



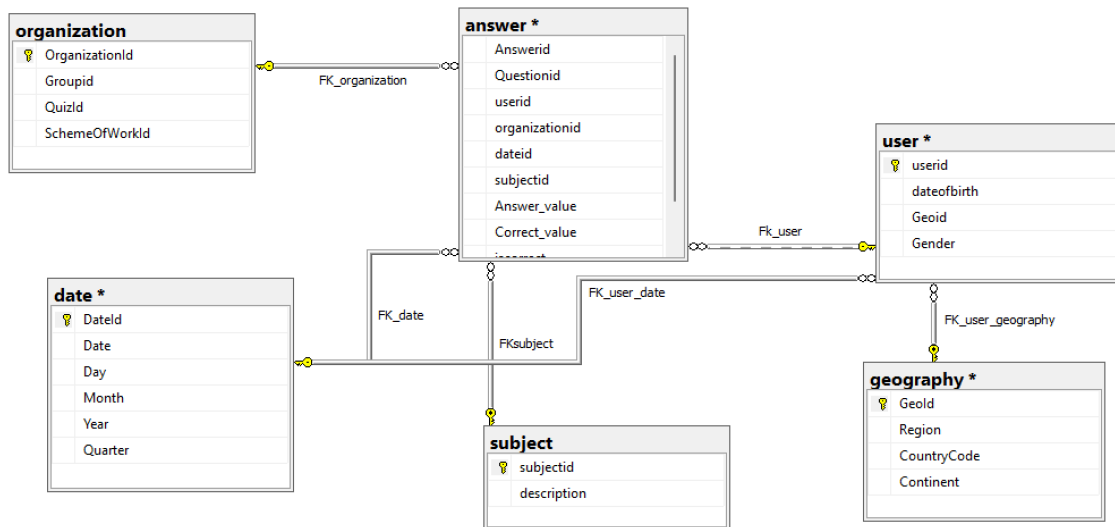
UNIVERSITÀ DI PISA

LUCA PALLA - 533605

GIULIA CALVO - 544434

LDS- GROUP 21\_DB: FIRST MIDTERM

## ASSIGNMENT 0



Following the first assignment we first set up the whole schema on Sql Management studio to respect the in the explanation of the project. We set up all the keys and the data type for each table. To do our work easily this assignment was strictly connected with the output of assignment 1.

## ASSIGNMENT 1

At first, we focused our efforts in creating those tables which didn't require external keys: these tables are Organization, Date, Geography and Subject. For each table we added an artificial incremental primary key and duplicated were dropped.

In particular for Date, special primary key was implemented: it is to the form “*number-a*” when it refers to date of answer and “*number-b*” when refers to date of birth (i.e. 1-a, 1-b).

The main properties of the tables are:

- Organization:
  - rows: 24640
  - features: OrganizationId, GroupId, QuizId, SchemeOfWorkId.
- Date:
  - rows: 596
  - features: DateId, Date, Day, Month, Year, Quarter

- Geography:
  - rows: 76
  - features: GeoId, Region, CountryName, Continent
- Subject:
  - rows: 412
  - features: SubjectId, Description.

After this task, also User, the last tables linked to the fact table was constructed. It doesn't have an artificial primary key because in our opinion it was not a problem manage the existent key also for future upload or update.

The properties are the following:

- User:
  - rows: 13630
  - features: UserId, DateOfBirthId, GeoId, Gender.

The last point concerning development tables in python regarded the fact table "Answer". In this case all the rows were included and therefore all primary keys of other tables were added as foreign keys.

Between the Answer's features one of the most important is "IsCorrect" which was calculated from the comparison of AnswerValue and CorrectAnswer. When these two features matched (the answer is actually correct) and 1 was assigned to IsCorrect, otherwise 0.

At the end of our work the Answer table have these properties:

- Answer:
  - rows: 538835
  - features: AnswerId, QuestionId, UserId, OrganizationId, DateId, SubjectId, AnswerValue, CorrectAnswer, IsCorrect, Confidence.

The next phase regards the upload of the content using a python script.

## ASSIGNMENT 2

At this point, knowing all the info about the tables and organizing them in Sql Management studio we started with population of data from the csv created.

In WORKING\_CODE-iterative you can find in the first rows the connection with our database and the creation of a cursor; right after we decided to create a “loading” function that took what is in the dictionary and upload it in the tables depending on the final table of destination of that row; at the end using an empty dictionary we uploaded all the tables restarting kernel

We choose the empty dictionary to “control” in a certain way what we were doing; indeed, the last part could have been uploaded iteratively with a for loop.

At this point everything was on the server in order to be used for part 2 of the project.