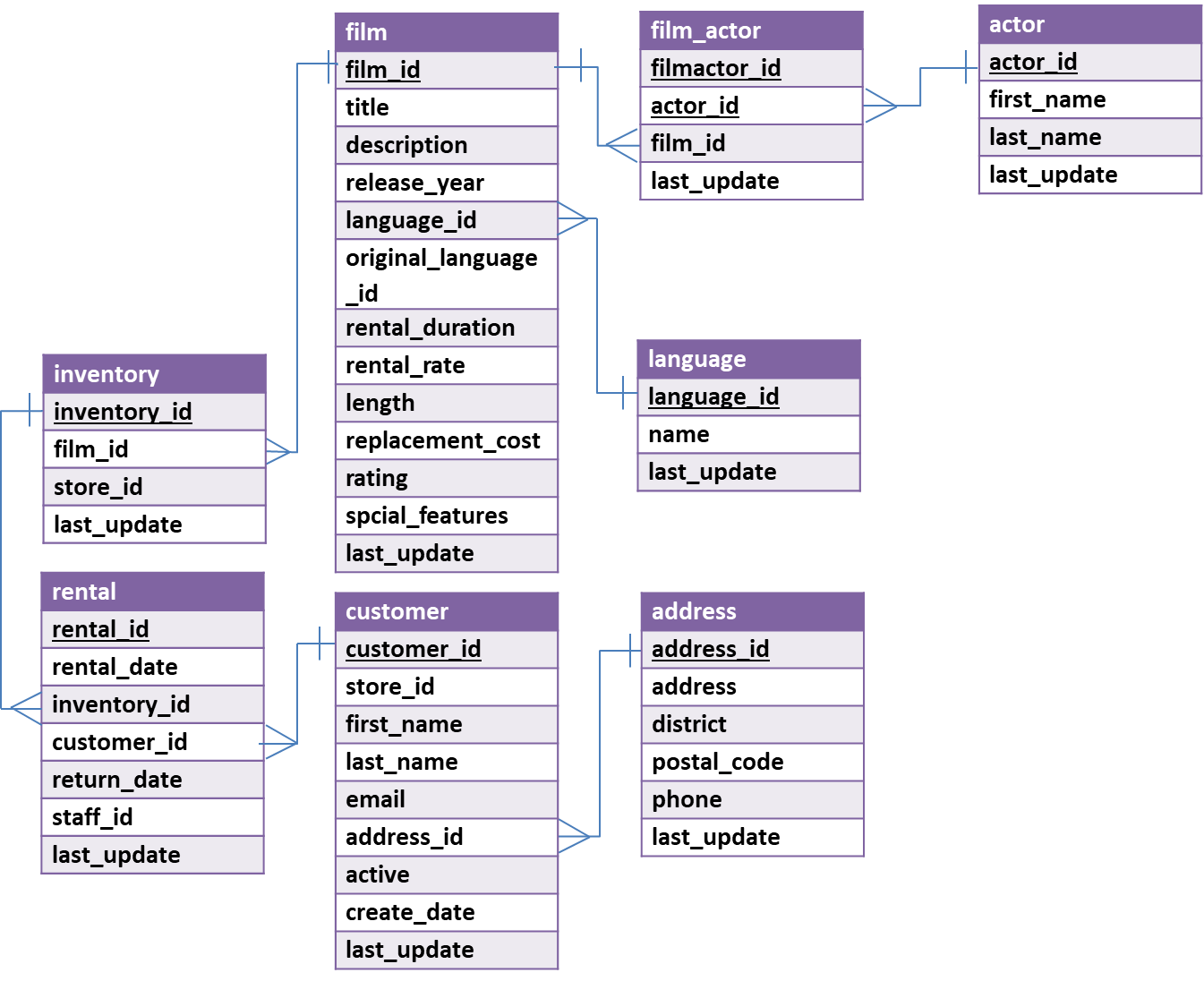
**In-class Activity #3: Working with SQL, Part 1**

**Getting Information out of a Database**

|  |
| --- |
| **Submission Instructions**  Submit your solutions for Part 2 of this exercise as a word or pdf file through **Canvas>Assignments>To-Do.** |

To do this exercise, you will be working with a movie rental database. The schema for this database is provided on the accompanying document. All of the tables are in a schema called simply “**moviedb**” (not m0moviedb, m1moviedb, etc.). You can’t write to any of the tables – you can only use SELECT statements to read from them (so don’t worry about causing any damage).

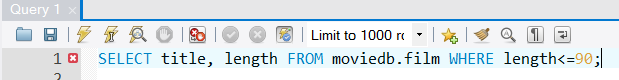
Spend some time looking at the schema carefully. The field names are pretty self-explanatory. For example, here are three tables:



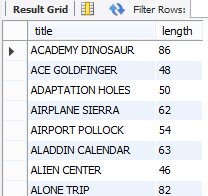
You can see that a film has a title, description, rating, and length (among other things). You can also see that an actor has a first name and a last name. The film\_actor table implements the many-to-many relationship between actor and film (i.e., a film can have more than one actor, and an actor can be in more than one film). You’ll also notice that data types are listed for each field, but they should be pretty obvious – for example, first\_name is a VARCHAR because it is a string value.

**Part 1: Try out some simple queries**

In the Query 1 pane, type the following (make sure you do not have any typo). And then click the Execute SQL Script button (the lightning bolt):



In the output tab, you’ll see this:



**Part 2: Write queries on your own (Submit your solutions for Part 2 only on Canvas)**

Create the SQL SELECT query to answer each of the questions below. Some of the questions can be answered by querying one table; others will require joining multiple tables to get the answer.

For each question you’ll need to write down **(1)** **the SQL query** (which you can copy and paste it from SQL Workbench) and **(2) the answer you get as a result of the query** (which you can copy and paste from the results).

1. Display all different ratings of movies in the database.

*Display: ratings*

1. How many customers have a last name that starts with ‘SH’ or ‘TH’?

*Display: number of customers*

1. Are R movies, on average, longer than PG movies? Prove it!

*Display: rating and average length*

1. What is the title and length of the longest movie in the database (assume there’s only one with the longest value and use LIMIT)?

*Display: title and length*

1. How many movies are there in each rating which has running time greater than 60? Sort the results based on the number of movies.

*Display: rating and number of movies*

1. How many movies are in French?

*Display: number of movies*

(HINT #1: The name field in the language table contains the film language names.

HINT #2: Surround the table name language with back quotes. (optional))

1. What is the postal code of a customer named “Ruth Martinez”?

*Display: postal code*

1. How many inventories are there for each film? Return the value for films that have no inventory.

*Display: film id and number of inventories*

(HINT #1: Use left join.)

**ANSWER SHEET**

| **Question** | | **SQL Query** | **Results/Answer from MySQL Workbench** |
| --- | --- | --- | --- |
| 1 | Display all different ratings of movies in the database. |  |  |
| 2 | How many customers have a last name that starts with ‘SH’ or ‘TH’? |  |  |
| 3 | Are R movies, on average, longer than PG movies? Prove it! |  |  |
| 4 | What is the title and length of the longest movie in the database? |  |  |
| 5 | How many movies are there in each rating which has running time greater than 60? Sort the results based on the number of movies. |  |  |
| 6 | How many movies are in French? |  |  |
| 7 | What is the postal code of a customer named “Ruth Martinez”? |  |  |
| 8 | How many inventories are there for each film? |  | (only show first 5 rows) |