In [131]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import sklearn
import matplotlib.pyplot as plt
```

In [132]:
```
Batsman_data= pd.read_csv('C:/Users/Harshita/OneDrive/Desktop/Batsman_Data.csv')
```

In [133]:
```
Batsman_data.head()
```

Out[133]:

	Unnamed: 0	Bat1	Runs	BF	SR	4s	6s	Opposition	Ground	Start Date	Match_ID	Batsman	Player_ID
0	1	DNB	-	-	-	-	-	v India	Nagpur	18 Dec 2009	ODI # 2933	Oshane Thomas	49619
1	2	DNB	-	-	-	-	-	v India	Kolkata	24 Dec 2009	ODI # 2935	Oshane Thomas	49619
2	3	DNB	-	-	-	-	-	v India	Delhi	27 Dec 2009	ODI # 2936	Oshane Thomas	49619
3	4	DNB	-	-	-	-	-	v Bangladesh	Dhaka	4 Jan 2010	ODI # 2937	Oshane Thomas	49619
4	5	DNB	-	-	-	-	-	v India	Dhaka	5 Jan 2010	ODI # 2938	Oshane Thomas	49619

In [134]:
```
Bowler_Data=pd.read_csv('C:/Users/Harshita/OneDrive/Desktop/Bowler_Data.csv')
```

In [135]:
```
Bowler_Data.head()
```

Out[135]:

	Unnamed: 0	Overs	Mdns	Runs	Wkts	Econ	Ave	SR	Opposition	Ground	Start Date	Match_ID	Bowler	Player_ID
0	1	8.0	0	57	0	7.12	-	-	v India	Nagpur	18 Dec 2009	ODI # 2933	Suranga Lakmal	49619
1	2	10.0	0	55	2	5.50	27.50	30.0	v India	Kolkata	24 Dec 2009	ODI # 2935	Suranga Lakmal	49619
2	3	-	-	-	-	-	-	-	v India	Delhi	27 Dec 2009	ODI # 2936	Suranga Lakmal	49619
3	4	9.0	1	63	2	7.00	31.50	27.0	v Bangladesh	Dhaka	4 Jan 2010	ODI # 2937	Suranga Lakmal	49619
4	5	8.0	1	48	0	6.00	-	-	v India	Dhaka	5 Jan 2010	ODI # 2938	Suranga Lakmal	49619

In [136]:
```
Ground_Averages=pd.read_csv('C:/Users/Harshita/OneDrive/Desktop/Ground_Averages.csv')
```

In [137]:
```
Ground_Averages.head()
```

Out[137]:

	Ground	Span	Mat	Won	Tied	NR	Runs	Wkts	Balls	Ave	RPO
0	Eden Gardens, Kolkata - India	2013-2017	4	4	0	0	2161	72	2297	30.01	5.64
1	Feroz Shah Kotla, Delhi - India	2013-2019	4	4	0	0	1789	75	2331	23.85	4.60
2	Melbourne Cricket Ground - Australia	2013-2019	15	15	0	0	7656	217	8482	35.28	5.41
3	Saurashtra Cricket Association Stadium, Rajkot...	2013-2015	2	2	0	0	1163	26	1200	44.73	5.81
4	Adelaide Oval - Australia	2013-2019	10	10	0	0	4863	157	5645	30.97	5.16

In [138]:
```
odi_match_results=pd.read_csv('C:/Users/Harshita/OneDrive/Desktop/ODI_Match_Results.csv')
```

In [139]:
```
odi_match_results.head()
```

Out[139]:

	Unnamed: 0	Result	Margin	BR	Toss	Bat	Opposition	Ground	Start Date	Match_ID	Country	Country_ID
0	418	won	85 runs	NaN	lost	1st	v India	Kolkata	3 Jan 2013	ODI # 3315	Pakistan	7
1	692	lost	85 runs	NaN	won	2nd	v Pakistan	Kolkata	3 Jan 2013	ODI # 3315	India	6
2	419	lost	10 runs	NaN	lost	2nd	v India	Delhi	6 Jan 2013	ODI # 3316	Pakistan	7
3	693	won	10 runs	NaN	won	1st	v Pakistan	Delhi	6 Jan 2013	ODI # 3316	India	6
4	121	lost	107 runs	NaN	lost	2nd	v Australia	Melbourne	11 Jan 2013	ODI # 3317	SriLanka	8

In [140]:
```
odi_match_totals=pd.read_csv('C:/Users/Harshita/OneDrive/Desktop/ODI_Match_Totals.csv')
```

In [141]:
```
odi_match_totals.head()
```

Out[141]:

	Unnamed: 0	Score	Overs	RPO	Target	Inns	Result	Opposition	Ground	Start Date	Match_ID	Country	Country_ID
0	412	250	48.3	5.15	NaN	1	won	v India	Kolkata	3 Jan 2013	ODI # 3315	Pakistan	7
1	680	165	48.0	3.43	251.0	2	lost	v Pakistan	Kolkata	3 Jan 2013	ODI # 3315	India	6
2	413	157	48.5	3.21	168.0	2	lost	v India	Delhi	6 Jan 2013	ODI # 3316	Pakistan	7
3	681	167	43.4	3.82	NaN	1	won	v Pakistan	Delhi	6 Jan 2013	ODI # 3316	India	6
4	117	198	40.0	4.95	306.0	2	lost	v Australia	Melbourne	11 Jan 2013	ODI # 3317	SriLanka	8

In [142]:
```
wc_players=pd.read_csv('C:/Users/Harshita/OneDrive/Desktop/WC_players.csv')
```

In [143]:
```
wc_players.head()
```

Out[143]:

	Player	ID	Country
0	Gulbadin Naib (c)	352048	Afghanistan
1	Rashid Khan (vc)	793463	Afghanistan
2	Aftab Alam	440963	Afghanistan
3	Asghar Afghan	320652	Afghanistan
4	Dawlat Zadran	516561	Afghanistan

In [144]:
```
Batsman_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11149 entries, 0 to 11148
Data columns (total 13 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   Unnamed: 0  11149 non-null  int64 
 1   Bat1        11149 non-null  object
 2   Runs        11149 non-null  object
 3   BF          11149 non-null  object
 4   SR          11149 non-null  object
 5   4s          11149 non-null  object
 6   6s          11149 non-null  object
 7   Opposition  11149 non-null  object
 8   Ground      11149 non-null  object
 9   Start Date  11149 non-null  object
 10  Match_ID    11149 non-null  object
 11  Batsman     11149 non-null  object
 12  Player_ID   11149 non-null  int64 
dtypes: int64(2), object(11)
memory usage: 1.1+ MB
```

In [145]:
```
Bowler_Data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11118 entries, 0 to 11117
Data columns (total 14 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   Unnamed: 0  11118 non-null  int64 
 1   Overs       11118 non-null  object
 2   Mdns        11118 non-null  object
 3   Runs        11118 non-null  object
 4   Wkts        11118 non-null  object
 5   Econ        11118 non-null  object
 6   Ave         11118 non-null  object
 7   SR          11118 non-null  object
 8   Opposition  11118 non-null  object
 9   Ground      11118 non-null  object
 10  Start Date  11118 non-null  object
 11  Match_ID    11118 non-null  object
 12  Bowler      11118 non-null  object
 13  Player_ID   11118 non-null  int64 
dtypes: int64(2), object(12)
memory usage: 1.2+ MB
```

In [146]:
```
Ground_Averages.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 106 entries, 0 to 105
Data columns (total 11 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   Ground  106 non-null    object 
 1   Span    106 non-null    object 
 2   Mat     106 non-null    int64  
 3   Won     106 non-null    int64  
 4   Tied    106 non-null    int64  
 5   NR      106 non-null    int64  
 6   Runs    106 non-null    int64  
 7   Wkts    106 non-null    int64  
 8   Balls   106 non-null    int64  
 9   Ave     106 non-null    float64
 10  RPO     106 non-null    float64
dtypes: float64(2), int64(7), object(2)
memory usage: 9.2+ KB
```

In [147]:
```
odi_match_results.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1322 entries, 0 to 1321
Data columns (total 12 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Unnamed: 0  1322 non-null   int64
 1   Result      1322 non-null   object
 2   Margin      1322 non-null   object
 3   BR          606 non-null    float64
 4   Toss        1322 non-null   object
 5   Bat         1322 non-null   object
 6   Opposition  1322 non-null   object
 7   Ground      1322 non-null   object
 8   Start Date  1322 non-null   object
 9   Match_ID    1322 non-null   object
 10  Country     1322 non-null   object
 11  Country_ID  1322 non-null   int64
dtypes: float64(1), int64(2), object(9)
memory usage: 124.1+ KB
```

In [148]:
```
odi_match_totals.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1296 entries, 0 to 1295
Data columns (total 13 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   Unnamed: 0  1296 non-null   int64  
 1   Score       1296 non-null   object 
 2   Overs       1296 non-null   float64
 3   RPO         1296 non-null   object 
 4   Target      620 non-null    float64
 5   Inns        1296 non-null   int64  
 6   Result      1296 non-null   object 
 7   Opposition  1296 non-null   object 
 8   Ground      1296 non-null   object 
 9   Start Date  1296 non-null   object 
 10  Match_ID    1296 non-null   object 
 11  Country     1296 non-null   object 
 12  Country_ID  1296 non-null   int64  
dtypes: float64(2), int64(3), object(8)
memory usage: 131.8+ KB
```

In [149]:
```
wc_players.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 152 entries, 0 to 151
Data columns (total 3 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   Player   152 non-null    object
 1   ID       152 non-null    int64 
 2   Country  152 non-null    object
dtypes: int64(1), object(2)
memory usage: 3.7+ KB
```

In [150]:
```
Batsman_data.isnull().sum()
```

Out[150]:
```
Unnamed: 0    0
Bat1          0
Runs          0
BF            0
SR            0
4s            0
6s            0
Opposition    0
Ground        0
Start Date    0
Match_ID      0
Batsman       0
Player_ID     0
dtype: int64
```

In [151]:
```
Bowler_Data.isnull().sum()
```

Out[151]:
```
Unnamed: 0    0
Overs         0
Mdns          0
Runs          0
Wkts          0
Econ          0
Ave           0
SR            0
Opposition    0
Ground        0
Start Date    0
Match_ID      0
Bowler        0
Player_ID     0
dtype: int64
```

In [152]:
```
Ground_Averages.isnull().sum()
```

Out[152]:
```
Ground    0
Span      0
Mat       0
Won       0
Tied      0
NR        0
Runs      0
Wkts      0
Balls     0
Ave       0
RPO       0
dtype: int64
```

In [153]:
```
odi_match_results.isnull().sum()
```

Out[153]:
```
Unnamed: 0      0
Result          0
Margin          0
BR            716
Toss            0
Bat             0
Opposition      0
Ground          0
Start Date      0
Match_ID        0
Country         0
Country_ID      0
dtype: int64
```

In [154]:
```
odi_match_totals.isnull().sum()
```

Out[154]:
```
Unnamed: 0      0
Score           0
Overs           0
RPO             0
Target        676
Inns            0
Result          0
Opposition      0
Ground          0
Start Date      0
Match_ID        0
Country         0
Country_ID      0
dtype: int64
```

In [155]:
```
wc_players.isnull().sum()
```

Out[155]:
```
Player     0
ID         0
Country    0
dtype: int64
```

In [156]:
```
Ground_Averages.shape
```

Out[156]:
```
(106, 11)
```

In [157]:
```
odi_match_results.shape
```

Out[157]:
```
(1322, 12)
```

In [158]:
```
odi_match_totals.shape
```

Out[158]:
```
(1296, 13)
```

In [159]:
```
Bowler_Data.shape
```

Out[159]:
```
(11118, 14)
```

In [160]:
```
Batsman_data.shape
```

Out[160]:
```
(11149, 13)
```

In [161]:
```
# missing value in odi_match_results
odi_match_results['Start Date']= pd.to_datetime(odi_match_results['Start Date'])
odi_match_results.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1322 entries, 0 to 1321
Data columns (total 12 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Unnamed: 0  1322 non-null   int64
 1   Result      1322 non-null   object
 2   Margin      1322 non-null   object
 3   BR          606 non-null    float64
 4   Toss        1322 non-null   object
 5   Bat         1322 non-null   object
 6   Opposition  1322 non-null   object
 7   Ground      1322 non-null   object
 8   Start Date  1322 non-null   datetime64[ns]
 9   Match_ID    1322 non-null   object
 10  Country     1322 non-null   object
 11  Country_ID  1322 non-null   int64
dtypes: datetime64[ns](1), float64(1), int64(2), object(8)
memory usage: 124.1+ KB
```

In [162]:
```
odi_match_results.head()
```

Out[162]:

	Unnamed: 0	Result	Margin	BR	Toss	Bat	Opposition	Ground	Start Date	Match_ID	Country	Country_ID
0	418	won	85 runs	NaN	lost	1st	v India	Kolkata	2013-01-03	ODI # 3315	Pakistan	7
1	692	lost	85 runs	NaN	won	2nd	v Pakistan	Kolkata	2013-01-03	ODI # 3315	India	6
2	419	lost	10 runs	NaN	lost	2nd	v India	Delhi	2013-01-06	ODI # 3316	Pakistan	7
3	693	won	10 runs	NaN	won	1st	v Pakistan	Delhi	2013-01-06	ODI # 3316	India	6
4	121	lost	107 runs	NaN	lost	2nd	v Australia	Melbourne	2013-01-11	ODI # 3317	SriLanka	8

In [163]:
```
# removing null values from odi_match_results
prob1_var= odi_match_results[['Toss', 'Result', 'Ground']]
prob1_var.dropna()
```

Out[163]:

	Toss	Result	Ground
0	lost	won	Kolkata
1	won	lost	Kolkata
2	lost	lost	Delhi
3	won	won	Delhi
4	lost	lost	Melbourne
...	...	...	...
1317	won	won	Nottingham
1318	-	aban	Edinburgh
1319	won	-	Belfast
1320	lost	-	Leeds
1321	won	-	Leeds

1322 rows × 3 columns

In [164]:
```
# grouping the data to the required stadium
prob1= prob1_var.query('Ground == "The Oval"')
prob1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 34 entries, 83 to 1303
Data columns (total 3 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   Toss    34 non-null     object
 1   Result  34 non-null     object
 2   Ground  34 non-null     object
dtypes: object(3)
memory usage: 1.1+ KB
```

In [165]:
```
# the winning results from the dataframe
tempvar = prob1.loc[prob1['Result']=='won']
tvar = tempvar['Toss'].value_counts()
tvar
```

Out[165]:
```
won     10
lost     5
Name: Toss, dtype: int64
```

In [166]:
```
# we group the data according to the Hamilton
prob2 = odi_match_results[['Bat', 'Ground', 'Result', 'Opposition']]
prob2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1322 entries, 0 to 1321
Data columns (total 4 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   Bat         1322 non-null   object
 1   Ground      1322 non-null   object
 2   Result      1322 non-null   object
 3   Opposition  1322 non-null   object
dtypes: object(4)
memory usage: 41.4+ KB
```

In [167]:
```
prob2.head()
```

Out[167]:

	Bat	Ground	Result	Opposition
0	1st	Kolkata	won	v India
1	2nd	Kolkata	lost	v Pakistan
2	2nd	Delhi	lost	v India
3	1st	Delhi	won	v Pakistan
4	2nd	Melbourne	lost	v Australia

In [168]:
```
#removing null values
prob2.dropna()
prob2 = prob2.query('Ground == "Hamilton"')
prob2
```

Out[168]:

	Bat	Ground	Result	Opposition
40	2nd	Hamilton	won	v England
41	1st	Hamilton	lost	v New Zealand
239	1st	Hamilton	won	v New Zealand
240	2nd	Hamilton	lost	v West Indies
249	1st	Hamilton	won	v India
250	2nd	Hamilton	lost	v New Zealand
257	2nd	Hamilton	won	v India
258	1st	Hamilton	lost	v New Zealand
382	1st	Hamilton	n/r	v New Zealand
383	2nd	Hamilton	n/r	v South Africa
442	2nd	Hamilton	won	v New Zealand
443	1st	Hamilton	lost	v Sri Lanka
487	1st	Hamilton	won	v Zimbabwe
532	2nd	Hamilton	won	v Ireland
535	2nd	Hamilton	won	v Bangladesh
536	1st	Hamilton	lost	v New Zealand
696	1st	Hamilton	won	v Australia
697	2nd	Hamilton	lost	v New Zealand
855	1st	Hamilton	won	v Australia
856	2nd	Hamilton	lost	v New Zealand
862	2nd	Hamilton	won	v New Zealand
863	1st	Hamilton	lost	v South Africa
872	1st	Hamilton	lost	v New Zealand
873	2nd	Hamilton	won	v South Africa
1043	1st	Hamilton	lost	v New Zealand
1044	2nd	Hamilton	won	v Pakistan
1081	2nd	Hamilton	won	v England

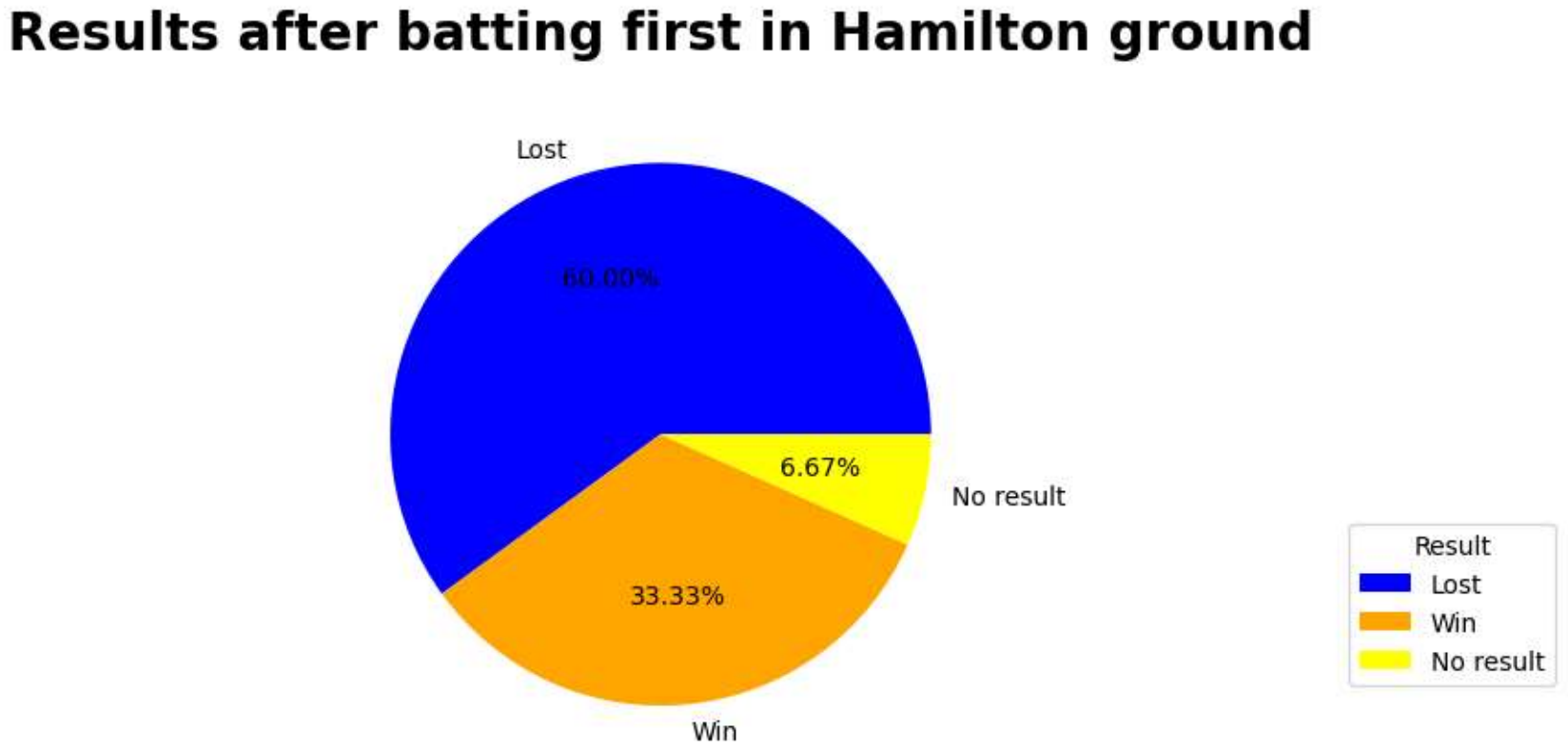
	Bat	Ground	Result	Opposition
1082	1st	Hamilton	lost	v New Zealand
1242	2nd	Hamilton	won	v India
1243	1st	Hamilton	lost	v New Zealand

In [169]:
```
prob2_b1 = prob2.loc[prob2['Bat']=='1st']
prob2_b1['Result'].value_counts()
```

Out[169]:
```
lost    9
won     5
n/r     1
Name: Result, dtype: int64
```

In [170]:
```
#plotting the graph
fig = figsize =(30, 20)
label = ['Lost', 'Win', 'No result']
colors = ['blue', 'orange', 'yellow']
plt.pie(prob2_b1['Result'].value_counts(), autopct='%1.2f%%', labels=label, textprops = dict(color ="black"),
plt.title('Results after batting first in Hamilton ground', size = 20, weight ="bold", pad=18)
plt.legend(title ="Result", loc ="center left", bbox_to_anchor =(1.5, 0, 0.5, 0.5))
```

Out[170]: <matplotlib.legend.Legend at 0x1ecaf3da3d0>

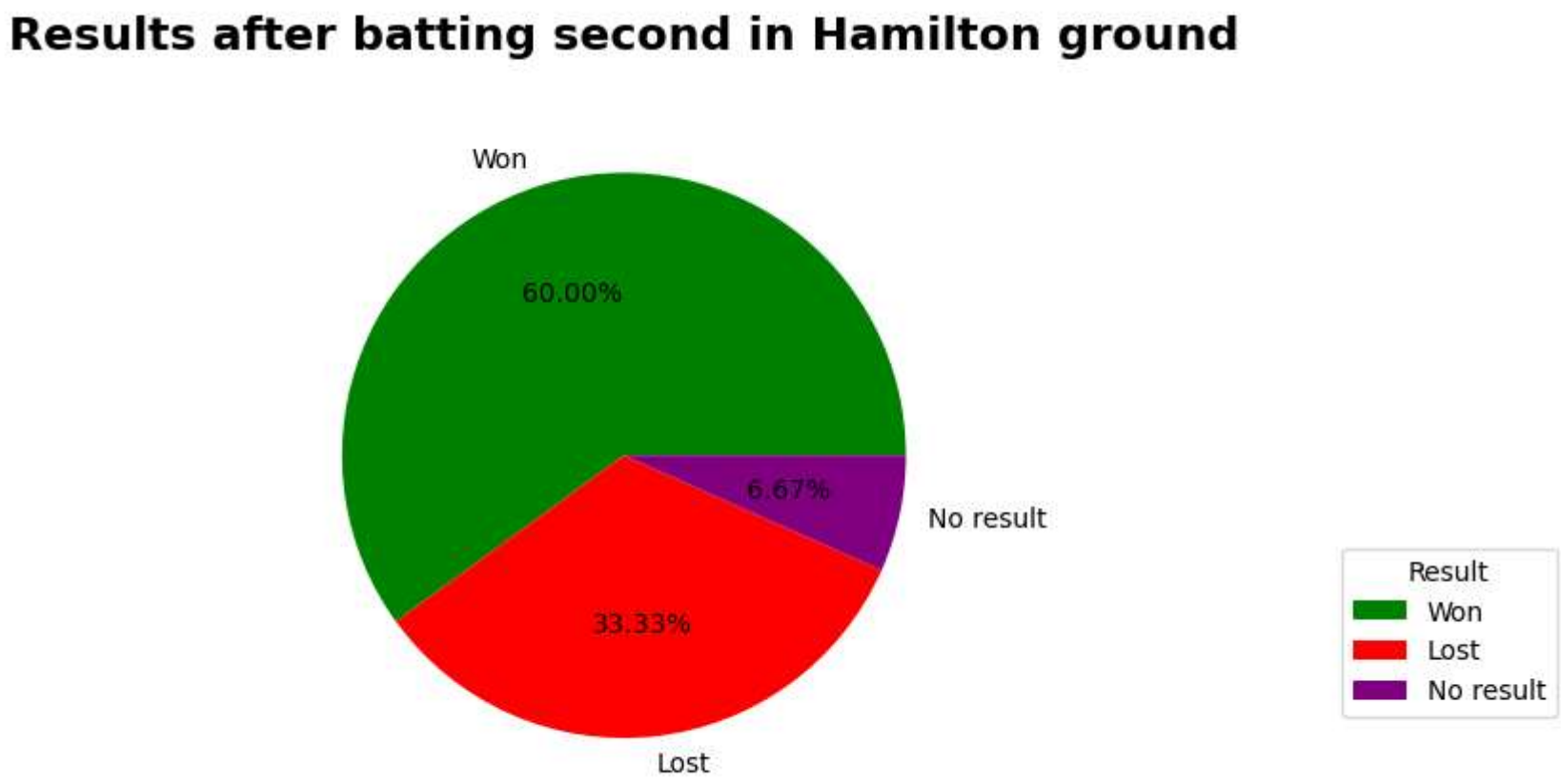


In [171]:
```
#Grouping the data according to second batting
prob2_b2 = prob2.loc[prob2['Bat']=='2nd']
#Counting how many of these second batters have won and lost
prob2_b2['Result'].value_counts()
```

Out[171]:
```
won     10
lost     4
n/r      1
Name: Result, dtype: int64
```

In [172]:
```
fig = figsize =(30, 20)
label = ['Won', 'Lost', 'No result']
colors = ['green', 'red', 'purple']
plt.pie(prob2_b1['Result'].value_counts(), autopct='%1.2f%%', labels=label, textprops = dict(color ="black"),
plt.title('Results after batting second in Hamilton ground', size = 17, weight ="bold", pad=20)
plt.legend(title ="Result", loc ="center left", bbox_to_anchor =(1.5, 0, 0.5, 0.5))
```

Out[172]: <matplotlib.legend.Legend at 0x1ecaf2fdb10>



Thus, we can predict that teams which have batted first have higher chances of winning than teams which have batted second

In [173]:
```
odi_match_results.head()
```

Out[173]:

	Unnamed: 0	Result	Margin	BR	Toss	Bat	Opposition	Ground	Start Date	Match_ID	Country	Country_ID
0	418	won	85 runs	NaN	lost	1st	v India	Kolkata	2013-01-03	ODI # 3315	Pakistan	7
1	692	lost	85 runs	NaN	won	2nd	v Pakistan	Kolkata	2013-01-03	ODI # 3315	India	6
2	419	lost	10 runs	NaN	lost	2nd	v India	Delhi	2013-01-06	ODI # 3316	Pakistan	7
3	693	won	10 runs	NaN	won	1st	v Pakistan	Delhi	2013-01-06	ODI # 3316	India	6
4	121	lost	107 runs	NaN	lost	2nd	v Australia	Melbourne	2013-01-11	ODI # 3317	SriLanka	8

In [174]:
```
prob3 = odi_match_totals[['Target', 'Result', 'Ground']]
prob3 = prob3.dropna()
prob3
```

Out[174]:

	Target	Result	Ground
1	251.0	lost	Kolkata
2	168.0	lost	Delhi
4	306.0	lost	Melbourne
6	326.0	lost	Rajkot
8	171.0	won	Adelaide
...	...	...	...
1288	293.0	won	Dublin
1290	210.0	won	Dublin (Malahide)
1292	341.0	won	Nottingham
1293	211.0	-	Belfast
1294	352.0	-	Leeds

620 rows × 3 columns

In [175]:
```
#Grouping according to Matches played at Oval
prob3g = prob3.query('Ground == "The Oval"')
#Grouping according to winners at Oval
prob3g = prob3g.loc[prob3g['Result']=='won']
#Mean score of the wiining team
mean_score = prob3g['Target'].mean()
prob3g
```

Out[175]:

	Target	Result	Ground
82	171.0	won	The Oval
91	234.0	won	The Oval
94	294.0	won	The Oval
105	176.0	won	The Oval
718	308.0	won	The Oval
893	306.0	won	The Oval
907	322.0	won	The Oval
916	192.0	won	The Oval
967	253.0	won	The Oval
1083	215.0	won	The Oval

In [176]:
```
mean_score
```

Out[176]:
```
247.1
```

In [177]:
```
print("The Average Score for this ground is: ")
print(prob3['Target'].mean())
print("The Highest Score for this ground is: ")
print(prob3g['Target'].max())
print("The Lowest Score for this ground is: ")
print(prob3['Target'].min())
```

```
The Average Score for this ground is:
253.44032258064516
The Highest Score for this ground is:
322.0
The Lowest Score for this ground is:
68.0
```

Country Analysis

In [178]:
```
prob6 = odi_match_results.query('Country == "India"')
prob6 = prob6.query('Opposition == "v Pakistan"')
prob6.head()
```

Out[178]:

	Unnamed: 0	Result	Margin	BR	Toss	Bat	Opposition	Ground	Start Date	Match_ID	Country	Country_ID
1	692	lost	85 runs	NaN	won	2nd	v Pakistan	Kolkata	2013-01-03	ODI # 3315	India	6
3	693	won	10 runs	NaN	won	1st	v Pakistan	Delhi	2013-01-06	ODI # 3316	India	6
100	701	won	8 wickets	17.0	won	2nd	v Pakistan	Birmingham	2013-06-15	ODI # 3372	India	6
281	734	lost	1 wickets	2.0	lost	1st	v Pakistan	Dhaka	2014-03-02	ODI # 3479	India	6
489	758	won	76 runs	NaN	won	1st	v Pakistan	Adelaide	2015-02-15	ODI # 3602	India	6

Top 10 Batsmen with their highest matches

In [179]:
```
# missing value
Batsman_data= Batsman_data[~Batsman_data['Bat1'].isin(['TDNB','DNB','absent','sub'])]
```

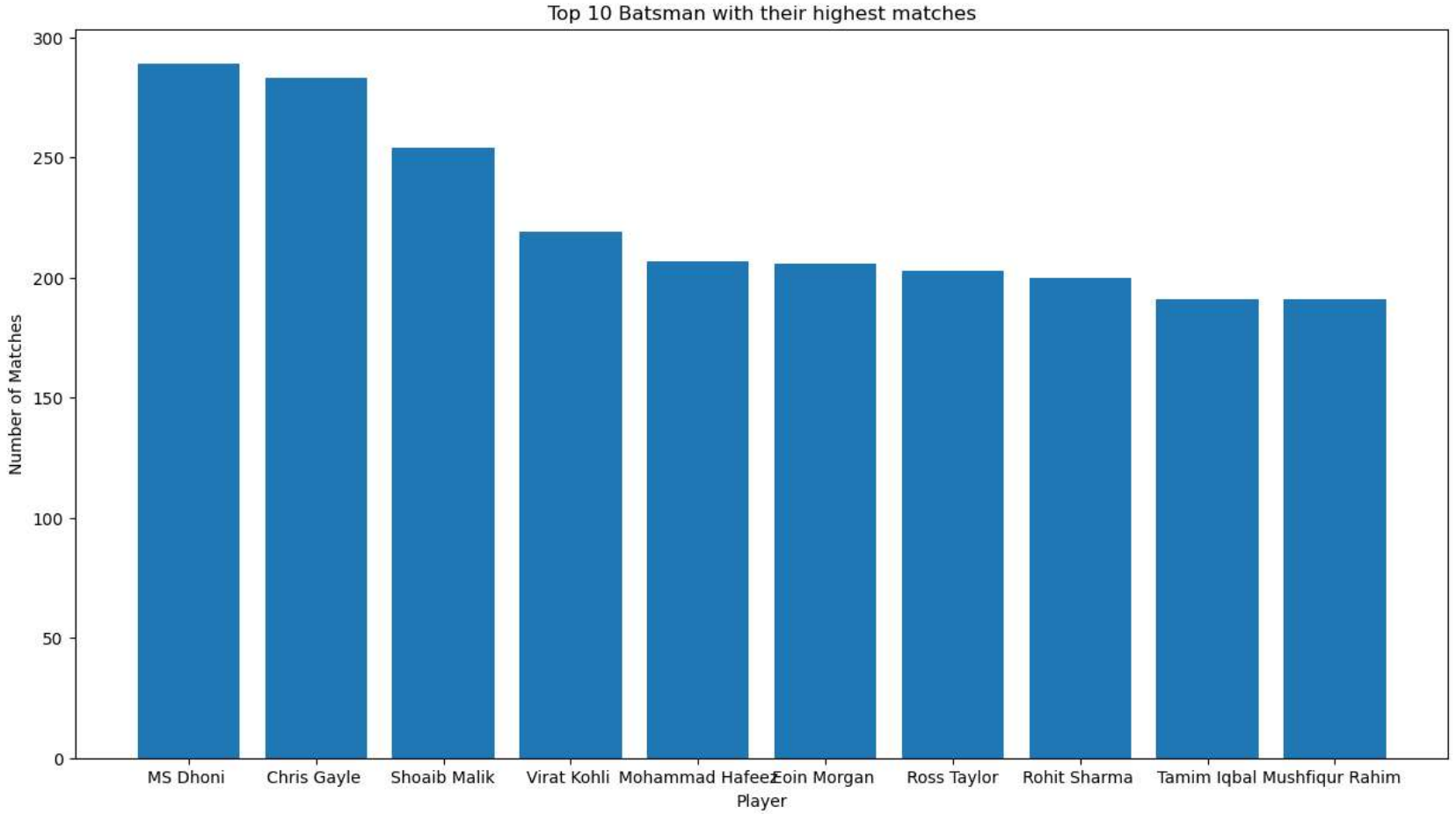
In [180]:
```
Batsman_data
```

Out[180]:

	Unnamed: 0	Bat1	Runs	BF	SR	4s	6s	Opposition	Ground	Start Date	Match_ID	Batsman	Player_ID
5	6	0*	0	8	0.00	0	0	v India	Dhaka	10 Jan 2010	ODI # 2941	Oshane Thomas	49619
6	7	0*	0	0	-	0	0	v England	The Oval	28 Jun 2011	ODI # 3165	Oshane Thomas	49619
9	10	1*	1	3	33.33	0	0	v England	Nottingham	6 Jul 2011	ODI # 3169	Oshane Thomas	49619
10	11	0*	0	2	0.00	0	0	v Australia	Pallekele	10 Aug 2011	ODI # 3175	Oshane Thomas	49619
11	12	0	0	2	0.00	0	0	v Pakistan	Dubai (DSC)	11 Nov 2011	ODI # 3212	Oshane Thomas	49619
...	...	...	...	...	...	...	...	...	...	...	...	...	...
11142	11143	0*	0	2	0.00	0	0	v Bangladesh	Abu Dhabi	23 Sep 2018	ODI # 4045	Gulbadin Naib	352048
11143	11144	15	15	46	32.60	1	0	v India	Dubai (DSC)	25 Sep 2018	ODI # 4046	Gulbadin Naib	352048
11144	11145	46	46	61	75.40	6	1	v Ireland	Dehradun	28 Feb 2019	ODI # 4100	Gulbadin Naib	352048
11145	11146	3	3	6	50.00	0	0	v Ireland	Dehradun	2 Mar 2019	ODI # 4101	Gulbadin Naib	352048
11146	11147	1	1	8	12.50	0	0	v Ireland	Dehradun	8 Mar 2019	ODI # 4108	Gulbadin Naib	352048

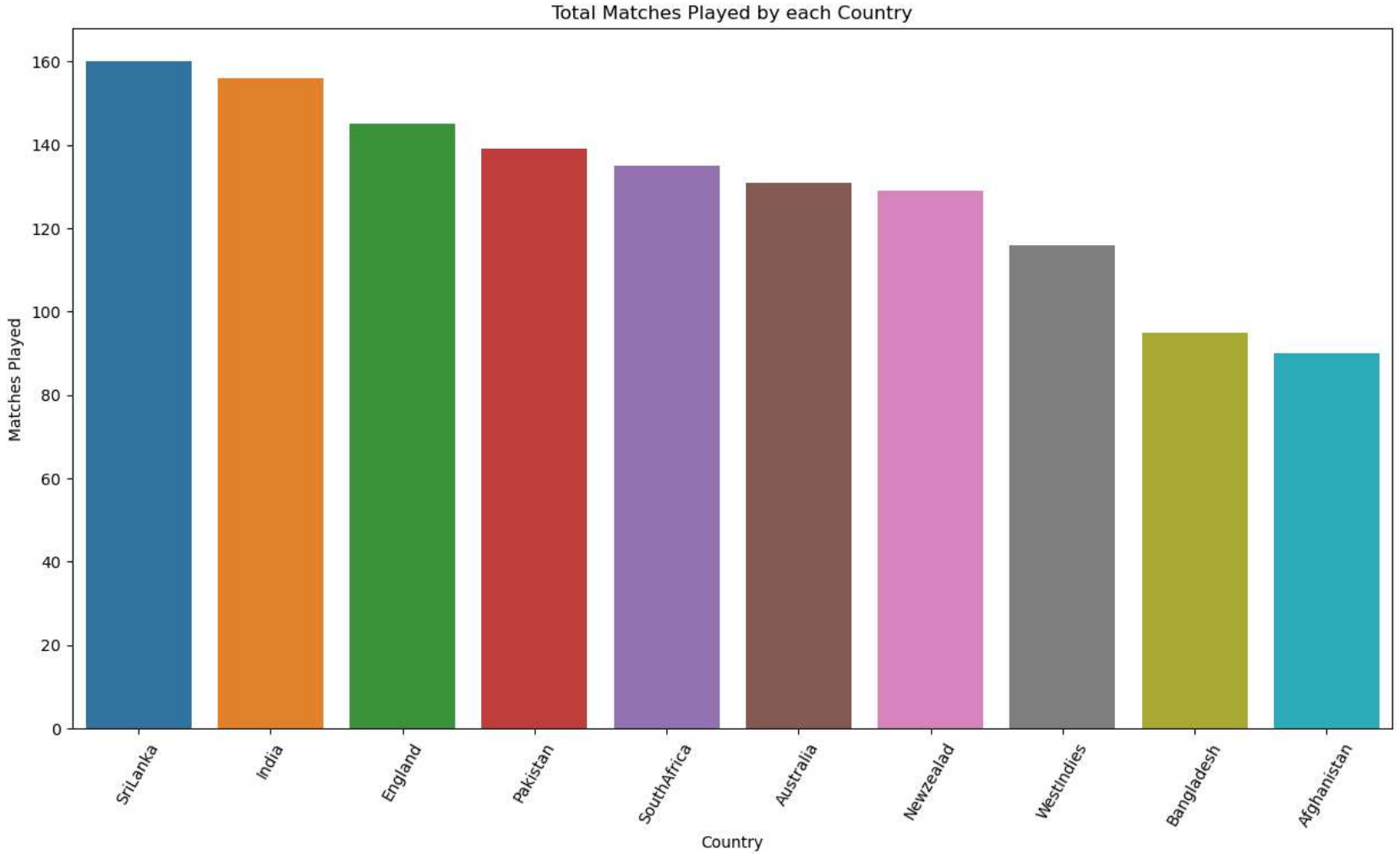
8918 rows × 13 columns

In [181]:
```
Top_10_players= Batsman_data['Batsman'].value_counts()[:10]
plt.figure(figsize=(15,8))
plt.bar(Top_10_players.index,Top_10_players.values)
plt.xlabel('Player')
plt.ylabel('Number of Matches')
plt.title('Top 10 Batsman with their highest matches')
plt.show()
```



In [182]:
```
Team_Matches = odi_match_totals.Country.value_counts().reset_index()
plt.figure(figsize=(15,8))
sns.barplot(x = "index", y = "Country", data = Team_Matches).set_title("Total Matches Played by each Country"
plt.xlabel("Country")
plt.ylabel("Matches Played")
plt.xticks(rotation = 60)
```

Out[182]:
```
(array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9]),
 [Text(0, 0, 'SriLanka'),
  Text(1, 0, 'India'),
  Text(2, 0, 'England'),
  Text(3, 0, 'Pakistan'),
  Text(4, 0, 'SouthAfrica'),
  Text(5, 0, 'Australia'),
  Text(6, 0, 'Newzealad'),
  Text(7, 0, 'WestIndies'),
  Text(8, 0, 'Bangladesh'),
  Text(9, 0, 'Afghanistan')])
```


Total Matches Played by each Country
Matches Played
0
20
40
60
80
100
120
140
160
SriLanka
India
England
Pakistan
SouthAfrica
Australia
Newzealad
Westindies
Bangladesh
Afghanistan
Country

In [183]:
```
# India vs pakistan match result
India_vs_Pakistan = odi_match_totals[odi_match_totals.Country == "Pakistan"]\
[odi_match_totals.Opposition.str.contains("India")]
India_vs_Pakistan = India_vs_Pakistan.Result.value_counts().reset_index()
sns.barplot(x = "index", y = "Result", data = India_vs_Pakistan).set_title("Pakistan against India")
plt.xlabel("Pakistan")
```

```
C:\Users\Harshita\AppData\Local\Temp\ipykernel_18788\1963163686.py:2: UserWarning: Boolean Series key will
be reindexed to match DataFrame index.
  India_vs_Pakistan = odi_match_totals[odi_match_totals.Country == "Pakistan"]\
```

Out[183]:
```
Text(0.5, 0, 'Pakistan')
```