

COSC 344

Assignment 2

Overview

In this assignment, you will turn your ER diagram into a set of Oracle tables, normalize your design, and populate your database.

Due date for assignment 2

~~Friday, 27 August at 4 PM~~ **Extended to Friday, 10 September at 4pm**

Description

1. Relational Schema

Revise your ER diagram based on the feedback for Assignment 1, and then use the 7-step algorithm introduced in Lecture 4 to convert the revised ER diagram to a relation schema. The result should be similar to the diagram discussed in Lecture 4. Underline the primary key, and show all attributes except the derived attributes. As we usually don't store the value for derived attributes, there is no need to consider them for schema creation and normalization. For referential integrity constraints, draw an arrow from the foreign key terminating on the primary key it refers to.

2. Normalization

Turn your schema diagram into a set of tables in BCNF. You should go to 1NF, 2NF, 3NF, then BCNF. It might happen that a relation may be already in a given normal form (e.g 3NF). In such a case, you just explain why it is in that normal form. Explain your steps in a manner similar to the example given in the attached sample for the assignment report. If you cannot get to BCNF, explain why. Your write-up should be reasonable to follow and include no jumps in logic – it's possible that your BCNF schema may be a 'good representation' of your miniworld, but 'not be correct' based on your initial ERD.

3. Create Database

Create an SQL script which will create your tables in their BCNF form. The CREATE TABLE commands should be as complete as possible. This means where possible they should:

1. specify domains if some subset of normal data types is appropriate
2. specify the primary key
3. specify uniqueness, not null, and/or a default where appropriate
4. referential integrity constraints where appropriate

Depending upon your constraints, the order of table creation may be important.

Put a series of DROP TABLE commands ahead of the CREATE TABLE commands in your script so that the database can be reloaded without any error. Again the order may matter.

Populate your database with a series of INSERT commands. These should follow the CREATE TABLE commands. If at all possible, the INSERT commands for a given table should be grouped together and followed by a COMMIT.

The script described above must be called `load.sql` and is a deliverable. It should execute without errors. The allowed exception is a DROP TABLE error when the table does not exist.

Teamwork Model

The team leader coordinates the allocation of the tasks, and makes sure the task allocation is fair to every team member. For example, each member should map at least two entity types and two relationship types. Each member should normalize at least one relation and create at least one table.

At the end of the report there should be a summary of the teamwork, showing which part was done by which member, and which parts have been discussed by all group members. You can add any comment on the teamwork in your group.

Assignment submission

Each team needs to submit your assignment via Blackboard. You can find the link to the assignment submission page, named “Assignment submission ” in the course menu of the COSC344 paper in Blackboard.

You need to submit the following two files:

1. the `load.sql` file;
2. Assignment report that contains
 - Revised ERD and explain what has been changed. If there is no change, just include the one you designed for Assignment 1.
 - Relational schema diagram to show integrity constraints (refer to the example given in slide 23 of Lecture 3).
 - Normalization description.

You can resubmit before the due date if you wish -- your last submission is the one that will be marked.