In [3]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()
```

In [4]:
```
df=pd.read_csv('Social_Network_Ads.csv')
```

In [5]:
```
df
```

Out[5]:

	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 [6]:
```
df.isnull().sum()
```

Out[6]:
```
User ID            0
Gender             0
Age                0
EstimatedSalary    0
Purchased          0
dtype: int64
```

In [14]:
```
df1=df.drop(['User ID'],axis=1)
```

In [15]:
```
df1.head(10)
```

Out[15]:

	Gender	Age	EstimatedSalary	Purchased
0	Male	19	19000	0
1	Male	35	20000	0
2	Female	26	43000	0
3	Female	27	57000	0
4	Male	19	76000	0
5	Male	27	58000	0
6	Female	27	84000	0
7	Female	32	150000	1
8	Male	25	33000	0
9	Female	35	65000	0

In [17]:
```
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
```

In [18]:
```
enc=LabelEncoder()
```

In [21]:
```
df1['Gender']=enc.fit_transform(df1['Gender'])
```

In [23]:
```
df1.head()
```

Out[23]:

	Gender	Age	EstimatedSalary	Purchased
0	1	19	19000	0
1	1	35	20000	0
2	0	26	43000	0
3	0	27	57000	0
4	1	19	76000	0

In [25]:
```
x=df1.drop(labels=['Purchased'],axis=1)
```

In [26]:
```
y=df1['Purchased']
```

In [27]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.25,random_state=10)
```

In [28]:
```
x_train.shape, x_test.shape
```

Out[28]:
```
((300, 3), (100, 3))
```

In [29]:
```
logr = LogisticRegression()
logr.fit(x_train,y_train)
```

Out[29]:
```
LogisticRegression()
```

In [30]:
```
logr.score(x_test,y_test)
```

Out[30]:
```
0.69
```

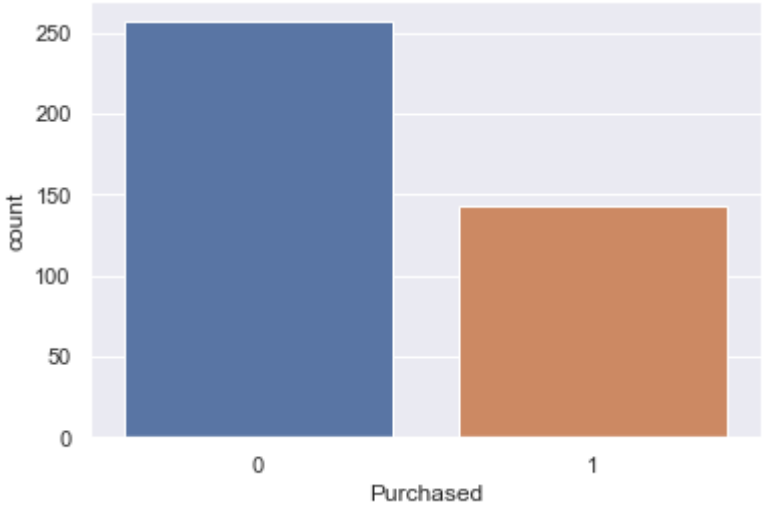
In [31]:
```
y_pred = logr.predict(x_test)
```

In [32]:
```
y_pred
```

Out[32]:
```
array([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, 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, 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], dtype=int64)
```

In [33]:
```
sns.countplot(x='Purchased',data=df1)
```

Out[33]:
```
<AxesSubplot:xlabel='Purchased', ylabel='count'>
```



In [34]:
```
from sklearn.metrics import confusion_matrix
```

In [35]:
```
confusion_matrix(y_test,y_pred)
```

Out[35]:
```
array([[69,  0],
       [31,  0]], dtype=int64)
```

In [36]:
```
from sklearn.metrics import classification_report as cr
print(cr(y_test,y_pred))
```

```
              precision    recall  f1-score   support

           0       0.69      1.00      0.82        69
           1       0.00      0.00      0.00        31

    accuracy                           0.69       100
   macro avg       0.34      0.50      0.41       100
weighted avg       0.48      0.69      0.56       100
```

```
C:\Users\Joshua Deshmukh\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.9_qbz5n2kfra8p0\LocalCache\local-packages\Pytho
n39\site-packages\sklearn\metrics\_classification.py:1318: UndefinedMetricWarning: Precision and F-score are ill-defined and bein
g set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
C:\Users\Joshua Deshmukh\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.9_qbz5n2kfra8p0\LocalCache\local-packages\Pytho
n39\site-packages\sklearn\metrics\_classification.py:1318: UndefinedMetricWarning: Precision and F-score are ill-defined and bein
g set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
C:\Users\Joshua Deshmukh\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.9_qbz5n2kfra8p0\LocalCache\local-packages\Pytho
n39\site-packages\sklearn\metrics\_classification.py:1318: UndefinedMetricWarning: Precision and F-score are ill-defined and bein
g set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

In [38]:
```
from statsmodels.stats.outliers_influence import variance_inflation_factor
variables = df[['Gender','EstimatedSalary','Age']]
vif = pd.DataFrame()
vif["VIF"] = [variance_inflation_factor(variables.values, i) for i in range(variables.shape[1])]
vif["features"] = variables.columns
vif
```

Out[38]:

	VIF	features
0	1.783132	Gender
1	4.601775	EstimatedSalary
2	5.122407	Age

In [ ]: