Analysis of Cricket matches

In [22]:

```
from pyspark.sql import *

# Creating a schema Row
Match = Row('Match_id','Team1','Team2','Winning_Team','Runs_scored_by_winner','Locatio
n')

# Creating enteries for the dataframe
match1 = Match(1,'IND','AUS','IND',300,'Mumbai')
match2 = Match(2,'SA','ENG','ENG',350,'Delhi')
match3 = Match(3,'BANG','KENIA','BANG',400,'Australia')
match4 = Match(4,'IND','SA','SA',200,'Pakistan')
match5 = Match(5,'IND','WI','WI',250,'India')
match6 = Match(6,'AUS','ITLAY','AUS',180,'India')
match7 = Match(7,'SA','WI','WI',313,'Sri Lanka')
match8 = Match(8,'IND','WI','IND',325,'Delhi')
match9 = Match(9,'IND','AUS','IND',400,'Mumbai')
match10 = Match(10,'IND','WI','WI',363,'Chennai')
```

In [23]:

```
# Showing the schema
Match
```

```
<Row(Match_id, Team1, Team2, Winning_Team, Runs_scored_by_winner, Locatio
n)>
```

In [24]:

```
# Showing match1
match1
```

```
Row(Match_id=1, Team1='IND', Team2='AUS', Winning_Team='IND', Runs_scored_
by_winner=300, Location='Mumbai')
```

In [25]:

```
# Creating a dataframe as df1 from a list
matches = [match1,match2,match3,match4,match5,match6,match7,match8,match9,match10]
df1 = spark.createDataFrame(matches)
df1.show()
```

```
+--------+-----+-----+------------+--------------------+---------+
|Match_id|Team1|Team2|Winning_Team|Runs_scored_by_winner| Location|
+--------+-----+-----+------------+--------------------+---------+
|       1|  IND|  AUS|         IND|                 300|   Mumbai|
|       2|   SA|  ENG|         ENG|                 350|    Delhi|
|       3| BANG|KENIA|        BANG|                 400|Australia|
|       4|  IND|   SA|          SA|                 200| Pakistan|
|       5|  IND|   WI|          WI|                 250|    India|
|       6|  AUS|ITLAY|         AUS|                 180|    India|
|       7|   SA|   WI|          WI|                 313|Sri Lanka|
|       8|  IND|   WI|         IND|                 325|    Delhi|
|       9|  IND|  AUS|         IND|                 400|   Mumbai|
|      10|  IND|   WI|          WI|                 363|  Chennai|
+--------+-----+-----+------------+--------------------+---------+
```

In [26]:

```
# selecting the fields from dataframe order by the winning team
df1.select('Match_id','Winning_Team','Runs_scored_by_winner').orderBy(df1.Winning_Team.
desc()).show()
```

```
+--------+------------+--------------------+
|Match_id|Winning_Team|Runs_scored_by_winner|
+--------+------------+--------------------+
|       5|          WI|                 250|
|       7|          WI|                 313|
|      10|          WI|                 363|
|       4|          SA|                 200|
|       9|         IND|                 400|
|       1|         IND|                 300|
|       8|         IND|                 325|
|       2|         ENG|                 350|
|       3|        BANG|                 400|
|       6|         AUS|                 180|
+--------+------------+--------------------+
```

In [28]:

```
# grouping the Winning team and finding the maximum scored
df1.groupby('Winning_Team').agg({'Runs_scored_by_winner':'max'}).show()
```

```
+------------+--------------------------+
|Winning_Team|max(Runs_scored_by_winner)|
+------------+--------------------------+
|         AUS|                       180|
|          WI|                       363|
|          SA|                       200|
|         ENG|                       350|
|         IND|                       400|
|        BANG|                       400|
+------------+--------------------------+
```

In [29]:

```
# Count of matches a team has won
df1.groupby('Winning_Team').count().show()
```

```
+------------+-----+
|Winning_Team|count|
+------------+-----+
|         AUS|    1|
|          WI|    3|
|          SA|    1|
|         ENG|    1|
|         IND|    3|
|        BANG|    1|
+------------+-----+
```

In [ ]:

```

```