Name: Christian Schmidt

**Table of Contents**

[Project Direction Overview 2](#_Toc69865130)

[Use Cases and Fields 3](#_Toc69865131)

[Structural Database Rules 6](#_Toc69865132)

[Conceptual Entity-Relationship Diagram 8](#_Toc69865133)

[Full DBMS Physical ERD 9](#_Toc69865134)

[History Table and Trigger 13](#_Toc69865135)

[Index Identification and Creations 15](#_Toc69865136)

[Stored Procedure Execution and Explanations 17](#_Toc69865137)

[Question Identification and Explanations 20](#_Toc69865138)

[Query Executions and Explanations 21](#_Toc69865139)

[Data Visualizations 23](#_Toc69865140)

[Summary and Reflection 26](#_Toc69865141)

# Project Direction Overview

I would like to develop a fantasy football database that tracks player’s, teams, and coaches scores, stats, and historical performance. My goal for this database is to help me win my fantasy football league, by picking the best possible lineup in the draft under a time-sensitive situation. This database will be primarily for me, but I may share my results for a nominal fee.

When using my database, I will be able to answer questions that normally would require extensive research before the draft. An example would be If it were my 10th pick and I wanted to research undervalued TEs, I would have to visit multiple websites, track how every team utilizes their TEs, look at historical performances, find out who may take targets away from them, all within 2 minutes! With my database, I can streamline the process and sort for a low draft score, Teams TE usage, historical performances, and target distribution of current players on that TEs team. All this information will allow me to make an informed decision in a stressful and time-dependent window during my pick in the draft.

The database will store information in two ways, players, and teams. For players, I plan on storing data on:

* Coach: This will include important information such as W/L, points scored for, points scored against, etc. My vision is that this detail will give me a better idea of what the coach has to offer that may boost a team’s overall performance. I can also see whether this coach prefers running the ball or throwing it so that I can pinpoint which player will do the best in their offense.
* Stats: Rushing Yards per game, Pass TDs, etc.… as well as prior season performances for all players. This is the meat and potatoes section, and I will use it for multiple different calculations such as scoring total, the average score per game, etc.…
* Player identity: Team, # of years in the league, Name, Position, Injury (Y/N). This will help answer multiple questions like what the schedule will be for this player, is he a rookie, is he injured, how many games has he missed in his career, etc.?
* Draft score: I will most likely base this one on a journalist prediction, this will be a number designated for a player likelihood of being drafted early or late.
* Target %: this is an important stat as the more opportunity a player has, the higher chance that they score more points.

There may be more data points that I wish to add later, but for now, this is what I will be basing my database on. Since I plan on only allowing myself to use this database, I would not need to limit access to this database for any other users.

# Use Cases and Fields

1. Use Case 1: Player Fantasy Points Load
   * Question: which RB player has scored the most points in the 2020 NFL season?
     1. I will log into my database.
     2. I select the player’s name, total fantasy points scored, and filter by the year 2020.
     3. I will then select the position that I want, in this care RB.
     4. I will then have this player high up on my list of players to draft.
2. Use Case 2: Player Draft Score Search
   * Question: I am picking 2nd overall. Which player has the highest consensus to go #2 in the fantasy football draft?
     1. I will log into my database.
     2. I will select the player’s name and only filter by draft score.
     3. I will then look for the 2nd highest-rated player.
     4. I will decide whether I agree with this consensus by searching for this player's stats.
     5. If I agree, I will then select this player with my second overall pick.
3. Use Case 3: Player Depth Chart look up.
   * Question: It is the 10th round and I need a handcuff to my best RB, what other RBs are on that player’s team?
     1. I will log into my database.
     2. I will filter by position RB and by Team.
     3. I will then include a draft score to see who is most likely to be second on the depth chart.
     4. I can then pick up the correct player to protect my team from injuries.
4. Use Case 4: Team Performance search
   * Question: What team had the worst points against last season?
     1. I will log into my database.
     2. I am looking for defensive Team stats, I will focus only on D/ST.
     3. This I am considering all scoring methods; I will look to arrange so I find a team with both the worst defense and offense are organized at top.
        1. This would involve me filtering by defense stats such as points scored against and total TDs surrendered, but then also look at Team stats such as lowest points scored, and least number of TDs scored.
     4. I will then find players who are playing against this team as more favorable by cross-referencing a player's team with their schedule.
5. Use Case 5: Coach Offensive Performance look up.
   * Question: Which coach has scored the most TDs on offense over the last three years and what is their current team?
     1. I will log into my database.
     2. I will look at Coaching stats and filter by TDs scored and filter by the last 3 years.
     3. I will then find which coaches had performed the best and see what team they are currently coaching.
        1. I can use this information to then feel at offensive players on that team more favorably.

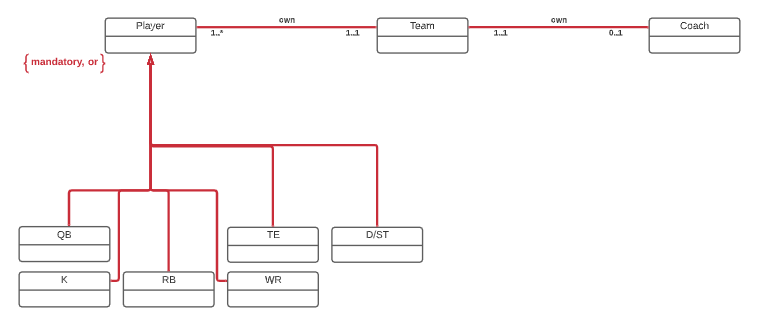
|  |  |  |
| --- | --- | --- |
| Field | What it Stores | Why it is needed |
| Player\_id | This will be the primary key for every player and coach’s name | This is important so that I can create a relational database while using a key that will be used across different data tables. |
| Name | The player/coach’s name | To correctly identify the individual. |
| Age | Age of the player | To correctly identify the individual |
| Team | Team Name | To correctly identify which team this player plays for (includes Free Agent) Will be a foreign key |
| Injured | Whether this player is currently injured or not | It may either lower my favorability on them or even eliminate them if they will not be available for this upcoming season |
| Position | QB,RB,TE,WR,K,D/ST | In fantasy football, you can only play so many players from each position, so it is important to find the best players possible for each position listed |
| Season | Year | This will help sort statistics yearly |
| Draft Score | Journalist reviewed draft position | This is a score that most websites that help with fantasy football has. All players are ranked from 1st (best player) to last (worst player). I will download these numbers from a reputable website |
| “Stats” | Amalgamation of stats | This is a broad field and is a placeholder for numerous columns that will show position stats, as well as historical scoring and averages. This will include everything from offense to defense and will be expanded further when I create new columns with calculations used to generate new insights. |
| Games Played (GP) | Shows how many games the player has played | Important stat that highlights experience + games missed due to injuries. I would need to decide whether to add another attribute to include “games missed”, but I am currently not for it as I believe this database is strictly focused on the performance of the players during a season, regardless of playing time. The injury Boolean attribute also would be my primary stat to use for availability. |
| Rookie | Yes or No | Particularly important attribute. I need to find players who may have a high draft score but have no stats. This is because of their potential contributions to the team that cannot be quantified currently as they have no games played in the NFL. They may have great college stats, but often, the best college players with incredible stat lines may not be great players in the NFL i.e., a Bust. This would highlight the player as a high risk/high reward, rather than someone I should just avoid. |

# Structural Database Rules

Here are my structural database rules. Keep in mind to simplify this database, I will not consider the minimum/maximum number of players allowed on a roster, due to practice squad/reserve players which may change roster construction frequently. I will also consider “Free Agent” as a team for a player, to avoid confusion:

* A Player **must be** **owned by one** Team.
  + No player can play for two teams, they may switch teams during a season, but will always be on a single team. Thus, it is a 1:1 relationship.
  + All players will have a team assigned to them. If a player is not on an NFL team, they will be listed as a “Free Agent”. For this database, I will consider Free Agent as a team.
    - I am changing this from a subtype of “Team” from my previous iteration. I realize now that there really is not a need for a second set of tables when discussing Free Agent players. Most times, a Free Agent does not contribute enough to justify a spot on an already limited roster selection for a fantasy football team. There are rare occurrences, but I could simply prioritize NFL players and filter out “Free Agents” from the team attributes. Another point is that Free Agent players will NOT have a draft score. Why would you draft a player that currently has a projected score of 0! They will not play until they are on a team, therefore it would just make sense to put Free Agent as a Team
* A Team **may own** **at least** one player, but most likely many.
  + Rosters in the NFL usually have around 53 players during the season give or take some injury reserved/practice and reserve additions.
* A Coach is owned **by only one** Team.
  + No coach can play for two teams, they may switch teams during a season, but will always be on a single team.
* A Team **must** own only one coach or none.
  + A Team will always fire a coach and hire a new one or upgrade an assistant coach to head coach status as an “interim” coach.
    - I Included none since I will now include Free Agent into “Team” and there is no coach for Free Agent.
* A Player may be in an Offense Position or a D/ST
  + I will make two tables to properly define a player’s position. I think this is important to separate as offense position scores points on offense, while D/ST scores points on defense. Therefore, each group has different ways to calculate points and needs different fields such as points scored against for D/ST vs. PPR scoring.
* A Player can be a K, WR, TE, QB, RB, several of these or none.
  + These are the offensive positions that can score in normal PPR leagues like the one I am in. Players can have multiple positions, but for the sake of my league and this database, the only positions available for a player are the ones listed above and D/ST, which is a special case.
  + I will also use the position subtype to prioritize stats accordingly. So, a WR would not include an attribute such as Field Goal attempts that is more important for a kicker, while a kicker would not have receiving yards as an attribute.
* A player may only be a D/ST or not.
  + There is not an either-or. D/ST is equivalent to an entire team’s defense and special teams, so every team has exactly one. One caveat is that it excludes Kickers (K) as they are a special position that generate points due to their own individual contributions to scoring. A player like an RB or WR can play and even score points on special teams, but that is scored purely on whether they score a TD or not so I would not need to create a special attribute for Special Teams contributions
  + This is a bit peculiar about fantasy leagues, as a team’s defense can score points for you. So, in essence, a “Team” could be a “Player”. To this database, D/ST which stands for Defense/Special Teams will be considered a player.
  + Since this is a group of players on defense for a team, this position will never include a secondary position like QB or TE. This position is made up of several which can include the Kicker or a WR even, but they only score points if they specifically do something points. If, for example, a WR who also plays special teams returns a punt for a TD, not only does the D/ST get points, but that WR also scores points.

# Conceptual Entity-Relationship Diagram



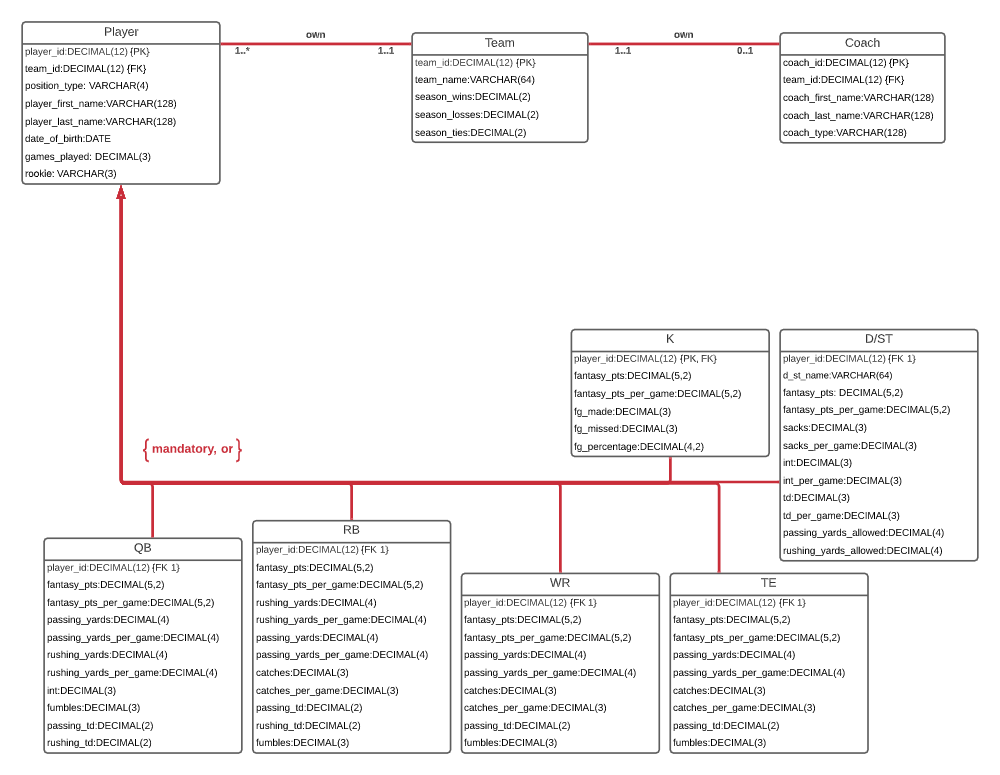
Here are the associative structural database rules used in the ERD.

* A Player **must be** **owned by one** Team.
* A Team **may own** **at least** one player, but most likely many.
* A Coach is owned **by only one** Team.
* A Team **must** own only one coach or none.
* A Player must be a K, WR, TE, QB, RB, or D/T.

There are three entities: Team and two others under which are Player and Coach. Both relationships are mandatory in be the database. Team to Coach is an optional 1:1 relationship and Coach to Team is a 1:1 relationship. Player to Team is a 1:1 relationship, however, Team to player is a 1:M relationship.

I removed the two entities under Team from my last iteration – Free Agent and NFL Team – as this relationship is no longer needed. I also removed the entity under Player, Offense Position, as this addition added unnecessary complexity as I wrote the initial SQL coding. The positions listed now all-together, QB, RB, WR, TE, K, and D/ST are a mandatory and disjointed relationship to the Player Table. ERD now captures both the structural database rules and use specialization-generalization.

# Full DBMS Physical ERD



I started with my associate relationships of Team/Coach and Team/Player.

Team/Coach is a 1:1 relationship where one team can have on coach and one coach belongs to only on team.

Team/Player is a 1:M relationship where one Team must have at least one player, likely has many and a player can only belong to one team.

For best practice. Each table has a synthetic key, with the datatype DECIMAL(12) to allow for multiple records in my database. For Team, I kept the Foreign Keys of both the Coach and Player Primary Key as it is a clear 1:1 relationship for both to Team. I used own for the relationship name as it made the most sense to me as there are contractual obligations for both coaches and players and that the team owns these individuals for several years.

For specialization-generalization relationship, I have one layers to my player entity. I know longer have D/ST separated out under an additional layer. D/ST scoring is still different from offensive positions. I will simply create separate attributes for this table compared to other positions so there is no need for a separate entity to use the proper stats and fantasy points calculations that would not work for other positions. From there I separated out the offense positions for a similar reason.

Going back to specialization-generalization relationship, each additional entity under Player has a primary and foreign key of position\_id which references what position a player play. To simplify my data table, only a player’s primary position is listed about the position\_id. If I were to include it, I would have to create new positions such as WR/RB to accommodate. With these additional mappings. The DBMS physical has all the relationships in the conceptual ERD I made prior.

Regarding attributes, I asked myself several questions to narrow down what I would need to include.

* What fields are obvious?
  + Storing the coaches, players, and team names, as well as the overall stats are the simple attributes that come to mind. Along with stats, players Age is also important when considering experience, usage in the offense, and whether they can keep at advance ages.
* What will separate my database from others.
  + The fact that I am not just focused on a Player stat, but also encompassing a Team and Coaches stat I think will separate my database. This is because I want to focus not only on how the player performs, but what if that player is joining a new team that did poorly the year before? Or does the Coach offense score a lot of points? These types of questions can be answered so it is important to separate out coaches, teams, and player stats to get a more comprehensive picture.
* What type of user interface would I be using?
  + I think this database can function on its own, but I would love to link this database to an R/Python application that could create outstanding visual data for even quicker decisions.

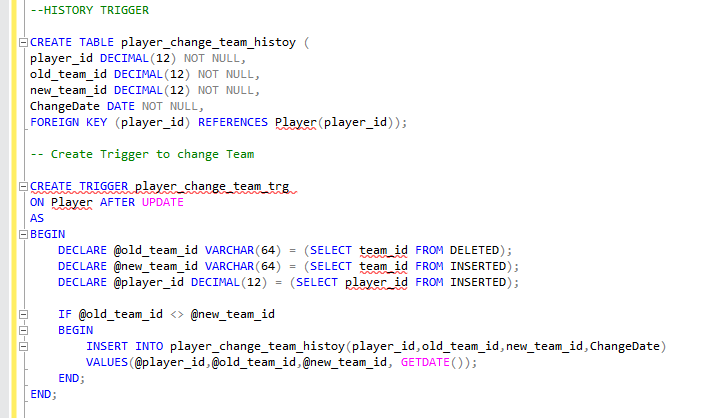
As for datatypes:

|  |  |  |
| --- | --- | --- |
| Field | Datatype | Reasoning |
| Position\_name | VARCHAR(16) | Simple field which will be K, QB, RB, WR, TE, or D/ST |
| (player,team,coach,position)\_id | DECIMAL(12) | Since I want this database to last for a long time, I want to make sure there is enough room for new players, while keeping a historical database for players/coaches who eventually retire |
| player\_first\_name, player\_last\_name,  coach\_last\_name,  coach\_first\_name | VARCHAR(128) | Many football players in today’s NFL come from diverse backgrounds, with well over 70% from minority groups. Therefore, I want to increase the character length to include longer names from various countries. |
| d\_st\_name, team\_name, offense name | VARCHAR(64) | The name of the football team will be the same for both, but d\_st will include player stats as it this is the defense/special teams for that football team. As for offense name, those are the full name of the position tables. |
| Date\_of\_birth | DATE | To avoid having to constantly update this field, I have changed age to Date\_of\_birth. |
| Position\_id | DECIMAL(2) | There is only so many positions to include. If I ever want to expand to include players on Defense separately, I will need at least two-digit limit |
| Fantasy\_pts  Fantasy\_pts\_per\_game | DECIMAL(5,2) | Stat which showed how well this player did during the season, as well as the avg score the player achieved in each game. |

|  |  |  |
| --- | --- | --- |
| Fg\_made.  Fg\_missed.  Fg\_percentage | DECIMAL(3)  DECIMAL(4,2) | K stats. How many FG did the kicker make and how many did they miss along with the percentage =(made/made+missed) |
| Passing\_yards  Passing\_yards\_per\_game  Rushing\_yards  Rushing\_yards\_per\_game | DECIMAL(4) | Passing yard stat will mean different things for different positions. QB will be the number of yards *thrown****,*** WR/RB/TE will be the number of yards *caught*.  Rushing yards will be the number of yards gained by running the ball. |
| Int  Int\_per\_game  Fumbles  Fumbles\_per\_game | DECIMAL(3) | How many turnovers did the player give up/did the defense get |
| Passing\_td  Rushing\_td  Td  Td\_per\_game | DECIMAL(2)  DECIMAL(3) | I must distinguish between passing td and rushing td, as the stat matters differently based on the position. For D, since there are several ways, they can score, I used td, and td\_per\_game as an effective way to see how a defense does on a per game basis. |
| Catches  Catches\_per\_game | DECIMAL(3) | Number of catches an offensive player caught in a season as well as a per game basis |
| Passing\_yards\_allowed  Rushing\_yards\_allowed | DECIMAL(4) | Defense stat where the more yards allowed to the opposition offense, the lower the fantasy points score |
| Games\_played | DECIMAL(3) | Number of games this player has played in the season |
| rookie | VARCHAR(3) | Yes or no. determines if the player is a rookie or not |
| Season\_wins  Season\_losses  Season\_ties | DECIMAL(2) | The teams prior season record |
| Coach\_type | VARCHAR(128) | The coach’s position (Defensive-Coordinator, Head Coach, etc…) |

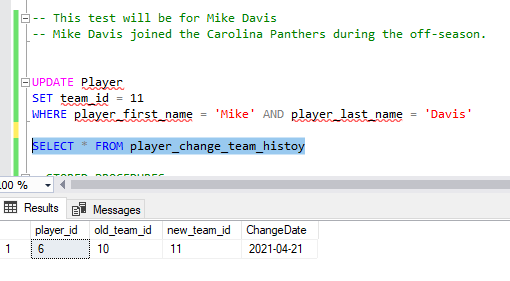
As for normalization, I found no normalization that I could do that would address any sort of redundancy or tentative relationship.

# History Table and Trigger

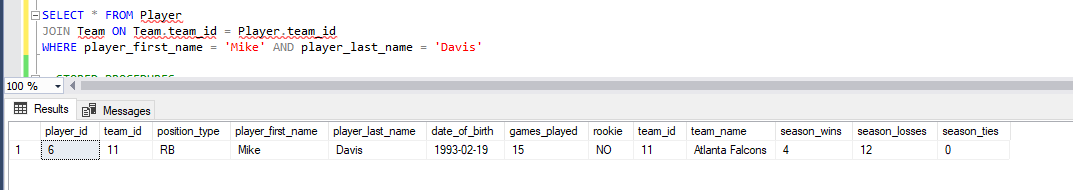


This History table and trigger will help me update players who have changed teams during the offseason. The ideal situation is that no player suddenly gets traded/cut and moves on to a new team, but now with this trigger, I will be able to update my database accordingly.

In this Use case, the Player “Mike Davis” had played for the Carolina Panthers in the 2020 season. After the end of the season, Mike changed teams in Free Agency and joined the Atlanta Falcons. I need to update the player so that he is now linked to his new team. The code to update is down below, along with the history table to show that this change was made



Now I will show the result using a JOIN statement.

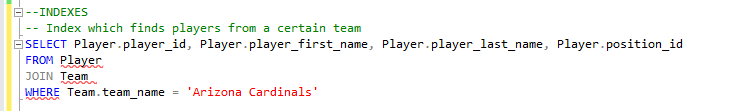


As you can see, Mike Davis is officially an Atlanta Falcon.

# Index Identification and Creations

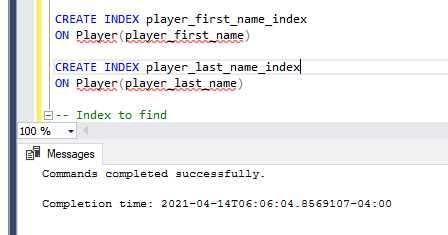
Replace this with indexes identifications useful to your database, explanations as to why they help, and screenshots of their creations.

So, to figure out the best indexes to use, I want to follow-up on one of my use cases which was to find players on a certain team. The query would look something like this.

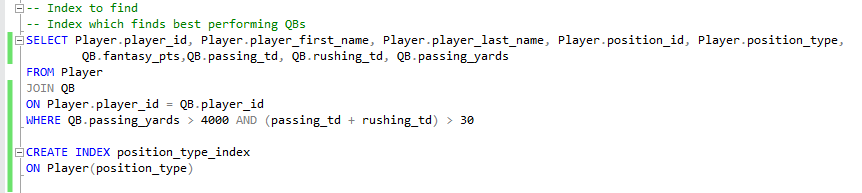


This query specifically helps me find players who are on a certain team. In this case I am trying to find player who belong to the “Arizona Cardinals” because I like how their offense looks going into the new season. I want to find just the players and their names + positions, while narrowing down my search by using WHERE statement. Given this, I would want to create a Unique index for team name and non-unique player name (two players may have similar first or last names) which would look like this.





Another use case came to mind where I wanted to find the best performing players for a position. The code is down below.



This query specifically narrowed down the list of QBs I wanted to observe based on several key performance attributes. A good benchmark on how well a QB did in a season is Passing Yards and Touchdowns. Therefore, I wanted to narrow above 4000-yard season with a combined rushing and passing yard score over 30.

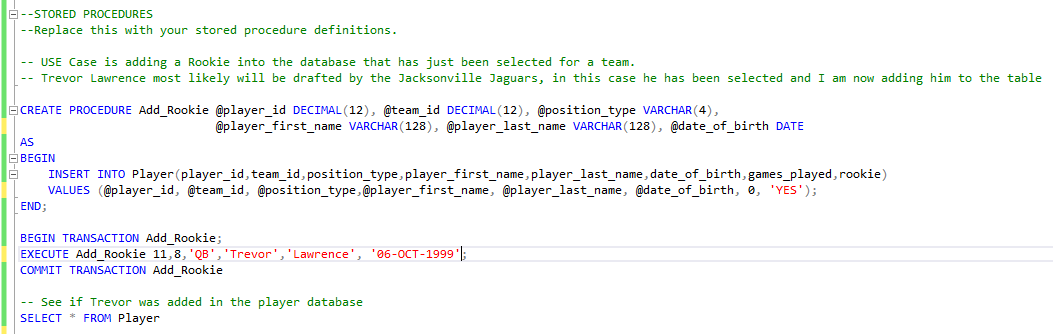
In the process, I realized I could use and index for position type. The reason is that although I am combing the QB table, what if I wanted to search player performances with similar attributes from more than just one table? WR and TE have the same attributes in my database, so If I wanted to search for the best WR/TE as a combined group, I may want an index if I wanted to just look at WR stats using the same query. This works for the other position groups as well and will help me when I Join multiple tables to not have to create an entirely new query.

# Stored Procedure Execution and Explanations

My stored procedure is the addition of Rookie players after the NFL draft. My history table is used to track when players have changed teams during Free Agency. However, players who are drafted are entirely new to the league and have no history of stats or experience. Therefore, I need to create a procedure that will allow me to add Rookies easily and without any issue. I only need to adjust the Players table as there will be no stats available.

My use case is the QB out of University of Clemson, Trevor Lawrence. Trevor Lawrence is considered a generational quarterback. Therefore, he is almost certain to be picked 1st overall in the upcoming draft. Furthermore, the team that is 1st to draft is the Jacksonville Jaguars. In this case, I will add Trevor Lawrence as a player who will play for the Jacksonville Jaguars.

The code is down below.



And here is the result.



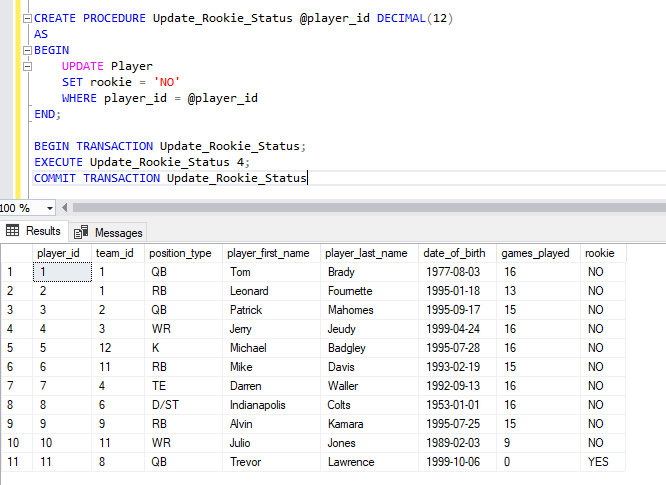
I know have added Trevor Lawrence as a rookie to the table.

The next procedure I created was to update players after the draft that are no longer rookies. Once the 2021 Draft class has been drafted, they are the rookies. Players in my table like Jerry Jeudy are no longer Rookies, which must be adjusted.

In this case, I need to create a procedure that will update the player rookie status from “YES” to “NO”.

The goal of this is to maintain accurate records, with newly drafted players categorized as Rookies.

Here is the code:

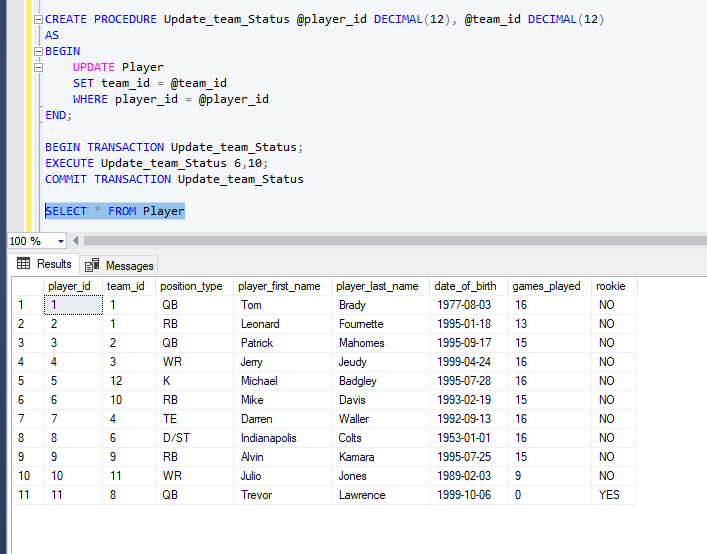


As you can see, Jerry Jeudy is no longer a Rookie, with currently Trevor Lawrence being the only rookie in my table.

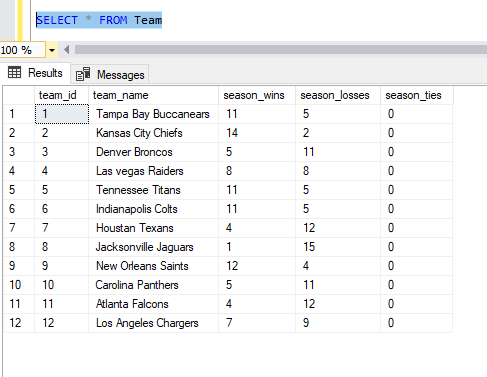
The last stored procedure I need is to update a player who may change teams DURING the season. This will be helpful as players who are traded or cut/resigned by a new team will need to have their Player table updated accordingly.

This use case will be hypothetical. For example, imagine Mike Davis has had such a poor season that he was traded by the Atlanta Falcons back to the Carolina Panthers. I would require a procedure that would streamline this change to avoid unnecessary coding.

Here is the code down below.



As you can see, Mike Davis is no back on team\_id 10, which means he is back on the Carolina Panthers



# Question Identification and Explanations

Although questions 1-6 are remarkably similar, this is a quite common scripts you would run during the draft and even after. As you would want to check every single time you are up to draft to see the top players you would want to choose from. The next question is more specific, as you may want to care about other factors, other than if the player is good or not. Next

Question 1: Top QBs

Explanation is simple, I want to find QBs who managed to score more than 200 fantasy points in a season. I consider 200 to be the lowest possible score for a player to be in my fantasy team for this position.

Question 2: Top RBs

I want to find RBs who managed to score more than 150 fantasy points in a season. I consider 150 to be the lowest possible score for a player to be in my fantasy team for this position.

Question 3: Top WRs

I want to find WRs who managed to score more than 100 fantasy points in a season. I consider 100 to be the lowest possible score for a player to be in my fantasy team for this position.

Question 4: Top TEs

I want to find TEs who managed to score more than 100 fantasy points in a season. I consider 100 to be the lowest possible score for a player to be in my fantasy team for this position.

Question 5: Top D/STs

I want to find D/STs who managed to score more than 100 fantasy points in a season. I consider 100 to be the lowest possible score for a player (which is a team’s whole defense) to be in my fantasy team for this position.

Question 6: Top Ks

I want to find Ks who managed to score more than 100 fantasy points in a season. I consider 100 to be the lowest possible score for a player to be in my fantasy team for this position.

Question 7: Which QB had a great season, did not miss a single game, and played on a 10+ winning team?

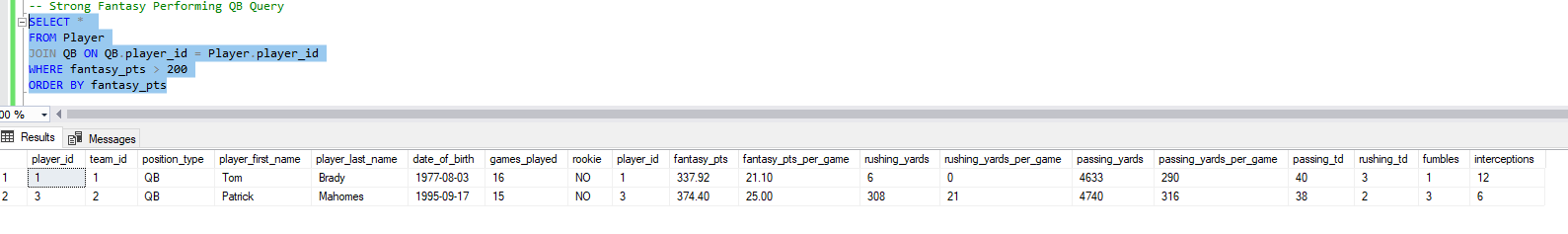
This question is useful, as I want to know which QB I can trust to play well, but also play on a team which is winning games. I want my players to be on winning teams because the generally put-up high scores, which lead to higher fantasy points. However, it is not just enough to be good, but also reliable. I want a player who did not miss a single game.

Question 8: which players only missed at most two games?

It is useful to maybe broaden my search, compared to Question 7. In general, I want to see players only miss games for minor injuries, as this could show players who may miss a game here or there but are reliable enough to play most games. So I want to find players who at most only missed two games

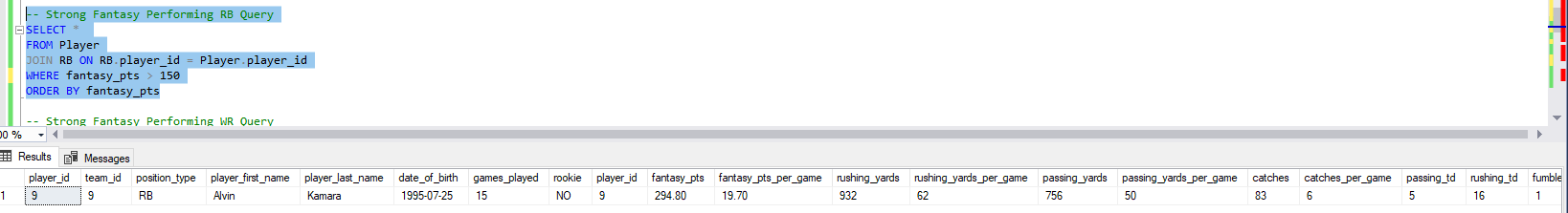
# Query Executions and Explanations

Question 1:



Both Tom Brady and Patrick Mahomes scored more than 200 fantasy points. I would find both players as possible options in my fantasy team.

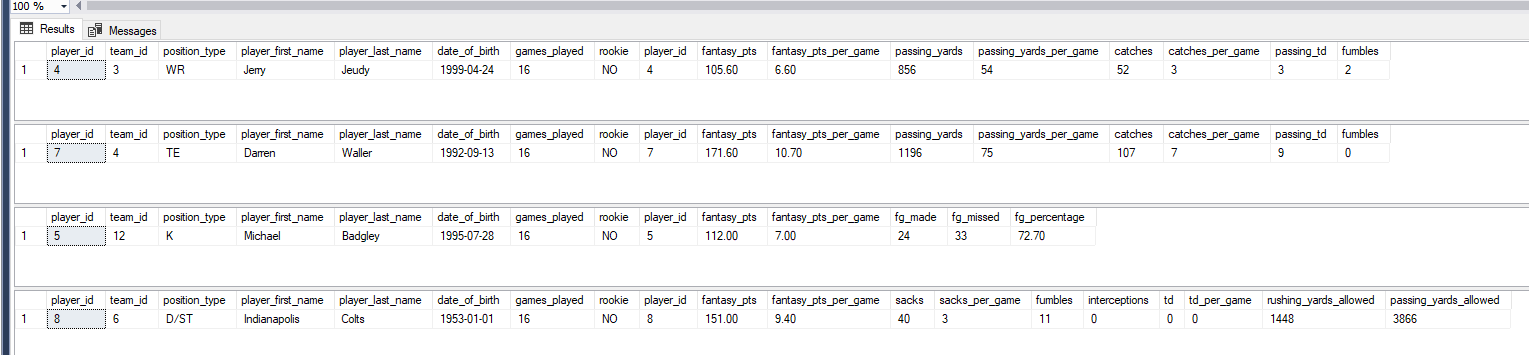
Question 2:



Alvin Kamara score more than 150 fantasy points. I would find him as a strong candidate for my fantasy team.

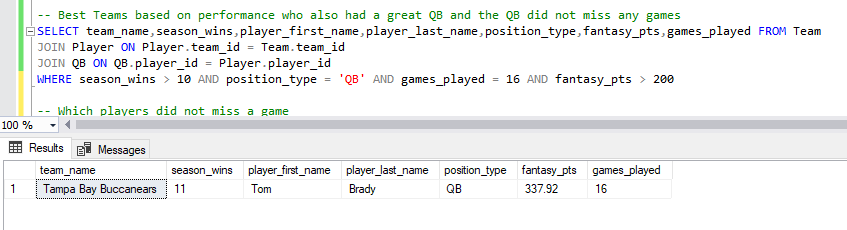
Question 3-6:





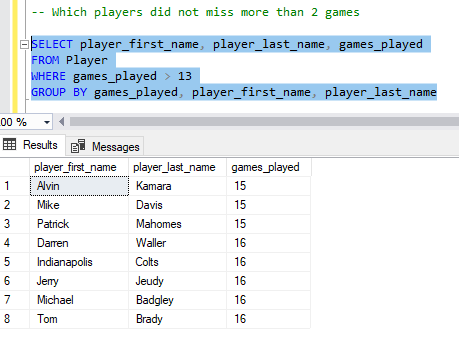
For Question 3-6, I find that Jerry Jeudy, Darren Waller, Michael Badgley, and the Colts D/ST are all viable options for my fantasy team as the scored more than the minimum amount I would expect from the WR, TE, D/ST, and the K positions

Question 7:



Tom Brady is the only player in my current database that was on a winning team, played QB, and did not miss a single game. If I had to compare both Patrick Mahomes and Tom Brady, I would need to consider that Tom Brady played all games. He was reliable while Mahomes still missed a game despite a strong fantasy performance.

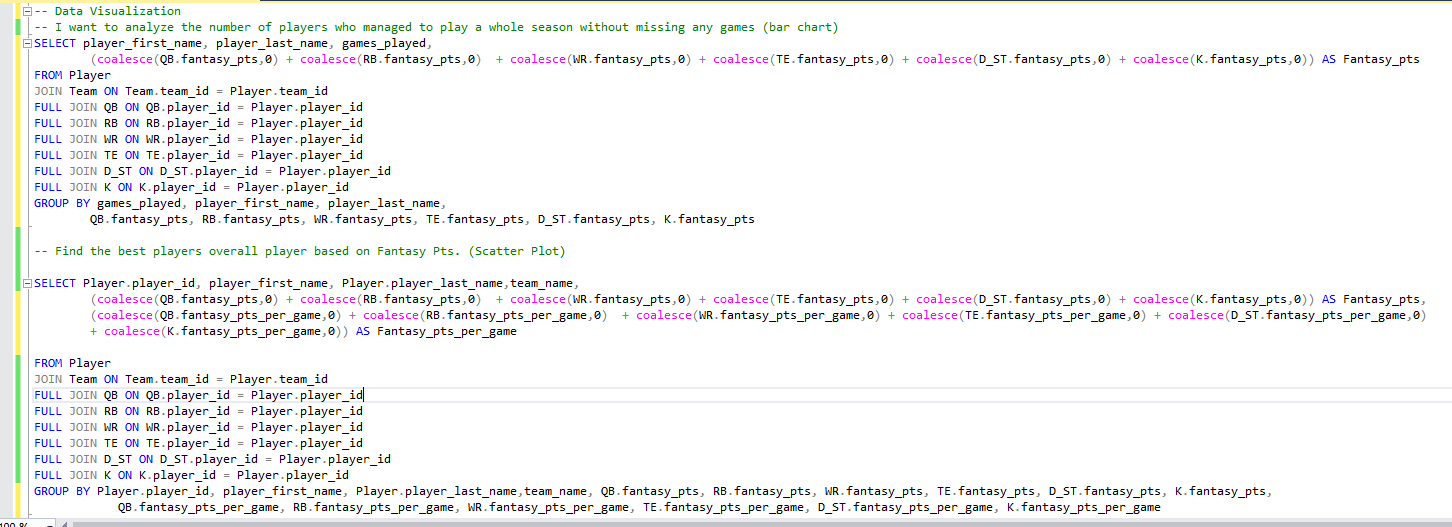
Question 8:

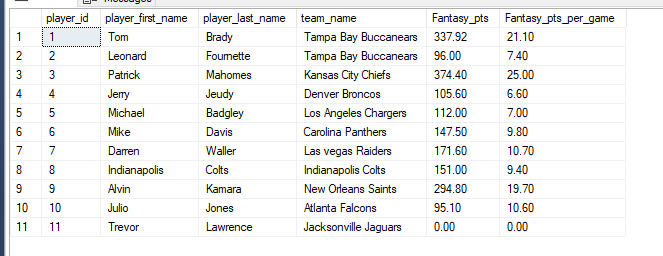


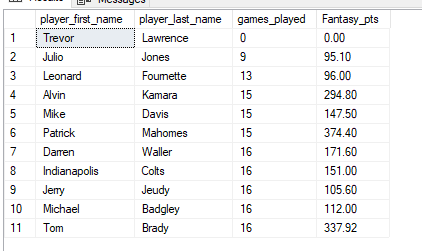
These players listed above managed to stay on the field without missing the field more than 2 times. This list would help filter out players that had an injury plagued season. As a result, I would consider players on this list higher than those that are not on this list.

# Data Visualizations

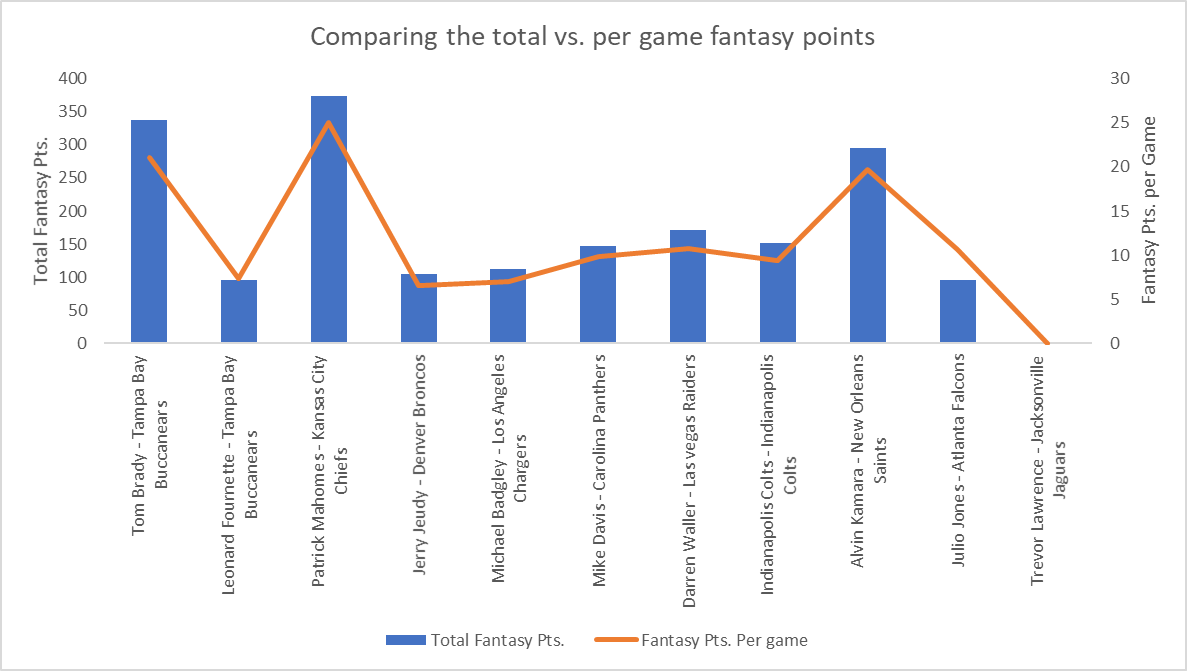
Here is the code for the resulting tables that I used for my data visualizations section



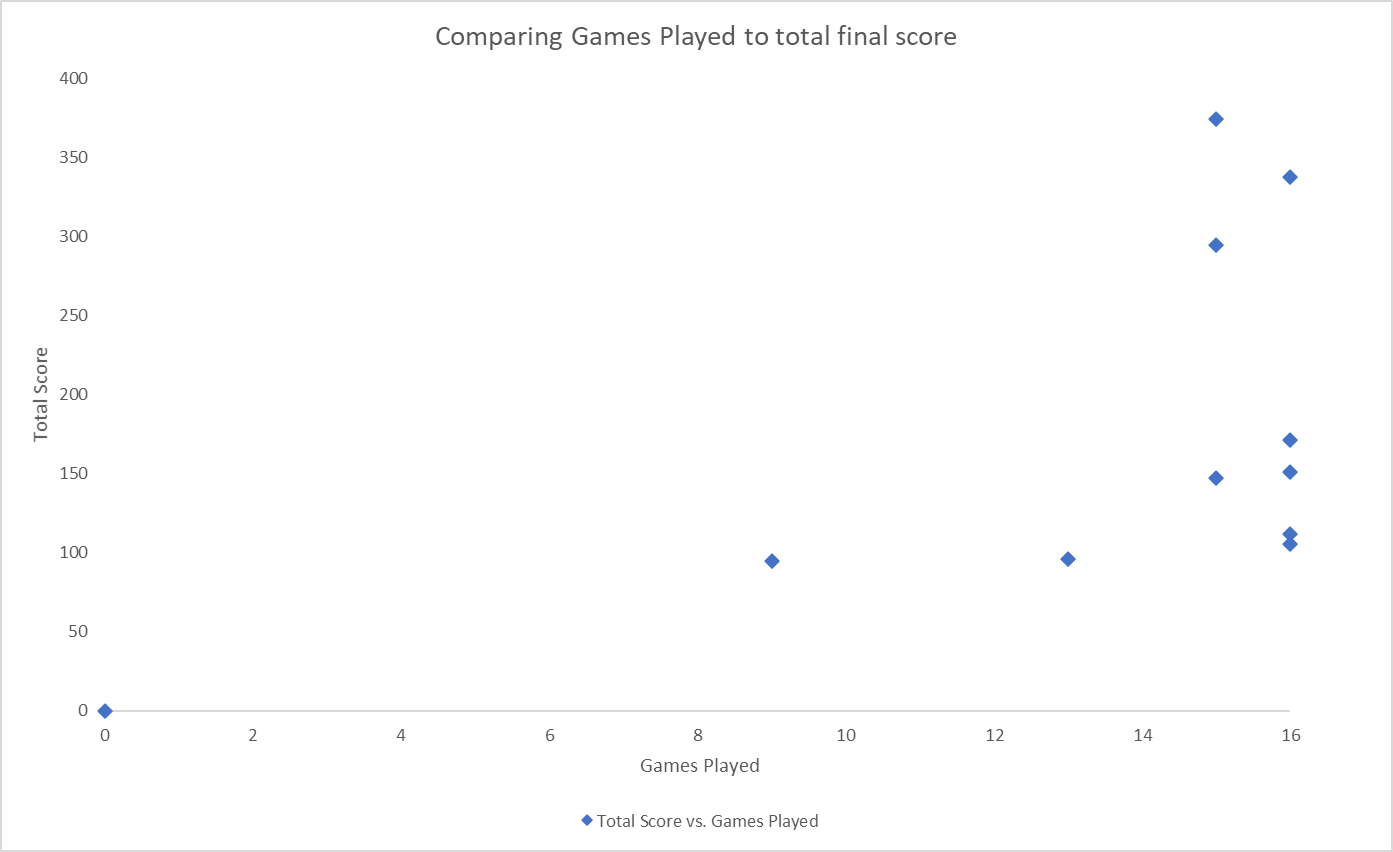




Here are the charts below:



I have a couple of observations regarding this graph. For one of the top performing players, only 3 players managed to score above a 200, while the rest scored within the range of 100-200. Although this is a small sample size, we can infer that it is much harder to have a great season than an average season. Another observation I see is that just because you are on the same team, one player can have a widely different outcome from another. Leonard Fournette never managed to surpass 100 pts during the 2020 season, while Tom Brady scored well over 300 pts! Both came from the same team that went on to reach the Super Bowl, so one can conclude that not all players will perform will in fantasy from winning teams. Lastly, pts per game has a direct correlation to how many pts you score in total. That said, Julio Jones is an exception then, as he has higher pts per game than other player, but a lower total point score. This would need to be investigated to find what the route cause of this problem is. I can use this information to determine which players may be ready for a bounce back season after missing a lot of games, as well as find healthy and high scoring players to pick in my fantasy team.



Moving on to number of games played, the main observation one can make is that players who play all their games, tend to have a higher score. You will not necessarily score more than a player who may have played only a couple fewer games, but there is a higher chance that players who play the full season will score the most points possible. This information shows that I should prioritize drafting healthy and great players, but not to limit my search if a player missed a couple of games.

# Summary and Reflection

My database is for my fantasy football league, which will help me pick players for the draft. During the draft process with such little time to choose, it can be hard to make the right decision. This tool will help me make the right choice and take the guesswork out of my lineup.

The data I collect will focus on three distinct entities: Coach, Team, and of course Player. Each entity will provide invaluable information that will ultimately help me choose the best players in the draft. Moving forward, there may be more fields to consider that I cannot think of right now. The data will be incorporated from multiple sources: data on players performance, data on team’s performance, and data on coach’s performance. In the future, I believe this database will be useful to perform more advanced mathematical and analytical observations using R or Python, which would be beyond the scope of this project and require additional time outside of this semester. Lastly, I believe there may be data that is not available elsewhere that I would have to calculate myself, such as calculate fantasy points, based on my fantasy league settings.