

Documentation

Group 33

- a. At first, we created all the tables that were indicated in the Milestone file. Only for loading the data of the courserRegistrations table there were some difficulties with grade, since it needed to be loaded as an int excepting null values.
We added all primary keys and foreign keys to the tables, as instructed in the file. However, we only kept the primary keys since the foreign keys do not make the queries run faster and it saves time for creating indices and views.
We created two indices, idx_student on StudentRegistrationsToDegrees (StudentRegistrationID), and idx_course on CourseOffers (CourseOfferId), since this helps speed up the performance of the queries.
We created two materialized views. One view to collect all grades per course a student is registered in, and one view to collect all the ects per student (for different degrees). This helps speed up the performance of the queries as well since we join a lot on these columns.
Furthermore, we also experimented with adding the columns StudentId to CourseRegistrations.
We also tried creating a hash index on StudenRegistrationId to speed up the first query. However, there wasn't enough time to see if it worked.
- b. After the extension of the deadline, submitting our files to the server kept giving the error 'Your result is error, Total = 1000 sec'. This was fixed at the date of the submission. Meaning we are not able to write down the exact numbers which show whether our performance speed increased or decreased. Furthermore, before and after the deadline Ubuntu gave us different speeding times at different days. Our queries took longer than at first. Therefore, these numbers are not reliable. It had to do with the amount of space within Ubuntu. One specific error looked like this: "could not write to hash-join temporary file: No space left on device", which occurred for query number 2 and 7.
We did notice that creating views and indexes speeded up the performance of the queries. We decided to leave out adding a column from a table to another table since using materialized views increase the performance speed of the queries more.