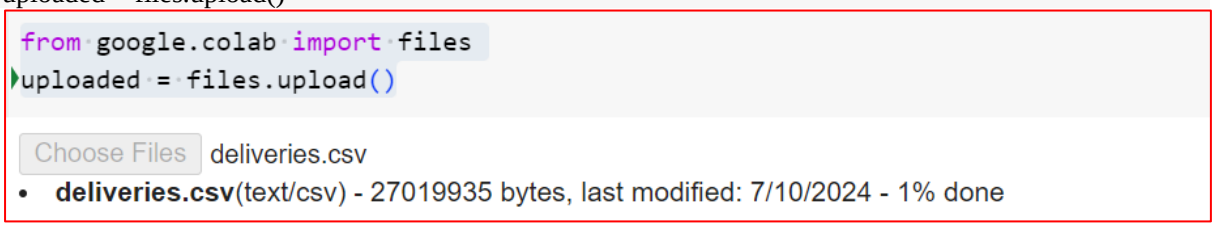
IPL Final KKR vs SRH 2024 Scorecard Building Using PySpark

```
!pip install pyspark findspark
```

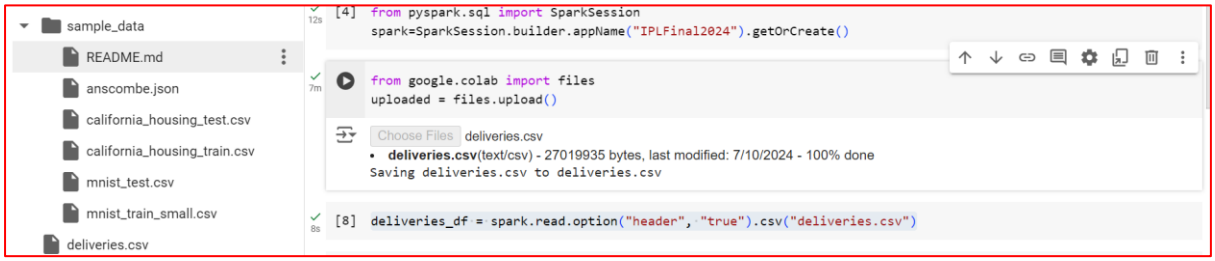

```
import findspark
findspark.init()
```

```
from pyspark.sql import SparkSession
spark=SparkSession.builder.appName("IPLFinal2024").getOrCreate()
```

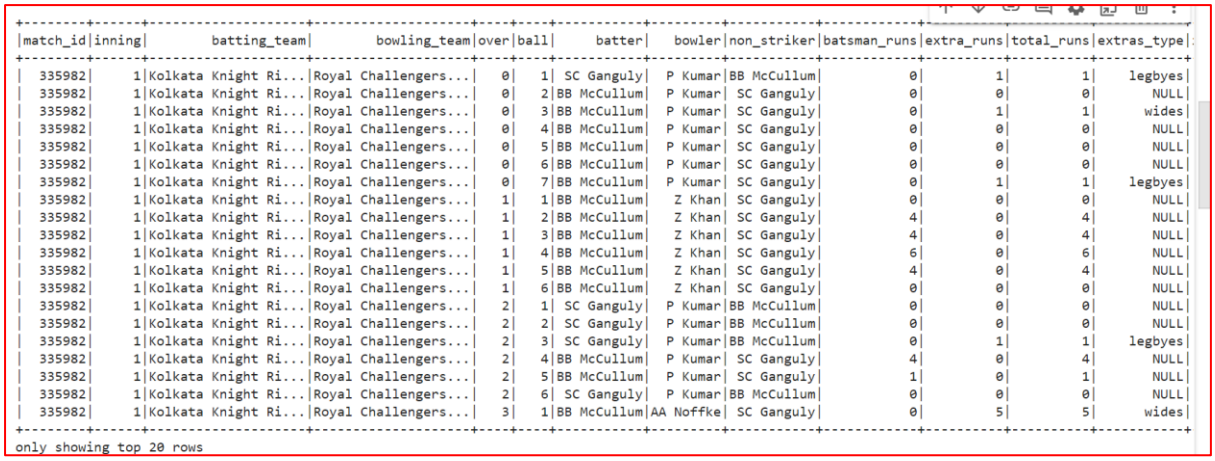
```
from google.colab import files
uploaded = files.upload()
```




```
deliveries_df = spark.read.option("header", "true").csv("deliveries.csv")
```

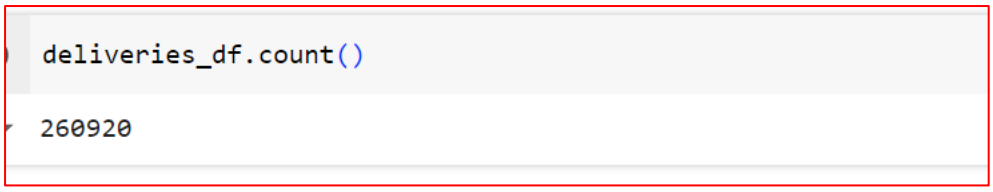



```
deliveries_df.show()
```



match_id	inning	batting_team	bowling_team	over	ball	batter	bowler	non_striker	batsman_runs	extra_runs	total_runs	extras_type
335982	1	Kolkata Knight Ri...	Royal Challengers...	0	1	SC Ganguly	P Kumar	BB McCullum	0	1	1	legbyes
335982	1	Kolkata Knight Ri...	Royal Challengers...	0	2	BB McCullum	P Kumar	SC Ganguly	0	0	0	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	0	3	BB McCullum	P Kumar	SC Ganguly	0	1	1	wides
335982	1	Kolkata Knight Ri...	Royal Challengers...	0	4	BB McCullum	P Kumar	SC Ganguly	0	0	0	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	0	5	BB McCullum	P Kumar	SC Ganguly	0	0	0	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	0	6	BB McCullum	P Kumar	SC Ganguly	0	0	0	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	0	7	BB McCullum	P Kumar	SC Ganguly	0	1	1	legbyes
335982	1	Kolkata Knight Ri...	Royal Challengers...	1	1	BB McCullum	Z Khan	SC Ganguly	0	0	0	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	1	2	BB McCullum	Z Khan	SC Ganguly	4	0	4	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	1	3	BB McCullum	Z Khan	SC Ganguly	4	0	4	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	1	4	BB McCullum	Z Khan	SC Ganguly	6	0	6	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	1	5	BB McCullum	Z Khan	SC Ganguly	4	0	4	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	1	6	BB McCullum	Z Khan	SC Ganguly	0	0	0	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	2	1	SC Ganguly	P Kumar	BB McCullum	0	0	0	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	2	2	SC Ganguly	P Kumar	BB McCullum	0	0	0	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	2	3	SC Ganguly	P Kumar	BB McCullum	0	1	1	legbyes
335982	1	Kolkata Knight Ri...	Royal Challengers...	2	4	BB McCullum	P Kumar	SC Ganguly	4	0	4	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	2	5	BB McCullum	P Kumar	SC Ganguly	1	0	1	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	2	6	SC Ganguly	P Kumar	BB McCullum	0	0	0	NULL
335982	1	Kolkata Knight Ri...	Royal Challengers...	3	1	BB McCullum	AA Noffke	SC Ganguly	0	5	5	wides

only showing top 20 rows

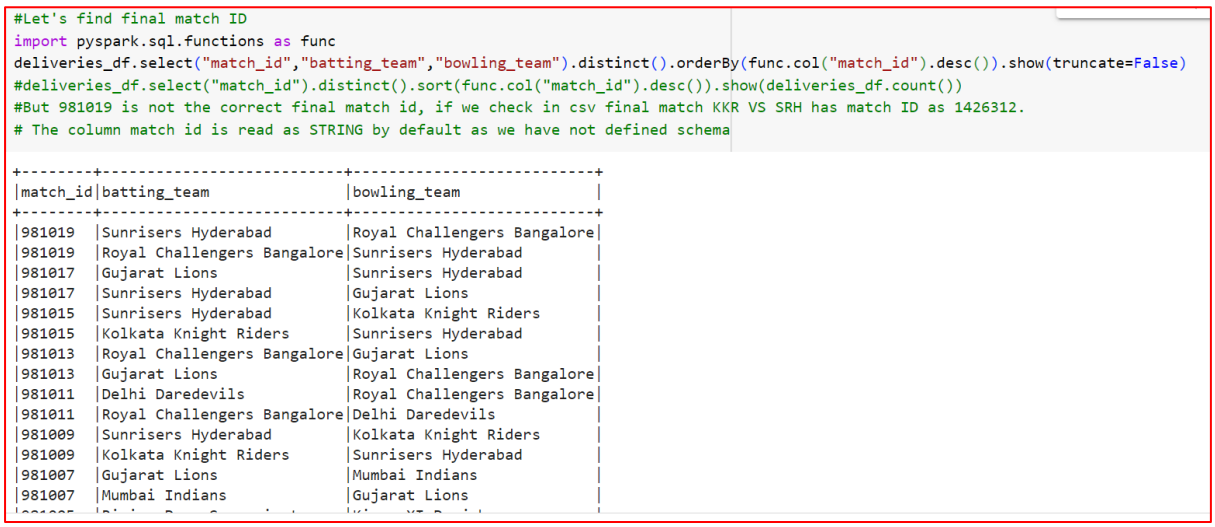


```
deliveries_df.count()
```

```
260920
```

```
deliveries_df.show(deliveries_df.count(),truncate=False) #df.show() Shows only top 20 rows
#truncate=False if not set it will truncate strings more than 20 chars eg. Royal Challegers ...
```

```
#Let's find final match ID
import pyspark.sql.functions as func
deliveries_df.select("match_id").distinct().orderBy(func.col("match_id").desc()).show()
#deliveries_df.select("match_id").distinct().sort(func.col("match_id").desc()).show(deliveries_df.count())
#But 981019 is not the correct final match id, if we check in csv final match KKR VS SRH has match ID as 1426312.
# The column match id is read as STRING by default as we have not defined schema
```



```
#Let's find final match ID
import pyspark.sql.functions as func
deliveries_df.select("match_id","batting_team","bowling_team").distinct().orderBy(func.col("match_id").desc()).show(truncate=False)
#deliveries_df.select("match_id").distinct().sort(func.col("match_id").desc()).show(deliveries_df.count())
#But 981019 is not the correct final match id, if we check in csv final match KKR VS SRH has match ID as 1426312.
# The column match id is read as STRING by default as we have not defined schema
```

match_id	batting_team	bowling_team
981019	Sunrisers Hyderabad	Royal Challengers Bangalore
981019	Royal Challengers Bangalore	Sunrisers Hyderabad
981017	Gujarat Lions	Sunrisers Hyderabad
981017	Sunrisers Hyderabad	Gujarat Lions
981015	Sunrisers Hyderabad	Kolkata Knight Riders
981015	Kolkata Knight Riders	Sunrisers Hyderabad
981013	Royal Challengers Bangalore	Gujarat Lions
981013	Gujarat Lions	Royal Challengers Bangalore
981011	Delhi Daredevils	Royal Challengers Bangalore
981011	Royal Challengers Bangalore	Delhi Daredevils
981009	Sunrisers Hyderabad	Kolkata Knight Riders
981009	Kolkata Knight Riders	Sunrisers Hyderabad
981007	Gujarat Lions	Mumbai Indians
981007	Mumbai Indians	Gujarat Lions

```
deliveries_df.printSchema()
```

```
root
 |-- match_id: string (nullable = true)
 |-- inning: string (nullable = true)
 |-- batting_team: string (nullable = true)
 |-- bowling_team: string (nullable = true)
 |-- over: string (nullable = true)
 |-- ball: string (nullable = true)
 |-- batter: string (nullable = true)
 |-- bowler: string (nullable = true)
 |-- non_striker: string (nullable = true)
 |-- batsman_runs: string (nullable = true)
 |-- extra_runs: string (nullable = true)
 |-- total_runs: string (nullable = true)
 |-- extras_type: string (nullable = true)
 |-- is_wicket: string (nullable = true)
 |-- player_dismissed: string (nullable = true)
 |-- dismissal_kind: string (nullable = true)
 |-- fielder: string (nullable = true)
```

```
#Let's define schema
from pyspark.sql.types import StringType, StructField
from pyspark.sql.types import *
int_col_list=["match_id","inning","over","ball","batsman_runs","extra_runs","total_runs","is_wicket"]

fields = StructType([ StructField("match_id", IntegerType(),nullable=True) ,  StructField("inning",
IntegerType(),nullable=True),
                     StructField("batting_team", StringType(),nullable=True),StructField("bowling_team",
StringType(),nullable=True),
                     StructField("over", IntegerType(),nullable=True),       StructField("ball",
IntegerType(),nullable=True),
                     StructField("batter", StringType(),nullable=True),      StructField("bowler",
StringType(),nullable=True),
                     StructField("non_striker", IntegerType(),nullable=True), StructField("batsman_runs",
IntegerType(),nullable=True),
                     StructField("extra_runs", IntegerType(),nullable=True), StructField("total_runs",
IntegerType(),nullable=True),
                     StructField("extras_type", StringType(),nullable=True), StructField("is_wicket",
IntegerType(),nullable=True),
                     StructField("player_dismissed", StringType(),nullable=True),StructField("dismissal_kind",
StringType(),nullable=True),
                     StructField("fielder", StringType(),nullable=True)
])

fields
```

```
#Let's define schema
from pyspark.sql.types import StringType, StructField
from pyspark.sql.types import *
int_col_list=["match_id","inning","over","ball","batsman_runs","extra_runs","total_runs","is_wicket"]

fields = StructType([ StructField("match_id", IntegerType(),nullable=True) ,  StructField("inning", IntegerType(),nullable=True),
                      StructField("batting_team", StringType(),nullable=True),StructField("bowling_team", StringType(),nullable=True),
                      StructField("over", IntegerType(),nullable=True),       StructField("ball", IntegerType(),nullable=True),
                      StructField("batter", StringType(),nullable=True),      StructField("bowler", StringType(),nullable=True),
                      StructField("non_striker", IntegerType(),nullable=True), StructField("batsman_runs", IntegerType(),nullable=True),
                      StructField("extra_runs", IntegerType(),nullable=True), StructField("total_runs", IntegerType(),nullable=True),
                      StructField("extras_type", StringType(),nullable=True), StructField("is_wicket", IntegerType(),nullable=True),
                      StructField("player_dismissed", StringType(),nullable=True),StructField("dismissal_kind", StringType(),nullable=True),
                      StructField("fielder", StringType(),nullable=True)
])

fields
```

```
StructType([StructField('match_id', IntegerType(), True), StructField('inning', IntegerType(), True), StructField('batting_team', StringType(), True),
StructField('bowling_team', StringType(), True), StructField('over', IntegerType(), True), StructField('ball', IntegerType(), True),
StructField('batter', StringType(), True), StructField('bowler', StringType(), True), StructField('non_striker', IntegerType(), True),
StructField('batsman_runs', IntegerType(), True), StructField('extra_runs', IntegerType(), True), StructField('total_runs', IntegerType(), True),
StructField('extras_type', StringType(), True), StructField('is_wicket', IntegerType(), True), StructField('player_dismissed', StringType(), True),
StructField('dismissal_kind', StringType(), True), StructField('fielder', StringType(), True)])
```

#Let's create dataframe with this schema

deliveries_df = spark.read.schema(fields).option("header",True).csv("deliveries.csv")

deliveries_df.printSchema()

```
#Let's create dataframe with this schema
deliveries_df = spark.read.schema(fields).option("header",True).csv("deliveries.csv")
deliveries_df.printSchema()
```

```
root
 |-- match_id: integer (nullable = true)
 |-- inning: integer (nullable = true)
 |-- batting_team: string (nullable = true)
 |-- bowling_team: string (nullable = true)
 |-- over: integer (nullable = true)
 |-- ball: integer (nullable = true)
 |-- batter: string (nullable = true)
 |-- bowler: string (nullable = true)
 |-- non_striker: integer (nullable = true)
 |-- batsman_runs: integer (nullable = true)
 |-- extra_runs: integer (nullable = true)
 |-- total_runs: integer (nullable = true)
 |-- extras_type: string (nullable = true)
 |-- is_wicket: integer (nullable = true)
 |-- player_dismissed: string (nullable = true)
 |-- dismissal_kind: string (nullable = true)
 |-- fielder: string (nullable = true)
```

#Let's find final match ID now

deliveries_df.select("match_id").distinct().orderBy(func.col("match_id").desc()).show(truncate=False)

```
#Let's find final match ID now
deliveries_df.select("match_id").distinct().orderBy(func.col("match_id").desc()).show(truncate=False)
```

```
+--------+
|match_id|
+--------+
|1426312 |
|1426311 |
|1426310 |
|1426309 |
|1426307 |
|1426306 |
|1426305 |
|1426303 |
|1426302 |
|1426300 |
|1426299 |
|1426298 |
|1426297 |
|1426296 |
|1426295 |
|1426294 |
|1426293 |
```

```
#Filter data only for final match id 1426312
import pyspark.sql.functions as func
ipl_final_df = deliveries_df.filter(func.col("match_id") == 1426312)

ipl_final_df.show(ipl_final_df.count(),truncate=False)
```

```
#Filter data only for final match id 1426312
import pyspark.sql.functions as func
ipl_final_df = deliveries_df.filter(func.col("match_id") == 1426312)

ipl_final_df.show(ipl_final_df.count(),truncate=False)
```

```
+--------+------+--------------------+--------------------+----+----+-------------------+-------------+-----------+------------+----------+-------
|match_id|inning|batting_team        |bowling_team        |over|ball|batter             |bowler       |non_striker|batsman_runs|extra_runs|total_r
+--------+------+--------------------+--------------------+----+----+-------------------+-------------+-----------+------------+----------+-------
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|0   |1   |Abhishek Sharma    |MA Starc     |NULL       |0           |0         |0
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|0   |2   |Abhishek Sharma    |MA Starc     |NULL       |0           |0         |0
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|0   |3   |Abhishek Sharma    |MA Starc     |NULL       |0           |0         |0
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|0   |4   |Abhishek Sharma    |MA Starc     |NULL       |2           |0         |2
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|0   |5   |Abhishek Sharma    |MA Starc     |NULL       |0           |0         |0
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|0   |6   |RA Tripathi        |MA Starc     |NULL       |1           |0         |1
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|1   |1   |RA Tripathi        |VG Arora     |NULL       |0           |0         |0
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|1   |2   |RA Tripathi        |VG Arora     |NULL       |0           |0         |0
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|1   |3   |RA Tripathi        |VG Arora     |NULL       |0           |0         |0
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|1   |4   |RA Tripathi        |VG Arora     |NULL       |0           |2         |2
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|1   |5   |RA Tripathi        |VG Arora     |NULL       |1           |0         |1
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|1   |6   |TM Head            |VG Arora     |NULL       |0           |0         |0
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|2   |1   |RA Tripathi        |MA Starc     |NULL       |4           |0         |4
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|2   |2   |RA Tripathi        |MA Starc     |NULL       |1           |0         |1
|1426312 |1     |Sunrisers Hyderabad |Kolkata Knight Riders|2   |3   |AK Markram         |MA Starc     |NULL       |0           |0         |0
```

✓ 1s completed at 12:53 PM

```
#We have to build the scorecard for both the innings, let's build it for first inning first.
first_inning_batting = ipl_final_df.filter(func.col("inning") == 1 )

first_inning_batting.show(first_inning_batting.count(), truncate = False)
```

```
#We have to build the scorecard for both the innings, let's build it for first inning first.
first_inning_batting = ipl_final_df.filter(func.col("inning") == 1 )

first_inning_batting.show(first_inning_batting.count(), truncate = False)
```

```
+--------+------+------------------+--------------------+----+----+------------------+------------+-----------+-----------+----------+----------+
|match_id|inning|batting_team      |bowling_team        |over|ball|batter            |bowler      |non_striker|batsman_runs|extra_runs|total_runs|
+--------+------+------------------+--------------------+----+----+------------------+------------+-----------+-----------+----------+----------+
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|0   |1   |Abhishek Sharma   |MA Starc    |NULL       |0          |0         |0         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|0   |2   |Abhishek Sharma   |MA Starc    |NULL       |0          |0         |0         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|0   |3   |Abhishek Sharma   |MA Starc    |NULL       |0          |0         |0         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|0   |4   |Abhishek Sharma   |MA Starc    |NULL       |2          |0         |2         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|0   |5   |Abhishek Sharma   |MA Starc    |NULL       |0          |0         |0         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|0   |6   |RA Tripathi       |MA Starc    |NULL       |1          |0         |1         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|1   |1   |RA Tripathi       |VG Arora    |NULL       |0          |0         |0         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|1   |2   |RA Tripathi       |VG Arora    |NULL       |0          |0         |0         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|1   |3   |RA Tripathi       |VG Arora    |NULL       |0          |0         |0         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|1   |4   |RA Tripathi       |VG Arora    |NULL       |0          |2         |2         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|1   |5   |RA Tripathi       |VG Arora    |NULL       |1          |0         |1         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|1   |6   |TM Head           |VG Arora    |NULL       |0          |0         |0         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|2   |1   |RA Tripathi       |MA Starc    |NULL       |4          |0         |4         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|2   |2   |RA Tripathi       |MA Starc    |NULL       |1          |0         |1         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|2   |3   |AK Markram        |MA Starc    |NULL       |0          |0         |0         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|2   |4   |AK Markram        |MA Starc    |NULL       |4          |0         |4         |
|1426312 |1     |Sunrisers Hyderabad|Kolkata Knight Riders|2   |5   |AK Markram        |MA Starc    |NULL       |0          |0         |0         |
```

#Find run scored by each batsman

```
batsman_run_df = first_inning_batting.filter(func.col("extras_type").isNull()
).groupBy(func.col("batter")).agg(func.sum("batsman_runs").alias("R"))

batsman_run_df.show()
```

```
#Find run scored by each batsman
batsman_run_df = first_inning_batting.groupBy(func.col("batter")).agg(func.sum("batsman_runs").alias("R"))

batsman_run_df.show()
```

```
+------------------+---+
|            batter|  R|
+------------------+---+
|           B Kumar|  0|
|         JD Unadkat|  4|
|          H Klaasen| 16|
|            TM Head|  0|
|        Abdul Samad|  4|
|    Abhishek Sharma|  2|
|Nithish Kumar Reddy| 13|
|      Shahbaz Ahmed|  8|
|         AK Markram| 20|
|         PJ Cummins| 24|
|        RA Tripathi|  9|
+------------------+---+
```

#Find balls faced by each batsman

```
ball_faced_df =
first_inning_batting.groupBy(func.col("batter")).agg(func.count("ball").alias("balls_faced"))
ball_faced_df.show()
```

```
#Find balls faced by each batsman
ball_faced_df = first_inning_batting.groupBy(func.col("batter")).agg(func.count("ball").alias("balls_faced"))
ball_faced_df.show()
```

```
+------------------+-----------+
|            batter|balls_faced|
+------------------+-----------+
|           B Kumar|          1|
|        JD Unadkat|         11|
|         H Klaasen|         17|
|           TM Head|          1|
|       Abdul Samad|          5|
|   Abhishek Sharma|          5|
|Nithish Kumar Reddy|         10|
|      Shahbaz Ahmed|          7|
|         AK Markram|         25|
|         PJ Cummins|         19|
|       RA Tripathi|         16|
+------------------+-----------+
```

```
#Find out number of bounderies hitted by each batsman
first_inning_batting.groupBy(func.col("batter")).agg(  func.count( func.when( func.col("batsman_runs") == 4 , 1 ) ).alias("4s") ) .show()
```

```
#Find out number of bounderies hitted by each batsman
first_inning_batting.groupBy(func.col("batter")).agg(  func.count( func.when( func.col("batsman_runs") == 4 , 1 ) ).alias("4s") ) .show()
```

```
+------------------+---+
|            batter| 4s|
+------------------+---+
|           B Kumar|  0|
|        JD Unadkat|  0|
|         H Klaasen|  1|
|           TM Head|  0|
|       Abdul Samad|  0|
|   Abhishek Sharma|  0|
|Nithish Kumar Reddy|  1|
|      Shahbaz Ahmed|  0|
|         AK Markram|  3|
|         PJ Cummins|  2|
|       RA Tripathi|  1|
+------------------+---+
```

```
#Find out number of 6s hitted by each batsman
first_inning_batting.groupBy(func.col("batter")).agg(  func.count( func.when( func.col("batsman_runs") == 6 , 1 ) ).alias("6s") ) .show()
```

```
+------------------+---+
|            batter| 6s|
+------------------+---+
|           B Kumar|  0|
|        JD Unadkat|  0|
|         H Klaasen|  0|
|           TM Head|  0|
|       Abdul Samad|  0|
|   Abhishek Sharma|  0|
|Nithish Kumar Reddy|  1|
|      Shahbaz Ahmed|  1|
|         AK Markram|  0|
|         PJ Cummins|  1|
|       RA Tripathi|  0|
+------------------+---+
```

```
#Let's build final scorecard now
Scorecard_df = first_inning_batting.filter(func.col("extras_type").isNull()
).groupBy(func.col("batter")).agg(func.sum("batsman_runs").alias("R"),
        func.count("ball").alias("balls"),
        func.count( func.when( func.col("batsman_runs") == 4 , 1 ) ).alias("4s"),
        func.count( func.when( func.col("batsman_runs") == 6 , 1 ) ).alias("6s"),
        func.round(func.sum(func.col("batsman_runs"))*100 / func.count("ball") , 2).alias("S/R")
```

```
    )
```

```
#Let's find batsman order
batsman_order = first_inning_batting.withColumn("over-ball", ( func.concat(
func.col("over"),func.lit("."),func.col("ball") ) )
.cast('Float')).groupBy("batter").agg(func.min("over-ball").alias("order")).orderBy("order")

batsman_order.show()
```

```
#Let's find batsman order
batsman_order = first_inning_batting.withColumn("over-ball", ( func.concat( func.col("over"),func.lit("."),func.col("ball") ) )
.cast('Float')).groupBy("batter").agg(func.min("over-ball").alias("order")).orderBy("order")

batsman_order.show()
```

```
+------------------+-----+
|            batter|order|
+------------------+-----+
|    Abhishek Sharma|  0.1|
|       RA Tripathi|  0.6|
|           TM Head|  1.6|
|         AK Markram|  2.3|
|Nithish Kumar Reddy|  4.3|
|          H Klaasen|  7.3|
|      Shahbaz Ahmed| 10.3|
|        Abdul Samad| 11.6|
|         PJ Cummins| 12.6|
|         JD Unadkat| 14.2|
|            B Kumar| 17.6|
+------------------+-----+
```

```
batting_order_df = batsman_order.withColumn("batting_order" ,
func.row_number().over(Window.orderBy("order"))  )

batting_order_df.show()
```

```
batting_order_df = batsman_order.withColumn("batting_order" , func.row_number().over(Window.orderBy("order"))  )

batting_order_df.show()
```

```
+------------------+-----+-------------+
|            batter|order|batting_order|
+------------------+-----+-------------+
|    Abhishek Sharma|  0.1|            1|
|       RA Tripathi|  0.6|            2|
|           TM Head|  1.6|            3|
|         AK Markram|  2.3|            4|
|Nithish Kumar Reddy|  4.3|            5|
|          H Klaasen|  7.3|            6|
|      Shahbaz Ahmed| 10.3|            7|
|        Abdul Samad| 11.6|            8|
|         PJ Cummins| 12.6|            9|
|         JD Unadkat| 14.2|           10|
|            B Kumar| 17.6|           11|
+------------------+-----+-------------+
```

```
#Let's build final scorecard now
Scorecard_df = first_inning_batting.filter(func.col("extras_type").isNull() ).groupBy(func.col("batter")).agg(func.sum("batsman_runs").alias("R"),
          func.count("ball").alias("balls"),
          func.count( func.when( func.col("batsman_runs") == 4 , 1 ) ).alias("4s"),
          func.count( func.when( func.col("batsman_runs") == 6 , 1 ) ).alias("6s"),
          func.round(func.sum(func.col("batsman_runs"))*100 / func.count("ball") , 2).alias("S/R")
        )
```

```
Scorecard_df.show()
```

```
+------------------+---+-----+---+---+------+
|            batter|  R|balls| 4s| 6s|   S/R|
+------------------+---+-----+---+---+------+
|           B Kumar|  0|    1|  0|  0|   0.0|
|        JD Unadkat|  4|   11|  0|  0| 36.36|
|         H Klaasen| 16|   17|  1|  0| 94.12|
|           TM Head|  0|    1|  0|  0|   0.0|
|       Abdul Samad|  4|    4|  0|  0| 100.0|
|   Abhishek Sharma|  2|    5|  0|  0|  40.0|
|Nithish Kumar Reddy| 13|   10|  1|  1| 130.0|
|     Shahbaz Ahmed|  8|    7|  0|  1|114.29|
|        AK Markram| 20|   23|  3|  0| 86.96|
|        PJ Cummins| 24|   17|  2|  1|141.18|
|       RA Tripathi|  9|   12|  1|  0|  75.0|
+------------------+---+-----+---+---+------+
```

Scorecard_df.join(batting_order_df, on = ["batter", "batter"], how = "inner").show()

```
Scorecard_df.join(batting_order_df, on = ["batter", "batter"], how = "inner").show()
```

```
+------------------+------------------+---+-----+---+---+------+-----+-------------+
|            batter|            batter|  R|balls| 4s| 6s|   S/R|order|batting_order|
+------------------+------------------+---+-----+---+---+------+-----+-------------+
|   Abhishek Sharma|   Abhishek Sharma|  2|    5|  0|  0|  40.0|  0.1|            1|
|       RA Tripathi|       RA Tripathi|  9|   12|  1|  0|  75.0|  0.6|            2|
|           TM Head|           TM Head|  0|    1|  0|  0|   0.0|  1.6|            3|
|        AK Markram|        AK Markram| 20|   23|  3|  0| 86.96|  2.3|            4|
|Nithish Kumar Reddy|Nithish Kumar Reddy| 13|   10|  1|  1| 130.0|  4.3|            5|
|         H Klaasen|         H Klaasen| 16|   17|  1|  0| 94.12|  7.3|            6|
|     Shahbaz Ahmed|     Shahbaz Ahmed|  8|    7|  0|  1|114.29| 10.3|            7|
|       Abdul Samad|       Abdul Samad|  4|    4|  0|  0| 100.0| 11.6|            8|
|        PJ Cummins|        PJ Cummins| 24|   17|  2|  1|141.18| 12.6|            9|
|        JD Unadkat|        JD Unadkat|  4|   11|  0|  0| 36.36| 14.2|           10|
|           B Kumar|           B Kumar|  0|    1|  0|  0|   0.0| 17.6|           11|
+------------------+------------------+---+-----+---+---+------+-----+-------------+
```

Scorecard_df.join(batting_order_df, ["batter", "batter"] , how = "inner").select(["batter","R","B","4s","6s","S/R","batting_order"]).show()

```
Scorecard_df.join(batting_order_df,  ["batter", "batter"] , how = "inner").select(["batter","R","B","4s","6s","S/R","batting_order"]).show()
```

```
+------------------+---+---+---+---+------+-------------+
|            batter|  R|  B| 4s| 6s|   S/R|batting_order|
+------------------+---+---+---+---+------+-------------+
|   Abhishek Sharma|  2|  5|  0|  0|  40.0|            1|
|       RA Tripathi|  9| 12|  1|  0|  75.0|            2|
|           TM Head|  0|  1|  0|  0|   0.0|            3|
|        AK Markram| 20| 23|  3|  0| 86.96|            4|
|Nithish Kumar Reddy| 13| 10|  1|  1| 130.0|            5|
|         H Klaasen| 16| 17|  1|  0| 94.12|            6|
|     Shahbaz Ahmed|  8|  7|  0|  1|114.29|            7|
|       Abdul Samad|  4|  4|  0|  0| 100.0|            8|
|        PJ Cummins| 24| 17|  2|  1|141.18|            9|
|        JD Unadkat|  4| 11|  0|  0| 36.36|           10|
|           B Kumar|  0|  1|  0|  0|   0.0|           11|
+------------------+---+---+---+---+------+-------------+
```

```
#Let's build bowling scorecard now
first_inning_bowling_df = ipl_final_df.groupBy("bowler") \
.agg(
```

```
   func.count( func.when(func.coalesce(func.col("extras_type"), func.lit("XYZ")) != "wides"
,1)).alias("balls_bowled"), # find over using this /6 & this
   # logic func.concat(func.floor(func.count("over")/6 ), func.lit('.') ,func.count("over")%6  ) .alias("O") ,
   func.sum(func.when(func.col("extras_type").isNull(), func.col("batsman_runs"))  # add wide to this
            .when(func.col("extras_type") == "wides", func.col("extra_runs"))
        ).alias("R"),
   func.sum(func.when(func.col("is_wicket") == 1, 1)).alias("W")
   )
first_inning_bowling_df.show()
```

```
+-------------+------------+---+----+
|       bowler|balls_bowled|  R|   W|
+-------------+------------+---+----+
| Harshit Rana|          24| 24|   2|
|      B Kumar|          12| 25|NULL|
|   JD Unadkat|           6|  9|NULL|
|    SP Narine|          24| 16|   1|
|     MA Starc|          18| 14|   2|
|   AD Russell|          15| 19|   3|
|  T Natarajan|          12| 29|NULL|
|     VG Arora|          18| 24|   1|
|Shahbaz Ahmed|          15| 22|   1|
|   AK Markram|           6|  5|NULL|
|     CV Varun|          12|  9|   1|
|   PJ Cummins|          12| 18|   1|
+-------------+------------+---+----+
```

```
first_inning_scorecard_df = \
first_inning_bowling_df.select( "bowler",
   func.concat(func.floor(func.col("balls_bowled")/6 ), func.lit('.') ,func.col("balls_bowled")%6  )
.alias("O"),
                        func.col("R"),
                         "W",
                         func.round( (func.col("R") / func.col("O")) , 2). alias("Econ")
                        )
first_inning_scorecard_df.show()
```

```
first_inning_scorecard_df = \
first_inning_bowling_df.select( "bowler",
    func.concat(func.floor(func.col("balls_bowled")/6 ), func.lit('.') ,func.col("balls_bowled")%6  ) .alias("O"),
                                func.col("R"),
                                 "W",
                                 func.round( (func.col("R") / func.col("O")) , 2). alias("Econ")
                                )
first_inning_scorecard_df.show()
```

```
+-------------+---+---+----+----+
|       bowler|  O|  R|   W|Econ|
+-------------+---+---+----+----+
| Harshit Rana|4.0| 24|   2| 6.0|
|      B Kumar|2.0| 25|NULL|12.5|
|   JD Unadkat|1.0|  9|NULL| 9.0|
|    SP Narine|4.0| 16|   1| 4.0|
|     MA Starc|3.0| 14|   2|4.67|
|   AD Russell|2.3| 19|   3|8.26|
|  T Natarajan|2.0| 29|NULL|14.5|
|     VG Arora|3.0| 24|   1| 8.0|
|Shahbaz Ahmed|2.3| 22|   1|9.57|
|   AK Markram|1.0|  5|NULL| 5.0|
|     CV Varun|2.0|  9|   1| 4.5|
|   PJ Cummins|2.0| 18|   1| 9.0|
+-------------+---+---+----+----+
```

```
maiden_over_df =
first_inning_batting.groupBy("bowler","over").agg(func.sum(func.col("total_runs")).alias("runs"),
                                                  func.count(func.col("over")).alias("balls"))

maiden_over_df = maiden_over_df.filter((func.col("runs") == 0) & (func.col("balls") ==6 ))\
                               .groupBy("bowler").agg(func.count("bowler").alias("M"))

maiden_over_df.show()
```

```
#let's find Maiden over now

maiden_over_df = first_inning_batting.groupBy("bowler","over").agg(func.sum(func.col("total_runs")).alias("runs"),
                                                                   func.count(func.col("over")).alias("balls"))

maiden_over_df = maiden_over_df.filter((func.col("runs") == 0) & (func.col("balls") ==6 ))\
                               .groupBy("bowler").agg(func.count("bowler").alias("M"))

maiden_over_df.show()
```

```
+------------+---+
|      bowler|  M|
+------------+---+
|Harshit Rana|  1|
+------------+---+
```

```
first_inning_final_df = first_inning_scorecard_df.join(maiden_over_df, on = ["bowler","bowler"],
how="left").fillna(value=0)

first_inning_final_df.show()
```

bowler	O	R	W	Econ	M
Harshit Rana	4.0	24	2	6.0	1
B Kumar	2.0	25	0	12.5	0
JD Unadkat	1.0	9	0	9.0	0
SP Narine	4.0	16	1	4.0	0
MA Starc	3.0	14	2	4.67	0
AD Russell	2.3	19	3	8.26	0
T Natarajan	2.0	29	0	14.5	0
VG Arora	3.0	24	1	8.0	0
Shahbaz Ahmed	2.3	22	1	9.57	0
AK Markram	1.0	5	0	5.0	0
CV Varun	2.0	9	1	4.5	0
PJ Cummins	2.0	18	1	9.0	0