Week 9 - Homework

For each question below, paste in the query or queries you used in order to answer the question. Also INCLUDE A SCREENSHOT of the results from your query.

Remeber to check the ER Diagram to understand where the different pieces of information you need live: https://www.jooq.org/sakila

1. What are the names of all the languages in the database (sorted alphabetically)?
2. Return the full names (first and last) of actors with “SON” in their last name, ordered by their first name.
3. Find all the addresses where the second address is not empty (i.e., contains some text), and return these second addresses sorted.
4. Return the first and last names of actors who played in a film involving a “Crocodile” and a “Shark”, along with the release year of the movie, sorted by the actors’ last names.
5. How many films involve a “Crocodile” and a “Shark”?
6. Find all the film categories in which there are between 55 and 65 films. Return the names of these categories and the number of films per category, sorted by the number of films.
7. What are the top 10 largest payments? Sort largest to smallest.
8. Which actors have the first name ‘Scarlett’? Sort by last name descending. 9 Which actors have the last name ‘Johansson’? Sort by first name ascending.
9. How many distinct actors last names are there?
10. Which last names are not repeated?
11. Which last names appear more than once?
12. Which actor has appeared in the most films?
13. Is ‘Academy Dinosaur’ available for rent from Store 1?
14. Insert a record to represent Mary Smith renting ‘Academy Dinosaur’ from Mike Hillyer at Store 1 today .
15. When is ‘Academy Dinosaur’ due?
16. What is that average running time of all the films in the sakila DB?
17. What is the average running time of films by category?

Bonus Question 1:

Create a new table called "theater". It should have at least 5 attributes: one primary key, two foreign keys (from existing tables), and two other attributes of your choosing. Insert at least 5 records into the table. Paste below the query you used to make the table, and a screenshot of the table with it's data (SELECT * from it when done).

Bonus Question 2:

Create a new table from an existing one. Change at least three of the data types of the new table. Sort the new table by any column of your choosing and delete the last 10 rows based on however you sorted. Include all the queries you used below. Prove that you completed the steps above.

Q1: What are the names of all the languages in the database (sorted alphabetically)?

select * from language order by name;

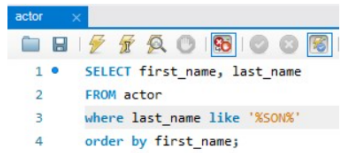
```
select * from language
order by name;
```

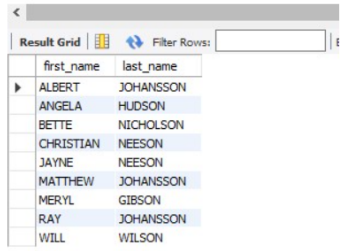
Result Grid | Filter Rows: | Edit:

language_id	name	last_update
1	English	2006-02-15 05:02:19
5	French	2006-02-15 05:02:19
6	German	2006-02-15 05:02:19
2	Italian	2006-02-15 05:02:19
3	Japanese	2006-02-15 05:02:19
4	Mandarin	2006-02-15 05:02:19

Q2: Return the full names (first and last) of actors with “SON” in their last name, ordered by their first name.

SELECT first_name, last_name FROM actor where last_name like '%SON%' order by first_name;

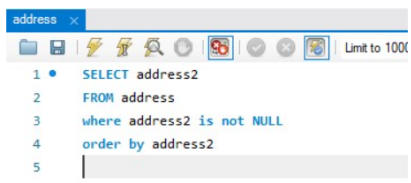


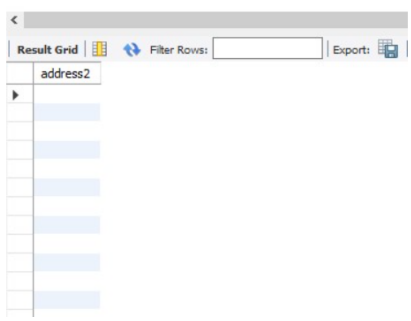


first_name	last_name
ALBERT	JOHANSSON
ANGELA	HUDSON
BETTE	NICHOLSON
CHRISTIAN	NEESON
JAYNE	NEESON
MATTHEW	JOHANSSON
MERYL	GIBSON
RAY	JOHANSSON
WILL	WILSON

q3: Find all the addresses where the second address is not empty (i.e., contains some text), and return these second addresses sorted.

SELECT address2 FROM address where address2 is not NULL order by address2





address2

q4: Return the first and last names of actors who played in a film involving a “Crocodile” and a “Shark”, along with the release year of the movie, sorted by the actors’ last names.

select distinct actor.first_name,actor.last_name, film.release_year From actor inner JOIN film_actor on film_actor.actor_id=actor.actor_id inner join film on film_actor.film_id=film.film_id where film.description like '%Shark%' and film.description like '%Crocodile%' order by actor.last_name

```
select  distinct actor.first_name,actor.last_name, film.release_year
From actor
inner JOIN film_actor on film_actor.actor_id=actor.actor_id
inner join film on film_actor.film_id=film.film_id
where film.description like '%Shark%' and film.description like '%Crocodile%'
order by actor.last_name
```

first_name	last_name	release_year
KIRSTEN	AKROYD	2006
KIM	ALLEN	2006
AUDREY	BAILEY	2006
JULIA	BARRYMORE	2006
VIVIEN	BASINGER	2006

q5: How many films involve a “Crocodile” and a “Shark”?

select count(film_id) from film where description like '%Shark%' and description like '%Crocodile%'

```
select count(film_id)
from film
where description like '%Shark%' and description like '%Crocodile%'
```

count(film_id)
10

q6: Find all the film categories in which there are between 55 and 65 films. Return the names of these categories and the number of films per category, sorted by the number of films.

select name , count(name) as films_per_category from category inner join film_category on film_category.category_id=category.category_id where film_category.film_id between 55 and 65 group by category.name;

```
select name , count(name) as films_per_category
from category
inner join film_category on film_category.category_id=category.category_id
where film_category.film_id between 55 and 65
group by category.name;
```

name	films_per_category
Sci-Fi	1
Action	1
Travel	1
Documentary	2
Children	1
Classics	1
Drama	2
Family	1
Horror	1

q7: What are the top 10 largest payments? Sort largest to smallest.

SELECT * FROM sakila.payment order by amount desc limit 10;

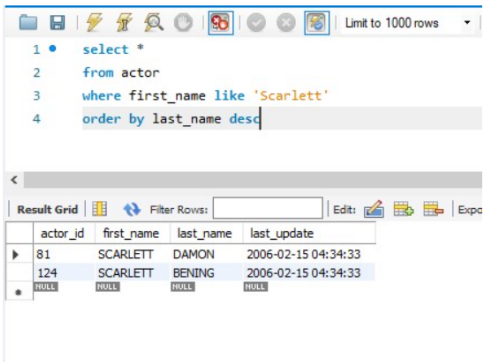
```
SELECT *
FROM sakila.payment
order by amount desc
limit 10;
```

payment_id	customer_id	staff_id	rental_id	amount	payment_date	last_update
342	13	2	8831	11.99	2005-07-29 22:37:41	2006-02-15 22:12:31
8272	305	1	2166	11.99	2005-06-17 23:51:21	2006-02-15 22:15:47
3146	116	2	14763	11.99	2005-08-21 23:34:00	2006-02-15 22:13:08
5550	204	2	15415	11.99	2005-08-22 23:48:56	2006-02-15 22:14:07
5280	195	2	16040	11.99	2005-08-23 22:19:33	2006-02-15 22:14:00
6409	237	2	11479	11.99	2005-08-02 22:18:13	2006-02-15 22:14:38
9803	362	1	14759	11.99	2005-08-21 23:28:58	2006-02-15 22:16:57
5281	196	2	106	11.99	2005-05-25 18:18:19	2006-02-15 22:14:00
15850	592	1	3973	11.99	2005-07-06 22:58:31	2006-02-15 22:23:25
15821	591	2	4383	11.99	2005-07-07 20:45:51	2006-02-15 22:23:18

q8: Which actors have the first name ‘Scarlett’? Sort by last name descending. 9 Which actors have the last name ‘Johansson’? Sort by first name ascending.

first part

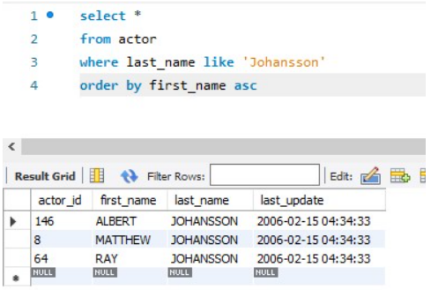
select * from actor where first_name like 'Scarlett' order by last_name desc



second part

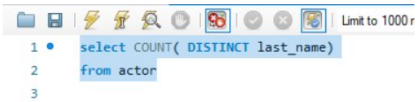
second part

select * from actor where last_name like 'Johansson' order by first_name asc



q9: How many distinct actors last names are there?

select COUNT(DISTINCT last_name) from actor



Result Grid | Filter Rows: | Export:

COUNT(DISTINCT last_name)
121

q10: Which last names are not repeated?

select last_name from actor group by last_name having count(last_name) <=1;

```
select last_name
from actor
group by last_name
having count(last_name) <=1;
```

Result Grid | Filter Rows:

last_name
ASTAIRE
BACALL
BALE
BALL
BARRYMORE
BASINGER
BERGEN

q11: Which last names appear more than once?

select last_name from actor group by last_name having count(last_name) > 1;

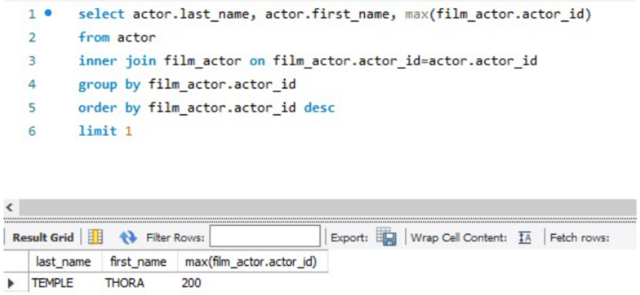
```
select last_name
from actor
group by last_name
having count(last_name) > 1;
```

Result Grid | Filter Rows:

last_name
AKROYD
ALLEN
BAILEY
BENING
BERRY
BOLGER
BRODY
CAGE
CHASE
CRAWFORD
CRONYN
DAVIS
DEAN
DEE
DEGENERES
DENCH
DEPP
DUKAKIS
FAWCETT

q12:Which actor has appeared in the most films ?

select actor.last_name, actor.first_name, max(film_actor.actor_id) from actor inner join film_actor on film_actor.actor_id=actor.actor_id group by film_actor.actor_id order by film_actor.actor_id desc limit 1



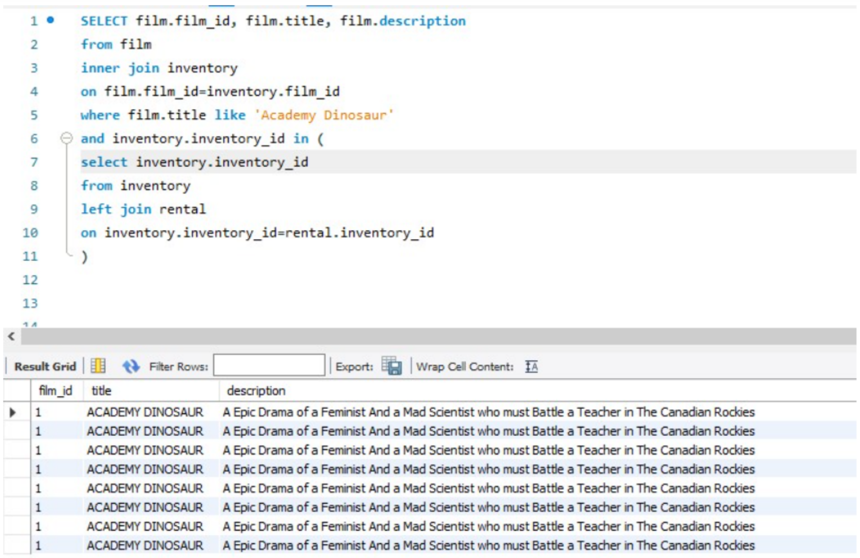
```
select actor.last_name, actor.first_name, max(film_actor.actor_id)
from actor
inner join film_actor on film_actor.actor_id=actor.actor_id
group by film_actor.actor_id
order by film_actor.actor_id desc
limit 1
```

last_name	first_name	max(film_actor.actor_id)
TEMPLE	THORA	200

q13: Is ‘Academy Dinosaur’ available for rent from Store 1?

Answer: Yes.

SELECT film.film_id, film.title, film.description from film inner join inventory on film.film_id=inventory.film_id where film.title like 'Academy Dinosaur' and inventory.inventory_id in (select inventory.inventory_id from inventory left join rental on inventory.inventory_id=rental.inventory_id)



```
SELECT film.film_id, film.title, film.description
from film
inner join inventory
on film.film_id=inventory.film_id
where film.title like 'Academy Dinosaur'
and inventory.inventory_id in (
select inventory.inventory_id
from inventory
left join rental
on inventory.inventory_id=rental.inventory_id
)
```

film_id	title	description
1	ACADEMY DINOSAUR	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies
1	ACADEMY DINOSAUR	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies
1	ACADEMY DINOSAUR	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies
1	ACADEMY DINOSAUR	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies
1	ACADEMY DINOSAUR	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies
1	ACADEMY DINOSAUR	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies
1	ACADEMY DINOSAUR	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies
1	ACADEMY DINOSAUR	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies

q14: Insert a record to represent Mary Smith renting ‘Academy Dinosaur’ from Mike Hillyer at Store 1 today

first getting the customer id for Mary Smith: customer_id=1

SELECT customer_id FROM customer where (first_name = 'Mary' AND last_name = 'Smith');

second getting info about the movie Academy Dinosaur: film_id=1,

SELECT * FROM sakila.film where title='Academy Dinosaur'

#third step, getting the inventory information: inventory_id = 1 and store_id=1, last_update= 2006-02-15 SELECT * FROM sakila.inventory where inventory_id=1;

finally knowing data about staff: staff_id=1 that is related to Mike Hillyer

SELECT * FROM sakila.staff where first_name like 'Mike' and last_name like 'Hillyer';

Right now we have enough data for each attributes we need to deal with:

We will enter data into table rental

we need:

rental_id=1

rental_date=2021-11-14

inventory_id=1

customer_id=1

staff_id=1

last_update=2021-11-14

#SQL INSERT INTO `sakila`.`rental` (`rental_id`, `rental_date`, `inventory_id`, `customer_id`, `return_date`, `staff_id`, `last_update`) VALUES (16050, 11/14/2021, 1, 1, 11/20/2021, 1, 11/14/2021);

rental_id	rental_date	inventory_id	customer_id	return_date	staff_id	last_update
16039	2005-08-23 22:18:51	545	78	2005-08-31 19:55:51	2	2006-02-15 21:30:53
16040	2005-08-23 22:19:33	3524	195	2005-09-02 02:19:33	2	2006-02-15 21:30:53
16041	2005-08-23 22:20:26	4116	121	2005-08-25 20:14:26	2	2006-02-15 21:30:53
16042	2005-08-23 22:20:40	629	131	2005-08-24 17:54:40	1	2006-02-15 21:30:53
16043	2005-08-23 22:21:03	3869	526	2005-08-31 03:09:03	2	2006-02-15 21:30:53
16044	2005-08-23 22:24:39	1312	468	2005-08-25 04:08:39	1	2006-02-15 21:30:53
16045	2005-08-23 22:25:26	772	14	2005-08-25 23:54:26	1	2006-02-15 21:30:53
16046	2005-08-23 22:26:47	4364	74	2005-08-27 18:02:47	2	2006-02-15 21:30:53
16047	2005-08-23 22:42:48	2088	114	2005-08-25 02:48:48	2	2006-02-15 21:30:53
16048	2005-08-23 22:43:07	2019	103	2005-08-31 21:33:07	1	2006-02-15 21:30:53
16049	2005-08-23 22:50:12	2666	393	2005-08-30 01:01:12	2	2006-02-15 21:30:53
16050	2021-11-14 14:51:50	1	1	0000-00-00 00:00:00	1	0000-00-00 00:00:00
NULL	NULL	NULL	NULL	NULL	NULL	NULL

q15: When is ‘Academy Dinosaur’ due?

select rental_date, rental_date + interval (select rental_duration from film where film_id = 1) day as due_date from rental where rental_id = (select rental_id from rental order by rental_id desc limit 1);

```
select rental_date,
       rental_date + interval
                     (select rental_duration from film where film_id = 1) day
                     as due_date
from rental
where rental_id = (select rental_id from rental order by rental_id desc limit 1);
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

rental_date	due_date
2021-11-14 14:51:50	2021-11-20 14:51:50

q16: What is that average running time of all the films in the sakila DB?

I treated running time as the length of the movie

select avg(length) from film;

```
select avg(length) from film;
```

Result Grid | Filter Rows: | Export:

avg(length)
115.2720

q17: What is the average running time of films by category?

select category.name, avg(length) as average_by_category from film join film_category on film_category.film_id=film.film_id join category on film_category.category_id=category.category_id group by category.name

```
select category.name, avg(length) as average_by_category
from film
join film_category
on film_category.film_id=film.film_id
join category
on film_category.category_id=category.category_id
group by category.name
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

name	average_by_category
Action	111.6094
Animation	111.0152
Children	109.8000
Classics	111.6667
Comedy	115.8276
Documentary	108.7500
Drama	120.8387
Family	114.7826
Foreign	121.6986
Games	127.8361
Horror	112.4821