Report.8.Sakila

Linh Duong

November 2017

1 List names of all the languages in the database (sorted alphabetically)?

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

```
mysql> select name from language
[    -> order by name;
+----------+
| name     |
+----------+
| English  |
| French   |
| German   |
| Italian  |
| Japanese |
| Mandarin |
+----------+
6 rows in set (0.00 sec)
```

2 List full names of actors with "GER" in their last name, ordered by their first name

```
select concat(first_name,'_',last_name) as fullname
from actor where last_name like '%GER%'
order by first_name;
```

```
mysql> select concat(first_name,' ',last_name) as fullname
    -> from actor where last_name like '%GER%'
[   -> order by first_name;
+-------------------+
| fullname          |
+-------------------+
| CAMERON ZELLWEGER |
| JULIA ZELLWEGER   |
| MICHAEL BOLGER    |
| MINNIE ZELLWEGER  |
| VAL BOLGER        |
| VIVIEN BASINGER   |
+-------------------+
6 rows in set (0.00 sec)
```

3 Find all the addresses where postal code starts with "57", and return addresses sorted.

select address **from** address
where postal_code **like** '57%'
order by address;

```
mysql> select address from address
    -> where postal_code like '57%'
[   -> order by address;
+----------------------------------+
| address                          |
+----------------------------------+
| 1029 Dzerzinsk Manor             |
| 1061 Ede Avenue                  |
| 1287 Xiangfan Boulevard          |
| 1309 Weifang Street              |
| 1515 Korla Way                   |
| 1697 Kowloon and New Kowloon Loop |
| 231 Kaliningrad Place            |
| 390 Wroclaw Way                  |
| 431 Szkesfehrvr Avenue           |
| 687 Alessandria Parkway          |
| 869 Shikarpur Way                |
+----------------------------------+
11 rows in set (0.00 sec)
```

4 How many films involve a "DWARF" in their titles?

select count(title) **from** film_list
where title **like** '%DWARF%';

```
mysql> select count(title) from film_list
[    -> where title like '%DWARF%';
+--------------+
| count(title) |
+--------------+
|            2 |
+--------------+
1 row in set (0.05 sec)
```

5 Find full names of actors who played in a film involving 'WAR' in title and longer than 2.5 hours, along with the title, run length and release year of the movie, sorted by the actors' last names.

```
select actors, film_list.title, film_list.length, release_year
from film_list join film on FID = film_id
where film_list.title like '%WAR%' and film.length > 150;
```

```
mysql> select actors, film_list.title, film_list.length, release_year
    -> from film_list join film on FID = film_id
    -> where film_list.title like '%WAR%' and film.length > 150;
+------------------------------------------------------------------------------------------------------+---------------------+--------+--------------+
| actors                                                                                               | title               | length | release_year |
+------------------------------------------------------------------------------------------------------+---------------------+--------+--------------+
| JULIA MCQUEEN, KIRK JOVOVICH, ADAM GRANT, CAMERON ZELLWEGER, IAN TANDY, JAYNE SILVERSTONE             | GLADIATOR WESTWARD   |    173 |         2006 |
| KARL BERRY, UMA WOOD, BOB FAWCETT, SPENCER DEPP, SUSAN DAVIS, PENELOPE MONROE, IAN TANDY              | LEATHERNECKS DWARFS  |    153 |         2006 |
| NICK WAHLBERG, KIRK JOVOVICH, ANGELINA ASTAIRE, WALTER TORN, WARREN NOLTE, RIVER DEAN, GREGORY GOODING, THORA TEMPLE | WARDROBE PHANTOM    |    178 |         2006 |
+------------------------------------------------------------------------------------------------------+---------------------+--------+--------------+
3 rows in set (0.05 sec)
```

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 descending

```
select name, count(film_id) as numberoffilms
from film_category
join category on film_category.category_id=category.category_id
group by film_category.category_id
having numberoffilms >=55 and numberoffilms <=65 ;
```

```
+----------+--------------+
| name     | numberoffilms |
+----------+--------------+
| Action   |           64 |
| Children |           60 |
| Classics |           57 |
| Comedy   |           58 |
| Drama    |           62 |
| Games    |           61 |
| Horror   |           56 |
| New      |           63 |
| Sci-Fi   |           61 |
| Travel   |           57 |
+----------+--------------+
10 rows in set (0.01 sec)
```

7   In how many film categories is the average di erence between the film replacement cost and the rental rate larger than 17?

```
select count(*) as result
from (
select category_id
from film_category
join film on film_category.film_id = film.film_id
group by film_category.category_id
having abs(avg(replacement_cost) − avg(rental_rate)) >17) as nocate;
```

```
+--------+
| result |
+--------+
|      8 |
+--------+
1 row in set (0.00 sec)
```

8   Find the address district(s) name(s) such that the minimal postal code in the district(s) is maximal over all the districts. Make sure your query ignores empty postal codes and district names.

```
select district , mincode
from
(select district , min(postal_code) as mincode
from address
```

```
where postal_code is not null
group by district) as result
group by district
order by mincode desc limit 0,1;


+----------+---------+
| district | mincode |
+----------+---------+
| Crdoba   | 99865   |
+----------+---------+
1 row in set (0.00 sec)
```

9 Find the names (first and last) of all the actors and customers whose first name is the same as the first name of the actor with ID 101 (exclude the actor with ID 101).

```
create view condi as
        select first_name from actor where actor_id = 101;

select concat(first_name,'_',last_name) as fullname
from actor
where first_name = (select * from condi)
and actor_id != 101
union
select concat(first_name,'_',last_name) as fullname
from customer
where first_name = (select * from condi);


+--------------+
| fullname     |
+--------------+
| SUSAN DAVIS  |
| SUSAN WILSON |
+--------------+
2 rows in set (0.00 sec)
```