In [85]:
```
import pandas as pd

from matplotlib import pyplot as plt

data=pd.read_csv(r"C:\Users\LENOVO\Desktop\ML\candy-data.csv")

data.shape
```

Out[85]: (85, 13)

In [61]:
```
print(data.isnull().sum())
```

```
competitorname      0
chocolate           0
fruity              0
caramel             0
peanutyalmondy      0
nougat              0
crispedricewafer    0
hard                0
bar                 0
pluribus            0
sugarpercent        0
pricepercent        0
winpercent          0
dtype: int64
```

In [62]:
```
data.describe()
```

Out[62]:

	chocolate	fruity	caramel	peanutyalmondy	nougat	crispedricewafer	
count	85.000000	85.000000	85.000000	85.000000	85.000000	85.000000	85.0
mean	0.435294	0.447059	0.164706	0.164706	0.082353	0.082353	0.1
std	0.498738	0.500140	0.373116	0.373116	0.276533	0.276533	0.3
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
25%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
50%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
75%	1.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.0
max	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.0

In [63]:
```
from sklearn.model_selection import train_test_split
```

In [65]:
```
X=data[["sugarpercent","pricepercent","winpercent"]].values
y=data["chocolate"].values
```

In [66]:
```
X_test, X_train, y_test, y_train=train_test_split(X,y,test_size=0.25, random_state=
```

In [67]:
```
from sklearn.linear_model import LogisticRegression
```

In [68]:
```
model=LogisticRegression()
```

In [69]:
```
model.fit(X_train,y_train)
```

Out[69]:
```
▾ LogisticRegression

LogisticRegression()
```

In [70]:
```
model.predict(X_test)
```

Out[70]:
```
array([1, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0,
       0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 1,
       1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 1, 0],
      dtype=int64)
```

In [71]:
```
#feature scalling
from sklearn.preprocessing import StandardScaler
sc= StandardScaler()
sc.fit(X_train)
X_train=sc.transform(X_train)
X_test=sc.transform(X_test)
```

In [73]:
```
model.fit(X_train,y_train)
```

Out[73]:
```
▾ LogisticRegression

LogisticRegression()
```

In [86]:
```
y_predicted=model.predict(X_test)
```

In [87]:
```
print(model.score(X_test,y_test))
```

```
0.8253968253968254
```

In [88]:
```
from sklearn.metrics import confusion_matrix
cm= confusion_matrix(y_test,y_predicted)
cm
```

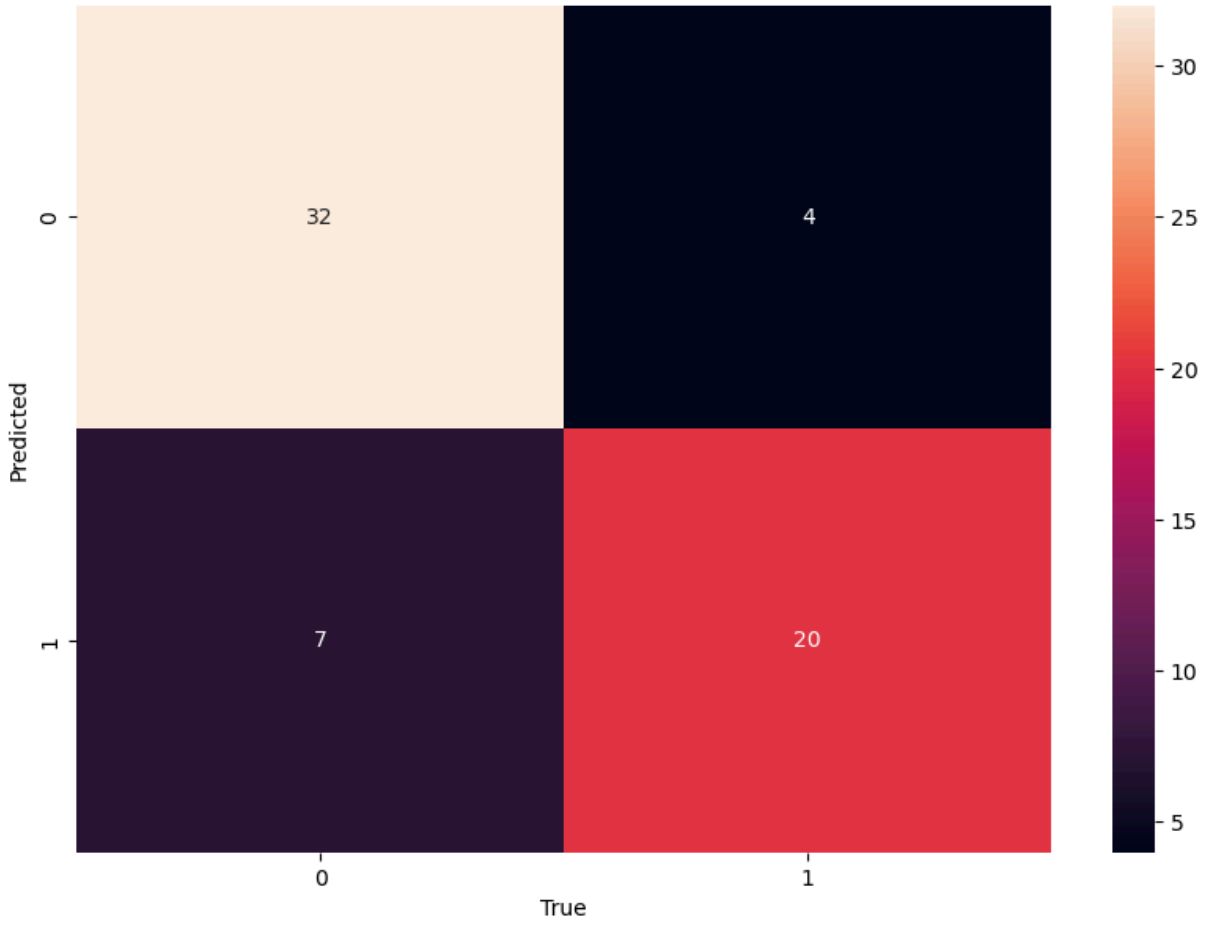
Out[88]:
```
array([[32,  4],
       [ 7, 20]], dtype=int64)
```

In [89]:
```
print(((54/63)*100))
```

```
85.71428571428571
```

In [90]:
```
import seaborn as sn
plt.figure(figsize=(10,7))
sn.heatmap(cm, annot=True)
plt.ylabel('Predicted')
plt.xlabel('True')
```

Out[90]: Text(0.5, 47.7222222222222, 'True')



In [ ]: