**Josh Holloway**

**mySQL Install**: <https://youtu.be/0cNOD1KMJ40>

# **Assignment-1: MySQL Countries**

**Queries**

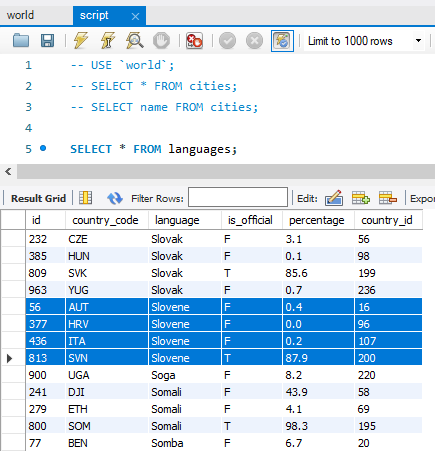
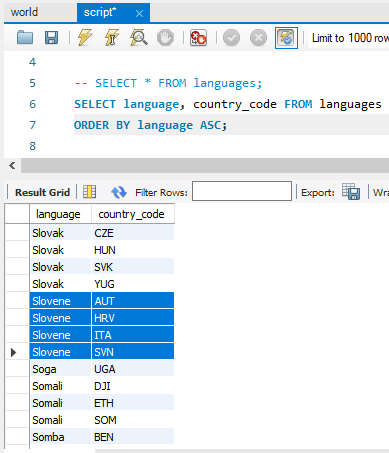
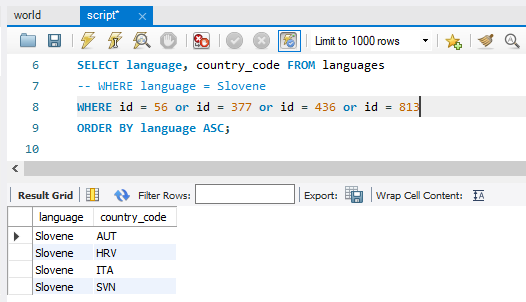
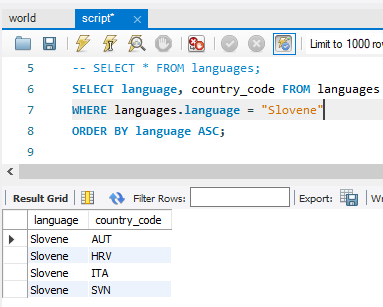
1. What query would you run to get all the **countries** that **speak Slovene**?

* Your query should return:
  + Name of the country
  + Language
  + Language percentage
* Your query should arrange the result by language percentage in descending order. (1)

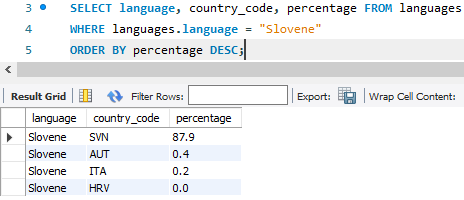
SELECT language, country\_code, percentage FROM languages

WHERE languages.language = "Slovene"

ORDER BY percentage DESC;

Steps taken to arrive at solution



2. What query would you run to display:

* **The total number of cities for each country**?
* Your query should return the **name of the country** and the **total number of cities**.
* Your query should arrange the result by the **number of cities** in **descending order**. (3)

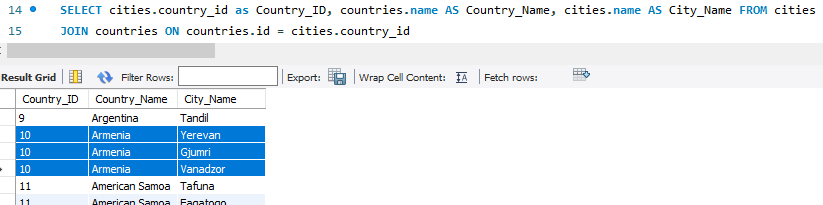
SELECT cities.country\_id as Country\_ID, countries.name AS Country\_Name, COUNT(\*) AS Number\_of\_Cities FROM cities

JOIN countries ON countries.id = cities.country\_id

GROUP BY countries.name

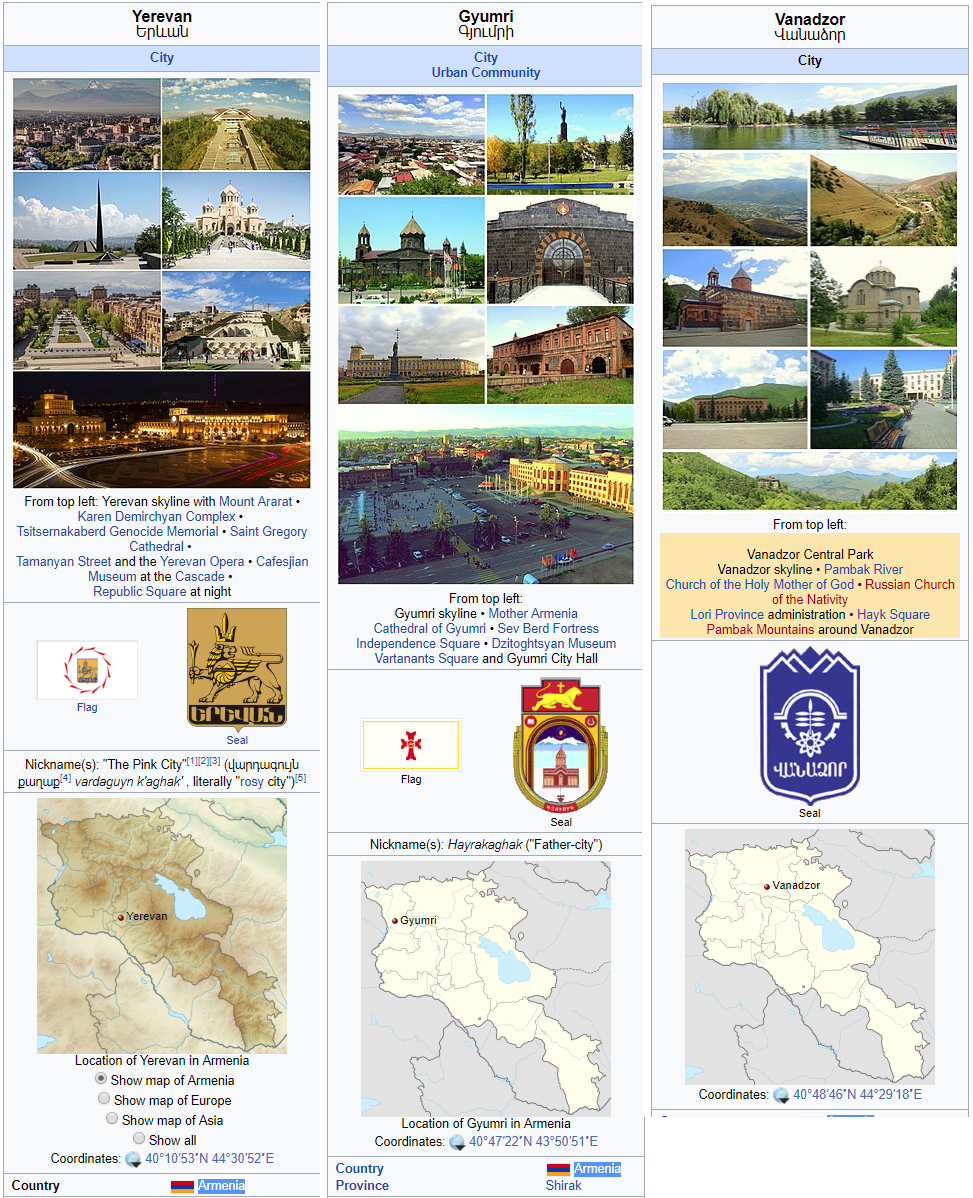
ORDER BY Number\_of\_Cities DESC;





- - - - - - - - - - - - - - - - - - - - - - - - - -





Cities used as a sanity check. [They are the 3 cities listed for Armenia]

3. What query would you run to get

* **all the cities**
* **in Mexico**
* **with a population of greater than 500,000**?
* Your query should arrange the result by population in descending order.

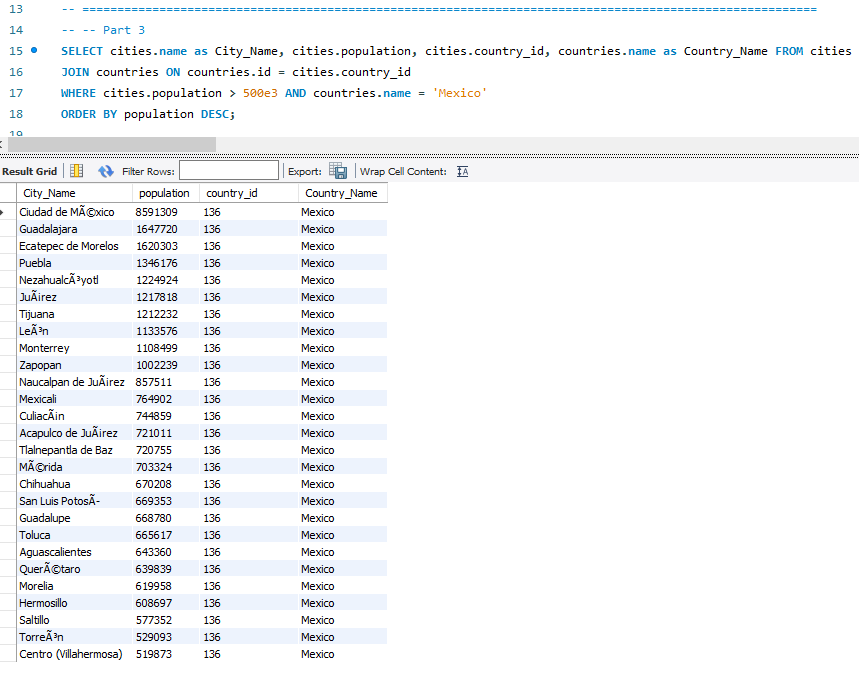
SELECT **cities**.name as City\_Name, **cities**.population, **cities**.country\_id, countries.name as Country\_Name FROM **cities**

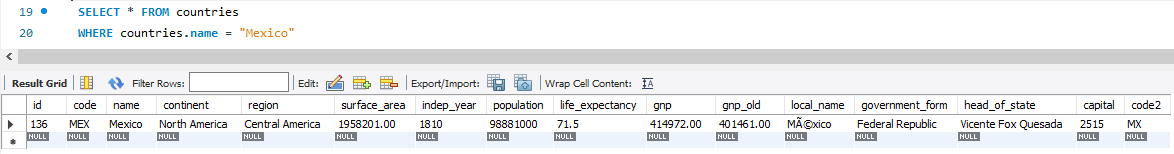
JOIN **countries** ON **countries**.id = **cities**.country\_id

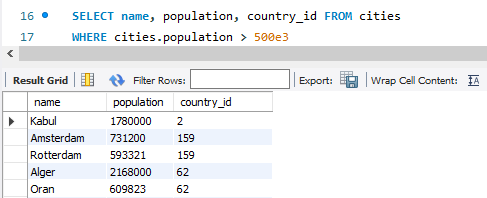
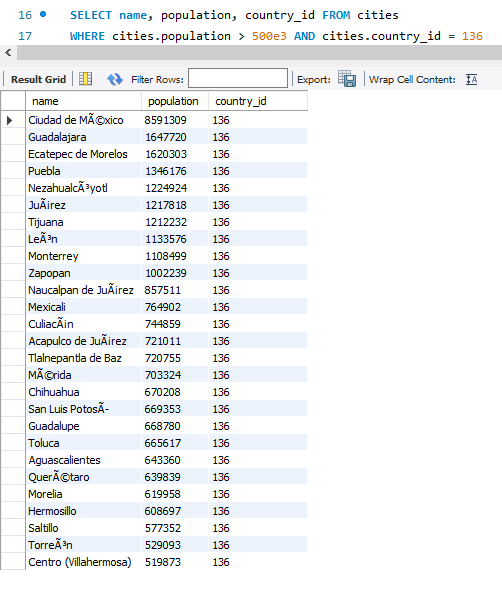
WHERE **cities**.population > 500e3 AND **countries**.name = 'Mexico'

ORDER BY population DESC;







Steps taken to work toward final solution.

4. What query would you run to get **all languages in each country with a percentage greater than 89%**?

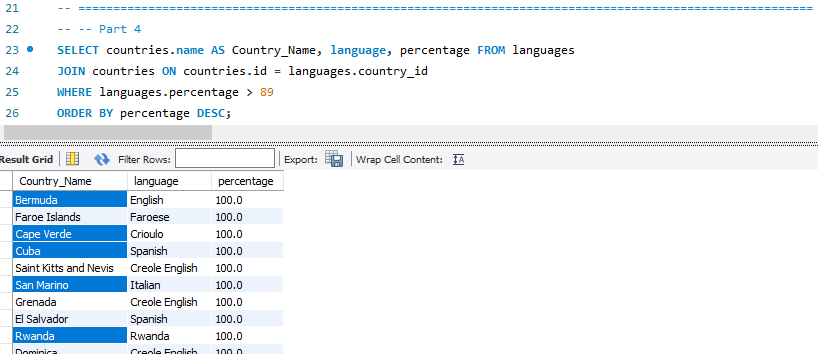
* Your query should arrange the result by percentage in descending order. (1)

SELECT countries.name AS Country\_Name, language, percentage FROM languages

JOIN countries ON countries.id = languages.country\_id

WHERE languages.percentage > 89

ORDER BY percentage DESC;

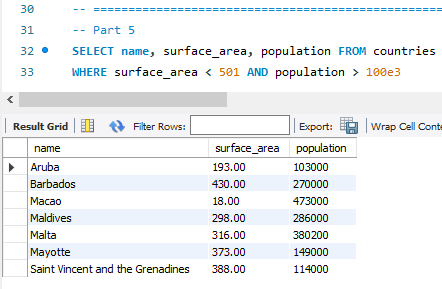


5. What query would you run to get:

* All the countries
* With Surface Area below 501
* And population greater than 100,000?

SELECT name, surface\_area, population FROM countries

WHERE surface\_area < 501 AND population > 100e3



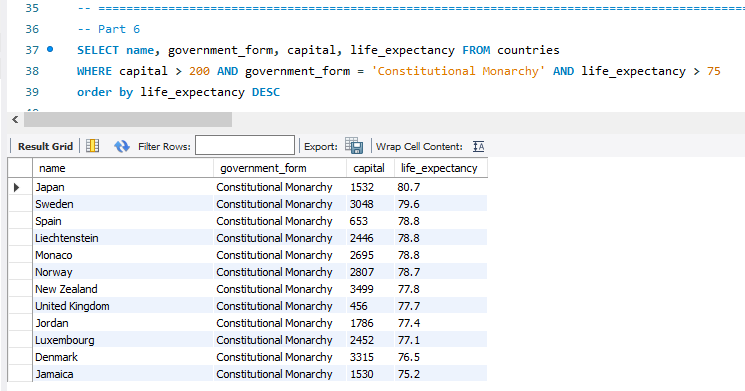
6. What query would you run to get:

* Countries
* With only Constitutional Monarchy
* With a capital greater than 200
* And a life expectancy greater than 75 years?

SELECT name, government\_form, capital, life\_expectancy FROM countries

WHERE capital > 200 AND government\_form = 'Constitutional Monarchy' AND life\_expectancy > 75

order by life\_expectancy DESC



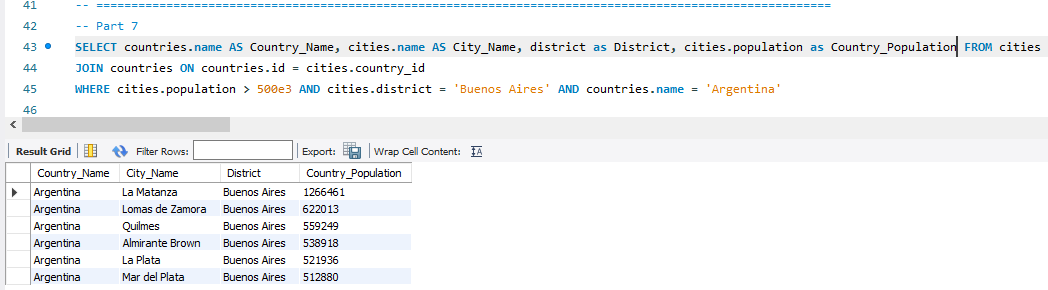
7. What query would you run to get:

* All the cities of Argentina
* Inside the Buenos Aires district
* And have the population greater than 500, 000?
* The query should return the Country Name, City Name, District and Population.

SELECT countries.name AS Country\_Name, cities.name AS City\_Name, district as District, cities.population as Country\_Population FROM cities

JOIN countries ON countries.id = cities.country\_id

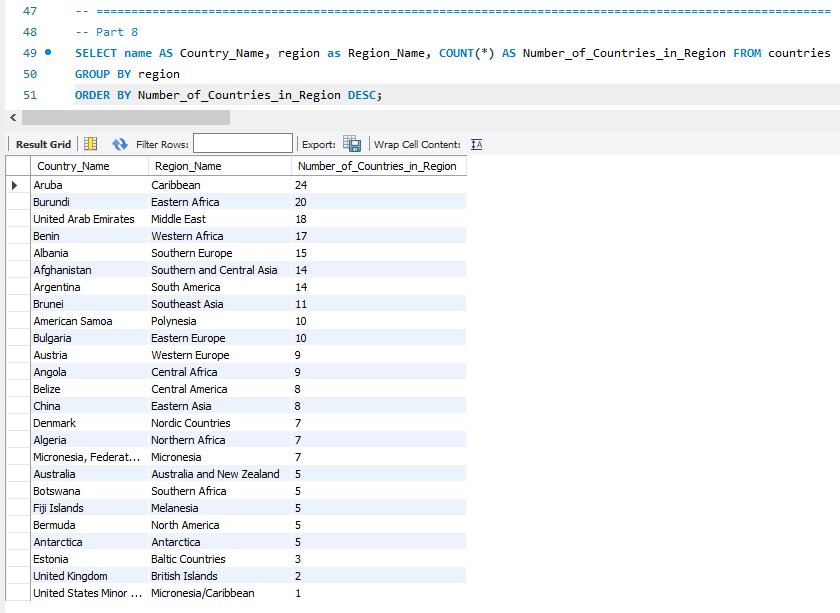
WHERE cities.population > 500e3 AND cities.district = 'Buenos Aires' AND countries.name = 'Argentina'

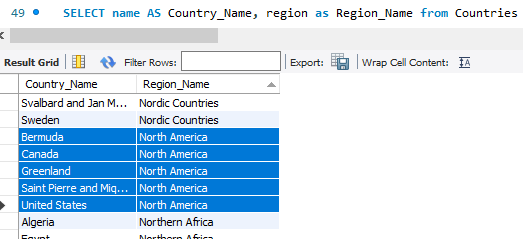


8. What query would you run to:

* Summarize the number of countries in each region?
* The query should display the name of the region and the number of countries.
* Also, the query should arrange the result by the number of countries in descending order. (2)







* **Showing that there are 5-countries in North-America region**
* **A few other regions were tested also**

# Assignment-2: ***Sakila***

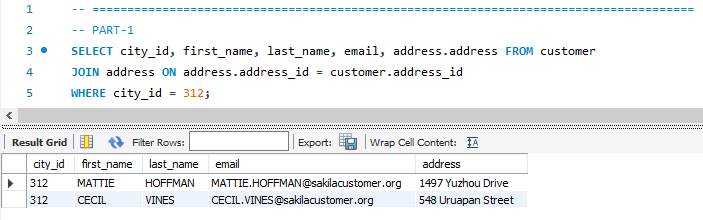
1. What query :

* All the customers
* Inside city\_id = 312
* Your query should return customer
  + first name
  + last name
  + email
  + address.

SELECT city\_id, first\_name, last\_name, email, address.address FROM customer

JOIN address ON address.address\_id = customer.address\_id

WHERE city\_id = 312;



2. What query would you run to get

* All comedy films
* Your query should return:
  + - film title
    - description
    - release year
    - rating
    - special features
    - genre (category)

SELECT film.film\_id, title AS Film\_Title, description, release\_year, rating, special\_features, name AS Genre FROM film\_category

JOIN film ON film.film\_id = film\_category.film\_id

JOIN category ON category.category\_id = film\_category.category\_id



3. What query would you run to get

* + All the films joined by actor\_id=5
* Your query should return
  + Actor id
  + Actor name
  + Film title
  + Description
  + Release year.

SELECT film\_actor.actor\_id, first\_name, last\_name, film.film\_id, title, film.description, film.release\_year FROM film\_actor

JOIN film ON film.film\_id = film\_actor.film\_id

JOIN actor ON actor.actor\_id = film\_actor.actor\_id

WHERE actor.actor\_id = 5



4. What query would you run to get:

* + All the customers in store\_id = 1
  + and inside these cities (1, 42, 312 and 459)
* Your query should return
  + first name
  + last name
  + email
  + address

SELECT \* FROM customer

JOIN address ON address.address\_id = customer.address\_id

JOIN city ON city.city\_id = address.city\_id

WHERE store\_id = 1 and city.city\_id = 1 or city.city\_id = 42 or city.city\_id = 312 or city.city\_id = 459



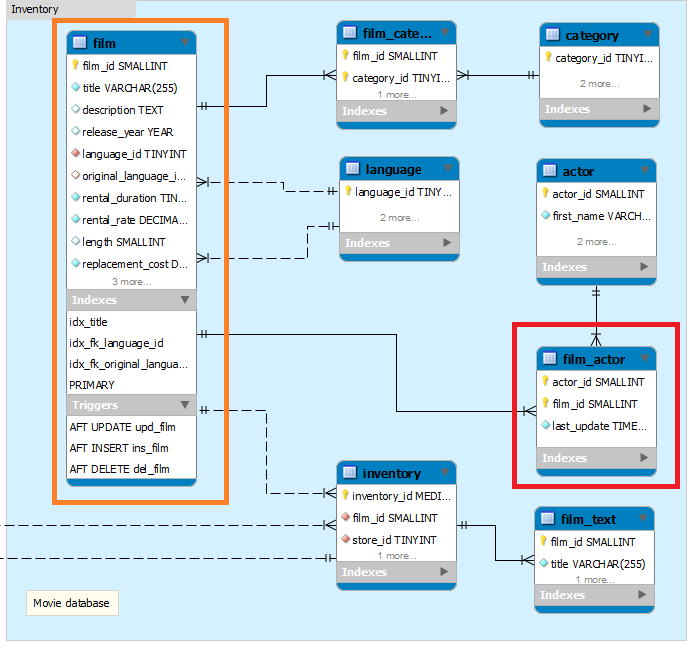
5. What query would you run to get

* + **all the films**
  + **with a "rating = G"**
  + and **"special feature = behind the scenes"**,
  + **joined by actor\_id = 15**
* Your query should return the film title
  + Description
  + Release Year
  + Rating
  + Special Feature.
* Hint: You may use LIKE function in getting the 'behind the scenes' part.

SELECT \* FROM film\_actor

JOIN film ON film.film\_id = film\_actor.film\_id

WHERE film\_actor.actor\_id LIKE 15 AND film.rating LIKE 'G' AND film.special\_features-- LIKE 'behind the scenes'



6. What query would you run to get

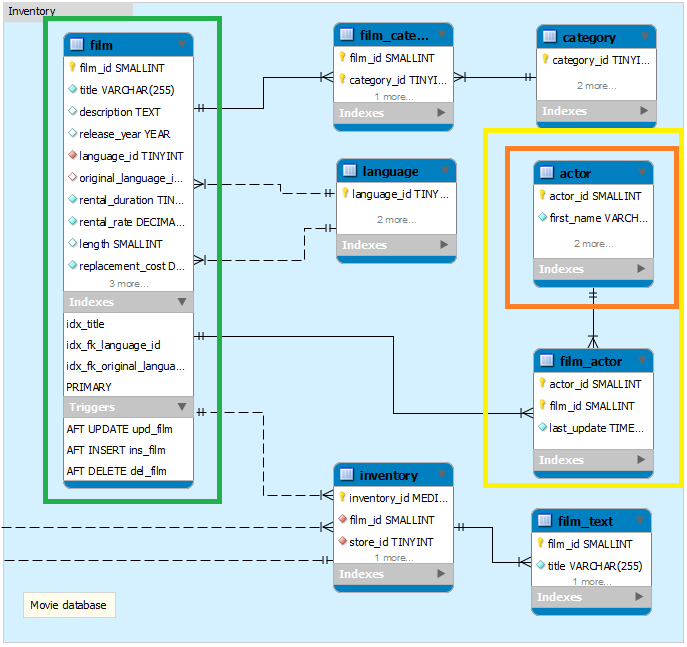
* + all the actors
  + that joined in the film\_id = 369
* Your query should return the
  + film\_id
  + title
  + actor\_id
  + actor\_name

SELECT film\_actor.film\_id, film.title, actor.actor\_id, actor.first\_name, actor.last\_name FROM film

JOIN film\_actor ON film\_actor.film\_id = film.film\_id

JOIN actor ON film\_actor.actor\_id = actor.actor\_id

WHERE film.film\_id = 369



7. What query would you run to get

* + all drama films
  + with a rental rate of 2.99
* Your query should return
  + film title
  + description
  + release year
  + rating
  + special features
  + genre

SELECT film.film\_id, title AS Film\_Title, description, release\_year, rating, special\_features, name AS Genre, rental\_rate FROM film\_category

JOIN film ON film.film\_id = film\_category.film\_id

JOIN category ON category.category\_id = film\_category.category\_id

WHERE rental\_rate = 2.99 AND name LIKE 'Drama'

8. What query would you run to get

* + all the action films
  + which are joined by SANDRA KILMER
* Your query should return
  + film title
  + description
  + release year
  + rating
  + special features
  + genre (category)
  + actor's first name
  + last name

SELECT actor.actor\_id, first\_name, last\_name,

film.film\_id, title AS Film\_Title, description, release\_year,

rating, special\_features, name AS Genre

FROM film\_category

-- Join 'category'-Table into 'film'-Table

JOIN film ON film.film\_id = film\_category.film\_id

JOIN category ON category.category\_id = film\_category.category\_id

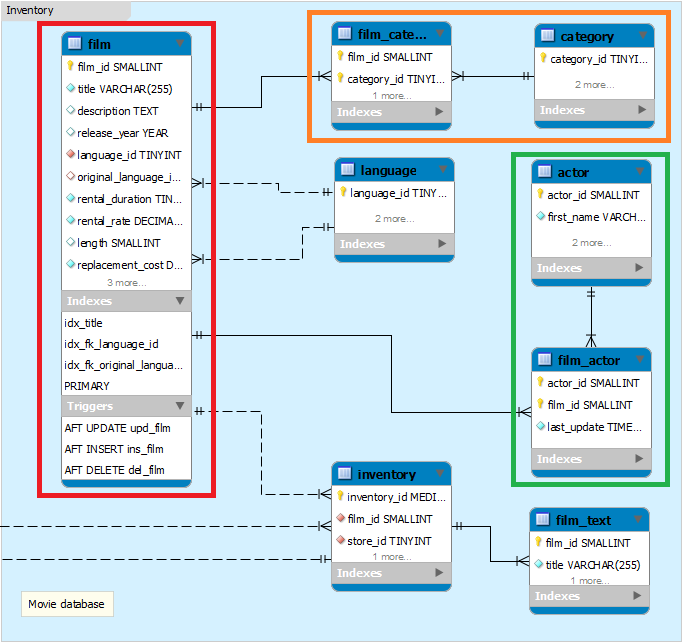
-- Join 'actor'-Table into 'film'-Table

JOIN film\_actor ON film\_actor.film\_id = film.film\_id

JOIN actor ON film\_actor.actor\_id = actor.actor\_id

-- Condition

WHERE first\_name LIKE 'Sandra' AND last\_name LIKE 'KILMER' AND name LIKE 'Action'

****