**Business problem**

The Major League Baseball (MLB) has been facing a decline in popularity in recent years. Our objective is to design a comprehensive database that not only helps maximize the revenue of the MLB but also helps in making data-driven decisions to improve the league's popularity. Thus, the MLB needs a centralized and structured database to efficiently manage its complex operations.

Here’s how our designed database can help also provide insights to strategize marketing campaigns, improve fan engagement, optimize operational costs, and enhance the league’s overall popularity.

**Table Design & Description**

I. Team and Personnel

1. players Table:

* Purpose: Stores information about each player in the league.
* Fields:
* players\_id: Unique identifier for each player (PK)
* team\_id: Foreign key referencing the team the player is associated with. (FK)
* last\_name & first\_name: The player's name.
* play\_position: The position the player plays in.
* age: Age of the player.
* years\_pro: Professional experience in years.
* salary: Salary of the player.
* nationality: Nationality of the player.
* contract\_start\_date & contract\_end\_date: Contract duration.

1. employees Table:

* Purpose: Contains details of employees working for the league.
* Fields:
* employee\_id: Unique identifier for each employee. (PK)
* department\_id: Foreign key referencing the department the employee works in. (FK)
* last\_name & first\_name: Employee names.
* job\_position: Position of the employee.
* age & years\_employed: Age and tenure.
* salary: Salary of the employee.

1. departments Table:

* Purpose: Keeps track of different departments within the league.
* Fields:
* department\_id: Unique identifier for each department. (PK)
* department\_name: Name of the department.
* budget: Budget allocated to the department.

1. teams Table:

* Purpose: Stores information about the teams in the league.
* Fields:
* team\_id: Unique identifier. (PK)
* team\_name: Name of the team.
* conference: Conference to which the team belongs.
* city: City of the team.
* founded\_year: Year when the team was founded.
* club\_owner: Owner of the team.
* stadium\_id: Foreign key for the stadium associated with the team. (FK)

II. Performance and Scheduling

1. team\_statistics Table:

* Purpose: Stores seasonal statistics for each team.
* Fields:
* team\_id & season\_year: Composite primary key. (PK)
* wins, losses, game\_ties: Performance metrics.
* points\_for, points\_against: Scoring metrics.
* profit: Profit made during the season.
* avg\_ticket\_sales & avg\_viewship: Audience metrics.

1. schedules Table:

* Purpose: Contains the schedule of matches.
* Fields:
* match\_id: Unique identifier for the match. (PK)
* match\_date: Date of the match.
* home\_team\_id & away\_team\_id: Teams playing. (FK)
* stadium\_id: Location of the match. (FK)
* broadcasting\_id: Broadcast information. (FK)

III. Broadcasting

1. broadcasting\_platforms Table:

* Purpose: Holds data about where the matches are broadcasted.
* Fields:
* broadcasting\_id: Unique identifier. (PK)
* channel\_name: Name of the broadcasting channel.
* season: Season year.
* total\_view: Total viewership.

IV. Financials and Sponsorships

1. sponsors Table:

* Purpose: Information about sponsor deals.
* Fields:
* sponsor\_id: Unique identifier. (PK)
* sponsor\_name: Name of the sponsor.
* phone\_number & email\_address: Contact info.
* season: Season of sponsorship.
* amount: Amount sponsored.

1. league\_financials Table:

* Purpose: Captures the league's overall financial health.
* Fields:
* season: Season year. (PK)
* revenue & cost: Financial metrics.

1. ticket\_sales Table:

* Purpose: Tracks ticket sales per match.
* Fields:
* match\_id: Foreign key for match. (FK)
* ticket\_price & tickets\_sold: Ticketing metrics.

V. Marketing and Engagement

1. stadium Table:

* Purpose: Contains details about stadiums.
* Fields:
* stadium\_id: Unique identifier. (PK)
* stadium\_name: Name of the stadium.
* city & state: Location.
* capacity & age: Stadium metrics.
* team\_owner: Foreign key for team ownership.

1. marketing Table:

* Purpose: Stores data about various marketing campaigns.
* Fields:
* campaign\_id: Unique identifier. (PK)
* campaign\_type: Type of campaign.
* platform & clicks: Engagement metrics.
* cost: Cost of the campaign.
* season: Season of the campaign.

1. fan\_engagement Table:

* Purpose: Measures levels of fan engagement.
* Fields:
* season: Season year. (PK)
* social\_mentions & average\_attendance: Engagement metrics.

1. players\_social\_media Table:

* Purpose: Tracks social media metrics for players.
* Fields:
* player\_id: Foreign key for players.
* twitter\_followers & instagram\_followers: Social media metrics.

VI. Health and Safety

1. injured\_players Table:

* Purpose: Keeps track of injured players.
* Fields:
* injured\_id: Unique identifier. (PK)
* player\_id: Foreign key for players. (FK)
* injury\_type & injury\_status: Injury details.
* treatment\_cost: Cost of treatment.
* injured\_date & estimated\_date\_out: Injury timings.

**Table Relationships & Rationale**

1. Players Table

This table has relationships with three different tables. Firstly, it has a one-to-many relationship with the injured player's table since only one player ID can be entered into a row for the injured player's tables. Also, that player ID can appear many times since a player can get injured more than once. Secondly, it has a many-to-one relationship as many players can all be a part of one single team. Lastly, it has a one-to-one relationship with the player's social media table as each player has one set of social media accounts that are entered into the player's social media table.

1. Departments and Employees Tables

These two tables each have one relationship and it is with each other. Specifically, the employees table a many-to-one relationship with departments. We chose the many-to-one relationship because there are a large number of employees who could all be working together in the same department.

1. Teams Table

In total, the team's table has 5 relationships with 4 other tables. Firstly, it has a one-to-many relationship with the player's table because one team is made up of a large number of players. Secondly, it has a one-to-one relationship with the stadium table since each team has one home stadium that they will host games at. Thirdly, it has a one-to-many relationship with team statistics as team statistics are taken each season meaning a team ID can appear multiple times. Lastly, it has two one-to-many relationships with the schedules table because only one team ID can be used for the home or away team ID.

1. Team Statistics Table

The team statistics table only has one relationship which is a many-to-one relationship with the team table

1. Schedules Table

The schedules table has 5 relationships with 4 different tables. Firstly, it has a many-to-one relationship with the stadium table because one stadium can be used for many different matches. Secondly, it has a many-to-one relationship with the broadcasting platforms table since a broadcasting platform can be used for a number of different matches. Thirdly, it has a one-to-one relationship with the ticket sales table since ticket sales data will be taken for each match ID. Lastly, it has two many-to-one relationships with the teams table since a team ID will be used for both the home and away teams. Only one team ID can be in this column however a team ID can appear multiple times in different matches.

1. Broadcasting Platforms Table

This table has a one-to-many relationship with the schedules table because a single broadcasting platform can be used for multiple matches.

1. Sponsors & League Financials

These two tables have no relationships with any other tables

1. Ticket SalesTable

The ticket sales table has a one-to-one relationship with the schedules table as ticket sales will be grouped by each match ID.

1. Stadiums Table

The stadiums table has one relationship with the teams table which is a one-to-one relationship. Each stadium is owned by one team.

1. Marketing & Fan Engagement Tables

These two tables have no relationships with any other tables

1. Players social media Table

This table has a one-to-one relationship with players as each player can have one set of social media accounts

1. Injured Players Table

This table has a many-to-one relationship with the players column because a player can be injured multiple times but only one player ID can be taken in for each injury report.