In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

In [32]:

```
kabaddi_data = pd.read_csv("prokabaddi.csv")
```

In [3]:

```
kabaddi_data.head()
```

Out[3]:

	match_no	stage	team1	team2	team1_score	team2_score	date	location	se
0	1	League	U Mumba	Jaipur Pink Panthers	45	31	26-07-2014	Dome@NSCI Svp Stadium, Mumbai, Mumbai	
1	2	League	Dabang Delhi KC	Bengaluru Bulls	28	47	27-07-2014	Dome@NSCI Svp Stadium, Mumbai, Mumbai	
2	3	League	Bengaluru Bulls	Puneri Paltan	40	37	28-07-2014	Dome@NSCI Svp Stadium, Mumbai, Mumbai	
3	4	League	U Mumba	Bengal Warriors	36	25	29-07-2014	Dome@NSCI Svp Stadium, Mumbai, Mumbai	
4	5	League	Puneri Paltan	Dabang Delhi KC	31	35	30-07-2014	Dome@NSCI Svp Stadium, Mumbai, Mumbai	

In [4]:

```
kabaddi_data.tail()
```

Out[4]:

	match_no	stage	team1	team2	team1_score	team2_score	date	location	se
785	33	Eliminator 1	UP Yoddha	Puneri Paltan	42	31	21-02-2022	Sheraton Grand Convention Center, Bangalore	
786	34	Eliminator 2	Gujarat Giants	Bengaluru Bulls	29	49	21-02-2022	Sheraton Grand Convention Center, Bangalore	
787	35	Semi Final	Patna Pirates	UP Yoddha	38	27	23-02-2022	Sheraton Grand Convention Center, Bangalore	
788	36	Semi Final	Dabang Delhi KC	Bengaluru Bulls	40	35	23-02-2022	Sheraton Grand Convention Center, Bangalore	
789	37	Final	Patna Pirates	Dabang Delhi KC	36	37	25-02-2022	Sheraton Grand Convention Center, Bangalore	

In [5]:

```
kabaddi_data.shape
```

Out[5]:

```
(790, 9)
```

In [7]:

```
kabaddi_data.columns
```

Out[7]:

```
Index(['match_no', 'stage', 'team1', 'team2', 'team1_score', 'team2_scor
e',
       'date', 'location', 'season'],
      dtype='object')
```

In [8]:

```
kabaddi_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 790 entries, 0 to 789
Data columns (total 9 columns):
 #   Column       Non-Null Count  Dtype 
---  ------       --------------  ----- 
 0   match_no     790 non-null    int64 
 1   stage        790 non-null    object
 2   team1        790 non-null    object
 3   team2        790 non-null    object
 4   team1_score  790 non-null    int64 
 5   team2_score  790 non-null    int64 
 6   date         790 non-null    object
 7   location     790 non-null    object
 8   season       790 non-null    int64 
dtypes: int64(4), object(5)
memory usage: 55.7+ KB
```

In [9]:

```
kabaddi_data.describe()
```

Out[9]:

	match_no	team1_score	team2_score	season
count	790.000000	790.000000	790.000000	790.000000
mean	37.984810	32.525316	32.713924	5.282278
std	26.209157	7.153309	7.810602	2.152622
min	0.000000	15.000000	15.000000	1.000000
25%	17.000000	27.250000	27.000000	4.000000
50%	33.000000	32.000000	32.000000	6.000000
75%	55.000000	37.000000	37.000000	7.000000
max	99.000000	69.000000	69.000000	8.000000

In [10]:

```
kabaddi_data.isnull().sum()
```

Out[10]:

```
match_no       0
stage          0
team1          0
team2          0
team1_score    0
team2_score    0
date           0
location       0
season         0
dtype: int64
```

In [11]:

```
kabaddi_data.nunique()
```

Out[11]:

```
match_no       100
stage           17
team1           13
team2           13
team1_score     41
team2_score     46
date           467
location        30
season           8
dtype: int64
```

In [12]:

```
kabaddi_data['team1'].unique()
```

Out[12]:

```
array(['U Mumba', 'Dabang Delhi KC', 'Bengaluru Bulls', 'Puneri Paltan',
       'Bengal Warriors', 'Jaipur Pink Panthers', 'Telugu Titans',
       'Patna Pirates', 'Gujarat Fortunegiants', 'Haryana Steelers',
       'UP Yoddha', 'Tamil Thalaivas', 'Gujarat Giants'], dtype=object)
```

In [33]:

```
U_Mumba_Matches_Team1 = kabaddi_data[kabaddi_data['team1'] == 'U Mumba']
```

In [14]:

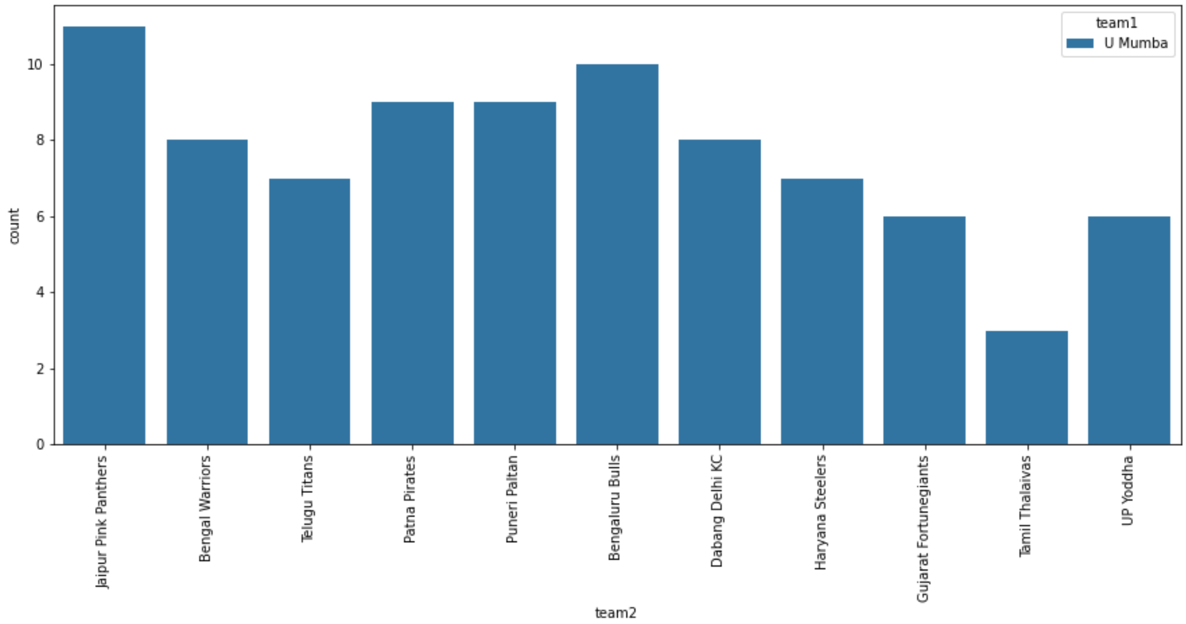
```
U_Mumba_Matches_Team1.head()
```

Out[14]:

	match_no	stage	team1	team2	team1_score	team2_score	date	location	seas
0	1	League	U Mumba	Jaipur Pink Panthers	45	31	26-07-2014	Dome@NSCI Svp Stadium, Mumbai, Mumbai	
3	4	League	U Mumba	Bengal Warriors	36	25	29-07-2014	Dome@NSCI Svp Stadium, Mumbai, Mumbai	
5	6	League	U Mumba	Telugu Titans	35	35	31-07-2014	Dome@NSCI Svp Stadium, Mumbai, Mumbai	
6	7	League	U Mumba	Patna Pirates	36	33	01-08-2014	Dome@NSCI Svp Stadium, Mumbai, Mumbai	
16	17	League	U Mumba	Puneri Paltan	44	28	11-08-2014	Thyagaraj Sports Complex, Delhi, Delhi	

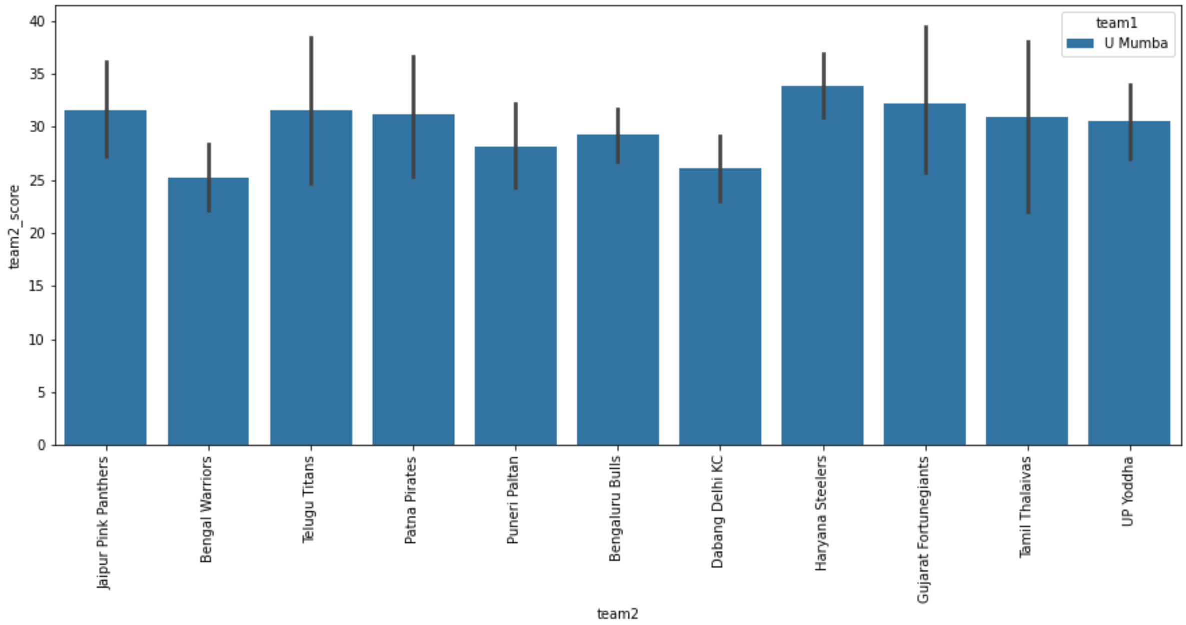
In [15]:

```
plt.figure(figsize=(15,6))
sns.countplot('team2',hue='team1', data = U_Mumba_Matches_Team1)
plt.xticks(rotation = 90)
plt.show()
```



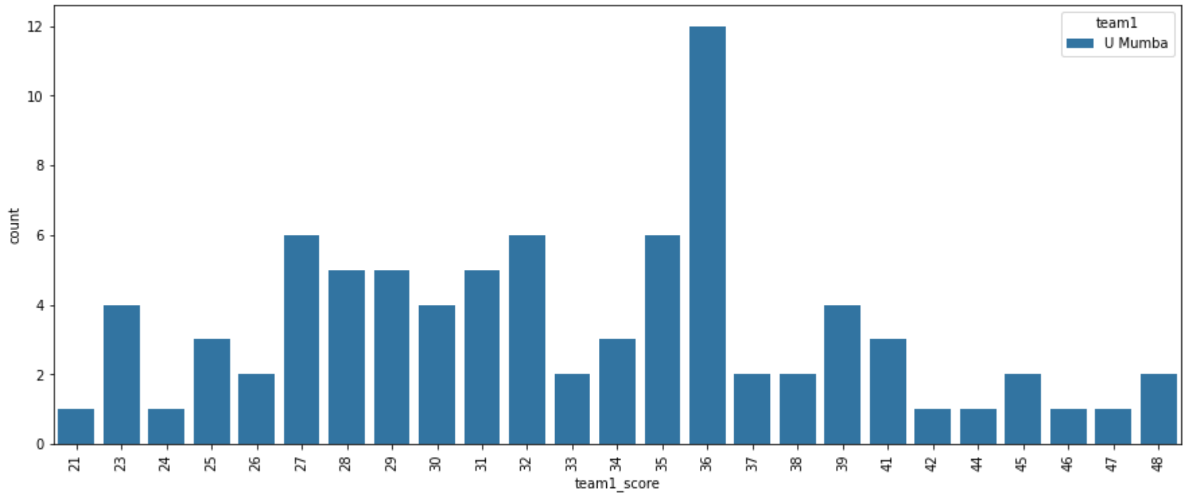
In [16]:

```
plt.figure(figsize=(15,6))
sns.barplot(x = 'team2', y ='team2_score',hue='team1', data = U_Mumba_Matches_Team1)
plt.xticks(rotation = 90)
plt.show()
```



In [17]:

```
plt.figure(figsize=(15,6))
sns.countplot('team1_score',hue='team1', data = U_Mumba_Matches_Team1)
plt.xticks(rotation = 90)
plt.show()
```



In [34]:

```
conditions = [U_Mumba_Matches_Team1['team1_score'] > U_Mumba_Matches_Team1['team2_score
              U_Mumba_Matches_Team1['team1_score'] < U_Mumba_Matches_Team1['team2_score
```

In [36]:

```
choices = ['Winner', 'Loose']
```

In [37]:

```
U_Mumba_Matches_Team1['result'] = np.select(conditions, choices, default='Tie')
```

In [21]:

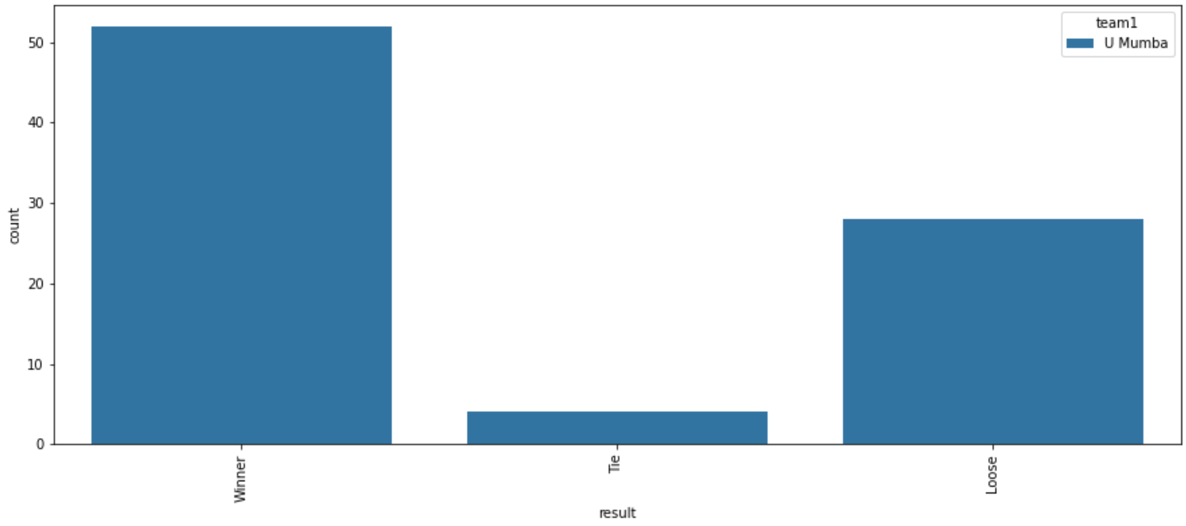
```
U_Mumba_Matches_Team1.head()
```

Out[21]:

	match_no	stage	team1	team2	team1_score	team2_score	date	location	seas
0	1	League	U Mumba	Jaipur Pink Panthers	45	31	26-07-2014	Dome@NSCI Svp Stadium, Mumbai, Mumbai	
3	4	League	U Mumba	Bengal Warriors	36	25	29-07-2014	Dome@NSCI Svp Stadium, Mumbai, Mumbai	
5	6	League	U Mumba	Telugu Titans	35	35	31-07-2014	Dome@NSCI Svp Stadium, Mumbai, Mumbai	
6	7	League	U Mumba	Patna Pirates	36	33	01-08-2014	Dome@NSCI Svp Stadium, Mumbai, Mumbai	
16	17	League	U Mumba	Puneri Paltan	44	28	11-08-2014	Thyagaraj Sports Complex, Delhi, Delhi	

In [22]:

```
plt.figure(figsize=(15,6))
sns.countplot(x = 'result',hue='team1', data = U_Mumba_Matches_Team1)
plt.xticks(rotation = 90)
plt.show()
```



In [38]:

```
from sklearn import preprocessing
label_encoder = preprocessing.LabelEncoder()
```

In [39]:

```
U_Mumba_Matches_Team1['result']= label_encoder.fit_transform(U_Mumba_Matches_Team1['resu
```

In [40]:

```
U_Mumba_Matches_Team1['team1']= label_encoder.fit_transform(U_Mumba_Matches_Team1['team1
```

In [41]:

```
U_Mumba_Matches_Team1['team2']= label_encoder.fit_transform(U_Mumba_Matches_Team1['team2
```

In [44]:

```
 x= U_Mumba_Matches_Team1.drop(['match_no', 'stage', 'team1_score', 'team2_score',
                                'date', 'location', 'season', 'result'], axis =1)
```

In [45]:

```
y = U_Mumba_Matches_Team1.result
```

In [46]:

```
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
```

In [47]:

```
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size = 0.2)
```

In [48]:

```
model = LogisticRegression()
model.fit(X_train, y_train)
```

Out[48]:

```
LogisticRegression()
```

In [49]:

```
y_pred = model.predict(X_test)
```

In [50]:

```
print("Training Accuracy :", model.score(X_train, y_train))
print("Testing Accuracy :", model.score(X_test, y_test))
```

```
Training Accuracy : 0.6268656716417911
Testing Accuracy : 0.5882352941176471
```