print(y_pred)

          [0 0 0 1 0 1 0 0 0 0 0 1 1 1 1 0 0 1 0 0 1 0 1 0 0 1 0 1 0 0 0 1 0 1 1 1 0
           1 0 1 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 1 0 0 0 0 0 0 1 0 0 0 1 1 1 1 0 0
           1 0 1 0 0 1 1 0 1 0 0 1 0 1 1 0 0 0 1 0 1 0 1 0 1 0 0 1 1 1 0 0 0 1 0 0 1
           0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0
           0 0 1 0 1 0 1 0 1 0 0 0 1 0 0 0 1 0 0 1 1 0 1 1 0 0 1 0 1 0 0 0 1 1 0 0 0
           0 0 1 0 0 0 0 0 1 0 1 0 0 1 1 0 1 0 1 1 0 0 0 1 0 0 0 1 1 1 0 0 0 0 0 0
           0 1 0 0 0 0 1 1]
```

```
In [17]:  from sklearn.metrics import confusion_matrix
```

In [18]:
```python
print(confusion_matrix(Y_test, y_pred))
```

```
[[113  33]
 [ 39  46]]
```

In [19]:
```python
print("Accuracy:", metrics.accuracy_score(Y_test,y_pred))
```

```
Accuracy: 0.6883116883116883
```

In [20]:
```python
clf = DecisionTreeClassifier(criterion="entropy", max_depth=3)
```

In [22]:
```python
clf = clf.fit(X_train,Y_train)
```

In [23]:
```python
y_pred = clf.predict(X_test)
```

In [25]:
```python
print("Accuracy:",metrics.accuracy_score(Y_test, y_pred))
```

```
Accuracy: 0.7705627705627706
```

In [ ]: