**Overview:**

We are designing a database that contains information about all MLB pitchers entering the 2023 MLB season. This database will contain pitcher biographical information, statistical metrics, team information, injuries, all-star berths, contract information, and playoff information.

In terms of relationships, we want to be able to list pitchers based on a variety of attributes. For example, on top of this high-level main database, we would like to be able to sort for all pitchers on one specific team, say, the Yankees.

**Entity sets:**

- **Pitcher** - entities representing each pitcher in the league.

- Attributes: pitcher\_id (primary key), first\_name, last\_name, height, weight, DOB, team\_name, debut\_date, pitching\_hand, age(), years\_in\_mlb()

- Constraints: None of the attributes can be NULL, and birth\_date and debut\_date must be valid dates

- **Team** - entities representing each team in the league.

- Attributes: team\_name (primary key), city, state, 2022\_wins, manager\_name, league

- Constraints: None of the attributes can be NULL and 2022\_wins must be a non-negative value

- **Country** - entities representing pitchers’ home countries.

- Attributes: country\_name (primary key), college

- **Contract** - entities representing the contract details of each pitcher.

- Attributes: contract\_id (primary key), annual\_salary, start\_date, end\_date, years\_remaining()

- Constraints: None of the attributes can be NULL, start\_date and end\_date must be valid dates, and annual\_salary and years\_remaining must be non-negative values

- **Statistics** - entities representing the various stats of a pitcher over a given period of time.

- Attributes: statistic\_id (primary key), games\_played, wins, losses, ERA, WHIP, innings\_pitched, win\_loss\_ratio

- Constraints: None of the attributes can be NULL and games\_played, innings\_pitched, wins, losses, ERA, WHIP, and win\_loss\_ratio must be

non-negative values

- **Playoffs** - entities representing a pitcher’s playoff appearances.

- Attributes: playoff\_ID (primary key), games\_played, wins, losses

- Constraints: None of the attributes can be NULL and games\_played, wins, and losses must be non-negative values.

- **All-Star** - entities representing the All-Star selections for each pitcher.

- Attributes: all\_star\_id (primary key), all\_star\_selections

- Constraints: None of the attributes can be NULL and all\_star\_selections must be a non-negative value.

- **Injuries** - entities representing current injuries incurred by a pitcher.

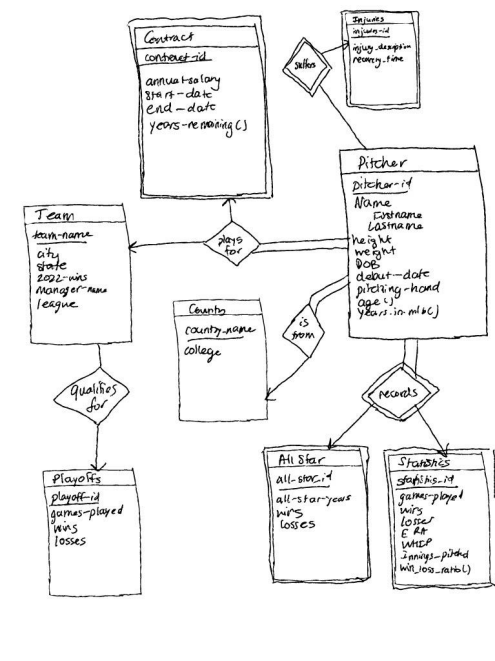
- Attributes: injury\_ID (primary key), injury\_description, recovery\_time - Constraints: Recovery\_time must be a non-negative value

**Relationship sets:**

- **Plays For:** a player must play for one team. A contract exists in this relationship as well. **- Records:** All pitchers record statistics, and may optionally also record all-star berths. **- Is From:** all pitchers are from one country.

**- Qualifies For:** a team may qualify for the playoffs (only one playoff bracket exists) **- Suffers:** a player may suffer an injury.

**ER Diagram**

****

**Data Plan**

We will be using datasets found on GitHub containing MLB pitcher information. These can either be imported to our database if possible, or can be manually inputted.

https://github.com/toddrob99/MLB-StatsAPI

This repository, for example, contains statistical information for all pitchers. There are many other similar repositories that will contain the information we need. If these are not sufficient, MLB.com and baseball-reference.com can provide any missed data.

**Web Front-End Option**

Interactions:

- Users can access the pitcher entity set and search for the statistics of any given pitcher - Users can also access a home country and teams to see the pitchers who belong to those sets

- Users can filter for multiple entities, for example, Yankee pitchers from the Dominican Republic

- Users can identify the injuries and all-star berths a player has

- Users can identify which teams qualified for the postseason

- Users can identify the contract details of a contract between a team and player

**Contingency Plan**

- If one of us drops the course, or something goes wrong, our shortened version of this database will simply contain current Yankees pitchers. This would remove the team entity, as well as the relationship “pitcher plays for team X,” and would significantly shorten the number of data entries in our database.

**SQL Schema:**

create table *pitcher*(

*pitcher\_id* **varchar(10)** primary key,

*first\_name* **varchar(20),**

*last\_name* **varchar(20),**

*height* **numeric,**

*weight* **numeric,**

*DOB* ***date,***

*debut\_date* ***date,***

*pitching\_hand* **varchar(10),**

*create function age()* **numeric,**

*create function years\_in\_mlb()* **numeric**

*foreign key team\_name references team* **varchar(10)**

*foreign key contract\_ID references contract* **varchar(10)**

*foreign key all\_star\_ID references all\_star* **varchar(10)**

*foreign key statistics\_ID references statistics* **varchar(10)**

*foreign key country\_name references country* **varchar(10)**

*foreign key injury\_ID references injuries* **varchar(10)** )

create table team(

*team\_name* **varchar(20)** primary key,

*city* **varchar(20),**

*state* **varchar(20),**

*2022\_wins* **numeric,**

*manager\_name* **varchar(20),**

*league* **varchar(2)**

*foreign key playoff\_ID references playoffs* **varchar(10)** )

create table country(

*country\_name* **varchar(20)** primary key,

*college* **varchar(20)**

)

create table contract(

*contract\_id* **varchar(10)** primary key,

*annual\_salary* **numeric,**

*start\_date* **date,**

end\_date **date,**

create function years\_remaining() **numeric,** *foreign key team\_name references team* **varchar(10)** )

create table statistics(

*statistics\_id* **varchar(10)** primary key,

*games\_played* **numeric,**

*wins* **numeric,**

*losses* **numeric,**

*ERA* **numeric(2,2),**

*WHIP* **numeric(2,2),**

*innings\_pitched* **numeric,**

*create function win\_loss\_ratio()* **numeric**

)

create table playoffs(

*playoff\_ID* **varchar(10)** primary key,

*games\_played* **numeric,**

*wins* **numeric,**

*losses* **numeric**

)

create table all\_star(

*all\_star\_ID* **varchar(10)** primary key

*all\_star\_years* **varchar(10),**

*wins* **numeric,**

*losses* **numeric**

)

create table injuries(

*injury\_ID* **varchar(10)** primary key,

*injury\_description* **varchar(50),**

recovery\_time **numeric**

)

create table plays\_for(

*pitcher\_ID,team\_name,contract\_ID* **varchar(30)** primary key, *foreign key pitcher\_id references pitcher NOT NULL* **varchar(10)**, *foreign key team\_name references team* **varchar(20),**

*foreign key contract\_id references contract* **varchar(10)**

)

create table is\_from(

*pitcher\_ID,country\_name* **varchar(30)** primary key, *foreign key pitcher\_id references pitcher NOT NULL* **varchar(10),** *foreign key country\_name references country* **varchar(20)** )

create records(

*pitcher\_ID,statistics\_ID,all\_star\_ID* **varchar(50)** primary key, *foreign key pitcher\_id references pitcher NOT NULL* **varchar(10),**

*foreign key statistics\_id references statistics* **varchar(10),**

*foreign key all\_star\_id references all\_star* **varchar(10)**

**)**

create qualifies\_for(

*team\_ID,playoff\_ID* **varchar(30)** primary key,

*foreign key team\_name references team* **varchar(20),**

*foreign key playoffs\_id references playoffs* **varchar(10)**

**)**

create suffers(

*pitcher\_ID,injury\_ID* **varchar(20)** primary key,

*foreign key injury\_id references injuries* **varchar(10),**

*foreign key pithcer\_id references pitcher* **varchar(10)**

**)**

**Revisions:**

- Modified our proposal structure based on the instructions document. - Incorporated a structured ER diagram and elaborated on our data plan. - Adjusted some entity sets and relationships based on feedback - Created schema based on these new entities/relationships - Declared primary and foreign keys