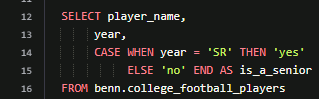
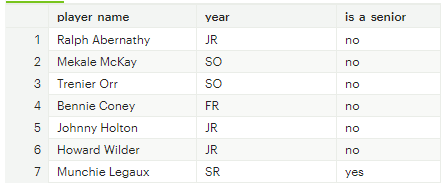
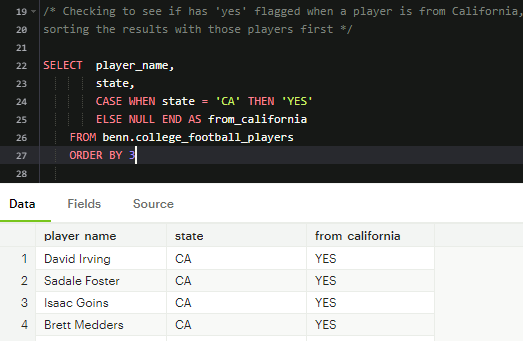


- Using the CASE statement as an excels version of if/then statement. Paired with WHEN statement. Every CASE statement must end with the END statement. Using the ELSE statement The CASE statement checks each row to see if the conditional statement year=’SR’ IS TRUE. The word “yes” gets printed for anywhere in the column that we have named is\_a\_senior. Where we see false, nothing happens in that row, leaving a null value in the is\_a\_senior column.

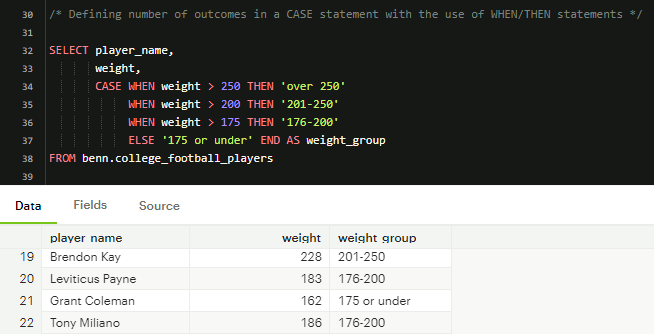




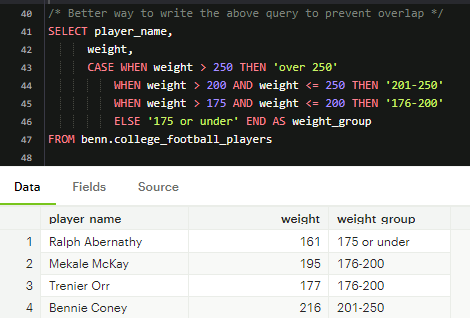
- Using NO with the ELSE statement, replaces null values



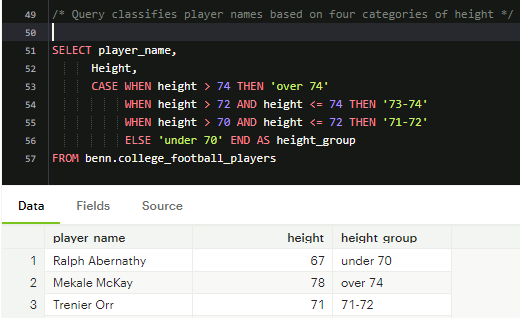
- In this query, if a player is from California, the column will be flagged ‘Yes’, and it will order the results by the from\_california column



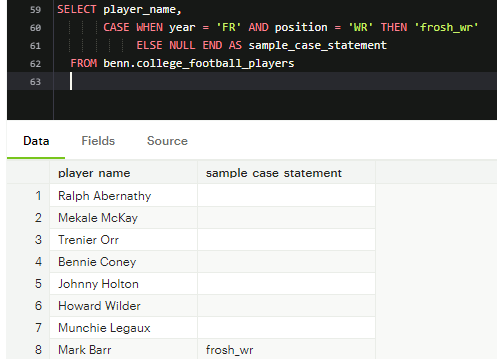
- Using the CASE statement in conjunction of the WHEN/THEN statement. This statement works in the order it is written. We did this query to identify which players weights fall under what weight group.



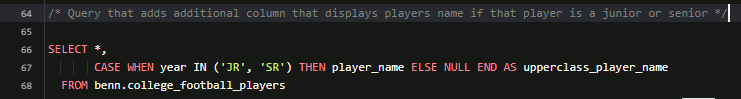
- Defining the values by using the ranges like this is the best practice .

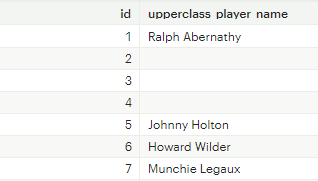


- This query includes the players names and a column named height\_group to classify the height into 4 different categories. We use the CASE WHEN statement followed by multiple WHEN statements. We observe the different height groups

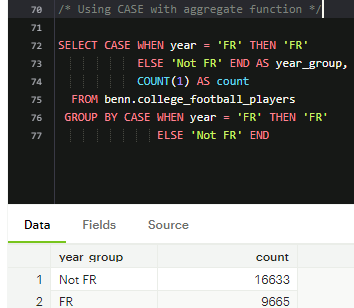


- Stringing multiple conditional statements with AND/OR

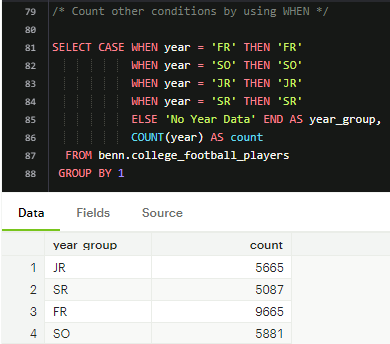




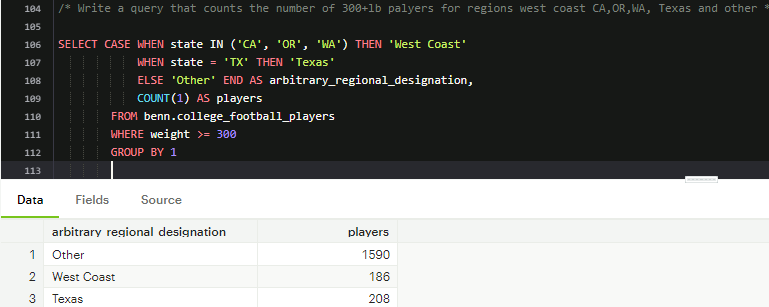
- This query was created to observe the names of players if they were juniors or seniors in a column named upperclass player name.



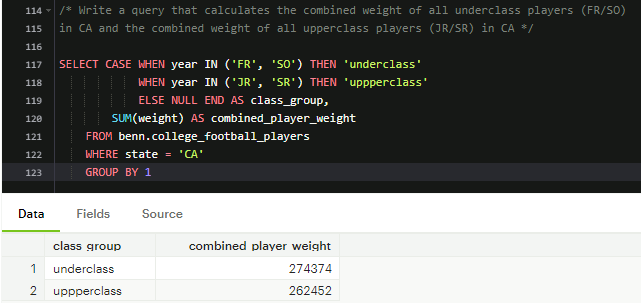
- We are looking for the count of FR and NOT FR, by pairing CASE with an aggregate function. COUNT by itself ignores nulls, the CASE statement is used to evaluate the condition and produce null or non-null values. We used NOT FR for null values.



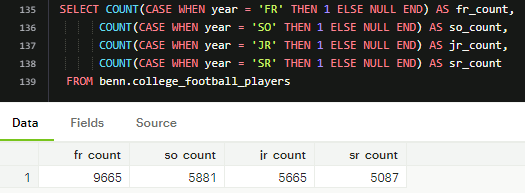
- This query we are exploring the count of year group, pairing the CASE statement with WHEN for many conditions



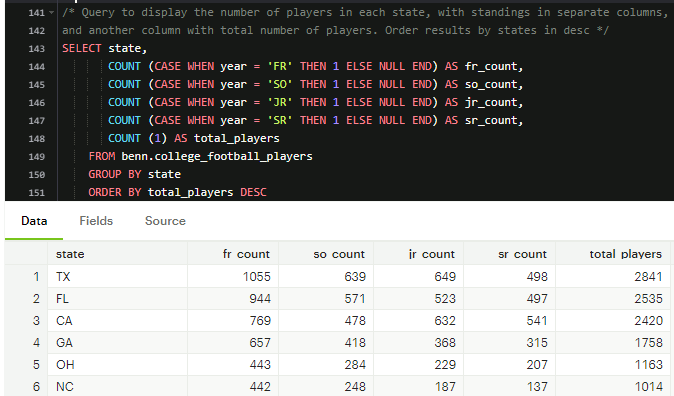
- In this query we explore the question of which states/regions have the amount of players where they weigh more than 300lbs.



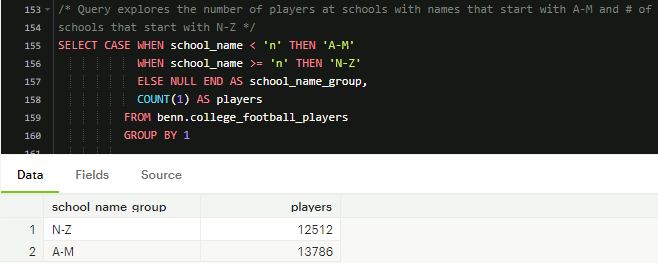
* In this query we answer the explore the calculated combined weight of underclass and upperclass. First we had to define the split of the group, then we use the SUM function to get the sum of combined weight and group by column 1.



- Re-orienting the data to show horizontally as a pivot table might.

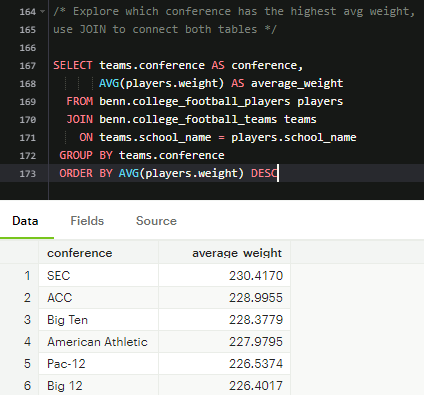


- Here we explore which states have the highest count of total players and break it down more by count of grade standing. Need the pair COUNT with CASE WHEN because we are looking for a numerical value

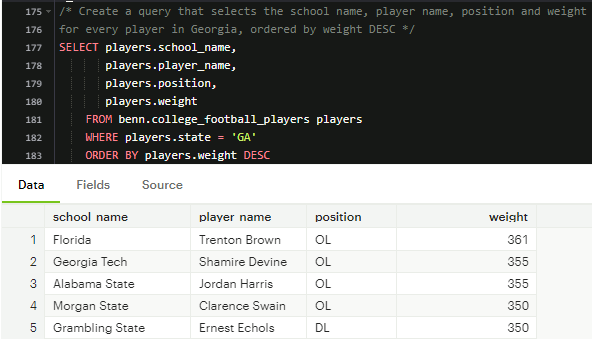


- Exploring the number of players at schools which names start with either A-M or N-Z

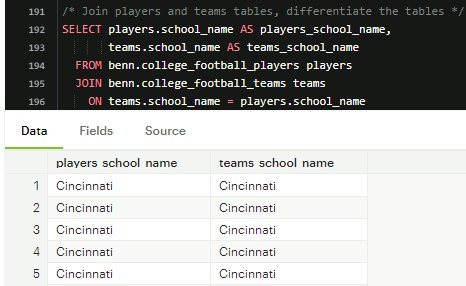
**The use of joins with dataset**



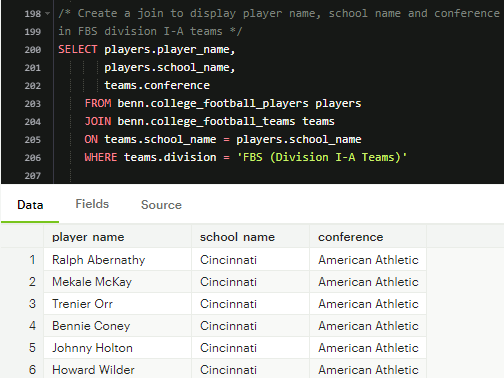
- Exploring which conference has the highest average weight. We use the JOIN clause to attach the football team table and the school name team. We then use GROUP BY clause with teams.conference to tie this data by conference



- Created a query that selects school name, player name, position, and weight for every player in Georgia. Observations include, player name Trenton Brown from Florida University with the weight of 361lbs is the largest weighted player



- We join the players table with the teams table. Since results can only support one column with a given name, we name the column individually to allow the columns to be independent.



- This query is written to show the relationship of player names, school names and conferences for schools in the FBS Division I-A teams. Join with player table and teams table had to be done because it has school name in the table as a common attribute, then we filter with WHERE for teams division.