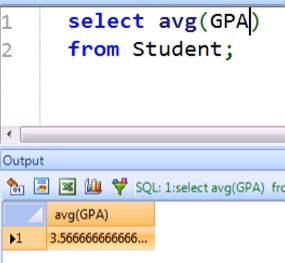
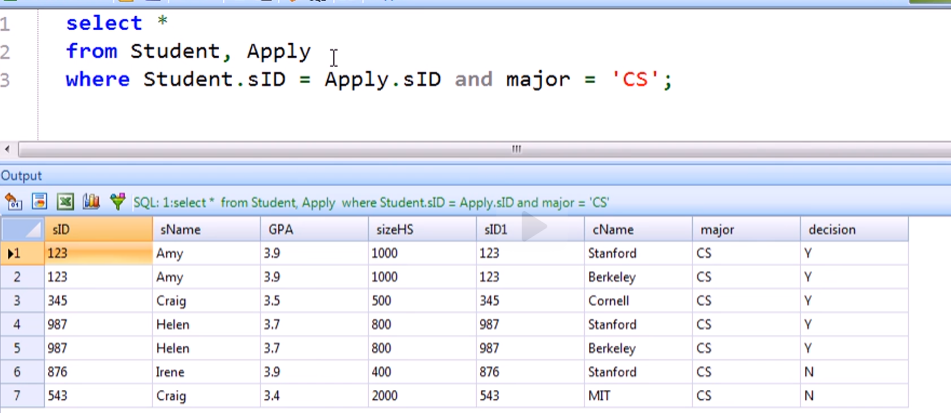


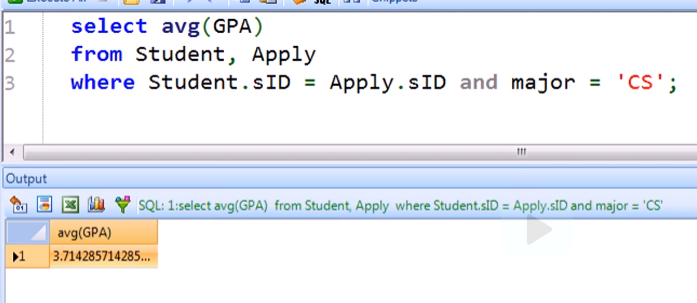
AVG

Getting the average GPA of all students:



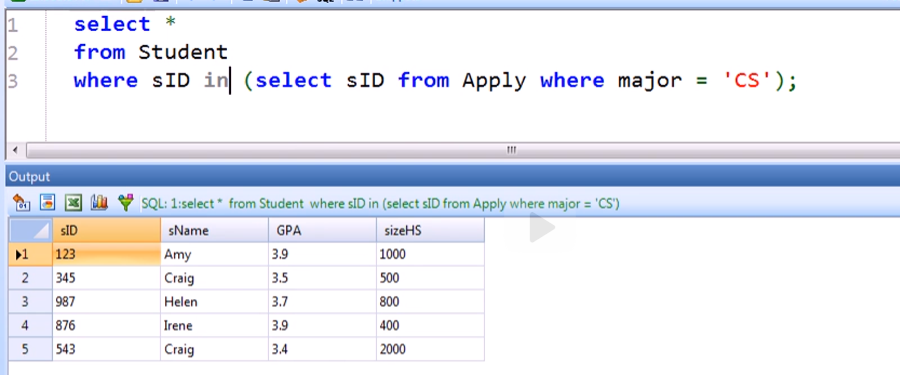
Getting the average GPA of students applying to CS





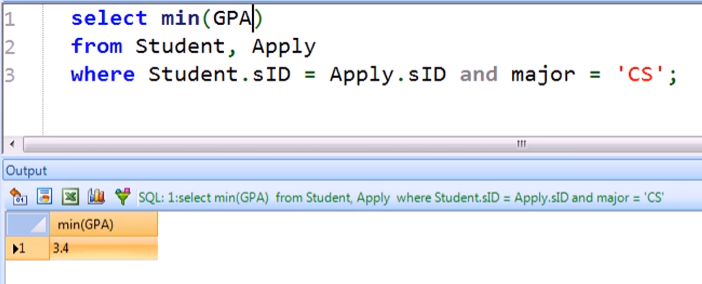
* Wrong! There are duplicate students because they apply in different college

Preventing duplicates in averaging:



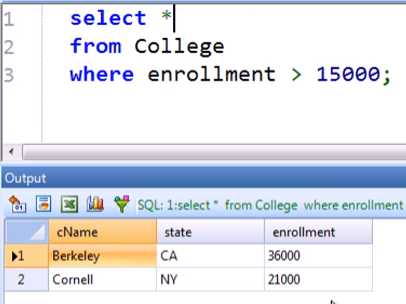
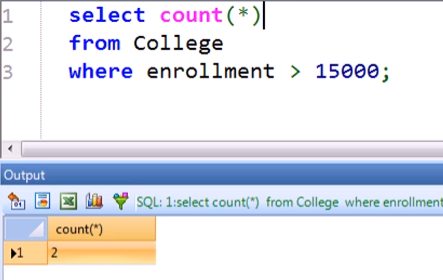


MIN

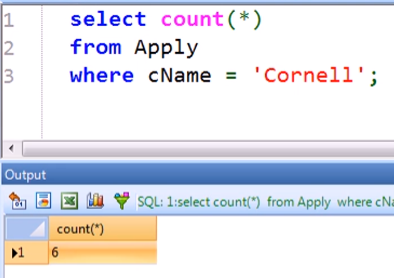
Getting the lowest GPA of students applying to CS

COUNT

Number of colleges bigger than 15000

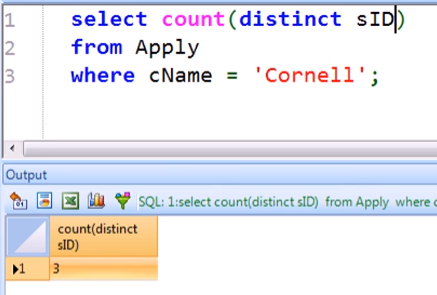
 

Number of students applying to Cornell

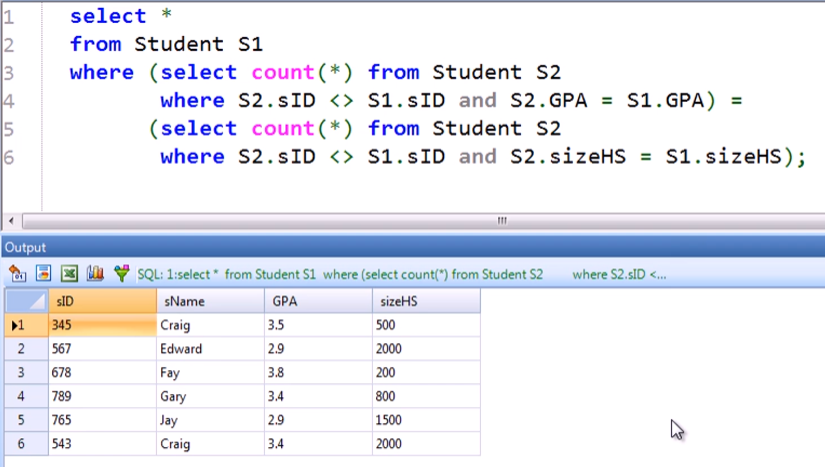


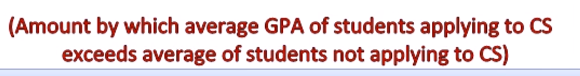
* Wrong! Similar to the averaging GPA where there 2 instances of students that applied to Cornell (maybe different majors)

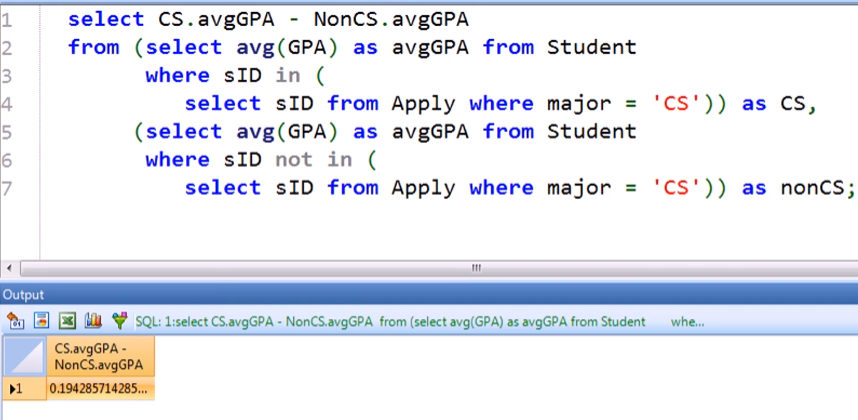
But this time we are not using subquery. We will use DISTINCT <attribute> inside the COUNT function



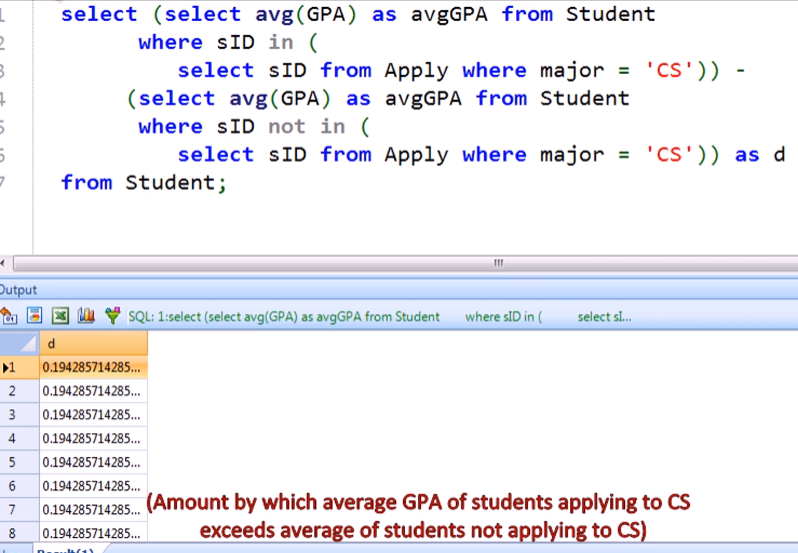








Rewriting the query above using subquery in SELECT clause

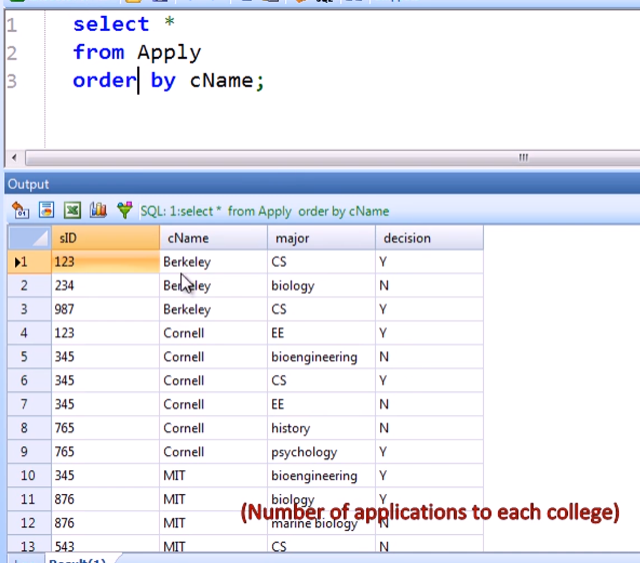


* We computed once for each tuple that is why we have many results

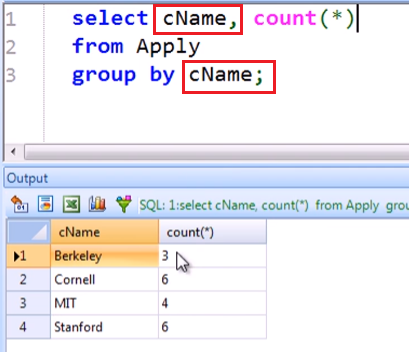
GROUP BY

* Only used in conjunction with aggregation

Initial data:

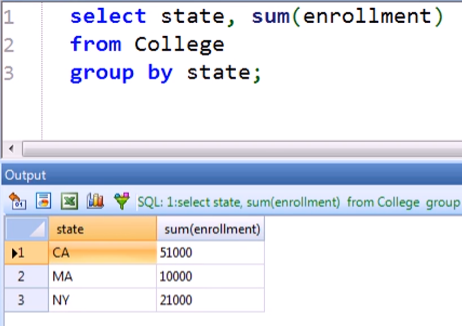


Applying COUNT and GROUP BY



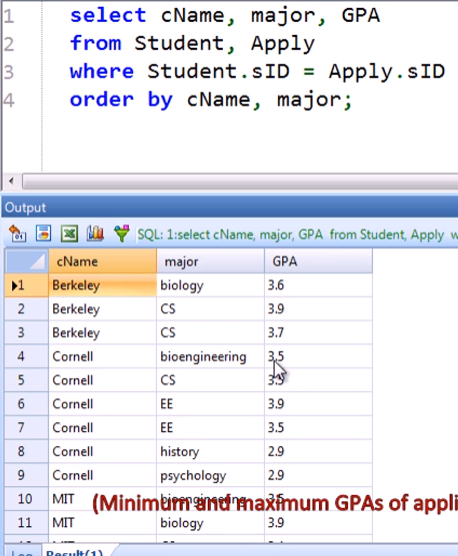
* Notice we can select columns that are in the group by clause

Next Example: College enrolments by state

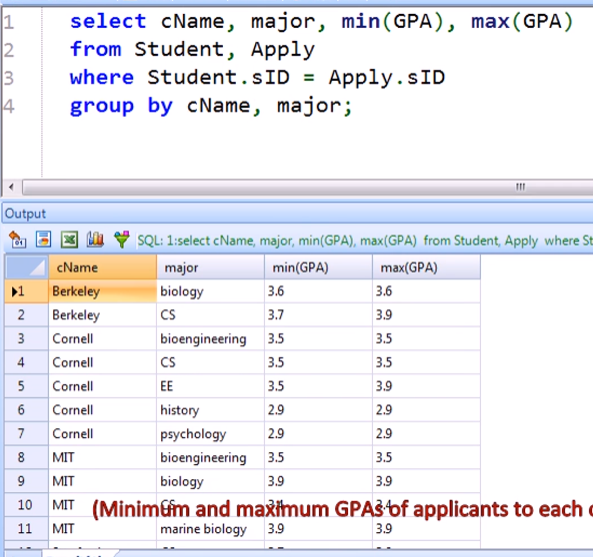


Next Example: Min and Max GPAs of applicants to each college and major

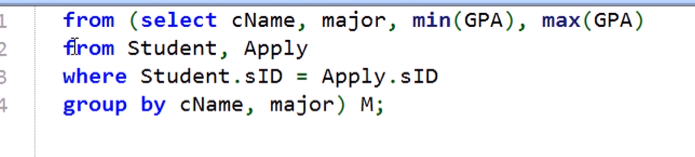
Initial data:



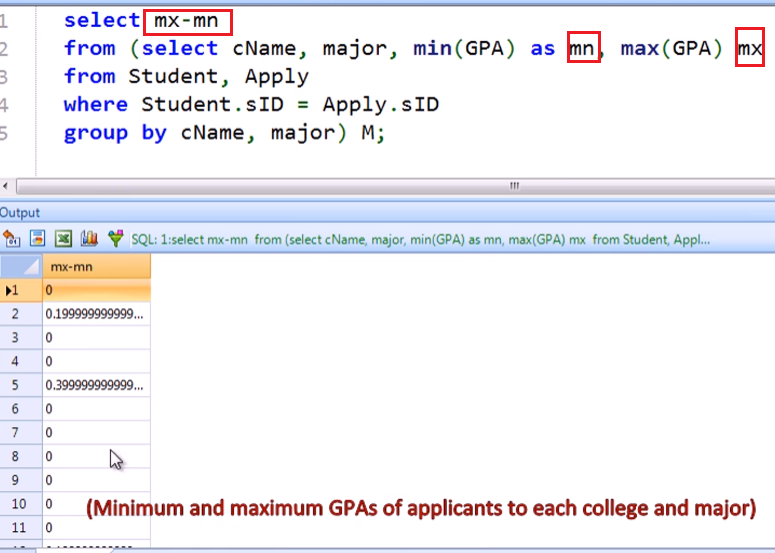
Applying the minimum and maximum:



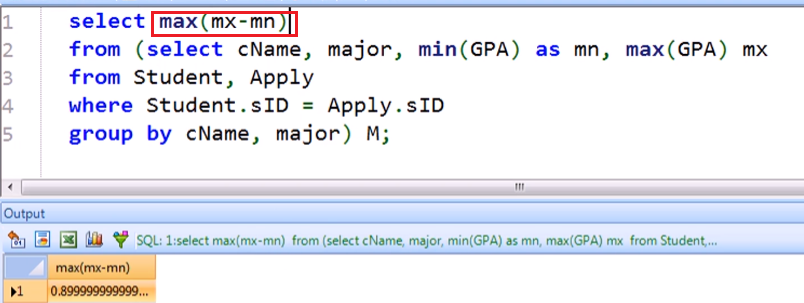
Using the result table above as our FROM table:



Then selecting the min and max from that result table to find the spread:

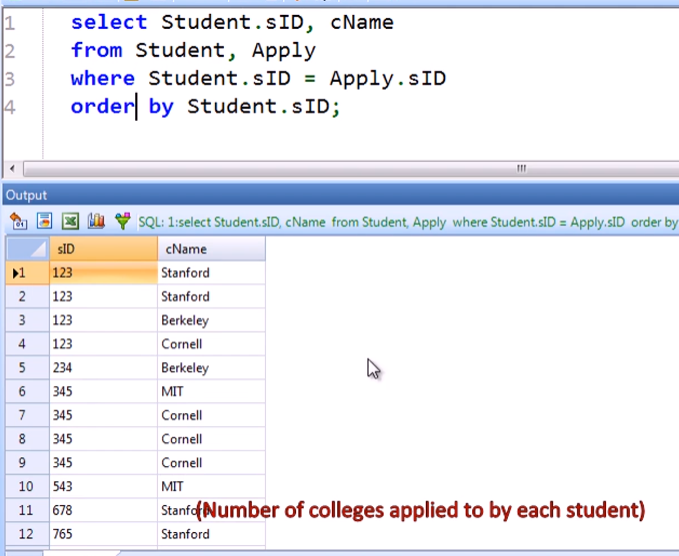


Then finding the maximum spread:

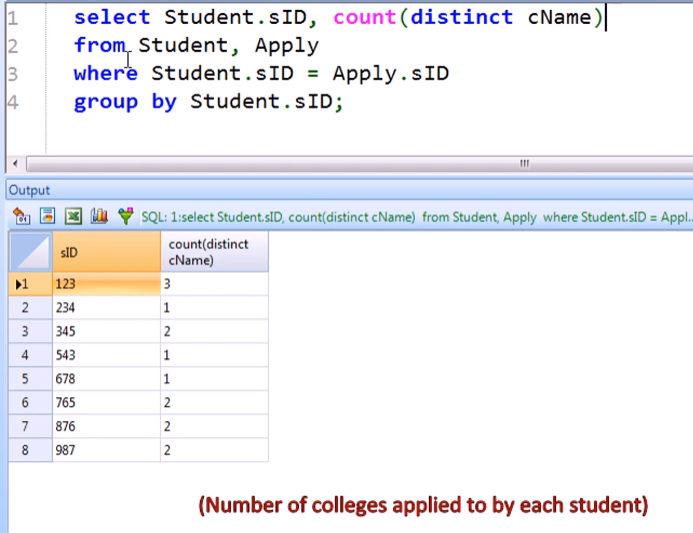


Finding Number of colleges applied to by each student

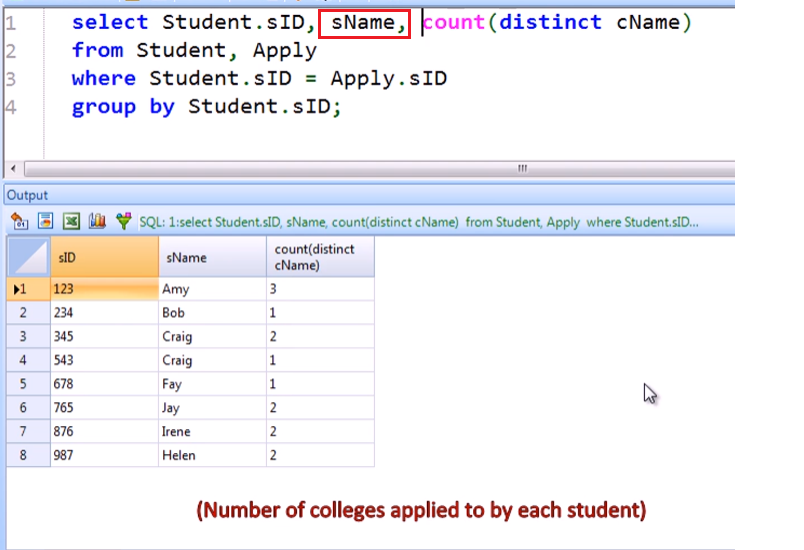
Initial query:



Apply COUNT

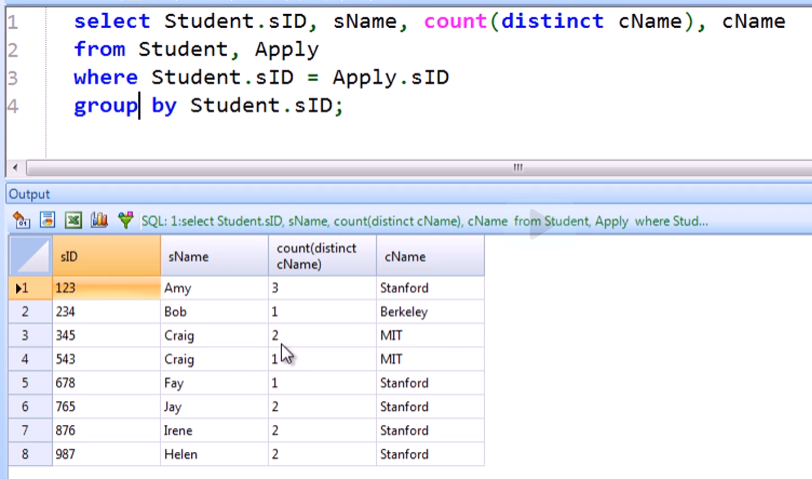


If we want also the student name to be displayed:

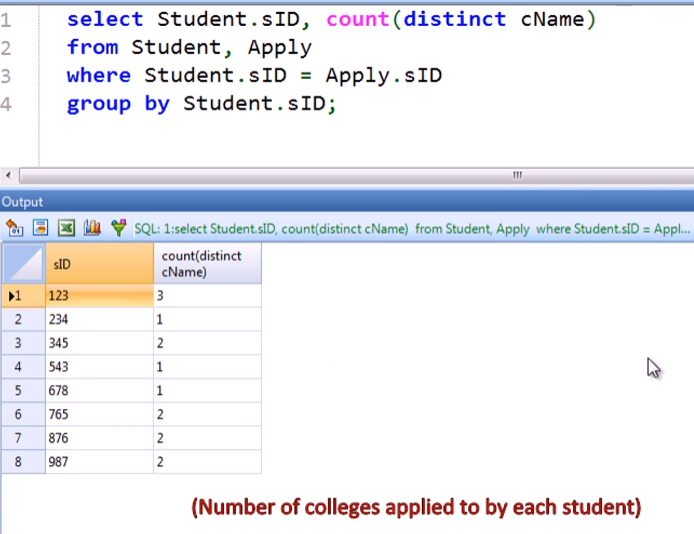


* This works here perfectly fine even if student name is not in the group by clause because we have unique values of student name per student id

If we add the college name in the SELECT clause, we will not get an error but we will not see the whole data

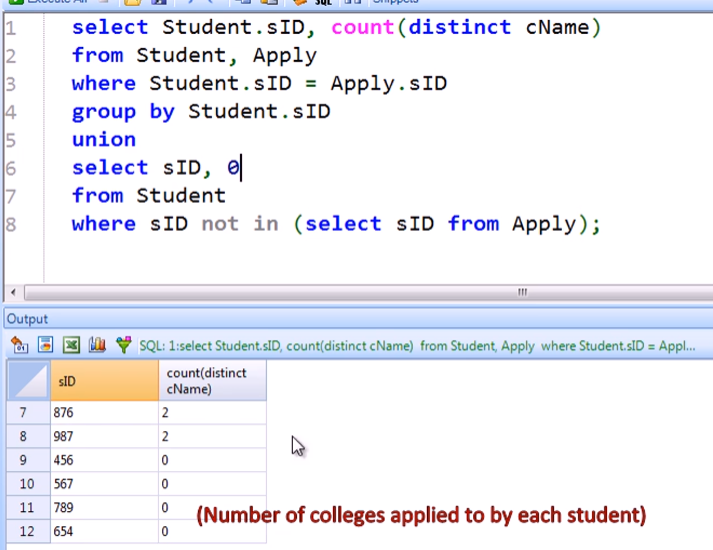


* For example, for Amy, she applied to 3 distinct colleges but Stanford is the only one that is being displayed



* Students who have not applied to any colleges will not be represented in our result

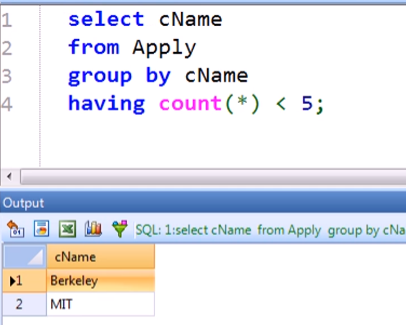
If we want to include those students:



HAVING clause

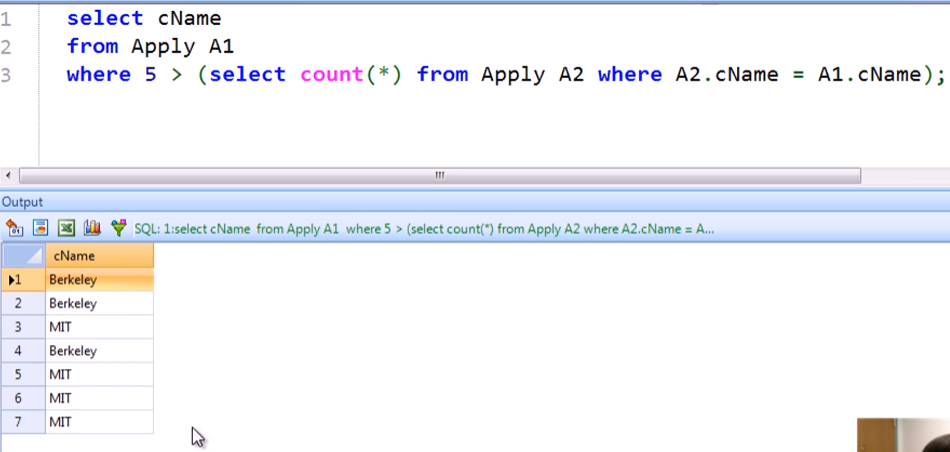
* Only used in conjunction with aggregation
* After the GROUP BY clause
* Check conditions that involve the ENTIRE group vs the WHERE clause that check conditions one record at a time



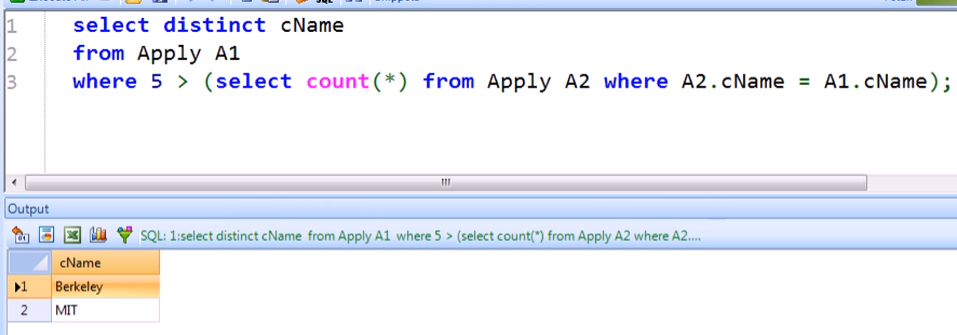


Rewriting without using GROUP BY and HAVING



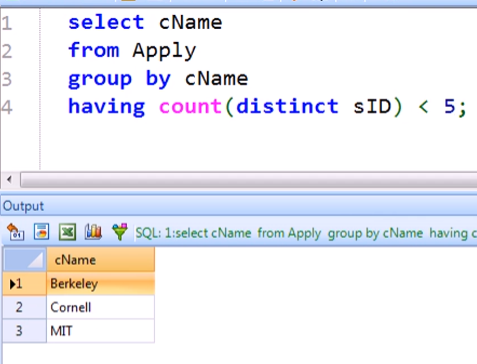


* This is because we are running for EACH application record (using WHERE)



* Just add distinct to remove dupes

Colleges that have fewer than 5 DISTINCT APPLICANTS



* Cornell have at least 5 applications but only have less than 5 applicants!



