EDS Assignment

NAME : JEET KSHIRSAGAR

ROLL NO. : CS7-05

PRN: 202401080041

DIV: CS7

1. Identify the team with the most former names

```python
import pandas as pd

former_names = pd.read_csv('/root/.cache/kagglehub/datasets/martj42/international-football-results-from-1872-to-2017/versions/93/former_names.csv')

team_name_counts = former_names.groupby('current').size()
most_former_names_team = team_name_counts.idxmax()
most_former_names_count = team_name_counts.max()

print(f"Team with the most former names: {most_former_names_team} ({most_former_names_count} former names)")
```

```
Team with the most former names: DR Congo (4 former names)
```

2. Year-wise team name changes

```python
import pandas as pd
former_names = pd.read_csv('/root/.cache/kagglehub/datasets/martj42/international-football-results-from-1872-to-2017/versions/93/former_names.csv')

former_names['start_date'] = pd.to_datetime(former_names['start_date'])
former_names['year'] = former_names['start_date'].dt.year

changes_per_year = former_names.groupby('year').size()
print(changes_per_year)
```

```
year
1882    1
1903    1
1905    1
1921    1
1924    2
1934    2
1937    1
1939    1
1945    1
1946    1
1947    2
1948    2
1950    1
1951    1
1952    2
1953    1
1957    2
1958    1
1959    1
1960    1
1963    1
1965    1
1968    1
1971    1
1979    1
1992    1
1993    2
dtype: int64
```

3. Top goal scorer

```python
import pandas as pd
#file_path = '/root/.cache/kagglehub/datasets/martj42/international-football-results-from-1872-to-2017/versions/93/former_names.csv'
# Assuming goalscorers.csv is in the same directory as former_names.csv:
file_path_goalscorers = '/root/.cache/kagglehub/datasets/martj42/international-football-results-from-1872-to-2017/versions/93/goalscorers.csv'
# Replace with the actual path if it's different

goalscorers = pd.read_csv(file_path_goalscorers) # Changed 'path_to/goalscorers.csv' to file_path_goalscorers


top_scorer = goalscorers['scorer'].value_counts().idxmax()
print(f"Top Scorer: {top_scorer}")
```

```
Top Scorer: Cristiano Ronaldo
```

4.Average goals per player

```python
import pandas as pd
avg_goals_per_player = goalscorers.groupby('scorer').size().mean()
print(f"Average goals per player: {avg_goals_per_player:.2f}")
```

```
Average goals per player: 3.09
```

5. Players with own goals

```python
import pandas as pd

own_goals = goalscorers[goalscorers['own_goal'] == True]
own_goal_counts = own_goals['scorer'].value_counts()
print(own_goal_counts)
```

```
scorer
Raio Piiroja          3
Gustavo Gómez         3
Ján Ďurica            3
Ragnar Klavan         3
Walid Abbas           3
                     ..
Mihails Zemļinskis    1
Juliano Belletti      1
Heriberto Morales     1
Cris                  1
Igor Mitreski         1
Name: count, Length: 771, dtype: int64
```

6. Team with most different players scoring

```python
import pandas as pd
team_scorer_counts = goalscorers.groupby('team')['scorer'].nunique().sort_values(ascending=False)
print(team_scorer_counts.head(1))
```

```
team
Brazil      272
Name: scorer, dtype: int64
```

7. Team with the most matches

```python
import pandas as pd
# Correct the file path:
results = pd.read_csv('/root/.cache/kagglehub/datasets/martj42/international-football-results-from-1872-to-2017/versions/93/results.csv')
# Assuming 'results.csv' is in the same directory as other dataset files

#Rest of the code remains the same
home_matches = results['home_team'].value_counts()
away_matches = results['away_team'].value_counts()
total_matches = home_matches.add(away_matches, fill_value=0)

most_matches_team = total_matches.idxmax()
print(f"Team with most matches: {most_matches_team}")
```

```
Team with most matches: Sweden
```

8. Match with biggest goal difference

```python
import pandas as pd
results['goal_diff'] = abs(results['home_score'] - results['away_score'])
biggest_victory = results.loc[results['goal_diff'].idxmax()]
print(biggest_victory)
```

```
date                            2001-04-11
home_team                        Australia
away_team                   American Samoa
home_score                              31
away_score                               0
tournament     FIFA World Cup qualification
city                         Coffs Harbour
country                          Australia
neutral                              False
goal_diff                               31
Name: 25438, dtype: object
```

9. Draw percentage

```python
import pandas as pd
draws = results[results['home_score'] == results['away_score']]
draw_percentage = (len(draws) / len(results)) * 100
print(f"Draw percentage: {draw_percentage:.2f}%")
```

```
Draw percentage: 22.72%
```

10. Win ratio for each team

```python
import pandas as pd

home_wins = results[results['home_score'] > results['away_score']]['home_team'].value_counts()
away_wins = results[results['away_score'] > results['home_score']]['away_team'].value_counts()
wins = home_wins.add(away_wins, fill_value=0)

win_ratio = (wins / total_matches).sort_values(ascending=False)
print(win_ratio.head())
```

```
Surrey            1.00
Elba Island       1.00
Maule Sur         1.00
Asturias          1.00
Canary Islands    0.75
Name: count, dtype: float64
```

11. Home vs Away wins

```python
import pandas as pd
home_wins_count = len(results[results['home_score'] > results['away_score']])
away_wins_count = len(results[results['away_score'] > results['home_score']])

print(f"Home Wins: {home_wins_count}")
print(f"Away Wins: {away_wins_count}")
```

```
Home Wins: 23645
Away Wins: 13608
```

12. Team with highest shootout win rate

```python
import pandas as pd
shootouts = pd.read_csv('/root/.cache/kagglehub/datasets/martj42/international-football-results-from-1872-to-2017/versions/93/shootouts.csv')
wins = shootouts['winner'].value_counts()

# Check the available columns in the DataFrame:
print(shootouts.columns)
# Assuming the column containing the losing team is named 'home_team' or 'away_team' or 'loser'
#Replace 'home_team' with the actual column name for losing team if different.
shootout_played = pd.concat([shootouts['winner'], shootouts['home_team']]).value_counts()

win_rate = (wins / shootout_played).sort_values(ascending=False)
print(win_rate.head(1))
```

```
Index(['date', 'home_team', 'away_team', 'winner', 'first_shooter'], dtype='object')
Anguilla     1.0
Name: count, dtype: float64
```

13. Minimum Victory

```python
import pandas as pd
results['goal_diff'] = abs(results['home_score'] - results['away_score'])
minimum_victory = results.loc[results['goal_diff'].idxmin()]
print(minimum_victory)
```

```
date              1872-11-30
home_team           Scotland
away_team            England
home_score                 0
away_score                 0
tournament          Friendly
city                 Glasgow
country             Scotland
neutral                False
goal_diff                  0
Name: 0, dtype: object
```

14. Most Shootouts

```
[ ]  import pandas as pd
     most_shootouts = shootout_played.sort_values(ascending=False).head(1)
     print(most_shootouts)
```

```
South Africa    31
Name: count, dtype: int64
```

```python
import kagglehub
import pandas as pd
import numpy as np

# Download the dataset
path = kagglehub.dataset_download("martj42/international-football-results-from-1872-to-2017")

# Print the path to check it
print("Path to dataset files:", path)

# Assuming shootouts.csv is within the downloaded dataset
shootouts_path = f"{path}/shootouts.csv"  # Update with the correct file name if needed

# Load the dataframe
shootouts = pd.read_csv(shootouts_path)

# Rest of the code remains the same
shootout_winners = shootouts['winner'].to_numpy()
(unique_winners, counts) = np.unique(shootout_winners, return_counts=True)
sorted_indices = np.argsort(-counts)
top_5 = unique_winners[sorted_indices][:5]

print("Top 5 penalty shootout winners:")
print(top_5)
```

```
Warning: Looks like you're using an outdated `kagglehub` version (installed: 0.3.11), please consider upgrading to the latest version (0.3.12).
Downloading from https://www.kaggle.com/api/v1/datasets/download/martj42/international-football-results-from-1872-to-2017?dataset_version_number=93...
100%|████████████| 1.15M/1.15M [00:00<00:00, 98.7MB/s]Extracting files...
Path to dataset files: /root/.cache/kagglehub/datasets/martj42/international-football-results-from-1872-to-2017/versions/93
Top 5 penalty shootout winners:
['Argentina' 'South Korea' 'Zambia' 'Egypt' 'South Africa']
```

```python
import pandas as pd
import numpy as np
# Correct path to your dataset
file_path = '/root/.cache/kagglehub/datasets/martj42/international-football-results-from-1872-to-2017/versions/93/former_names.csv'

# Load the dataset
former_names = pd.read_csv(file_path)

# Print the columns to verify
print(former_names.columns)

# Check the first few rows
print(former_names.head())

# Group by the correct column name: 'current'
team_name_counts = former_names.groupby('current').size()

# Find the team with the most former names
least_former_names_team = team_name_counts.idxmin()
least_former_names_count = team_name_counts.min()

print(f"Team with the least former names: {least_former_names_team}")
print(f"Number of former names: {least_former_names_count}")
```

```
Index(['current', 'former', 'start_date', 'end_date'], dtype='object')
          current               former  start_date    end_date
0            Benin              Dahomey  1959-11-08  1975-11-30
1     Burkina Faso          Upper Volta  1960-04-14  1984-08-04
2          Curaçao  Netherlands Antilles  1957-03-03  2010-10-10
3   Czechoslovakia              Bohemia  1903-04-05  1919-01-01
4   Czechoslovakia  Bohemia and Moravia  1939-01-01  1945-05-01
Team with the least former names: Benin
Number of former names: 1
```

18.Team with least matches

```python
import pandas as pd
# Correct the file path:
results = pd.read_csv('/root/.cache/kagglehub/datasets/martj42/international-football-results-from-1872-to-2017/versions/93/results.csv')
# Assuming 'results.csv' is in the same directory as other dataset files

#Rest of the code remains the same
home_matches = results['home_team'].value_counts()
away_matches = results['away_team'].value_counts()
total_matches = home_matches.add(away_matches, fill_value=0)

least_matches_team = total_matches.idxmin()
print(f"Team with least matches: {least_matches_team}")
```

```
Team with least matches: Asturias
```

19. Team with least former names

```python
import pandas as pd


former_names = pd.read_csv('/root/.cache/kagglehub/datasets/martj42/international-football-results-from-1872-to-2017/versions/93/former_names.csv')

team_name_counts = former_names.groupby('current').size()
least_former_names_team = team_name_counts.idxmin()
least_former_names_count = team_name_counts.min()

print(f"Team with the least former names: {least_former_names_team} ({least_former_names_count} former names)")
```

```
Team with the least former names: Benin (1 former names)
```

20. Top 10 Team wise score

```python
import pandas as pd
team_scorer_counts = goalscorers.groupby('team')['scorer'].nunique().sort_values()
print(team_scorer_counts.head(10))
```

```
team
Anguilla            2
French Guiana       2
South Sudan         2
Bhutan              3
Somalia             3
Seychelles          4
Eritrea             4
Brunei              4
Saarland            4
Yemen DPR           4
Name: scorer, dtype: int64
```