

COMPX323 Milestone 2

March 19, 2021

General Information

Milestone 2 consists of five parts and you have seven weeks to complete it (including the teaching recess). So, your group should aim to complete about one part each week.

Relational Schema

Develop the corresponding relational model (schema) for your complete ER-Design from Milestone 1. Mark key attributes by underlining.

Relations in Oracle

Select 2 or 3 main relations from Part 2 above and generate tables in Oracle. Declare both keys and foreign keys. You may also need to create several minor tables so that the foreign keys can work correctly. You may want to read through all of the tasks for Milestone 2 before deciding on which tables to select.

Test data sets

Create two data sets. One will be a small dataset containing realistic data, while the other one will contain more data, but can be filled with somewhat-realistic dummy values. Create your data sets in a way that allows you to easily re-load the data as needed.

Small dataset: For each relation, generate a file with about 5-10 meaningful records. Your documentation should include a screenshot showing table contents (generated with select *).

Large dataset: Either write a program (in a language of your choice) that generates large files with random data for your table, or use a data generator tool. The data needs to be sized and formatted similar to "regular" records (e.g. long strings). The data should consist of at least two related tables and at least *several hundred thousand* tuples in each table. Take care that the data conforms to the previously defined functional dependencies.

Documentation is to contain a description of how you created your data, as well as a screenshot that shows the successful loading of the data.

Application

Code up a simple interactive application for your project, using Java, C or another programming language of your choice (no scripting). Your program should present the user with the following action sequence:

- Some data input is required from the user.
- Depending on the input the system generates a query.
- The system access the database in order to issue whatever queries or modifications are required.
- Result data and/or an appropriate confirmation message are displayed to the user.

Your system input needs to be error proof with appropriate user messages. You need to explore the appropriate options for your programming language. Simple text boxes, drop down boxes, and buttons should be sufficient to capture most user input. The interface needn't be highly sophisticated (but have basic solid interaction design) as we are primarily concerned with the implemented functionality. The assumption is that this simple interface would be replaced with a web interface later (not part of your project).

Documentation is to include screenshots that show your application in action. Presentations of the software will be organised later.

Indexing

Indexing will be discussed in the week after the teaching recess.

Write a query that creates a high load on your database (when using the large dataset). Note that Oracle automatically creates an index for each key attribute. Therefore you don't need to explicitly create indexes for key attributes.

Explore creating indexes to speed up your query. Analyse the use of the index using the query plan. Documentation is to include a script that documents creation of the index, as well as the relevant performance measurements (time elapsed) for query processing with and without indexes. (you can use the command `set timing on` in `sqlplus`). Take care not to measure the cache effect.

Deadline: Friday 7th May