**import** pandas **as** pd

**import** numpy **as** np

**import** seaborn **as** sns

**import** matplotlib.pyplot **as** plt

**from** sklearn.neighbors **import** KNeighborsClassifier

**from** sklearn.metrics **import** accuracy\_score

In [74]:

df **=** pd**.**read\_csv('heart.csv')

In [75]:

df**.**head()

Out[75]:

|  | **age** | **sex** | **cp** | **trestbps** | **chol** | **fbs** | **restecg** | **thalach** | **exang** | **oldpeak** | **slope** | **ca** | **thal** | **target** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 63 | 1 | 3 | 145 | 233 | 1 | 0 | 150 | 0 | 2.3 | 0 | 0 | 1 | 1 |
| **1** | 37 | 1 | 2 | 130 | 250 | 0 | 1 | 187 | 0 | 3.5 | 0 | 0 | 2 | 1 |
| **2** | 41 | 0 | 1 | 130 | 204 | 0 | 0 | 172 | 0 | 1.4 | 2 | 0 | 2 | 1 |
| **3** | 56 | 1 | 1 | 120 | 236 | 0 | 1 | 178 | 0 | 0.8 | 2 | 0 | 2 | 1 |
| **4** | 57 | 0 | 0 | 120 | 354 | 0 | 1 | 163 | 1 | 0.6 | 2 | 0 | 2 | 1 |

In [76]:

sns**.**countplot(df['target'])

Out[76]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x1ed23599608>



In [77]:

x**=** df**.**iloc[:,0:13]**.**values

y**=** df['target']**.**values

In [78]:

**from** sklearn.model\_selection **import** train\_test\_split

x\_train, x\_test, y\_train, y\_test**=** train\_test\_split(x, y, test\_size**=** 0.25, random\_state**=**0)

In [79]:

**from** sklearn.preprocessing **import** StandardScaler

st\_x**=** StandardScaler()

x\_train**=** st\_x**.**fit\_transform(x\_train)

x\_test**=** st\_x**.**transform(x\_test)

In [80]:

error **=** []

*# Calculating error for K values between 1 and 30*

**for** i **in** range(1, 30):

knn **=** KNeighborsClassifier(n\_neighbors**=**i)

knn**.**fit(x\_train, y\_train)

pred\_i **=** knn**.**predict(x\_test)

error**.**append(np**.**mean(pred\_i **!=** y\_test))

plt**.**figure(figsize**=**(12, 6))

plt**.**plot(range(1, 30), error, color**=**'red', linestyle**=**'dashed', marker**=**'o',

markerfacecolor**=**'blue', markersize**=**10)

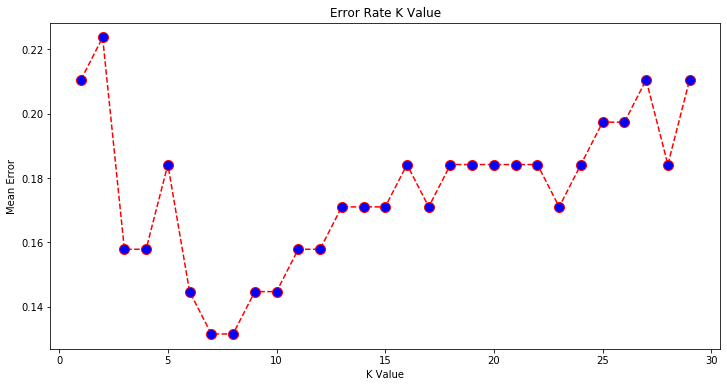
plt**.**title('Error Rate K Value')

plt**.**xlabel('K Value')

plt**.**ylabel('Mean Error')

print("Minimum error:-",min(error),"at K =",error**.**index(min(error))**+**1)

Minimum error:- 0.13157894736842105 at K = 7



In [86]:

classifier**=** KNeighborsClassifier(n\_neighbors**=**7)

classifier**.**fit(x\_train, y\_train)

Out[86]:

KNeighborsClassifier(n\_neighbors=7)

In [87]:

y\_pred**=** classifier**.**predict(x\_test)

In [88]:

**from** sklearn.metrics **import** confusion\_matrix

cm**=** confusion\_matrix(y\_test, y\_pred)

In [89]:

cm

Out[89]:

array([[26, 7],

[ 3, 40]], dtype=int64)

In [90]:

accuracy\_score(y\_test, y\_pred)

Out[90]:

0.868421052631579