Importing neccesary libraries

In [1]:

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

Resume

```
This notebook aims to predict the results of the FIFA Qatar 2022 Group Stage and Knockouts.
```

Importing data and models

In [2]:

```
ml_model = joblib.load('groups_stage_prediction.pkl')
```

In [3]:

```
last_team_scores = pd.read_csv('datos/team_scores.csv')
last_team_scores.head()
```

Out[3]:

	team	RK	Total Points	GK_score	def_score	mid_score	off_score
0	Brazil	1	1841.30	89.0	85.0	86.0	86.0
1	Belgium	2	1816.71	89.0	81.0	86.0	86.0
2	Argentina	3	1773.88	84.0	82.0	84.0	89.0
3	France	4	1759.78	87.0	84.0	87.0	88.0
4	England	5	1728.47	83.0	85.0	84.0	88.0

In [4]:

```
squad_stats = pd.read_csv('datos/squadstats26.csv')
squad_stats.head()
```

Out[4]:

	Nationality	Overall	Potential
0	Brazil	83.653846	85.346154
1	England	83.461538	85.923077
2	Germany	83.269231	85.384615
3	France	83.192308	85.346154
4	Spain	82.884615	84.500000

In [5]:

```
group_matches = pd.read_excel('datos/group_stage.xlsx')
round_16 = group_matches.iloc[48:56, :]
quarter_finals = group_matches.iloc[56:60, :]
semi_finals = group_matches.iloc[60:62, :]
final = group_matches.iloc[62:63, :]
second_final = group_matches.iloc[63:64, :]
group_matches = group_matches.iloc[:48, :]
group_matches.tail()
```

Out[5]:

```
          country1     country2  group
43      Costa Rica      Germany      e
44           Ghana      Uruguay      h
45  Korea Republic     Portugal      h
46          Serbia  Switzerland      g
47        Cameroon       Brazil      g
```

In [6]:

```
team_group = group_matches.drop(['country2'], axis=1)
team_group = team_group.drop_duplicates().reset_index(drop=True)
team_group = team_group.rename(columns = {"country1":"team"})
team_group.head()
```

Out[6]:

```
      team  group
0    Qatar      a
1  Senegal      a
2  England      b
3      USA      b
4   France      d
```

Defining some important functions we will use

In [7]:

```
def matches(g_matches):
    g_matches.insert(2, 'Overall1', g_matches['country1'].map(squad_stats.set_index('Nationality')['Overall']))
    g_matches.insert(3, 'Overall2', g_matches['country2'].map(squad_stats.set_index('Nationality')['Overall']))
    g_matches.insert(4, 'rank1', g_matches['country1'].map(last_team_scores.set_index('team')['RK']))
    g_matches.insert(5, 'rank2', g_matches['country2'].map(last_team_scores.set_index('team')['RK']))
    pred_set = []

    for index, row in g_matches.iterrows():
        if row['Overall1'] > row['Overall2'] and abs(row['Overall1'] - row['Overall2']) > 2:
            pred_set.append({'Team1': row['country1'], 'Team2': row['country2']})
        elif row['Overall2'] > row['Overall1'] and abs(row['Overall2'] - row['Overall1']) > 2:
            pred_set.append({'Team1': row['country2'], 'Team2': row['country1']})
        else:
            if row['rank1'] > row['rank2']:
                pred_set.append({'Team1': row['country1'], 'Team2': row['country2']})
            else:
                pred_set.append({'Team1': row['country2'], 'Team2': row['country1']})

    pred_set = pd.DataFrame(pred_set)
    pred_set.insert(2, 'Team1_FIFA_RANK', pred_set['Team1'].map(last_team_scores.set_index('team')['RK']))
    pred_set.insert(3, 'Team2_FIFA_RANK', pred_set['Team2'].map(last_team_scores.set_index('team')['RK']))
    pred_set.insert(4, 'Team1_Goalkeeper_Score', pred_set['Team1'].map(last_team_scores.set_index('team')['GK_score']))
    pred_set.insert(5, 'Team2_Goalkeeper_Score', pred_set['Team2'].map(last_team_scores.set_index('team')['GK_score']))
    pred_set.insert(6, 'Team1_Defense', pred_set['Team1'].map(last_team_scores.set_index('team')['def_score']))
    pred_set.insert(7, 'Team1_Offense', pred_set['Team1'].map(last_team_scores.set_index('team')['off_score']))
    pred_set.insert(8, 'Team1_Midfield', pred_set['Team1'].map(last_team_scores.set_index('team')['mid_score']))
    pred_set.insert(9, 'Team2_Defense', pred_set['Team2'].map(last_team_scores.set_index('team')['def_score']))
    pred_set.insert(10, 'Team2_Offense', pred_set['Team2'].map(last_team_scores.set_index('team')['off_score']))
    pred_set.insert(11, 'Team2_Midfield', pred_set['Team2'].map(last_team_scores.set_index('team')['mid_score']))
    return pred_set
```

In [8]:

```

def print_results(dataset, y_pred, matches, proba):
    results = []
    for i in range(dataset.shape[0]):
        print()
        if y_pred[i] == 1:
            print(matches.iloc[i, 0] + " vs. " + matches.iloc[i, 1] + " => Winner: " + dataset.iloc[i, 0])
            results.append({'result': dataset.iloc[i, 0]})
        else:
            print(matches.iloc[i, 0] + " vs. " + matches.iloc[i, 1] + " => Winner: " + dataset.iloc[i, 1])
            results.append({'result': dataset.iloc[i, 1]})
        try:
            print('Probability of ' + dataset.iloc[i, 0] + ' winning: ', '%.3f'%(proba[i][1]))
            print('Probability of ' + dataset.iloc[i, 1] + ' winning: ', '%.3f'%(proba[i][0]))
        except:
            print('Probability of ' + dataset.iloc[i, 1] + ' winning: ', '%.3f'%(proba[i][0]))
            print("")
    results = pd.DataFrame(results)
    matches = pd.concat([matches.group, results], axis=1)
    return results
```

In [9]:
```
dataset_groups = matches(group_matches)
dataset_groups.head()
```

Out[9]:

	Team1	Team2	Team1_FIFA_RANK	Team2_FIFA_RANK	Team1_Goalkeeper_Score	Team2_Goalkeeper_Score	Team1_Defense	Team1_Offense	Team1_Midfield	Team2_Defense	Team2_Offense	Team2_Midfield
0	Ecuador	Qatar	44	50	71.0	65.0	74.0	76.0	74.0	65.0	65.0	65.0
1	Netherlands	Senegal	8	18	81.0	83.0	85.0	83.0	84.0	79.0	81.0	79.0
2	IR Iran	England	20	5	73.0	83.0	69.0	75.0	69.0	85.0	88.0	84.0
3	Wales	USA	19	16	74.0	77.0	75.0	73.0	78.0	76.0	78.0	76.0
4	France	Australia	4	38	87.0	77.0	84.0	88.0	87.0	72.0	72.0	74.0

In [10]:

```
group_matches
```

Out[10]:

	country1	country2	Overall1	Overall2	rank1	rank2	group
0	Qatar	Ecuador	67.307692	71.500000	50	44	a
1	Senegal	Netherlands	75.076923	81.230769	18	8	a
2	England	IR Iran	83.461538	NaN	5	20	b
3	USA	Wales	NaN	73.538462	16	19	b
4	France	Australia	83.192308	71.038462	4	38	d
5	Denmark	Tunisia	76.692308	68.565217	10	30	d
6	Mexico	Poland	77.307692	76.115385	13	26	c
7	Argentina	Saudi Arabia	82.000000	68.846154	3	51	c
8	Belgium	Canada	80.538462	71.346154	2	41	f
9	Spain	Costa Rica	82.884615	67.947368	7	31	e
10	Germany	Japan	83.269231	73.307692	11	24	e
11	Morocco	Croatia	76.384615	78.769231	22	12	f
12	Switzerland	Cameroon	76.846154	72.653846	15	43	g
13	Uruguay	Korea Republic	78.346154	72.769231	14	28	h
14	Portugal	Ghana	82.807692	75.000000	9	61	h
15	Brazil	Serbia	83.653846	76.615385	1	21	g
16	Wales	IR Iran	73.538462	NaN	19	20	b
17	Qatar	Senegal	67.307692	75.076923	50	18	a
18	Netherlands	Ecuador	81.230769	71.500000	8	44	a
19	England	USA	83.461538	NaN	5	16	b
20	Tunisia	Australia	68.565217	71.038462	30	38	d
21	Poland	Saudi Arabia	76.115385	68.846154	26	51	c
22	France	Denmark	83.192308	76.692308	4	10	d
23	Argentina	Mexico	82.000000	77.307692	3	13	c
24	Japan	Costa Rica	73.307692	67.947368	24	31	e
25	Belgium	Morocco	80.538462	76.384615	2	22	f
26	Croatia	Canada	78.769231	71.346154	12	41	f
27	Spain	Germany	82.884615	83.269231	7	11	e
28	Cameroon	Serbia	72.653846	76.615385	43	21	g
29	Korea Republic	Ghana	72.769231	75.000000	28	61	h
30	Brazil	Switzerland	83.653846	76.846154	1	15	g
31	Portugal	Uruguay	82.807692	78.346154	9	14	h
32	Wales	England	73.538462	83.461538	19	5	b
33	IR Iran	USA	NaN	NaN	20	16	b
34	Ecuador	Senegal	71.500000	75.076923	44	18	a
35	Netherlands	Qatar	81.230769	67.307692	8	50	a
36	Australia	Denmark	71.038462	76.692308	38	10	d
37	Tunisia	France	68.565217	83.192308	30	4	d
38	Poland	Argentina	76.115385	82.000000	26	3	c

	country1	country2	Overall1	Overall2	rank1	rank2	group
39	Saudi Arabia	Mexico	68.846154	77.307692	51	13	c
40	Croatia	Belgium	78.769231	80.538462	12	2	f
41	Canada	Morocco	71.346154	76.384615	41	22	f
42	Japan	Spain	73.307692	82.884615	24	7	e
43	Costa Rica	Germany	67.947368	83.269231	31	11	e
44	Ghana	Uruguay	75.000000	78.346154	61	14	h
45	Korea Republic	Portugal	72.769231	82.807692	28	9	h
46	Serbia	Switzerland	76.615385	76.846154	21	15	g
47	Cameroon	Brazil	72.653846	83.653846	43	1	g

In [11]:

```
prediction_groups = ml_model.predict(dataset_groups)
proba = ml_model.predict_proba(dataset_groups)
results = print_results(dataset_groups, prediction_groups, group_matches, proba)
```

```
Costa Rica vs. Germany => Winner: Germany
Probability of Germany winning:  0.691
Probability of Costa Rica winning:  0.309

Ghana vs. Uruguay => Winner: Uruguay
Probability of Uruguay winning:  0.650
Probability of Ghana winning:  0.350

Korea Republic vs. Portugal => Winner: Portugal
Probability of Portugal winning:  0.646
Probability of Korea Republic winning:  0.354

Serbia vs. Switzerland => Winner: Switzerland
Probability of Serbia winning:  0.427
Probability of Switzerland winning:  0.573

Cameroon vs. Brazil => Winner: Brazil
Probability of Brazil winning:  0.713
Probability of Cameroon winning:  0.287
```

In [12]:

```
team_group['points'] = 0
for i in range(results.shape[0]):
    for j in range(team_group.shape[0]):
        if results.iloc[i, 0] == team_group.iloc[j, 0]:
            team_group.iloc[j, 2] += 3
```

In [13]:

```
print(team_group.groupby(['group','team']).mean().astype(int))
```

```
                      points
group team
a     Ecuador              3
      Netherlands          9
      Qatar                0
      Senegal              6
b     England              9
      IR Iran              0
      USA                  6
      Wales                3
c     Argentina            9
      Mexico               6
      Poland               3
      Saudi Arabia         0
d     Australia            0
      Denmark              6
      France               9
      Tunisia              3
e     Costa Rica           0
      Germany              6
      Japan                3
      Spain                9
f     Belgium              9
      Canada               0
      Croatia              3
      Morocco              6
g     Brazil               9
      Cameroon             3
      Serbia               0
      Switzerland          6
h     Ghana                0
      Korea Republic       3
      Portugal             6
      Uruguay              9
```

KNOCKOUTS

Round of 16

In [14]:

```
def winner_to_match(round, prev_match):
    round.insert(0, 'c1', round['country1'].map(prev_match.set_index('group')['result']))
    round.insert(1, 'c2', round['country2'].map(prev_match.set_index('group')['result']))
    round = round.drop(['country1','country2'], axis=1)
    round = round.rename(columns={'c1':'country1', 'c2':'country2'}).reset_index(drop=True)
    return round
def prediction_knockout(round):
    dataset_round = matches(round)
    prediction_round = ml_model.predict(dataset_round)
    proba_round = ml_model.predict_proba(dataset_round)
    results_round = print_results(dataset_round, prediction_round, round, proba_round)
    return results_round
```

In [15]:
```
round_of_16 = team_group[team_group['points'] > 5].reset_index(drop=True)
round_of_16['group'] = (4 - 1/3 * round_of_16.points).astype(int).astype(str) + round_of_16.group
round_of_16 = round_of_16.rename(columns = {"team":"result"})

round_16 = winner_to_match(round_16, round_of_16)
results_round_16 = prediction_knockout(round_16)
```

```
Netherlands vs. USA => Winner: Netherlands
Probability of USA winning:  0.288
Probability of Netherlands winning:  0.712

Argentina vs. Denmark => Winner: Argentina
Probability of Argentina winning:  0.520
Probability of Denmark winning:  0.480

Spain vs. Morocco => Winner: Spain
Probability of Spain winning:  0.521
Probability of Morocco winning:  0.479

Brazil vs. Portugal => Winner: Brazil
Probability of Portugal winning:  0.280
Probability of Brazil winning:  0.720

England vs. Senegal => Winner: England
Probability of England winning:  0.593
Probability of Senegal winning:  0.407
```

Quarter finals

In [16]:
```
okas = pd.concat([round_16, results_round_16], axis=1)
```

In [17]:
```
quarter_finals = winner_to_match(quarter_finals, okas)
results_quarter_finals = prediction_knockout(quarter_finals)
```

```
Netherlands vs. Argentina => Winner: Argentina
Probability of Netherlands winning:  0.352
Probability of Argentina winning:  0.648

Spain vs. Brazil => Winner: Brazil
Probability of Spain winning:  0.406
Probability of Brazil winning:  0.594

England vs. France => Winner: France
Probability of England winning:  0.355
Probability of France winning:  0.645

Belgium vs. Uruguay => Winner: Belgium
Probability of Belgium winning:  0.617
Probability of Uruguay winning:  0.383
```

Semi Final

In [18]:
```
okas = pd.concat([quarter_finals, results_quarter_finals], axis=1)
```

In [19]:

```
semi_finals = winner_to_match(semi_finals, okas)
results_finals = prediction_knockout(semi_finals)
```

```
Argentina vs. Brazil => Winner: Brazil
Probability of Argentina winning:  0.298
Probability of Brazil winning:  0.702

France vs. Belgium => Winner: France
Probability of France winning:  0.560
Probability of Belgium winning:  0.440
```

Final

In [20]:

```
okas = pd.concat([semi_finals, results_finals], axis=1)
```

In [21]:

```
final = winner_to_match(final, okas)
winner = prediction_knockout(final)
```

```
Brazil vs. France => Winner: Brazil
Probability of France winning:  0.414
Probability of Brazil winning:  0.586
```

Third Place

In [22]:

```
results_finals_3 = results_quarter_finals[~results_quarter_finals.result.isin(results_finals.result)]

semi_finals['group'] = semi_finals['group'].replace('y1','z1')
semi_finals['group'] = semi_finals['group'].replace('y2','z2')

results_finals_3 = results_finals_3.reset_index()
okas = pd.concat([semi_finals, results_finals_3], axis=1)
okas.drop('index', inplace=True, axis=1)
```

In [23]:

```
second_final = winner_to_match(second_final, okas)
third = prediction_knockout(second_final)
```

```
Argentina vs. Belgium => Winner: Belgium
Probability of Argentina winning:  0.417
Probability of Belgium winning:  0.583
```

In [24]:

```
winner = winner
second = results_finals[~results_finals.result.isin(winner.result)]
third = third
```

Tournament Table

In [25]:
```
def center_str(round):
    spaces = ['',' ','  ','   ','    ','     ',]
    for j in range(2):
        for i in range(round.shape[0]):
            if (13 - len(round.iloc[i, j])) % 2 == 0:
                round.iloc[i, j] = spaces[int((13 - len(round.iloc[i, j])) / 2)] + round.iloc[i, j] + spaces[int((13 - len(round.iloc[i, j])) / 2)]
            else:
                round.iloc[i, j] = spaces[int(((13 - len(round.iloc[i, j])) / 2) - 0.5)] + round.iloc[i, j] + spaces[int(((13 - len(round.iloc[i, j])) / 2) + 0.5)]
    return round

def center2(a):
    spaces = ['',' ','  ','   ','    ','     ','      ','       ','        ','         ','          ','           ','            ','             ','              ','               ','
    if (29 - len(a)) % 2 == 0:
        a = spaces[int((29 - len(a)) / 2)] + a + spaces[int((29 - len(a)) / 2)]
    else:
        a = spaces[int(((29 - len(a)) / 2) - 0.5)] + a + spaces[int(((29 - len(a)) / 2) + 0.5)]
    return a
```

In [26]:
```
round_16 = center_str(round_16)
quarter_finals = center_str(quarter_finals)
semi_finals = center_str(semi_finals)
final = center_str(final)
group_matches = center_str(group_matches)
```

In [27]:

```
print(round_16.iloc[0, 0]+'────┐                                                                                                                              ┌────'+round_16.iloc[4, 0])
print('                 │                                                                                                                              │')
print('                 ├────'+quarter_finals.iloc[0, 0]+'────┐                                                                                  ┌────'+quarter_finals.iloc[2, 0]+'────┤')
print('                 │                     │                                                                                  │                     │')
print(round_16.iloc[0, 1]+'────┘                     │                                                                                  │                     └────'+round_16.iloc[4, 1])
print('                                       ├────'+semi_finals.iloc[0, 0]+'────┐                                      ┌────'+semi_finals.iloc[1, 0]+'────┤')
print(round_16.iloc[1, 0]+'────┐                     │                      │                                     │                     │                     ┌────'+round_16.iloc[5, 0])
print('                 │                     │                      │                                     │                     │                     │')
print('                 ├────'+quarter_finals.iloc[0, 1]+'────┘                      │                                     │                     └────'+quarter_finals.iloc[2, 1]+'────┤')
print('                 │                                            │                                     │                                           │')
print(round_16.iloc[1, 1]+'────┘                                            │                                     │                                           └────'+round_16.iloc[5, 1])
print('                                                              ├────'+final.iloc[0, 0]+'vs.'+final.iloc[0, 1]+'────┤')
print(round_16.iloc[2, 0]+'────┐                                            │                                     │                                           ┌────'+round_16.iloc[6, 0])
print('                 │                                            │                                     │                                           │')
print('                 ├────'+quarter_finals.iloc[1, 0]+'────┐                      │                                     │                     ┌────'+quarter_finals.iloc[3, 0]+'────┤')
print('                 │                     │                      │                                     │                     │                     │')
print(round_16.iloc[2, 1]+'────┘                     │                      │                                     │                     │                     └────'+round_16.iloc[6, 1])
print('                                       ├────'+semi_finals.iloc[0, 1]+'────┘                                      └────'+semi_finals.iloc[1, 1]+'────┤')
print(round_16.iloc[3, 0]+'────┐                     │                                                                                  │                     ┌────'+round_16.iloc[7, 0])
print('                 │                     │                                                                                  │                     │')
print('                 ├────'+quarter_finals.iloc[1, 1]+'────┘                                                                                  └────'+quarter_finals.iloc[3, 1]+'────┤')
print('                 │                                                                                                                              │')
print(round_16.iloc[3, 1]+'────┘                                                                                                                              └────'+round_16.iloc[7, 1])
print("                                                                 "+center2("\U0001F947"+winner.iloc[0, 0]))
print("                                                                 "+center2("\U0001F948"+second.iloc[0, 0]))
print("                                                                 "+center2("\U0001F949"+third.iloc[0, 0]))
```

```
 Netherlands ────┐                                                                                                                              ┌────   England
                 │                                                                                                                              │
                 ├──── Netherlands ────┐                                                                                  ┌────   England   ────┤
                 │                     │                                                                                  │                     │
     USA     ────┘                     │                                                                                  │                     └────   Senegal
                                       ├────  Argentina   ────┐                                     ┌────   France    ────┤
  Argentina  ────┐                     │                      │                                     │                     │                     ┌────   France
                 │                     │                      │                                     │                     │                     │
                 ├────  Argentina  ────┘                      │                                     │                     └────   France    ────┤
                 │                                            │                                     │                                           │
   Denmark   ────┘                                            │                                     │                                           └────   Mexico
                                                              ├────   Brazil    vs.   France    ────┤
    Spain    ────┐                                            │                                     │                                           ┌────   Belgium
                 │                                            │                                     │                                           │
                 ├────    Spain    ────┐                      │                                     │                     ┌────   Belgium   ────┤
                 │                     │                      │                                     │                     │                     │
   Morocco   ────┘                     │                      │                                     │                     │                     └────   Germany
                                       ├────   Brazil     ────┘                                     └────   Belgium   ────┤
   Brazil    ────┐                     │                                                                                  │                     ┌────   Uruguay
                 │                     │                                                                                  │                     │
                 ├────   Brazil    ────┘                                                                                  └────   Uruguay   ────┤
                 │                                                                                                                              │
  Portugal   ────┘                                                                                                                              └──── Switzerland
                                                                             🥇Brazil
                                                                             🥈France
                                                                            🥉Belgium
```

In [ ]: