**Project 3: IPL Analysis using SQL**

The Indian Premier League (IPL) is one of the most popular cricket leagues in the world. A dataset containing information about IPL matches and player statistics is available for analysis. As a data analyst with SQL expertise, your objective is to perform data analysis on the IPL dataset to gain insights into player performance, team dynamics, and match outcomes.

Segment 1: Database - Tables, Columns, Relationships

* Identify the tables in the dataset and their respective columns.
* Determine the number of rows in each table within the schema.
* Handle any missing or inconsistent values in the dataset.

Segment 2: Team Analysis

* Determine the number of matches played by each team in the dataset.
* Analyse the win-loss ratio for each team in IPL history.
* Calculate the average total runs scored by each team in all matches.
* Identify the team with the highest number of sixes in a single season.
* Determine the team that has won the most IPL titles.

Segment 3: Player Performance Analysis

* Identify the top five players with the most runs scored in IPL history.
* Determine the average strike rate for batsmen who have played at least 50 matches.
* Analyse the distribution of player dismissals (caught, bowled, etc.) in the dataset.
* Calculate the average economy rate for bowlers who have bowled at least 100 overs.
* Identify the top three bowlers with the most wickets in a single season.

Segment 4: Segment 4: Match Analysis

* Determine the total number of matches played in each season.
* Analyse the distribution of match outcomes for each team (wins, ties, no results) in the dataset.
* Calculate the average winning margin (runs or wickets) for all matches.
* Identify the top three venues with the highest average runs scored per match.
* Determine the team that has won the most matches by a narrow margin (less than 10 runs or 2 wickets).

Segment 5: Player Performance Comparison

* Compare the batting performance of players in home matches versus away matches and identify any significant differences.
* Analyse the bowling performance of players against left-handed batsmen versus right-handed batsmen and identify any performance variations.
* Identify the players who have shown consistent improvement in their performance metrics over multiple IPL seasons.

Segment 6: Team Dynamics and Strategy

* Analyse the relationship between a team's batting order and their overall run rate in matches.
* Determine the effectiveness of teams in successfully chasing targets in different match scenarios (e.g., high target, low target, tight finish).
* Identify the teams that have shown the most effective use of powerplay overs and analyse its impact on their match results.
* Analyse the distribution of match outcomes (wins, losses, ties) based on the team batting first or second. Identify any patterns or trends that could provide insights into successful match strategies for teams.
* Which IPL season had the highest overall run rate? Analyze the factors contributing to the high-scoring matches and the impact on viewership and team strategies.

Segment 7: SQL Concepts

* Use subqueries to find the players who have scored more than 500 runs in a single season.
* Implement joins to retrieve the player information along with their team details.
* Utilise aggregate functions to calculate the average strike rate for each team.
* Apply window functions to rank the teams based on their total runs scored in a season.
* Create stored procedures to calculate the net run rate for each team in a specific season.
* Identify the tables and columns that should be indexed to improve query performance.
* Analyse the impact of adding or removing indexes on query execution time.
* Evaluate the performance improvement of queries after using common table expressions (CTEs).
* Identify any potential bottlenecks in the database schema and suggest optimizations to mitigate them.

Evaluation pointers:

* The tasks are correctly identified and executed.
* The solution output matches the expected output.
* The query is optimised and syntactically correct.
* Proper aliases are used
* If required any, appropriate comments are written.
* The code is written concisely with appropriate indentations.