**SQL Script**

**create database cricket;**

**use cricket;**

***\*\* CREATE TABLE COMMAND \*\****

**CREATE TABLE match\_data (**

**match\_id INT,**

**season VARCHAR(255),**

**start\_date VARCHAR(255),**

**venue VARCHAR(255),**

**innings INT,**

**ball INT,**

**batting\_team VARCHAR(255),**

**bowling\_team VARCHAR(255),**

**striker VARCHAR(255),**

**non\_striker VARCHAR(255),**

**bowler VARCHAR(255),**

**runs\_off\_bat INT,**

**extras INT,**

**wides INT,**

**noballs INT,**

**byes INT,**

**legbyes INT,**

**penalty INT,**

**wicket\_type VARCHAR(255),**

**player\_dismissed VARCHAR(255),**

**other\_wicket\_type VARCHAR(255),**

**cricsheet\_id INT**

**);**

***\*\* CONVERTING DATE COLUMN TO DATE FORMAT \*\****

**update match\_data**

**set start\_date = str\_to\_date(start\_date, '%d-%m-%Y');**

**set global local\_infile=true;**

**SET GLOBAL local\_infile=1;**

***\*\* LOAD DATA TO MATCH\_DATA\*\****

**LOAD DATA LOCAL INFILE "C:/Users/jwala/Downloads/Interview Assignment- Data Analyst/match\_data.csv"**

**INTO TABLE match\_data**

**COLUMNS TERMINATED BY ','**

**OPTIONALLY ENCLOSED BY '"'**

**ESCAPED BY '"'**

**LINES TERMINATED BY '\n'**

**IGNORE 1 LINES;**

***\*\* CREATE TABLE COMMAND FOR match\_info TABLE \*\****

**CREATE TABLE match\_info (**

**id INT,**

**season VARCHAR(255),**

**city VARCHAR(255),**

**match\_date VARCHAR(255),**

**team1 VARCHAR(255),**

**team2 VARCHAR(255),**

**toss\_winner VARCHAR(255),**

**toss\_decision VARCHAR(255),**

**result VARCHAR(255),**

**dl\_applied int,**

**winner VARCHAR(255),**

**win\_by\_runs INT,**

**win\_by\_wickets INT,**

**player\_of\_match VARCHAR(255),**

**venue VARCHAR(255),**

**umpire1 VARCHAR(255),**

**umpire2 VARCHAR(255),**

**umpire3 VARCHAR(255)**

**);**

**SELECT \* FROM match\_info;**

***\*\* LOAD DATA INTO match\_info TABLE\*\****

**LOAD DATA LOCAL INFILE "C:/Users/jwala/Downloads/Interview Assignment- Data Analyst/match\_info\_data.csv"**

**INTO TABLE match\_info**

**COLUMNS TERMINATED BY ','**

**OPTIONALLY ENCLOSED BY '"'**

**ESCAPED BY '"'**

**LINES TERMINATED BY '\n'**

**IGNORE 1 LINES;**

**update match\_info**

**set match\_date = str\_to\_date(match\_date, '%d-%m-%Y');**

**-- 1. Batter**

**## overrall top 10 batsman**

**DROP view top\_10\_batsman\_overrall;**

**create view top\_10\_batsman\_overrall AS**

**SELECT ipl\_bat.\* FROM**

**(**

**select striker,**

**count(distinct match\_id) as total\_matches\_played,**

**sum(runs\_off\_bat) as total\_runs\_scored,**

**ceil(sum(runs\_off\_bat) /count(distinct match\_id)) as batting\_avg,**

**dense\_rank() OVER(order by sum(runs\_off\_bat) DESC) AS run\_rnk**

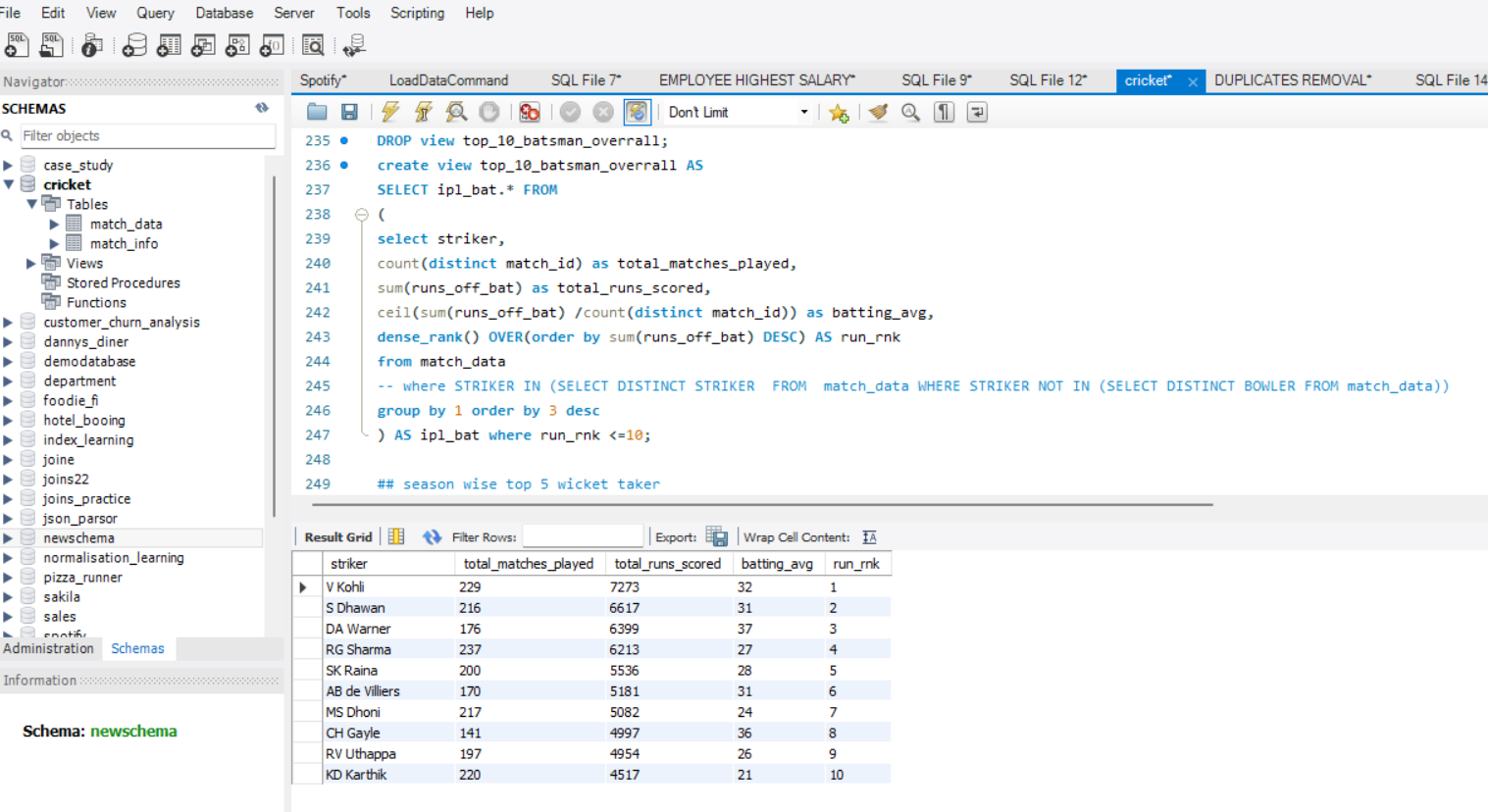
**from match\_data**

**-- where STRIKER IN (SELECT DISTINCT STRIKER FROM match\_data WHERE STRIKER NOT IN (SELECT DISTINCT BOWLER FROM match\_data))**

**group by 1 order by 3 desc**

**) AS ipl\_bat where run\_rnk <=10;**

**\*\* OUTPUT\*\***

****

**-- #############################################################################**

**-- 2. Bowler**

**-- ## overrall top 10 bowler**

**DROP VIEW top\_10\_bowler\_overrall;**

**create view top\_10\_bowler\_overrall AS**

**SELECT ipl\_wick.\* FROM**

**(**

**select bowler,**

**count(distinct match\_id) as total\_matches\_played,**

**count(case when wicket\_type is not NULL then 1 end) as total\_wickets\_taken,**

**ceil(count(case when wicket\_type is not NULL then 1 end) /count(distinct match\_id)) as wick\_avg,**

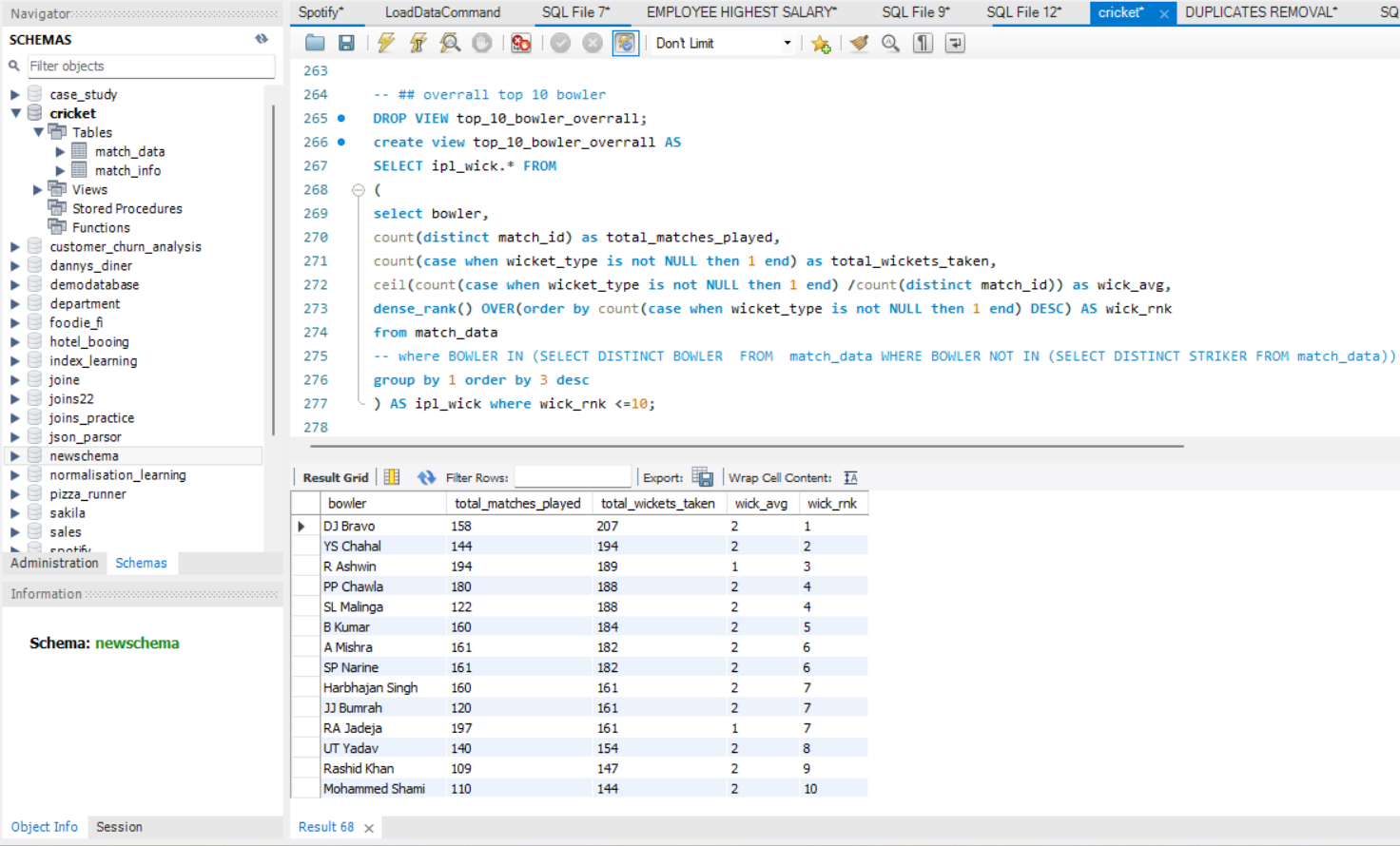
**dense\_rank() OVER(order by count(case when wicket\_type is not NULL then 1 end) DESC) AS wick\_rnk**

**from match\_data**

**-- where BOWLER IN (SELECT DISTINCT BOWLER FROM match\_data WHERE BOWLER NOT IN (SELECT DISTINCT STRIKER FROM match\_data))**

**group by 1 order by 3 desc**

**) AS ipl\_wick where wick\_rnk <=10;**

****

**-- 3. Batting all-rounder**

**SELECT**

**m1.striker, COUNT(DISTINCT m1.match\_id) as total\_match\_played,**

**COALESCE(SUM(m1.runs\_off\_bat), 0) AS total\_runs,**

**COALESCE(wickets\_taken.wicket\_count, 0) AS wickets\_taken,**

**ceil(COALESCE(SUM(m1.runs\_off\_bat), 0)/COUNT(DISTINCT m1.match\_id)) as batting\_avg,**

**ceil(COALESCE(wickets\_taken.wicket\_count, 0)/COUNT(DISTINCT m1.match\_id)) as wicket\_avg**

**FROM**

**match\_data m1**

**LEFT JOIN (**

**SELECT**

**bowler,**

**COUNT(wicket\_type) AS wicket\_count**

**FROM**

**match\_data**

**WHERE**

**wicket\_type IS NOT NULL**

**GROUP BY**

**bowler**

**) wickets\_taken ON m1.striker = wickets\_taken.bowler**

**GROUP BY**

**m1.striker**

**having wickets\_taken > (SELECT count(case when wicket\_type is not NULL then 1 end)/ COUNT(DISTINCT BOWLER) AS avg\_wkt FROM match\_data)**

**order by total\_match\_played desc, total\_runs desc, wickets\_taken desc, batting\_avg desc, wicket\_avg desc;**

**A screenshot of a computer

Description automatically generated**

**-- 4. Bowling all-rounder**

**SELECT m1.bowler,**

**COUNT(m1.wicket\_type) AS wicket\_count ,**

**runs\_scored.total\_runs AS run\_scored**

**FROM**

**match\_data m1**

**LEFT JOIN (**

**SELECT striker,**

**COALESCE(SUM(runs\_off\_bat), 0) AS total\_runs**

**FROM match\_data**

**GROUP BY striker**

**) runs\_scored ON m1.bowler = runs\_scored.striker**

**GROUP BY m1.bowler**

**order by wicket\_count desc, run\_scored desc;**

**A screenshot of a computer

Description automatically generated**