Data Analytics II

1. Implement logistic regression using Python/R to perform classification on Social_Network_Ads.csv dataset.
2. Compute Confusion matrix to find TP, FP, TN, FN, Accuracy, Error rate, Precision, Recall on the given dataset

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

In [2]:
```
df=pd.read_csv("/home/mca01/Downloads/Social_Network_Ads.csv")
```

In [3]:
```
df.head()
```

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

In [4]:
```
df.describe()
```

Out[4]:

	User ID	Age	EstimatedSalary	Purchased
count	4.000000e+02	400.000000	400.000000	400.000000
mean	1.569154e+07	37.655000	69742.500000	0.357500
std	7.165832e+04	10.482877	34096.960282	0.479864
min	1.556669e+07	18.000000	15000.000000	0.000000
25%	1.562676e+07	29.750000	43000.000000	0.000000
50%	1.569434e+07	37.000000	70000.000000	0.000000
75%	1.575036e+07	46.000000	88000.000000	1.000000
max	1.581524e+07	60.000000	150000.000000	1.000000

In [5]:
```
df.isnull().sum()
```

Out[5]:
```
User ID            0
Gender             0
Age                0
EstimatedSalary    0
Purchased          0
dtype: int64
```

In [6]:
```
df.shape
```

Out[6]:
```
(400, 5)
```

In [7]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 400 entries, 0 to 399
Data columns (total 5 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   User ID          400 non-null    int64
 1   Gender           400 non-null    object
 2   Age              400 non-null    int64
 3   EstimatedSalary  400 non-null    int64
 4   Purchased        400 non-null    int64
dtypes: int64(4), object(1)
memory usage: 15.8+ KB
```

In [8]:
```
x = df.iloc[:,2:4]
y = df.iloc[:,4]
```

In [9]:
```
print(x)
```

```
     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 x 2 columns]
```

In [10]:
```
from sklearn.model_selection import train_test_split
```

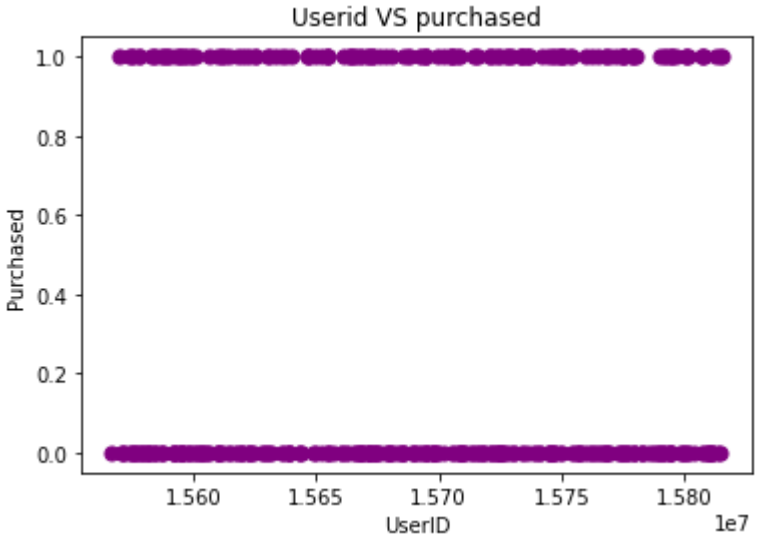
In [11]:
```
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.25, ra
```

In [12]:
```
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix,ConfusionMatrixDisplay,classifi
```

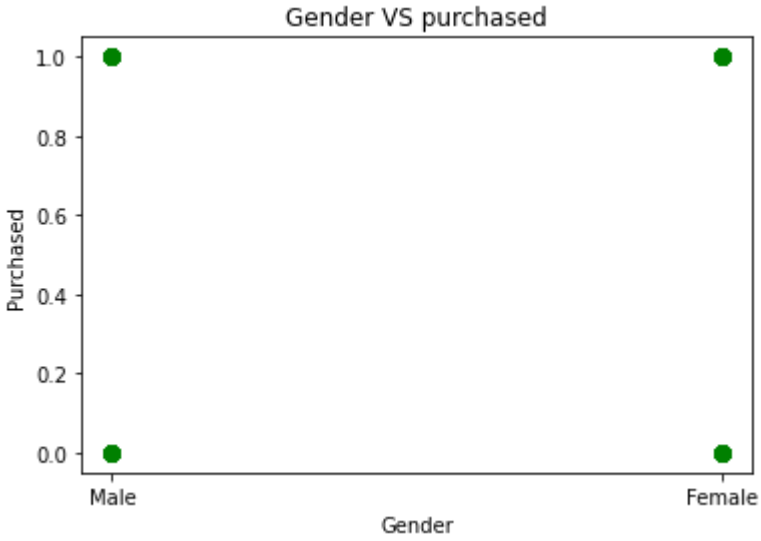
In [13]:
```
scale = StandardScaler()
x_train = scale.fit_transform(x_train)
x_test = scale.transform(x_test)
```

In [14]:
```
lr = LogisticRegression(random_state=0, solver='lbfgs')
lr.fit(x_train, y_train)
pred = lr.predict(x_test)
```

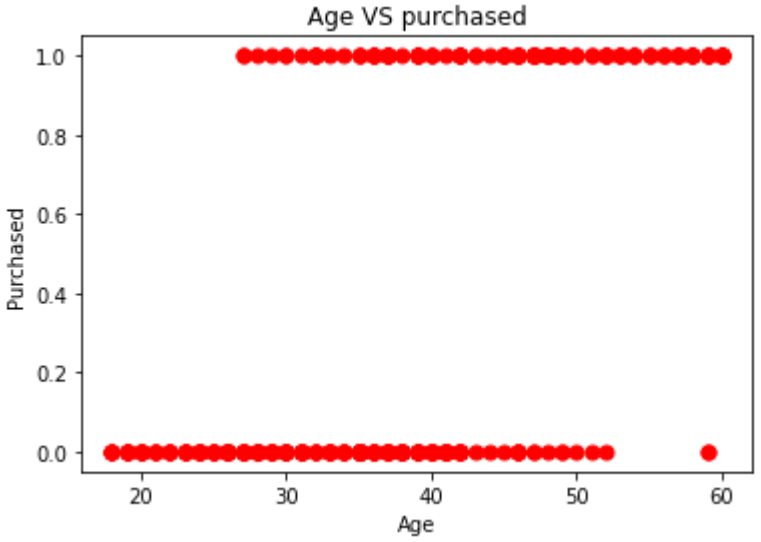
In [15]:
```
x1=df.iloc[:, 0].values
y1=df.iloc[:, 4].values
mtp.scatter(x1,y1,color='purple',s=50)
mtp.xlabel('UserID')
mtp.ylabel('Purchased')
mtp.title('Userid VS purchased')
mtp.show()
```



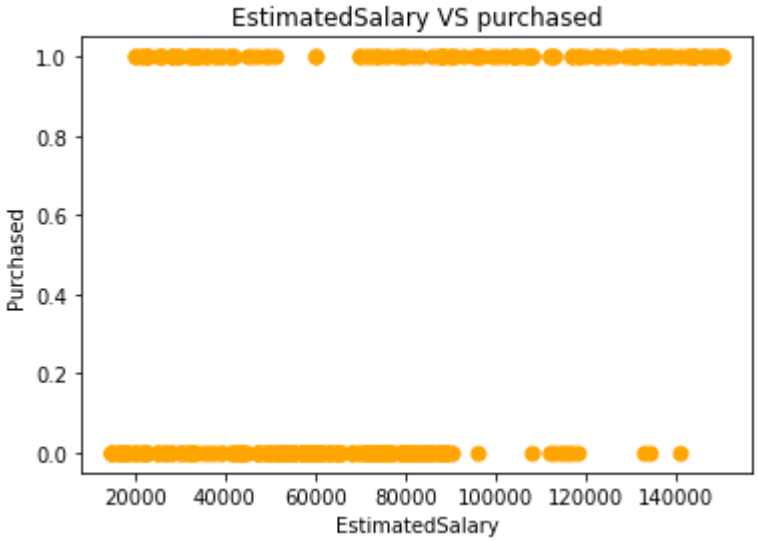
In [16]:
```
x2=df.iloc[:, 1].values
y2=df.iloc[:, 4].values
mtp.scatter(x2,y2,color='green',s=50)
mtp.xlabel('Gender')
mtp.ylabel('Purchased')
mtp.title('Gender VS purchased')
mtp.show()
```



In [17]:
```
x3=df.iloc[:, 2].values
y3=df.iloc[:, 4].values
mtp.scatter(x3,y3,color='red',s=50)
mtp.xlabel('Age')
mtp.ylabel('Purchased')
mtp.title('Age VS purchased')
mtp.show()
```



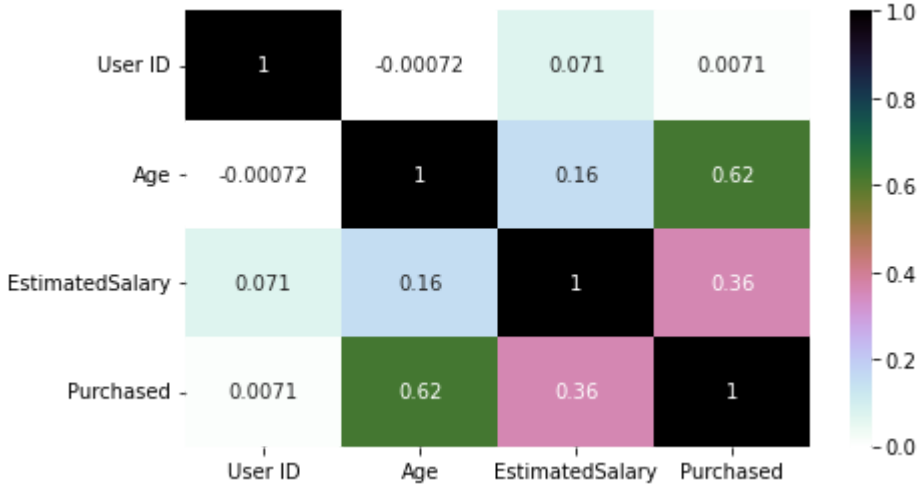
In [18]:
```
x4=df.iloc[:, 3].values
y4=df.iloc[:, 4].values
mtp.scatter(x4,y4,color='orange',s=50)
mtp.xlabel('EstimatedSalary')
mtp.ylabel('Purchased')
mtp.title('EstimatedSalary VS purchased')
mtp.show()
```



In [20]:
```
!pip install seaborn
```

```
Defaulting to user installation because normal site-packages is not writeabl
e
Collecting seaborn
  Downloading seaborn-0.13.2-py3-none-any.whl (294 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 294.9/294.9 KB 3.4 MB/s eta 0:0
0:00[31m3.3 MB/s eta 0:00:01
Requirement already satisfied: numpy!=1.24.0,>=1.20 in /usr/lib/python3/dist
-packages (from seaborn) (1.21.5)
Requirement already satisfied: pandas>=1.2 in /usr/lib/python3/dist-packages
(from seaborn) (1.3.5)
Requirement already satisfied: matplotlib!=3.6.1,>=3.4 in /usr/lib/python3/d
ist-packages (from seaborn) (3.5.1)
Installing collected packages: seaborn
Successfully installed seaborn-0.13.2
```

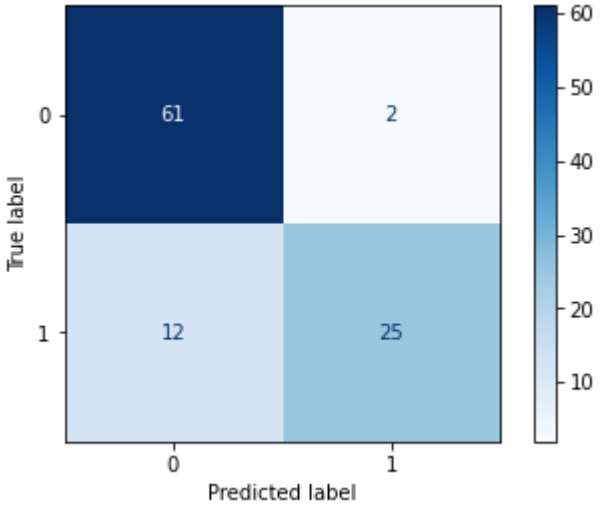
In [21]:
```
import seaborn as sns
mtp.figure(figsize=(7,4))
sns.heatmap(df.corr(),annot=True,cmap='cubehelix_r')
mtp.show()
```



In [22]:
```
matrix = confusion_matrix(y_test, pred, labels= lr.classes_)

conf_matrix = ConfusionMatrixDisplay(confusion_matrix=matrix,display_labels=

conf_matrix.plot(cmap=mtp.cm.Blues)
mtp.show()
```



In [ ]:
```
```