libraries

```
In [55]:  import pandas as pd
          import numpy as np
          from matplotlib import pyplot as plt
```

```
In [2]:  data=pd.read_csv('C:/Users/91733/Downloads/User_Data (1).csv')
```

```
In [3]:  data
```

Out[3]:

	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19	19000	0
1	15810944	Male	35	20000	0
2	15668575	Female	26	43000	0
3	15603246	Female	27	57000	0
4	15804002	Male	19	76000	0
...	...	...	...	...	...
395	15691863	Female	46	41000	1
396	15706071	Male	51	23000	1
397	15654296	Female	50	20000	1
398	15755018	Male	36	33000	0
399	15594041	Female	49	36000	1

400 rows × 5 columns

```
In [4]:  x=data.iloc[:,[2,3]].values
```

In [5]: `x`

```
[      25,   79000],
[      27,   54000],
[      30,  135000],
[      31,   89000],
[      24,   32000],
[      18,   44000],
[      29,   83000],
[      35,   23000],

[      27,   58000],
[      24,   55000],
[      23,   48000],
[      28,   79000],
[      22,   18000],
[      32,  117000],
[      27,   20000],
[      25,   87000],
[      23,   66000],
[      32,  120000],
[      59,   83000],
```

In [21]: `y=data.iloc[:,4].values`

In [22]: `y`

Out[22]:
```
array([0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1,
       0, 1, 1, 1, 0, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 0, 0, 0, 1, 1, 0,
       1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 1, 1, 0,
       1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1,
       0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0, 1,
       1, 0, 0, 1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 1,
       0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0,
       1, 0, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1,
       0, 1, 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1,
       1, 1, 0, 1], dtype=int64)
```

In [23]: `from sklearn.model_selection import train_test_split`

In [24]: `x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.25)`

In [31]:
```python
x_train.shape
```

Out[31]: (300, 2)

In [30]:
```python
x_test.shape
```

Out[30]: (100, 2)

In [32]:
```python
y_test.shape
```

Out[32]: (100,)

In [29]:
```python
y_train.shape
```

Out[29]: (300,)

In [34]:
```python
from sklearn.preprocessing import StandardScaler
```

In [35]:
```python
stx=StandardScaler()
```

In [36]:
```python
x_train=stx.fit_transform(x_train)
```

In [37]:
```python
x_train
```

```
       [-1.38222303, -0.37306406],
       [-1.19209779, -1.59213637],
       [-1.66741589, -1.56379982],
       [-0.33652521,  0.10805672],
       [-1.47728865, -0.2319819 ],
       [ 0.23385651, -0.74203983],
       [-0.52665245, -0.31699156],
       [-0.33652521,  1.04316291],
       [ 0.89930184,  1.35486498],
       [-0.33652521,  2.14828842],
       [ 1.84993804, -0.68536672],
       [-1.00197055,  0.44809533],
       [ 0.80423822, -1.1670881 ],
       [ 1.75487442, -1.2804343 ],
       [-0.71677969,  0.50476844],
       [ 1.84993804, -0.94039569],
       [-1.95260675,  0.30641258],
       [ 0.13879289,  0.10805672],
       [-0.33652521, -0.94039569],
       [ 0.32892013,  0.53310499],
```

In [38]:
```python
x_test=stx.fit_transform(x_test)
```

In [39]: x_test

Out[39]: array([[-0.60889392, 0.59377514],
                [-0.71037624, 0.66128795],
                [-1.52223479, -1.43160908],
                [1.62371711, 2.72042857],
                [-0.5074116 , 0.72880075],
                [0.60889392, 0.49250593],
                [-1.52223479, 0.28996751],
                [-1.31927015, 0.52626233],
                [-1.11630551, -1.43160908],
                [0.30444696, -0.14886574],
                [0.   , -0.01384013],
                [-1.42075247, 0.08742909],
                [-1.72519943, 0.08742909],
                [-0.60889392, -0.75648101],
                [0.40592928, 0.39123672],
                [1.42075247, -1.26282706],
                [-0.81185856, -1.6341475],
                [1.11630551, -1.12780145],
                [-0.71037624, 0.79631356],
                [-0.5074116 , -0.55394258],
                [-0.91334088, -1.02653223],
                [0.   , -0.52018618],
                [1.21778783, -0.62145539],
                [-0.71037624, -0.21637855],
                [0.60889392, 0.2224547],
                [1.82668175, 1.30265962],
                [-0.10148232, -0.75648101],
                [1.11630551, 0.56001874],
                [-0.5074116 , 0.45874953],
                [-0.40592928, 0.08742909],
                [-1.11630551, -0.35140416],
                [2.23261103, -0.18262214],
                [-0.91334088, -0.45267337],
                [0.30444696, 1.57271085],
                [-0.60889392, 0.59377514],
                [0.   , 0.2562111],
                [0.60889392, -0.08135293],
                [0.30444696, -0.52018618],
                [-1.52223479, -0.55394258],
                [-1.62371711, 0.15494189],
                [-0.5074116 , -0.11510934],
                [-0.91334088, 0.49250593],
                [-1.42075247, 0.76255716],
                [1.42075247, 0.28996751],
                [-0.40592928, 0.1886983],
                [2.33409335, 1.20139041],
                [-0.60889392, 2.78794138],
                [0.5074116 , 2.58540295],
                [-0.20296464, -1.26282706],
                [0.   , 0.76255716],
                [1.0148232 , -1.46536548],
                [0.   , -0.14886574],
                [1.82668175, 2.61915936],
                [1.0148232 , -0.6889682],

```
       [ 1.11630551,  1.03260839],
       [-0.60889392, -1.26282706],
       [ 0.81185856,  1.57271085],
       [ 0.30444696,  0.49250593],
       [ 0.71037624,  0.45874953],
       [-1.21778783, -1.53287829],
       [ 0.30444696, -0.52018618],
       [ 0.        ,  0.2224547 ],
       [ 0.91334088, -0.89150662],
       [ 0.20296464,  0.42499312],
       [ 0.71037624, -0.01384013],
       [ 0.20296464,  0.1886983 ],
       [-1.72519943, -0.45267337],
       [-0.30444696, -1.6003911 ],
       [ 1.31927015, -0.82399381],
       [ 1.31927015,  1.03260839],
       [-0.60889392, -0.75648101],
       [-0.91334088, -0.75648101],
       [ 1.0148232 ,  0.45874953],
       [ 0.        ,  0.1886983 ],
       [ 2.13112871, -0.18262214],
       [ 0.60889392, -0.48642978],
       [ 0.10148232, -0.08135293],
       [-1.62371711, -1.33033987],
       [-0.30444696,  1.74149286],
       [ 0.30444696, -0.21637855],
       [ 0.71037624,  0.49250593],
       [-1.72519943, -0.7227246 ],
       [ 0.40592928,  0.1886983 ],
       [-0.40592928, -1.06028864],
       [ 0.71037624,  2.82169778],
       [ 1.42075247,  0.79631356],
       [ 0.20296464, -0.41891697],
       [-1.31927015, -0.08135293],
       [ 1.11630551, -1.46536548],
       [ 0.        ,  0.2224547 ],
       [-0.40592928,  0.35748031],
       [ 0.        , -1.46536548],
       [-1.62371711, -1.49912189],
       [ 0.20296464, -0.28389135],
       [ 0.20296464,  0.93133918],
       [-0.20296464, -0.48642978],
       [-0.81185856, -0.28389135],
       [-0.20296464, -1.16155785],
       [-1.11630551, -1.29658346],
       [ 1.82668175,  0.56001874]])
```

In [40]:
```python
from sklearn.ensemble import RandomForestClassifier
```

In [42]:
```python
classifier=RandomForestClassifier(n_estimators=10,criterion="entropy")
```

In [43]:
```python
classifier.fit(x_train,y_train)
```

Out[43]:
```
            ▼                RandomForestClassifier
RandomForestClassifier(criterion='entropy', n_estimators=10)
```

In [48]:
```python
y_pred=classifier.predict(x_test)
```

In [49]:
```python
y_pred
```

Out[49]:
```
array([1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0,
       1, 0, 1, 1, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1,
       0, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0,
       0, 0, 1, 1, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0,
       1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1], dtype=int64)
```

In [51]:
```python
from sklearn.metrics import confusion_matrix
```

In [52]:
```python
cm=confusion_matrix(y_test,y_pred)
```

In [53]:
```python
cm
```

Out[53]:
```
array([[57, 21],
       [ 1, 21]], dtype=int64)
```

In [54]:
```python
#78 are correct
#22 are wrong
```

In [ ]: