**TESTVAGRANT ASSIGNMENT**

**readme File:**

PROBLEM TO SOLVE :

1. Create a small data structure that holds details - Name of the team, points they have earned and result of last 5 matches as per above image:

Description:

Firstly we created two Tables a) Team\_Name : It have Team Id, Team Name ,Total Matches played and Total Points earned.

Query to Make Table:

1)

CREATE TABLE Team\_Name(

TeamID int NOT NULL,

TeamName varchar(255) NOT NULL,

TotalMatches int NOT NULL,

Points int NOT NULL,

PRIMARY KEY(TeamID)

);

INSERT INTO Team\_Name (TeamID,TeamName,TotalMatches,Points)

VALUES (1,'GT',14,20),

(2,'RR',14,18),

(3,'LSG',14,18),

(4,'RCB',14,16),

(5,'DC',14,14),

(6,'PBKS',14,14),

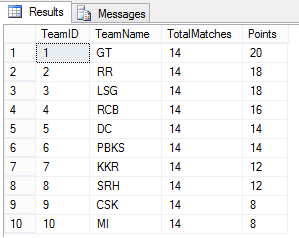
(7,'KKR',14,12),

(8,'SRH',14,12),

(9,'CSK',14,8),

(10,'MI',14,8);

SELECT \* FROM Team\_Name



Description:

b)Team\_Matches :It have Team Id, Match Id and Is Win column. Match Id refers to the last 5 matches played by the team and Is Win refers to the result of the match played by the team. 1 means the team have won their corresponding match and 0 means the team have lose their corresponding match.

2)

CREATE TABLE Team\_Matches(

TeamID int NOT NULL,

MatchID int NOT NULL,

IsWin int NOT NULL

);

INSERT INTO Team\_Matches (TeamID,MatchID,IsWin)

VALUES (1,1,0),(1,2,1),(1,3,1),(1,4,0),(1,5,0),

(2,1,1),(2,2,1),(2,3,0),(2,4,1),(2,5,0),

(3,1,1),(3,2,0),(3,3,0),(3,4,1),(3,5,1),

(4,1,1),(4,2,0),(4,3,1),(4,4,1),(4,5,0),

(5,1,0),(5,2,1),(5,3,1),(5,4,0),(5,5,1),

(6,1,1),(6,2,0),(6,3,1),(6,4,0),(6,5,1),

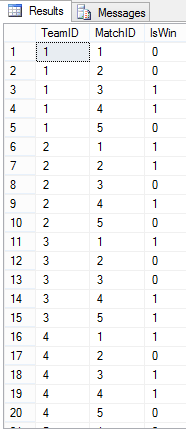
(7,1,0),(7,2,1),(7,3,1),(7,4,0),(7,5,1),

(8,1,0),(8,2,1),(8,3,0),(8,4,0),(8,5,0),

(9,1,0),(9,2,0),(9,3,0),(9,4,1),(9,5,0),

(10,1,1),(10,2,0),(10,3,1),(10,4,0),(10,5,1);

SELECT \* FROM Team\_Matches



1. Programmatically retrieve the teams that have 2 consecutive losses.

Description:

Here we have declared two params a) @ConsecutiveNumber: It states that the how much Consecutive number of matches particular team has won or lost. b)@IsWin : It states that the the team has won or lost the matches(0 for loss and 1 for Win).

So as the question statement is given we want to get the teams that are having consecutive 2 losses so in param @ConsecutiveNumber we have put value 2 and in @IsWin we have put the value 0 so we want the teams that have lost.

declare @ConsecutiveNumber INT=2, @IsWin BIT=0

;with Result1 as

(

SELECT DISTINCT tn.TeamName,Table1.TeamId ,IsWin, COUNT(Table1.TEAMID) OVER (PARTITION BY Table1.TEAMID,A-B,ISWIN) AS C\_Result FROM

(

SELECT TeamId,MatchID,IsWin,ROW\_NUMBER() OVER (PARTITION BY TeamID ORDER BY MatchID) AS A,

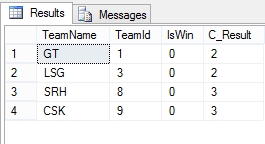
ROW\_NUMBER() OVER (PARTITION BY TeamID,ISWIN ORDER BY MatchID) AS B

FROM Team\_Matches

) AS Table1 inner join Team\_Name TN on Table1.TeamID=TN.TeamID

)

SELECT \* FROM Result1 WHERE C\_Result>=@ConsecutiveNumber AND IsWin=@IsWin



1. Generalize the same solution, so that we could get teams that have n consecutive losses/wins.

Description:

Here we have declared two params a) @ConsecutiveNumber: It states that the how much Consecutive number of matches particular team has won or lost. b)@IsWin : It states that the the team has won or lost the matches(0 for loss and 1 for Win).

This is the generalized query for getting the teams which have consecutively lost or won the games which can be n number upto 5 because we have result of last 5 matches of all teams.

Then we are running subquery to get the desired output and having join on our procedure with our table Team\_Name to retrieve the name of the team we are looking for having n consecutive wins /losses.

Also if we are looking for teams with consecutive 3 losses we are also getting the team which also have lost 2 consecutive matches.

declare @ConsecutiveNumber INT=n1, @IsWin BIT=n2

;with Result1 as

(

SELECT DISTINCT tn.TeamName,Table1.TeamId ,IsWin, COUNT(Table1.TEAMID) OVER (PARTITION BY Table1.TEAMID,A-B,ISWIN) AS C\_Result FROM

(

SELECT TeamId,MatchID,IsWin,ROW\_NUMBER() OVER (PARTITION BY TeamID ORDER BY MatchID) AS A,

ROW\_NUMBER() OVER (PARTITION BY TeamID,ISWIN ORDER BY MatchID) AS B

FROM Team\_Matches

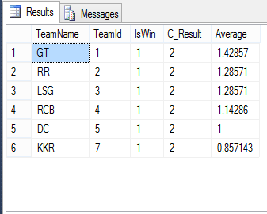
) AS Table1 inner join Team\_Name TN on Table1.TeamID=TN.TeamID

)

SELECT \* FROM Result1 WHERE C\_Result>=@ConsecutiveNumber AND IsWin=@IsWin

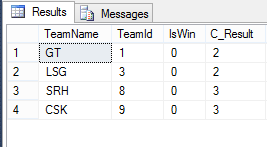
Screenshot Description:

Here we have put @ConsecutiveNumber=2 and @IsWin=1.So we are getting all the teams which have consecutively won their 2 matches.



Screenshot Description:

Here we have put @ConsecutiveNumber=2 and @IsWin=0.So we are getting all the teams which have consecutively lost their 2 matches. And also the teams with consecutive 3 losses as they have also lost their consecutive 2 matches.



1. Once the above is done, Calculate the average points of these filtered teams:

Description:

Using below query first we created a temporary table #AvgDetailss and then we inserted the average of the particular team by dividing Points to the Total Matches and the we joined the temporary table created to the Team\_Name table so we can get Average in one table only.

1)

CREATE TABLE #AvgDetailss (TeamID int,Average VARCHAR(25))

insert into #AvgDetailss

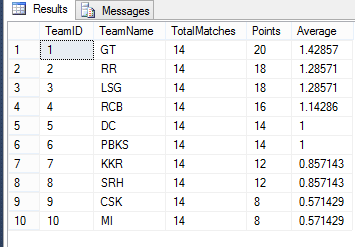
select TeamID, CAST(Points AS FLOAT)/TotalMatches AS Average from Team\_Name

select \* from #AvgDetailss

UPDATE c

SET c.Average = a.Average

FROM Team\_Name c inner join #AvgDetailss a on c.TeamID = a.TeamID



Description:

We just here added the Average column from Team\_Name table using Inner Join and displayed the Average of filtered teams from consective wins/losses.

2)

declare @ConsecutiveNumber INT=2, @IsWin BIT=0

;with Result1 as

(

SELECT DISTINCT tn.TeamName,Table1.TeamId ,IsWin, COUNT(Table1.TEAMID) OVER (PARTITION BY Table1.TEAMID,A-B,ISWIN) AS C\_Result,tn.Average FROM

(

SELECT TeamId,MatchID,IsWin,ROW\_NUMBER() OVER (PARTITION BY TeamID ORDER BY MatchID) AS A,

ROW\_NUMBER() OVER (PARTITION BY TeamID,ISWIN ORDER BY MatchID) AS B

FROM Team\_Matches

) AS Table1 inner join Team\_Name TN on Table1.TeamID=TN.TeamID

)

SELECT \* FROM Result1 WHERE C\_Result>=@ConsecutiveNumber AND IsWin=@IsWin

