In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:
```
f = pd.read_csv("C:\\Users\\dell\\Downloads\\Compressed\\Churn_Modelling.csv")
df = pd.DataFrame(f)
```

In [3]:
```
df
```

Out[3]:

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
0	1	15634602	Hargrave	619	France	Female	42	2	0.00	1	1	1	101348.88	1
1	2	15647311	Hill	608	Spain	Female	41	1	83807.86	1	0	1	112542.58	0
2	3	15619304	Onio	502	France	Female	42	8	159660.80	3	1	0	113931.57	1
3	4	15701354	Boni	699	France	Female	39	1	0.00	2	0	0	93826.63	0
4	5	15737888	Mitchell	850	Spain	Female	43	2	125510.82	1	1	1	79084.10	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9995	9996	15606229	Obijiaku	771	France	Male	39	5	0.00	2	1	0	96270.64	0
9996	9997	15569892	Johnstone	516	France	Male	35	10	57369.61	1	1	1	101699.77	0
9997	9998	15584532	Liu	709	France	Female	36	7	0.00	1	0	1	42095.58	1
9998	9999	15682355	Sabbatini	772	Germany	Male	42	3	75075.31	2	1	0	92888.52	1
9999	10000	15628319	Walker	792	France	Female	28	4	130142.79	1	1	0	38190.78	0

10000 rows × 14 columns

In [19]:
```
dummies = pd.get_dummies(df.Geography)
```

In [20]:
```
df_1 = pd.concat([df,dummies],axis='columns')
```

Out[21]:

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	France	Germany	Spain
0	1	15634602	Hargrave	619	France	Female	42	2	0.00	1	1	1	101348.88	1	1	0	0
1	2	15647311	Hill	608	Spain	Female	41	1	83807.86	1	0	1	112542.58	0	0	0	1
2	3	15619304	Onio	502	France	Female	42	8	159660.80	3	1	0	113931.57	1	1	0	0
3	4	15701354	Boni	699	France	Female	39	1	0.00	2	0	0	93826.63	0	1	0	0
4	5	15737888	Mitchell	850	Spain	Female	43	2	125510.82	1	1	1	79084.10	0	0	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9995	9996	15606229	Obijiaku	771	France	Male	39	5	0.00	2	1	0	96270.64	0	1	0	0
9996	9997	15569892	Johnstone	516	France	Male	35	10	57369.61	1	1	1	101699.77	0	1	0	0
9997	9998	15584532	Liu	709	France	Female	36	7	0.00	1	0	1	42095.58	1	1	0	0
9998	9999	15682355	Sabbatini	772	Germany	Male	42	3	75075.31	2	1	0	92888.52	1	0	1	0
9999	10000	15628319	Walker	792	France	Female	28	4	130142.79	1	1	0	38190.78	0	1	0	0

10000 rows × 17 columns

In [22]:
```
df = df.drop(['Geography'],axis = 'columns')
```

In [25]:
```
from sklearn.preprocessing import LabelEncoder
```

In [27]:
```
labelencoder = LabelEncoder()
```

In [29]:
```
df.Gender = labelencoder.fit_transform(df.Gender)
```

In [30]:
```
df
```

Out[30]:

	RowNumber	CustomerId	Surname	CreditScore	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	France	Germany	Spain
0	1	15634602	Hargrave	619	0	42	2	0.00	1	1	1	101348.88	1	1	0	0
1	2	15647311	Hill	608	0	41	1	83807.86	1	0	1	112542.58	0	0	0	1
2	3	15619304	Onio	502	0	42	8	159660.80	3	1	0	113931.57	1	1	0	0
3	4	15701354	Boni	699	0	39	1	0.00	2	0	0	93826.63	0	1	0	0
4	5	15737888	Mitchell	850	0	43	2	125510.82	1	1	1	79084.10	0	0	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9995	9996	15606229	Obijiaku	771	1	39	5	0.00	2	1	0	96270.64	0	1	0	0
9996	9997	15569892	Johnstone	516	1	35	10	57369.61	1	1	1	101699.77	0	1	0	0
9997	9998	15584532	Liu	709	0	36	7	0.00	1	0	1	42095.58	1	1	0	0
9998	9999	15682355	Sabbatini	772	1	42	3	75075.31	2	1	0	92888.52	1	0	1	0
9999	10000	15628319	Walker	792	0	28	4	130142.79	1	1	0	38190.78	0	1	0	0

10000 rows × 16 columns

In [32]:
```
y = df.Exited
```

In [35]:
```
X = df.drop(['RowNumber','CustomerId','Surname','Exited'],axis = 'columns')
```

In [36]:
```
X
```

Out[36]:

	CreditScore	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	France	Germany	Spain
0	619	0	42	2	0.00	1	1	1	101348.88	1	0	0
1	608	0	41	1	83807.86	1	0	1	112542.58	0	0	1
2	502	0	42	8	159660.80	3	1	0	113931.57	1	0	0
3	699	0	39	1	0.00	2	0	0	93826.63	1	0	0
4	850	0	43	2	125510.82	1	1	1	79084.10	0	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...
9995	771	1	39	5	0.00	2	1	0	96270.64	1	0	0
9996	516	1	35	10	57369.61	1	1	1	101699.77	1	0	0
9997	709	0	36	7	0.00	1	0	1	42095.58	1	0	0
9998	772	1	42	3	75075.31	2	1	0	92888.52	0	1	0
9999	792	0	28	4	130142.79	1	1	0	38190.78	1	0	0

10000 rows × 12 columns

In [37]:
```
from sklearn.model_selection import train_test_split
```

In [38]:
```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size = 0.2,random_state = 0)
```

In [39]:
```
from sklearn.preprocessing import StandardScaler
```

In [40]:
```
sc = StandardScaler()
```

In [41]:
```
X_train = sc.fit_transform(X_train)
```

In [42]:
```
X_test = sc.fit_transform(X_test)
```

In [60]:
```
import tensorflow as tf
```

In [61]:
```
classifier = tf.keras.models.Sequential()
```

In [64]:
```
classifier.add(tf.keras.layers.Dense(units=6,activation='relu'))
```

In [66]:
```
classifier.add(tf.keras.layers.Dense(units=6,activation='relu'))
```

In [67]:
```
classifier.add(tf.keras.layers.Dense(units=1,activation='sigmoid'))
```

In [68]:
```
classifier.compile(optimizer='adam',loss='binary_crossentropy',metrics = ['accuracy'])
```

In [69]:
```
classifier.fit(X_train,y_train,batch_size=30,epochs = 100)
```

```
Epoch 1/100
267/267 [==============================] - 1s 749us/step - loss: 0.5798 - accuracy: 0.7576
Epoch 2/100
267/267 [==============================] - 0s 883us/step - loss: 0.4918 - accuracy: 0.7954
Epoch 3/100
267/267 [==============================] - 0s 792us/step - loss: 0.4468 - accuracy: 0.7983
Epoch 4/100
267/267 [==============================] - 0s 811us/step - loss: 0.4237 - accuracy: 0.8046
Epoch 5/100
267/267 [==============================] - 0s 800us/step - loss: 0.4067 - accuracy: 0.8238
Epoch 6/100
267/267 [==============================] - 0s 780us/step - loss: 0.3912 - accuracy: 0.8331
Epoch 7/100
267/267 [==============================] - 0s 777us/step - loss: 0.3799 - accuracy: 0.8378
Epoch 8/100
267/267 [==============================] - 0s 763us/step - loss: 0.3721 - accuracy: 0.8438
Epoch 9/100
267/267 [==============================] - 0s 756us/step - loss: 0.3661 - accuracy: 0.8471
Epoch 10/100
267/267 [==============================] - 0s 752us/step - loss: 0.3612 - accuracy: 0.8505
Epoch 11/100
267/267 [==============================] - 0s 762us/step - loss: 0.3578 - accuracy: 0.8525
Epoch 12/100
267/267 [==============================] - 0s 766us/step - loss: 0.3546 - accuracy: 0.8546
Epoch 13/100
267/267 [==============================] - 0s 759us/step - loss: 0.3527 - accuracy: 0.8550
Epoch 14/100
267/267 [==============================] - 0s 759us/step - loss: 0.3515 - accuracy: 0.8571
Epoch 15/100
267/267 [==============================] - 0s 763us/step - loss: 0.3496 - accuracy: 0.8594
Epoch 16/100
267/267 [==============================] - 0s 747us/step - loss: 0.3484 - accuracy: 0.8590
Epoch 17/100
267/267 [==============================] - 0s 776us/step - loss: 0.3473 - accuracy: 0.8597
Epoch 18/100
267/267 [==============================] - 0s 782us/step - loss: 0.3463 - accuracy: 0.8586
Epoch 19/100
267/267 [==============================] - 0s 764us/step - loss: 0.3457 - accuracy: 0.8620
Epoch 20/100
267/267 [==============================] - 0s 766us/step - loss: 0.3451 - accuracy: 0.8605
Epoch 21/100
267/267 [==============================] - 0s 738us/step - loss: 0.3442 - accuracy: 0.8608
Epoch 22/100
267/267 [==============================] - 0s 791us/step - loss: 0.3438 - accuracy: 0.8610
Epoch 23/100
267/267 [==============================] - 0s 780us/step - loss: 0.3435 - accuracy: 0.8610
Epoch 24/100
267/267 [==============================] - 0s 772us/step - loss: 0.3428 - accuracy: 0.8605
Epoch 25/100
267/267 [==============================] - 0s 781us/step - loss: 0.3421 - accuracy: 0.8635
Epoch 26/100
267/267 [==============================] - 0s 772us/step - loss: 0.3418 - accuracy: 0.8614
Epoch 27/100
267/267 [==============================] - 0s 749us/step - loss: 0.3415 - accuracy: 0.8614
Epoch 28/100
267/267 [==============================] - 0s 778us/step - loss: 0.3411 - accuracy: 0.8619
Epoch 29/100
267/267 [==============================] - 0s 748us/step - loss: 0.3409 - accuracy: 0.8621
Epoch 30/100
267/267 [==============================] - 0s 791us/step - loss: 0.3407 - accuracy: 0.8619
Epoch 31/100
267/267 [==============================] - 0s 761us/step - loss: 0.3404 - accuracy: 0.8619
Epoch 32/100
267/267 [==============================] - 0s 776us/step - loss: 0.3400 - accuracy: 0.8616
Epoch 33/100
267/267 [==============================] - 0s 778us/step - loss: 0.3398 - accuracy: 0.8616
Epoch 34/100
267/267 [==============================] - 0s 785us/step - loss: 0.3395 - accuracy: 0.8621
Epoch 35/100
267/267 [==============================] - 0s 771us/step - loss: 0.3392 - accuracy: 0.8648
Epoch 36/100
267/267 [==============================] - 0s 766us/step - loss: 0.3387 - accuracy: 0.8618
Epoch 37/100
267/267 [==============================] - 0s 773us/step - loss: 0.3389 - accuracy: 0.8634
Epoch 38/100
267/267 [==============================] - 0s 760us/step - loss: 0.3384 - accuracy: 0.8624
Epoch 39/100
267/267 [==============================] - 0s 751us/step - loss: 0.3381 - accuracy: 0.8621
Epoch 40/100
267/267 [==============================] - 0s 769us/step - loss: 0.3376 - accuracy: 0.8629
Epoch 41/100
267/267 [==============================] - 0s 772us/step - loss: 0.3373 - accuracy: 0.8624
Epoch 42/100
267/267 [==============================] - 0s 760us/step - loss: 0.3373 - accuracy: 0.8621
Epoch 43/100
267/267 [==============================] - 0s 747us/step - loss: 0.3367 - accuracy: 0.8634
Epoch 44/100
267/267 [==============================] - 0s 747us/step - loss: 0.3364 - accuracy: 0.8637
Epoch 45/100
267/267 [==============================] - 0s 755us/step - loss: 0.3360 - accuracy: 0.8629
Epoch 46/100
267/267 [==============================] - 0s 752us/step - loss: 0.3359 - accuracy: 0.8629
Epoch 47/100
267/267 [==============================] - 0s 770us/step - loss: 0.3349 - accuracy: 0.8636
Epoch 48/100
267/267 [==============================] - 0s 753us/step - loss: 0.3351 - accuracy: 0.8626
Epoch 49/100
267/267 [==============================] - 0s 752us/step - loss: 0.3348 - accuracy: 0.8634
Epoch 50/100
267/267 [==============================] - 0s 781us/step - loss: 0.3343 - accuracy: 0.8646
Epoch 51/100
267/267 [==============================] - 0s 777us/step - loss: 0.3338 - accuracy: 0.8633
Epoch 52/100
267/267 [==============================] - 0s 745us/step - loss: 0.3339 - accuracy: 0.8644
Epoch 53/100
267/267 [==============================] - 0s 770us/step - loss: 0.3341 - accuracy: 0.8639
Epoch 54/100
267/267 [==============================] - 0s 757us/step - loss: 0.3336 - accuracy: 0.8648
Epoch 55/100
267/267 [==============================] - 0s 767us/step - loss: 0.3334 - accuracy: 0.8641
Epoch 56/100
267/267 [==============================] - 0s 767us/step - loss: 0.3332 - accuracy: 0.8651
Epoch 57/100
267/267 [==============================] - 0s 771us/step - loss: 0.3329 - accuracy: 0.8639
Epoch 58/100
267/267 [==============================] - 0s 762us/step - loss: 0.3327 - accuracy: 0.8639
Epoch 59/100
267/267 [==============================] - 0s 787us/step - loss: 0.3328 - accuracy: 0.8636
Epoch 60/100
267/267 [==============================] - 0s 777us/step - loss: 0.3323 - accuracy: 0.8646
Epoch 61/100
267/267 [==============================] - 0s 765us/step - loss: 0.3326 - accuracy: 0.8652
Epoch 62/100
267/267 [==============================] - 0s 775us/step - loss: 0.3320 - accuracy: 0.8660
Epoch 63/100
267/267 [==============================] - 0s 767us/step - loss: 0.3320 - accuracy: 0.8649
Epoch 64/100
267/267 [==============================] - 0s 762us/step - loss: 0.3318 - accuracy: 0.8655
Epoch 65/100
267/267 [==============================] - 0s 770us/step - loss: 0.3316 - accuracy: 0.8656
Epoch 66/100
267/267 [==============================] - 0s 777us/step - loss: 0.3317 - accuracy: 0.8635
Epoch 67/100
267/267 [==============================] - 0s 777us/step - loss: 0.3312 - accuracy: 0.8648
Epoch 68/100
267/267 [==============================] - 0s 765us/step - loss: 0.3312 - accuracy: 0.8651
Epoch 69/100
267/267 [==============================] - 0s 779us/step - loss: 0.3313 - accuracy: 0.8659
Epoch 70/100
267/267 [==============================] - 0s 773us/step - loss: 0.3311 - accuracy: 0.8651
Epoch 71/100
267/267 [==============================] - 0s 765us/step - loss: 0.3309 - accuracy: 0.8687
Epoch 72/100
267/267 [==============================] - 0s 768us/step - loss: 0.3307 - accuracy: 0.8660
Epoch 73/100
267/267 [==============================] - 0s 758us/step - loss: 0.3306 - accuracy: 0.8645
Epoch 74/100
267/267 [==============================] - 0s 780us/step - loss: 0.3303 - accuracy: 0.8664
Epoch 75/100
267/267 [==============================] - 0s 724us/step - loss: 0.3303 - accuracy: 0.8665
Epoch 76/100
267/267 [==============================] - 0s 659us/step - loss: 0.3303 - accuracy: 0.8652
Epoch 77/100
267/267 [==============================] - 0s 657us/step - loss: 0.3301 - accuracy: 0.8677
Epoch 78/100
267/267 [==============================] - 0s 783us/step - loss: 0.3302 - accuracy: 0.8674
Epoch 79/100
267/267 [==============================] - 0s 761us/step - loss: 0.3299 - accuracy: 0.8677
Epoch 80/100
267/267 [==============================] - 0s 774us/step - loss: 0.3296 - accuracy: 0.8671
Epoch 81/100
267/267 [==============================] - 0s 754us/step - loss: 0.3295 - accuracy: 0.8669
Epoch 82/100
267/267 [==============================] - 0s 754us/step - loss: 0.3294 - accuracy: 0.8662
Epoch 83/100
267/267 [==============================] - 0s 719us/step - loss: 0.3294 - accuracy: 0.8666
Epoch 84/100
267/267 [==============================] - 0s 738us/step - loss: 0.3292 - accuracy: 0.8679
Epoch 85/100
267/267 [==============================] - 0s 749us/step - loss: 0.3292 - accuracy: 0.8677
Epoch 86/100
267/267 [==============================] - 0s 773us/step - loss: 0.3291 - accuracy: 0.8670
Epoch 87/100
267/267 [==============================] - 0s 746us/step - loss: 0.3289 - accuracy: 0.8669
Epoch 88/100
267/267 [==============================] - 0s 770us/step - loss: 0.3287 - accuracy: 0.8683
Epoch 89/100
267/267 [==============================] - 0s 765us/step - loss: 0.3287 - accuracy: 0.8683
Epoch 90/100
267/267 [==============================] - 0s 767us/step - loss: 0.3285 - accuracy: 0.8676
Epoch 91/100
267/267 [==============================] - 0s 768us/step - loss: 0.3286 - accuracy: 0.8679
Epoch 92/100
267/267 [==============================] - 0s 780us/step - loss: 0.3285 - accuracy: 0.8670
Epoch 93/100
267/267 [==============================] - 0s 749us/step - loss: 0.3284 - accuracy: 0.8675
Epoch 94/100
267/267 [==============================] - 0s 783us/step - loss: 0.3284 - accuracy: 0.8680
Epoch 95/100
267/267 [==============================] - 0s 773us/step - loss: 0.3285 - accuracy: 0.8670
Epoch 96/100
267/267 [==============================] - 0s 762us/step - loss: 0.3280 - accuracy: 0.8674
Epoch 97/100
267/267 [==============================] - 0s 742us/step - loss: 0.3280 - accuracy: 0.8673
Epoch 98/100
267/267 [==============================] - 0s 767us/step - loss: 0.3278 - accuracy: 0.8669
Epoch 99/100
267/267 [==============================] - 0s 775us/step - loss: 0.3280 - accuracy: 0.8666
Epoch 100/100
267/267 [==============================] - 0s 766us/step - loss: 0.3277 - accuracy: 0.8681
<keras.callbacks.History at 0x2881b022200>
```

In [69]:
```
y_pred = classifier.predict(X_test)
```

In [72]:
```
y_pred = (y_pred > 0.5)
```

In [74]:
```
y_pred
```

Out[74]:
```
array([[False],
       [False],
       [False],
       ...,
       [False],
       [False],
       [False]])
```

In [75]:
```
from sklearn.metrics import confusion_matrix
```

In [76]:
```
cm = confusion_matrix(y_test,y_pred)
```

In [77]:
```
cm
```

Out[77]:
```
array([[1533,   62],
       [ 205,  200]], dtype=int64)
```

In [79]:
```
(1533 + 200) / 2000
```

Out[79]:
```
0.8665
```

In [ ]: