

Project 2: Using SQL and MySQL

CS 347 - Spring 2026

The goals of this project are to install and obtain a working instance of the MySQL RDBMS, to create a sample database on this instance, and then to write and execute diverse SQL queries. Each step below will indicate what you are asked to submit as deliverables of the project (all via the Canvas assignment).

1. Installing MySQL

We are interested in installing two products for now: MySQL Community Server and MySQL Workbench.

- The [MySQL Community Downloads](#) web page shows the products available to the community. Notice the different connectors listed, which are used when we want to access a database from an application that we are developing.
- Choose each product to download, and then the OS of your computer (where you have full admin privileges) and the specific format that applies to you (based on your machine's architecture).
- Complete the download and installation of the two products listed above.
- For more information on MySQL products, features, and functions, as well as URLs for its family of products, see the [Useful Resources](#) page on Canvas. The first slide deck available there has detailed instructions for installing on MS Windows.

Two other sources of information when working with MySQL are:

- The [MySQL Bugs](#) database
- The [MySQL Forums](#) community (specially the 'Install & Repo' section)

Submit a text file (**report2.txt**) where you report your experience completing these installations (did you hit any snags? how easy was getting MySQL running?), on the Canvas assignment. Read more about this report file in step 2.

2. Creating the Sakila database (schema)

The Sakila database is intended to provide a standard schema that can be used as an example while learning SQL in general as well as features of MySQL such as Views, Stored Procedures, and Triggers. It is fully documented at <https://dev.mysql.com/doc/sakila/en/>. The ER diagram is attached to this Canvas assignment, as you probably noticed.

- Access our course repository: [cs347 repository](#)
- Download these two files: `sakila-database.sql` and `sakila-data.sql`
- Create the Sakila schema and its database objects by running `sakila-database.sql`

- Populate the database tables by running `sakila-data.sql`
- To run SQL code on MySQL you can use either the MySQL CLI or the MySQL Workbench application
- Review both SQL files used and document (* see below) the different types of database objects that were created, as well as the method used for populating the database

If there is a data type in the CREATE TABLE statements that you don't recognize, look it up in the online [MySQL documentation](#). This link takes you to a specific page in the Reference Manual, and from there you can navigate to other sections as needed (e.g., SQL syntax).

*: Add to the report file, from step 1, your observations on the SQL code you used to create and populate the database.

3. Creating and running SQL queries/statements

Create SQL statements for the following tasks, and store them in a (text) SQL file named **project2.sql**. Notice the requests for comments in this file, given in the tasks below.

Submit this SQL file on the Canvas assignment.

- Write queries to get a row count for **five of the tables** in the database (5 queries).
- Write queries to sample a few rows (showing **only a few of the columns**) from **five of the tables**, using the LIMIT clause (5 queries).
- Write a query (on any **one** table) that uses both a WHERE clause and an ORDER BY clause. Add a short comment above your SQL statement to describe the query.
- Write an INSERT statement to add a row into any **one** of the tables (*: make sure it is not a **View** by looking at its CREATE statement in the file `sakila-database.sql`). Add a short comment above your SQL statement to describe the statement.
- Write an UPDATE statement to update one or more rows from any **one** of the tables (see * above). Add a short comment above your SQL statement to describe the statement.
- Write a DELETE statement to delete one or more rows from any **one** of the tables (see * above). Add a short comment above your SQL statement to describe the statement.

Submit screen shots **of the results** of each of the above tasks (query executions), they may be combined into a pdf or zip file, **results2.pdf/zip**, on the Canvas assignment.

4. See the grading rubric on the next page.

Notice we expect 3 files in your submission (.txt, .sql, .pdf/.zip).

Project 2 Rubric

Download and install the two MySQL products. Observations written in the report2.txt file.	20
-10 no observations for each product -5 incomplete observations for each product	
Create the database based on <code>sakila-database.sql</code> . Populate the tables from the <code>sakila-data.sql</code> file. Observations also written in the report2.txt file.	20
-20 missing Sakila database (per the screen shots below) -5 incomplete observations about each SQL file	
Run five queries to get row counts of five tables in the database. Code is to be found in the project2.sql file. Output of these (and below) queries is to be found in the results2 file.	15
-3 for each missing query and output	
Run five queries to sample the data from five tables using the <code>LIMIT</code> clause.	15
-3 for each missing query and output	
Run a query that includes a <code>WHERE</code> clause and an <code>ORDER BY</code> clause. Include a short comment above your query to explain its function.	10
-4 missing WHERE clause -4 missing ORDER BY clause -3 missing output -2 missing comment or if comment is not descriptive	
Run three other CRUD operations. Include a short comment above each statement to explain their function.	
<ul style="list-style-type: none"> • An <code>INSERT</code> statement into a table • An <code>UPDATE</code> statement on a table • A <code>DELETE</code> statement on a table -5 for each incorrect statement -3 for each missing output -2 for each missing comment or if comment is not descriptive	20
Total Credit:	100