In [1]:

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
```

In [2]:

```
df=pd.read_csv("/home/ubuntu/Downloads/Social_Network_Ads.csv")
df
```

Out[2]:

	Age	EstimatedSalary	Purchased
0	19	19000	0
1	35	20000	0
2	26	43000	0
3	27	57000	0
4	19	76000	0
...	...	...	...
395	46	41000	1
396	51	23000	1
397	50	20000	1
398	36	33000	0
399	49	36000	1

400 rows × 3 columns

In [3]:

```
df.isnull().sum()
```

Out[3]:

```
Age                0
EstimatedSalary    0
Purchased          0
dtype: int64
```

In [4]:

```
x=df.drop("Purchased",axis=1)
x
```

Out[4]:

	Age	EstimatedSalary
0	19	19000
1	35	20000
2	26	43000
3	27	57000
4	19	76000
...	...	...
395	46	41000
396	51	23000
397	50	20000
398	36	33000
399	49	36000

400 rows × 2 columns

In [5]:

```
y=df.Purchased
y
```

Out[5]:

```
0      0
1      0
2      0
3      0
4      0
      ..
395    1
396    1
397    1
398    0
399    1
Name: Purchased, Length: 400, dtype: int64
```

In [6]:

```
from sklearn.model_selection import train_test_split
xtrain, xtest, ytrain, ytest =train_test_split(x, y, test_size =0.2,random_state =
```

In [7]:

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
xtrain = sc.fit_transform(xtrain)
xtest = sc.transform(xtest)
```

In [8]:

```
xtrain
```

Out[8]:

```
array([[ 1.92295008e+00,  2.14601566e+00],
       [ 2.02016082e+00,  3.78719297e-01],
       [-1.38221530e+00, -4.32498705e-01],
       [-1.18779381e+00, -1.01194013e+00],
       [ 1.92295008e+00, -9.25023920e-01],
       [ 3.67578135e-01,  2.91803083e-01],
       [ 1.73156642e-01,  1.46942725e-01],
       [ 2.02016082e+00,  1.74040666e+00],
       [ 7.56421121e-01, -8.38107706e-01],
       [ 2.70367388e-01, -2.87638347e-01],
       [ 3.67578135e-01, -1.71750061e-01],
       [-1.18475597e-01,  2.20395980e+00],
       [-1.47942605e+00, -6.35303205e-01],
       [-1.28500455e+00, -1.06988428e+00],
       [-1.38221530e+00,  4.07691369e-01],
       [-1.09058306e+00,  7.55356227e-01],
       [-1.47942605e+00, -2.00722133e-01],
       [ 9.50842613e-01, -1.06988428e+00],
```

In [9]:

```
xtest
```

Out[9]:

```
array([[-7.98950822e-01,  4.94607583e-01],
       [-2.12648508e-02, -5.77359062e-01],
       [-3.12897090e-01,  1.46942725e-01],
       [-7.98950822e-01,  2.62831011e-01],
       [-3.12897090e-01, -5.77359062e-01],
       [-1.09058306e+00, -1.44652121e+00],
       [-7.01740076e-01, -1.59138156e+00],
       [-2.15686344e-01,  2.14601566e+00],
       [-1.96547978e+00, -5.58617754e-02],
       [ 8.53631867e-01, -7.80163563e-01],
       [-7.98950822e-01, -6.06331134e-01],
       [-9.93372315e-01, -4.32498705e-01],
       [-1.18475597e-01, -4.32498705e-01],
       [ 7.59458956e-02,  2.04886868e-01],
       [-1.77105829e+00,  4.65635512e-01],
       [-6.04529329e-01,  1.36376973e+00],
       [-1.18475597e-01,  2.04886868e-01],
       [-1.86826903e+00,  4.36663440e-01],
       [ 1.63131784e+00,  1.74040666e+00],
       [-3.12897090e-01, -1.38857706e+00],
       [-3.12897090e-01, -6.64275277e-01],
       [ 8.53631867e-01,  2.14601566e+00],
       [ 2.70367388e-01, -5.48386991e-01],
       [ 8.53631867e-01,  1.01610487e+00],
       [-1.47942605e+00, -1.21474464e+00],
       [ 1.04805336e+00,  2.05909944e+00],
       [-9.93372315e-01,  4.94607583e-01],
       [-8.96161568e-01,  2.91803083e-01],
       [-1.18475597e-01, -2.29694204e-01],
       [-6.04529329e-01,  4.65635512e-01],
       [-1.67384754e+00,  5.23579655e-01],
       [-1.18475597e-01,  2.62831011e-01],
       [ 1.82573933e+00, -2.87638347e-01],
       [-1.18475597e-01, -4.90442848e-01],
       [-1.38221530e+00, -3.45582490e-01],
       [-1.96547978e+00, -5.19414919e-01],
       [-1.57663679e+00,  3.20775154e-01],
       [-4.10107836e-01, -7.80163563e-01],
       [-7.01740076e-01, -1.04091221e+00],
       [ 1.04805336e+00, -9.82968063e-01],
       [-1.09058306e+00,  5.23579655e-01],
       [ 2.70367388e-01, -5.19414919e-01],
       [-1.09058306e+00,  4.07691369e-01],
       [-3.12897090e-01, -1.44652121e+00],
       [ 4.64788881e-01,  1.21890937e+00],
       [-1.09058306e+00, -3.45582490e-01],
       [-1.18475597e-01,  2.91803083e-01],
       [ 1.33968560e+00,  5.81523798e-01],
       [-1.18779381e+00, -1.15680049e+00],
       [ 1.04805336e+00,  4.65635512e-01],
       [ 1.82573933e+00,  1.50863009e+00],
       [-4.10107836e-01, -1.30166085e+00],
       [-3.12897090e-01, -3.74554562e-01],
       [-4.10107836e-01,  1.30582558e+00],
       [ 2.02016082e+00,  5.23579655e-01],
```

```
        [ 6.59210374e-01, -1.09885635e+00],
        [-8.96161568e-01,  3.78719297e-01],
        [-1.18779381e+00,  2.91803083e-01],
        [ 1.04805336e+00, -1.21474464e+00],
        [-1.47942605e+00, -1.44652121e+00],
        [-6.04529329e-01, -1.50446535e+00],
        [ 2.11737157e+00, -8.09135634e-01],
        [-1.86826903e+00,  1.75914797e-01],
        [-2.15686344e-01,  8.42272441e-01],
        [-1.86826903e+00, -1.27268878e+00],
        [ 2.11737157e+00,  3.78719297e-01],
        [-1.38221530e+00,  5.52551726e-01],
        [-1.09058306e+00, -3.45582490e-01],
        [ 1.73156642e-01, -6.64275277e-01],
        [ 3.67578135e-01,  2.08236764e-03],
        [-6.04529329e-01,  2.31984809e+00],
        [-3.12897090e-01,  2.04886868e-01],
        [-1.57663679e+00, -2.00722133e-01],
        [ 6.59210374e-01, -1.38857706e+00],
        [-1.09058306e+00,  5.52551726e-01],
        [-1.96547978e+00,  3.49747226e-01],
        [ 3.67578135e-01,  2.62831011e-01],
        [ 1.73156642e-01, -2.87638347e-01],
        [ 1.43689635e+00, -1.04091221e+00],
        [ 8.53631867e-01,  1.07404901e+00]])
```

In [10]:

```
from sklearn.linear_model import LogisticRegression
lr = LogisticRegression()
lr.fit(xtrain, ytrain)
```

Out[10]:

▾ LogisticRegression

LogisticRegression()

In [11]:

```
ytrain_pred = lr.predict(xtrain)
ytest_pred = lr.predict(xtest)
```

In [12]:

```
ytrain_pred
```

Out[12]:

```
array([1, 1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0,
0,
       0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1,
1,
       0, 0, 1, 1, 1, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0,
1,
       0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0,
0,
       0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1,
1,
       0, 1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 1, 0, 1, 0, 0,
0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1,
0,
       0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1,
1,
       0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1, 0, 1,
0,
       1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0,
0,
       1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0,
0,
       0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1,
0,
       0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1,
0,
       0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0,
1,
       1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0])
```

In [13]:

```
ytest_pred
```

Out[13]:

```
array([0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,
1,
       0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0,
0,
       1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0,
1,
       0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 1])
```

In [14]:

```
print(lr.predict(sc.transform([[19,19000]])))       # transform is used because of
```

```
[0]

/home/ubuntu/.local/lib/python3.8/site-packages/sklearn/base.py:420: U
serWarning: X does not have valid feature names, but StandardScaler wa
s fitted with feature names
  warnings.warn(
```

In [15]:

```
lr.predict([[-7.98950822e-01,4.94607583e-01]])
```

Out[15]:

```
array([0])
```

In [16]:

```
lr.predict([[-2.15686344e-01,2.14601566e+00]])
```

Out[16]:

```
array([1])
```

In [17]:

```
from sklearn.metrics import confusion_matrix,classification_report,accuracy_score
matrix = confusion_matrix(ytest,ytest_pred)
print(matrix)
```

```
[[57  1]
 [ 5 17]]
```

In [18]:

```
score=accuracy_score(ytest,ytest_pred)
score
```

Out[18]:

```
0.925
```

In [19]:

```
cr=classification_report(ytest,ytest_pred)
print(cr)
```

```
              precision    recall  f1-score   support

           0       0.92      0.98      0.95        58
           1       0.94      0.77      0.85        22

    accuracy                           0.93        80
   macro avg       0.93      0.88      0.90        80
weighted avg       0.93      0.93      0.92        80
```

In [20]:

```
pip install seaborn
```

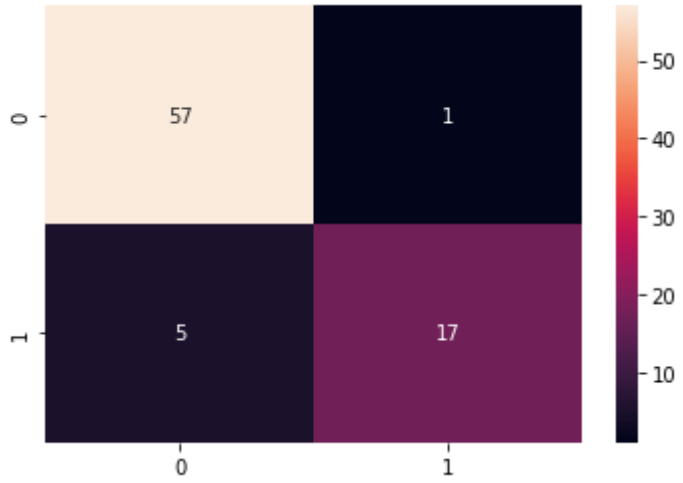
```
Defaulting to user installation because normal site-packages is not wr
iteable
Collecting seaborn
  Downloading seaborn-0.12.2-py3-none-any.whl (293 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 293.3/293.3 KB 1.3 MB/s e
ta 0:00:00[36m0:00:01[36m0:00:01:01
Requirement already satisfied: numpy!=1.24.0,>=1.17 in /home/ubuntu/.l
ocal/lib/python3.8/site-packages (from seaborn) (1.22.4)
Requirement already satisfied: matplotlib!=3.6.1,>=3.1 in /home/ubunt
u/.local/lib/python3.8/site-packages (from seaborn) (3.7.1)
Requirement already satisfied: pandas>=0.25 in /home/ubuntu/.local/li
b/python3.8/site-packages (from seaborn) (1.4.2)
Requirement already satisfied: kiwisolver>=1.0.1 in /home/ubuntu/.loca
l/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
 (1.4.4)
Requirement already satisfied: importlib-resources>=3.2.0 in /home/ubu
ntu/.local/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->
seaborn) (5.7.1)
Requirement already satisfied: cycler>=0.10 in /home/ubuntu/.local/li
b/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (0.1
1.0)
Requirement already satisfied: python-dateutil>=2.7 in /home/ubuntu/.l
ocal/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seabor
n) (2.8.2)
Requirement already satisfied: pyparsing>=2.3.1 in /home/ubuntu/.loca
l/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
 (3.0.8)
Requirement already satisfied: contourpy>=1.0.1 in /home/ubuntu/.loca
l/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
 (1.0.7)
Requirement already satisfied: packaging>=20.0 in /home/ubuntu/.local/
lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (2
1.3)
Requirement already satisfied: fonttools>=4.22.0 in /home/ubuntu/.loca
l/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
 (4.38.0)
Requirement already satisfied: pillow>=6.2.0 in /usr/lib/python3/dist-
packages (from matplotlib!=3.6.1,>=3.1->seaborn) (7.0.0)
Requirement already satisfied: pytz>=2020.1 in /home/ubuntu/.local/li
b/python3.8/site-packages (from pandas>=0.25->seaborn) (2022.1)
Requirement already satisfied: zipp>=3.1.0 in /home/ubuntu/.local/lib/
python3.8/site-packages (from importlib-resources>=3.2.0->matplotlib!=
3.6.1,>=3.1->seaborn) (3.8.0)
Requirement already satisfied: six>=1.5 in /usr/lib/python3/dist-packa
ges (from python-dateutil>=2.7->matplotlib!=3.6.1,>=3.1->seaborn) (1.1
4.0)
Installing collected packages: seaborn
Successfully installed seaborn-0.12.2
WARNING: You are using pip version 22.0.4; however, version 23.1 is av
ailable.
You should consider upgrading via the '/usr/bin/python3 -m pip install
--upgrade pip' command.
Note: you may need to restart the kernel to use updated packages.
```

In [22]:

```
import seaborn as sns
sns.heatmap(matrix,annot=True)
```

Out[22]:

<Axes: >



In [23]:

```
tn, fp, fn, tp = confusion_matrix(ytest,ytest_pred).ravel()
```

In [24]:

```
print(tn, fp, fn, tp)
```

```
57 1 5 17
```

In [25]:

```
print("Error rate:",(fp+fn)/(tn+fp+fn+tp))
```

```
Error rate: 0.075
```