

### Objective

The purpose of the Semester Database Project is to give you the opportunity to design and develop a database from start to finish.

Deliverable 1:	Write requirements and specifications for the project
Deliverable 2:	Design the ER, schema, and data dictionary
<b>Deliverable 3:</b>	<b>Implement the database, populate it with records, demonstrate its features, and provide documentation.</b>

Grading: The entire project is 20% of your final course grade (Deliverable 1 = 4%, Deliverable 2 = 7%, Deliverable 3 = 9%).

### Deliverable 3 Instructions

1. Refine the written specifications, ER diagram, relation diagram, and data dictionary for your database. You may add or modify entities and relationships as suggested in the grading comments, or as required to bring your project into the right scope of having between 8 to 12 tables. As in the previous deliverable:

- Your written specifications should be no longer than 1,500 words and should focus on the entities, attributes, and relationships that are important to your database.
- Your ER diagram should adhere to the notations we have used in class, should have a good layout and be easy to read, and should fit on one page.
- Your relation diagram should follow the notations we have used in class, including all relations, attributes, primary keys, and foreign key references.
- Your data dictionary should be broken into sections for each relation. For each relation, for each attribute, you should have a table with the attribute name, datatype, domain, and a brief description. The domain of each attribute should indicate the values it can have. This can be done either by specifying or enumerating the range, or in some cases, by indicating the amount of space for the datatype.

2. Implement your database in SQLite. Based on your ER and relation diagrams and data dictionary, create tables, specify primary and foreign keys, and include any other necessary integrity constraints such as checks. As part of Deliverable #3, turn in a list of the SQL DDL commands (e.g., create table commands) that you used to create your database.

3. Populate your database with data. You may do this either by writing SQL insert statements to load data, or by using the data entry or import features of SQLite. You need to load enough data to illustrate how your database works. At a minimum, load at least three records for each table. However, you may need to load more records to illustrate that particular queries work correctly. You do NOT need to include any information about the data loading as part of your report.

4. Write and run queries. Ideally, these queries should be based on the queries that you outlined in Parts 1 and 2, but if you need to make adjustments to your original queries, that is okay. Write and run the queries in SQLite on your database with the data loaded. You should run between 5 to 10 queries and your queries should exercise all parts of your database. You should be sure to include queries that involve joins, aggregate functions, and nested queries. **Part of your grade will be based on your choice of queries.** In other words, if all your queries are simple, your grade may be lower. For each query, clearly state the query in English, then list the SQL query you wrote for it, and finally, list the output from the database in response to the query. From SQLite, after you have run a query, you can select Actions → View CSV and then copy the results as text that you can paste into your word processing document. If you write a query that returns a large set of results (>20 items), only include the first 20 results in your report.

### **Files to Turn In**

You will need to turn in TWO files for this deliverable.

Turn in ONE PDF FILE that contains the following clearly labeled sections:

- 1) Database Specifications
- 2) ER Diagram
- 3) Relation Diagram
- 4) Data Dictionary
- 5) Implementation (create table statements)
- 6) Queries

In addition, you will turn in a file that contains a copy of your database. When you create your database in SQLite, it asks you to indicate a folder to save the database. In that folder, you will find a file with your database name with a .sqlite extension. For example, for the library database we used in class, I named it librarydb, so it would be stored in the file librarydb.sqlite. This .sqlite file is the second file that you should upload to Sakai.

### **Grading:**

Each group will get a grade on Part 3. I will be looking for clarity of your specifications, how well your specifications address issues from Parts 1 and 2, the clarity and accuracy of your ER and relation diagrams, and the quality and correctness of your data and queries.

### **How to turn in your assignment:**

Prepare your assignment using a word processor and diagram drawing software. Select ONE group member to submit your assignment. Save your assignment into a SINGLE PDF file. Your report should be in 10 or 12 point font. Name your file according to the following convention:

onyen1-onyen2-p3.pdf

Replace *onyen1* and *onyen2* with the actual Onyens of your group members (e.g. if I was in a group with Mickey Mouse, our assignment would be rcapra-mmouse-p3.pdf). The character between *youronyen* and the “p3.pdf” part should be a single minus sign (not an underscore). You could also call this character a dash. There should be no spaces or other characters in the filename. Files with names that do not follow this convention will not be graded.

You will also turn in a copy of your database as stored in the .sqlite file. Sakai will be configured so that you can include TWO attachments for your assignment.

ONLY ONE of your team members should submit your two files electronically through the Sakai by going to the Assignments area and finding the “P3” assignment. After you think you have submitted the assignment, I strongly recommend checking to be sure the files were uploaded correctly. For the pdf file, be sure that you can view it by clicking on it from within Sakai. If I cannot open or view your file, I cannot grade it.

If for some reason you need to re-submit your homework file, you must add a version number to your filename so that we will know which file is the most recent. Sakai is configured so that it will only accept 3 total submissions. Use the following file naming convention if you need to re-submit:

Your first submission:    onyen1-onyen2-p3.pdf  
Your second submission: onyen1-onyen2-p3-v2.pdf  
Your third submission:    onyen1-onyen2-p3-v3.pdf

Sakai is also configured with a due date and an “accept until” date. Submissions received after the due date (even just 1 minute!) will receive a 10% penalty per day. The “accept until” date is 5 days after the due date. Submissions will not be accepted after the “accept until” date and will have a score of zero recorded.