**Matthew Abraham**

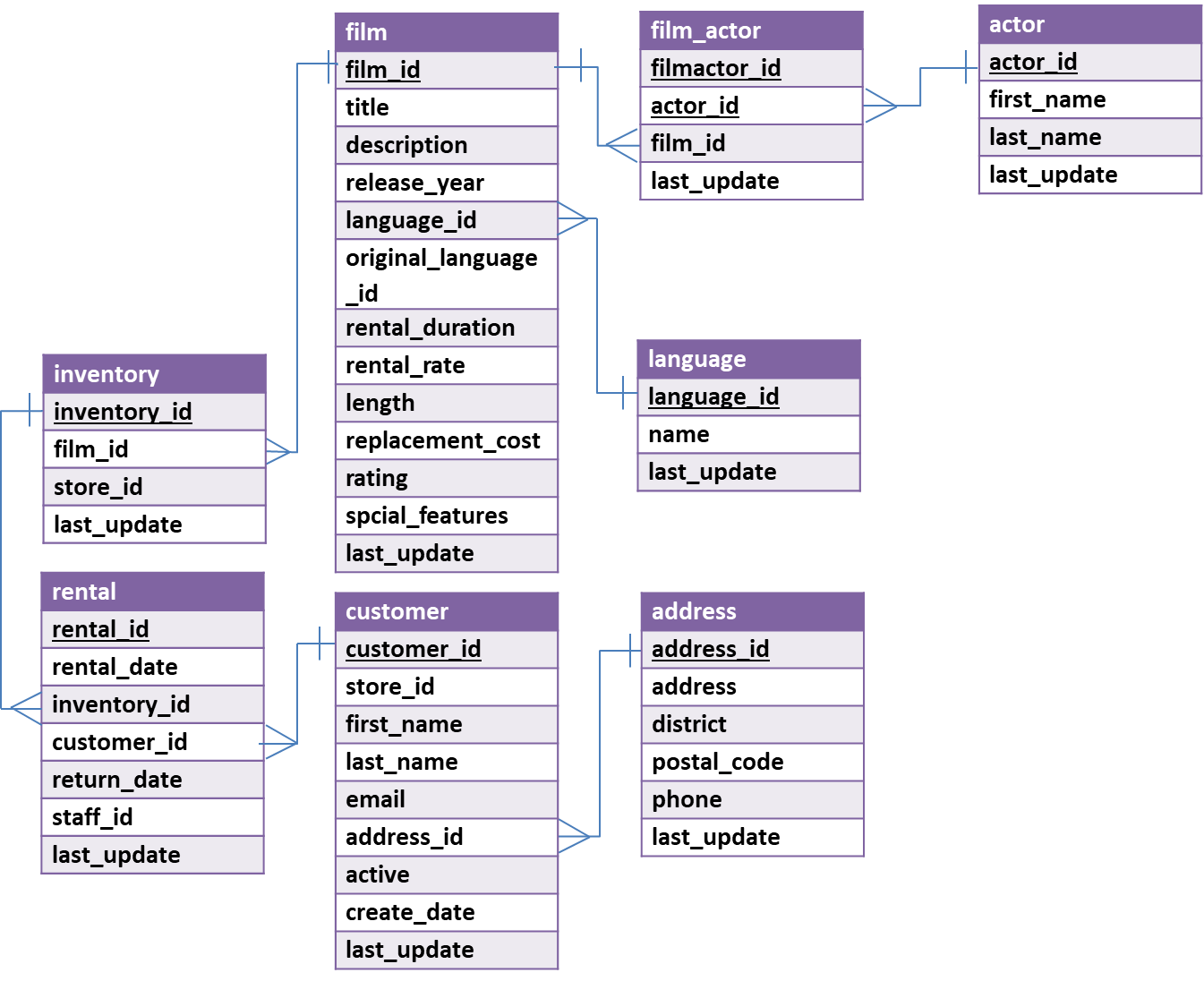
**In-class Activity #4: Working with SQL, Part 2**

**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 queries**

1. **A “subselect” statement**

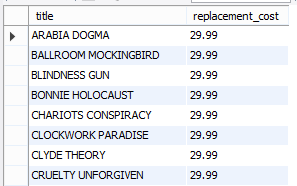
Finally, let’s do a query with a **subselect**:

SELECT title, replacement\_cost

FROM moviedb.film  
WHERE replacement\_cost = (SELECT MAX(replacement\_cost)

FROM moviedb.film);

This will return movies with the highest replacement cost (there are 53 of them), here’s the first few:



Here’s what’s going on:

|  |  |
| --- | --- |
| SELECT title, replacement\_cost  FROM moviedb.film | SELECTing the title and replacement cost of the movies from our film table |
| WHERE replacement\_cost = | Filtering based on replacement cost. We only want the movies where the cost is equal to its highest value |
| (SELECT MAX(replacement\_cost)  FROM moviedb.film) | We get the highest value with another SELECT statement that gets the MAXimum replacement cost from the film table |

**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. What is the title and length of the longest movie in the database? Assume there could be more than one movie with the longest value and use a *subselect* with the MAX() function to find the movies with the greatest length.

*Display: title and length*

*(Hint: You* ***should not use LIMIT*** *as below as* ***there******can******be******more******than******one*** *movie with the same longest value:*

SELECT title, length

FROM moviedb.film

ORDER BY length DESC LIMIT 1;)

1. In which films did Sandra Peck star (only return the first five)?

*Display: title*

1. How many times has the customer Adam Gooch rented?

*Display: number of movies*

(HINT: You need to use the customer table and the rental table.)

1. What are the two most popular first names of customers who live in postal code greater than 40000?

*Display: first name and the number of first names*

1. What’s the longest film among inventories in store 2?

(HINT: Use store\_id column.)

*Display: title*

1. What's the most expensive (in terms of rental rate) film by Penelope Guiness? And what is the rental rate?

*Display: title and rental rate*

1. How many different ratings are there in Film table?

*Display: number of different ratings*

**ANSWER SHEET**

| **Question** | | **SQL Query** | **Results/Answer from MySQL Workbench** |
| --- | --- | --- | --- |
| 1 | What is the title and length of the longest movie in the database? Assume there could be more than one movie with the longest value. | SELECT title, length FROM moviedb.film  WHERE length = (SELECT max(length) FROM moviedb.film) | 'CHICAGO NORTH','185'  'CONTROL ANTHEM','185'  'DARN FORRESTER','185'  'GANGS PRIDE','185'  'HOME PITY','185'  'MUSCLE BRIGHT','185'  'POND SEATTLE','185'  'SOLDIERS EVOLUTION','185'  'SWEET BROTHERHOOD','185'  'WORST BANGER','185' | |
| 2 | In which films did Sandra Peck star (only return the first five)? | SELECT title  FROM moviedb.film  JOIN moviedb.film\_actor ON film.film\_id = film\_actor.film\_id  JOIN moviedb.actor ON film\_actor.actor\_id = actor.actor\_id  WHERE first\_name = 'Sandra' AND last\_name = 'Peck' LIMIT 5; | 'ACADEMY DINOSAUR'  'BANG KWAI'  'BEETHOVEN EXORCIST'  'BEVERLY OUTLAW'  'BIRDS PERDITION' | |
| 3 | How many times has the customer Adam Gooch rented? | SELECT count(\*)  FROM moviedb.rental  JOIN moviedb.customer ON rental.customer\_id = customer.customer\_id  WHERE first\_name = 'Adam' AND last\_name = 'Gooch'; | 22 | |
| 4 | What are the two most common first names of customers who live in postal code greater than 40000? | SELECT first\_name, count(\*) FROM moviedb.customer  JOIN moviedb.address ON customer.address\_id = address.address\_id  WHERE postal\_code > 40000  GROUP BY first\_name  ORDER BY count(\*) DESC LIMIT 2 | 'JAMIE','2'  'LESLIE','2' | |
| 5 | What’s the longest film among inventories in store 2? | SELECT DISTINCT title  FROM moviedb.film  JOIN moviedb.inventory ON film.film\_id = inventory.film\_id  WHERE store\_id = 2 AND  length = (SELECT max(length) FROM moviedb.film JOIN moviedb.inventory ON film.film\_id = inventory.film\_id WHERE store\_id = 2); | 'CHICAGO NORTH'  'DARN FORRESTER'  'GANGS PRIDE'  'HOME PITY'  'MUSCLE BRIGHT'  'POND SEATTLE'  'SWEET BROTHERHOOD'  'WORST BANGER' | |
| 6 | What's the most expensive (in terms of rental rate) film by Penelope Guiness? And what is the rental rate? | SELECT title, rental\_rate  FROM moviedb.actor  JOIN moviedb.film\_actor ON actor.actor\_id = film\_actor.actor\_id  JOIN moviedb.film ON film.film\_id = film\_actor.film\_id  WHERE actor.first\_name = 'Penelope'  AND actor.last\_name = 'Guiness'  AND rental\_rate = (SELECT max(rental\_rate)  FROM moviedb.actor  JOIN moviedb.film\_actor ON actor.actor\_id = film\_actor.actor\_id  JOIN moviedb.film ON film.film\_id = film\_actor.film\_id  WHERE actor.first\_name = 'Penelope'  AND actor.last\_name = 'Guiness'); | 'ELEPHANT TROJAN','4.99'  'KING EVOLUTION','4.99'  'LADY STAGE','4.99'  'RULES HUMAN','4.99'  'WIZARD COLDBLOODED','4.99' | |
| 7 | How many different ratings are there in Film table? | SELECT COUNT(\*)  FROM (SELECT distinct rating FROM moviedb.film) as temp1; | 5 | |