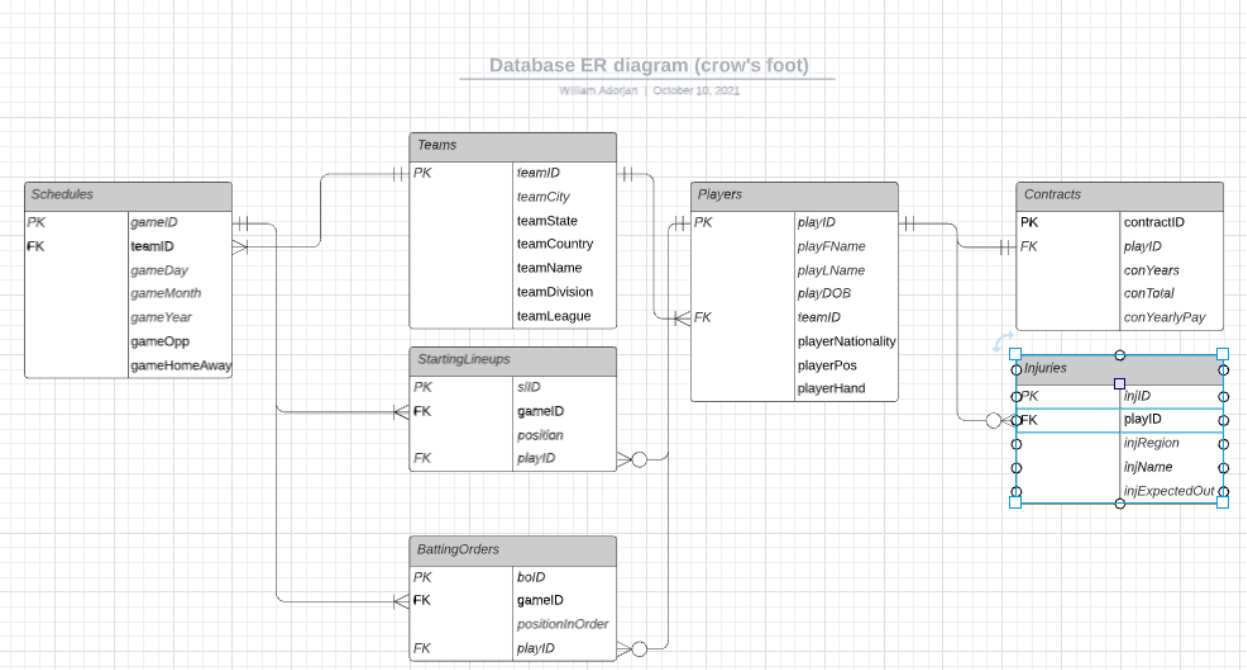

Schedules - This table contains data about every game to be played by any team in the organization. It includes day, month, and year to show the game date. This table also stores the opponent's name and whether the game will be played at home or away. There is one foreign key teamID and one primary key called gameID. This table is related to the teamID because a schedule instance could be for any one of the organization's multiple teams. This table is related to the batting order and starting lineup tables because you need a starting lineup and batting order for every game on the schedule. Both notations are one-to-many because there will be nine instances per game in both the starting lineup and batting order.

Teams - This table contains info about all the teams in the organization. The team's city, state, and country to show location. The team's name, division, and league for general info. The team's table has no foreign keys, and its primary key is teamID. This table is related to the players, batting orders, starting lineups, and schedules tables. There are many starting lineups/batting orders per team, but only one team per starting lineup/batting order instance. A one-to-many relationship to players because only one team per player but a team will have many players.

StartingLineups – This table is an instance relationship it connects a player to a spot on the starting lineup specific to one game. It contains gameID, position, and playID. Two foreign keys gameID, and playID. The primary key is slID. There is a one-to-many relationship to schedules because every starting lineup instance is specific to only one game, but each game will have nine players in the starting lineup. There is a one-to possibly many relationship to players because a player might not be good enough to be in the starting lineup but if they are good enough they will be in many starting lineups. Each instance in the starting lineup is specific to one player.

BattingOrders - This table is an instance relationship it connects a player to a spot in the batting order specific to one game. It contains gameID, positionInOrder, and playID. The positionInOrder is the player's spot in the lineup one through nine. There are two foreign keys gameID, and playID. The primary key is boID. There is a one-to-many relationship to schedules because every batting order instance is specific to only one game, but each game will have nine players in the batting order. There is a one-to possibly many relationship to players because a player might not be good enough to be in the batting order but if they are good enough, they will be in many starting lineups. Each instance in the batting order is specific to one player.

Players - This table holds general info specific to players. Including first and last name, date of birth, nationality, position, and their dominant hand will be stored in playerHand. There is one foreign key, teamID, which shows what team the player is on. The primary key is playID. The Players table is related to the starting lineups, batting orders, contracts, teams, and injuries table. The relationship to contracts is one to one because every player has only one contract, but every player has a contract. The relationship to Injuries is one to possibly many because a player can have anywhere from 0 to many injuries. The relationship to batting orders and starting lineups is one to possibly many because if the player is bad they will not be placed in any starting lineup or batting order, but if they are good they will be in many.

Contracts - This table holds specifics about player contracts. Including contract years, contract total value, and contract yearly pay. The primary key is contractID and the only foreign key is playerID. This table relates only to players because they are the only people getting paid. The relationship is one-to-one because there is one contract per player.

Injuries - This table holds data specific to player injuries. It includes the region that was injured, the name of the injury, and the expected time out of play. The primary key is injID and the only foreign key is playID. This table only relates to the employee database. The relationship to players is one to possibly many because there is one player per injury but not every player will be injured. Also a player could be injured multiple times throughout their career, hence the possibly many.