NAME: Atharav Sunil Charthad

ROLL NO.: ET2-07

PRN: 202401070229

Cricket World Cup Dataset

Find the total number of matches played.

```
[1] import pandas as pd
matches_df=pd.read_csv("/content/matches[1].csv")

[2] total_matches = len(matches_df)
total_matches

348
```

Identify all unique teams that participated.

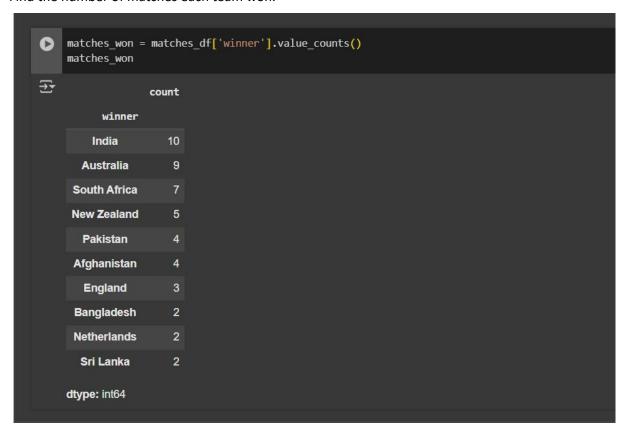
Find the number of unique cities where matches were hosted.

```
unique_cities = matches_df['city'].nunique()
unique_cities

10
```

List all the venues where matches were played.

Find the number of matches each team won.



Determine the most common toss decision (bat or field).

```
most_common_toss_decision = matches_df['toss_decision'].mode()[0]
most_common_toss_decision

field'
```

Identify the player with the most 'Player of the Match' awards.

```
top_player = matches_df['player_of_match'].value_counts().idxmax()
top_player

'Mohammed Shami'
```

Find the match with the highest winning margin by runs.



Find the match with the highest winning margin by wickets.



Count how many matches were decided by runs and how many by wickets.

```
runs_decided = matches_df['winner_runs'].notna().sum()
wickets_decided = matches_df['winner_wickets'].notna().sum()
runs_decided, wickets_decided

(np.int64(24), np.int64(24))
```

Find the city that hosted the most matches.

```
top_city = matches_df['city'].value_counts().idxmax()
top_city

'Ahmedabad'
```

Find how many times the toss winner also won the match.

```
toss_and_match_winner = (matches_df['toss_winner'] == matches_df['winner']).sum()

toss_and_match_winner

np.int64(19)
```

Identify the umpire who officiated the most matches.

```
umpires = pd.concat([matches_df['umpire1'], matches_df['umpire2']])
top_umpire = umpires.value_counts().idxmax()
top_umpire

'RK Illingworth'
```

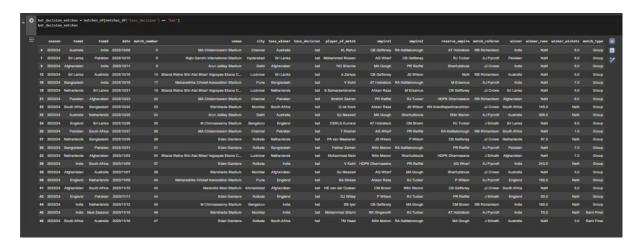
Calculate the average winning margin by runs.

```
avg_win_runs = matches_df['winner_runs'].mean()
avg_win_runs

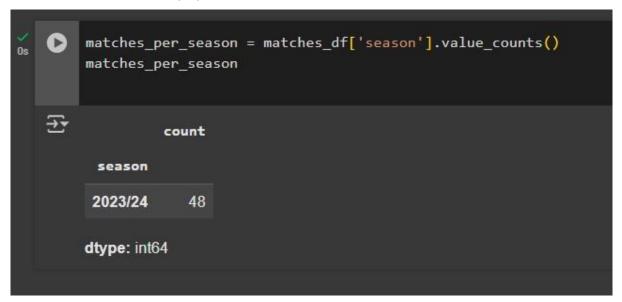
pp.float64(125.9166666666667)
```

Calculate the average winning margin by wickets.

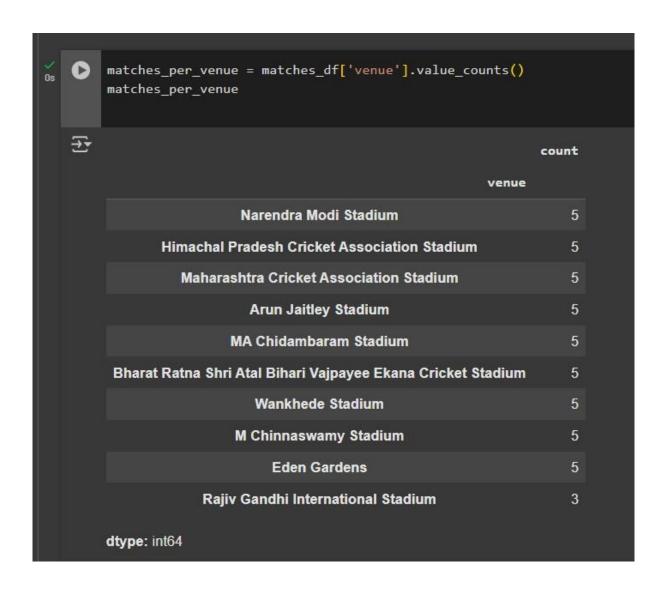
List all matches where the toss winner chose to bat.



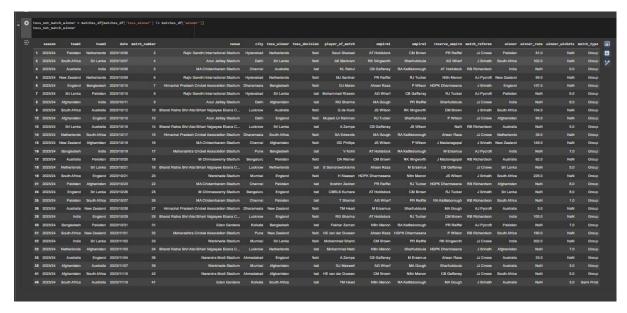
Find the number of matches played in each season.



Find how many matches were played at each venue.



Find matches where toss winner and match winner were different.



Find how many matches were "Group" matches.

```
group_matches = (matches_df['match_type'] == 'Group').sum()
group_matches

np.int64(45)
```

Find all cities that hosted more than 3 matches.

Find how many matches were played in Mumbai city.

```
matches_in_mumbai = (matches_df['city'] == 'Mumbai').sum()
matches_in_mumbai

pr.int64(5)
```

List all unique team names (no duplicates)