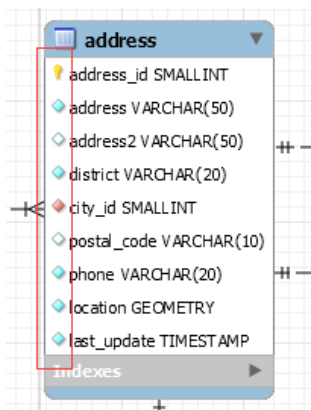


实验一报告

一、回答问题

请一边熟悉 sakila 数据库，一边回答以下问题：

1. sakila.mwb 模型中，表结构里每个字段前面的小标记分别表示什么意思？



| 标记 | 意义 |
|----|---------|
| ⚡ | 主键 |
| ◆ | 候选码 |
| ◇ | 可以为空的属性 |
| ◆ | 外键 |

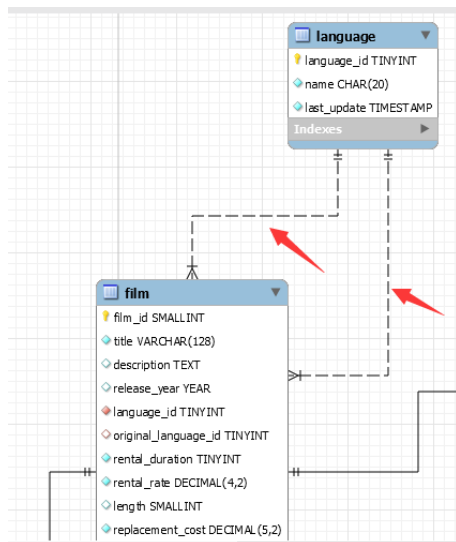
2. 图中哪部分体现影片-演员关系？换句话说，如果要找出演某个影片的演员名字，访问哪几张表可以获得信息？

通过 film(film_id), film_actor(actor_id), actor 表连接寻找

3. 如果已知某个顾客姓名，要找到他租借的所有影片名，需要访问哪几张表？

Customer(rental_id), rental(inventory_id), inventory(film_id), film

4. 为什么 film 和 language 表间会有 2 条线？



可以用两个外键对应另一张表的主键，用于对应一部电影同时存在的不同语言版本。

二、实验截图

(注意截图清晰，截图时需要体现 SQL 语句、执行结果、Output 窗口)

- 1、请列出所有商店的详细地址，显示商店 id，商店地址，所在区域，所在城市，所在国家；

```

1 select store_id, address, district, city, country
2 from store, address, city, country
3 where store.address_id=address.address_id and address.city_id=city.city_id and city.country_id=country.country_id
4
  
```

| | store_id | address | district | city | country |
|---|----------|--------------------|----------|------------|-----------|
| ▶ | 1 | 47 MySakila Drive | Alberta | Lethbridge | Canada |
| | 2 | 28 MySQL Boulevard | QLD | Woodridge | Australia |

- 2、哪些演员出演过影片《WEST LION》？请列出他的姓名；

```

1 select first_name, last_name
2 from actor, film_actor, film
3 where film.title="WEST LION" and film.film_id=film_actor.film_id and film_actor.actor_id=actor.actor_id;
4
  
```

| first_name | last_name |
|------------|-------------|
| ZERO | CAGE |
| CAMERON | STREEP |
| SEAN | WILLIAMS |
| CARY | MCCONAUGHEY |
| MORGAN | HOPKINS |
| CUBA | ALLEN |
| ALBERT | JOHANSSON |
| HUMPHREY | GARLAND |
| GREGORY | GOODING |

- 3、 找出租 DVD 花费最高的前 3 名，请列出他们的 first_name, last_name 和每个人花费的金额；

```

1 select first_name, last_name, sum(amount) as sum
2 from customer, rental, payment
3 where customer.customer_id=rental.customer_id and rental.rental_id=payment.rental_id
4 group by customer.customer_id
5 order by sum desc limit 3;

```

| first_name | last_name | sum |
|------------|-----------|--------|
| KARL | SEAL | 221.55 |
| ELEANOR | HUNT | 216.54 |
| CLARA | SHAW | 195.58 |

- 4、 哪个影片获得了总体最高的租金？请列出影片 id、影片名、总租金；

```

1 select film.film_id, title, sum(amount) as sum
2 from film, inventory, rental, payment
3 where film.film_id=inventory.film_id and inventory.inventory_id=rental.inventory_id and rental.rental_id=payment.rental_id
4 group by film_id order by sum desc limit 1;

```

| film_id | title | sum |
|---------|------------------|--------|
| 879 | TELEGRAPH VOYAGE | 231.73 |

- 5、 哪个演员出演的电影超过 40 部？ 请列出演员 id、演员名、出演的电影数；

```

1 select actor.actor_id, first_name, last_name, title, count(title) as cnt
2 from film, film_actor, actor
3 where film.film_id=film_actor.film_id and film_actor.actor_id=actor.actor_id
4 group by actor_id
5 having cnt>40

```

| actor_id | first_name | last_name | cnt |
|----------|------------|-----------|-----|
| 102 | WALTER | TORN | 41 |
| 107 | GINA | DEGENERES | 42 |

6. 请找出没有租借过电影《WEST LION》的顾客姓名

```

1 select customer.first_name, customer.last_name
2 from film, inventory, rental, customer
3 where film.film_id=inventory.film_id and inventory.inventory_id=rental.inventory_id and rental.customer_id=customer.customer_id and
4 customer.customer_id not in(
5 select customer.customer_id
6 from film, inventory, rental, customer
7 where film.film_id=inventory.film_id and inventory.inventory_id=rental.inventory_id and rental.customer_id=customer.customer_id
8 )
9 group by customer.customer_id

```

583 row(s) returned

| | first_name | last_name |
|---|------------|-------------|
| ▶ | JOEL | FRANCISCO |
| | GABRIEL | HARDER |
| | DIANNE | SHELTON |
| | NORMAN | CURRIER |
| | BEATRICE | ARNOLD |
| | GERALDINE | PERKINS |
| | VIRGIL | WOFFORD |
| | WILLIE | MARKHAM |
| | DEBRA | NELSON |
| | DARREN | WINDHAM |
| | ROBERT | BAUGHMAN |
| | HENRY | BILLINGSLEY |
| | SERGIO | STANFIELD |
| | FREDDIE | DUGGAN |
| | MARIE | TURNER |
| | MATTIE | HOFFMAN |
| | DWAYNE | OLVERA |
| | CARL | ARTIS |
| | NATHAN | RUNYON |
| | HECTOR | POINDEXTER |
| | TINA | SIMMONS |
| | SUSAN | WILSON |
| | REBECCA | SCOTT |
| | PENNY | NEAL |

7. 查询演过《FIRE WOLVES》和《JAWBREAKER BROOKLYN》这两部电影的演员，列出其姓名：

```
1 select actor.first_name, actor.last_name
2 from film, film_actor, actor
3 where film.film_id=film_actor.film_id and film_actor.actor_id=actor.actor_id and film.title='FIRE WOLVES' and actor.actor_id in(
4     select actor.actor_id
5     from film, film_actor, actor
6     where film.film_id=film_actor.film_id and film_actor.actor_id=actor.actor_id and film.title='JAWBREAKER BROOKLYN'
7 )
```

| | first_name | last_name |
|---|------------|-----------|
| ▶ | CHRISTIAN | AKROYD |

8. 统计每种类型的影片数，显示类型编号、类型名称、该类型影片数：

```
1 select category.category_id as category_id, name, count(film_id) as cnt
2 from category, film_category
3 where category.category_id=film_category.category_id
4 group by category_id
```

| | category_id | name | cnt |
|---|-------------|-------------|-----|
| ▶ | 1 | Action | 64 |
| | 2 | Animation | 66 |
| | 3 | Children | 60 |
| | 4 | Classics | 57 |
| | 5 | Comedy | 58 |
| | 6 | Documentary | 68 |
| | 7 | Drama | 62 |
| | 8 | Family | 69 |
| | 9 | Foreign | 73 |
| | 10 | Games | 61 |
| | 11 | Horror | 56 |
| | 12 | Music | 51 |
| | 13 | New | 63 |
| | 14 | Sci-Fi | 61 |
| | 15 | Sports | 74 |
| | 16 | Travel | 57 |

9. 有哪些影片是 2 个商店都有库存的？

```

1  select title, store.store_id
2  from film, inventory, store
3  where film.film_id=inventory.film_id and inventory.store_id=store.store_id
4  group by film.film_id having(max(store.store_id)<>min(store.store_id));

```

| | title | store_id |
|---|--------------------|----------|
| ▶ | ACADEMY DINOSAUR | 1 |
| | AFFAIR PREJUDICE | 1 |
| | AGENT TRUMAN | 1 |
| | AIRPLANE SIERRA | 1 |
| | ALABAMA DEVIL | 1 |
| | ALADDIN CALENDAR | 1 |
| | ALAMO VIDEOTAPE | 1 |
| | ALASKA PHANTOM | 1 |
| | ALIEN CENTER | 1 |
| | ALLEY EVOLUTION | 1 |
| | ALONE TRIP | 1 |
| | ALTER VICTORY | 1 |
| | AMADEUS HOLY | 1 |
| | AMERICAN CIRCUS | 1 |
| | AMISTAD MIDSUMMER | 1 |
| | ANACONDA CONFES... | 1 |
| | ANGELS LIFE | 1 |
| | ANNIE IDENTITY | 1 |
| | APACHE DIVINE | 1 |
| | ARACHNOPHOBIA R... | 1 |
| | ARIZONA BANG | 1 |
| | ARMAGEDDON LOST | 1 |
| | ATLANTIS CAUSE | 1 |
| | ATTACKS HATE | 1 |

85 16:46:48 select title, store.store_id from film, inventory, store where film.film_id=inventory.film_id and inventory.store_id=store.store_id group by film.film_id having(m... 563 row(s) returned

10. 查询单次租借影片时间最长的 3 位客户，列出其 first_name、last_name 和当次租借时长；

```

1 select customer.first_name, customer.last_name, TIMESTAMPDIFF(second, rental.rental_date, rental.return_date) as duration
2 from film, inventory, rental, customer
3 where film.film_id=inventory.film_id and inventory.inventory_id=rental.inventory_id and rental.customer_id=customer.customer_id
4 order by duration desc limit 3

```

| | first_name | last_name | duration |
|---|------------|-----------|----------|
| ▶ | ELAINE | STEVENS | 799140 |
| | MARTIN | BALES | 799140 |
| | JAMES | GANNON | 799080 |

11. 在 customer 表中新增一条数据，注意 customer 表与其他表的关系；

```

1 insert into customer
2 values(600, 1, 'Yikai', 'Li', 'xxxxxxxx@example.com', 1, 1, Now(), Now())

```

| | customer_id | store_id | first_name | last_name | email | address_id | active | create_date | last_update |
|---|-------------|----------|------------|-----------|----------------------|------------|--------|---------------------|---------------------|
| ▶ | 600 | 1 | Yikai | Li | xxxxxxxx@example.com | 1 | 1 | 2021-09-17 16:22:16 | 2021-09-17 16:22:16 |

12. 修改刚才在 customer 表中新增的那条数据；

```

1 update customer
2 set store_id=2
3 where customer_id=600;

```

| | customer_id | store_id | first_name | last_name | email | address_id | active | create_date | last_update |
|---|-------------|----------|------------|-----------|----------------------|------------|--------|---------------------|---------------------|
| ▶ | 600 | 2 | Yikai | Li | xxxxxxxx@example.com | 1 | 1 | 2021-09-17 16:22:16 | 2021-09-17 16:26:10 |

13. 删除第 11 步新增的那条数据。

```

1 delete * from customer where customer_id=600

```

65 16:28:35 delete from customer where customer_id=600 1 row(s) affected

三、思考题

- 1) 如果 insert 一条数据到 actor 表，但 actor_id 和已有数据重复，会发生什么？同学们请自己尝试一下。
发生报错，提示如下：
Error Code: 1062. Duplicate entry '201' for key 'actor.PRIMARY'
- 2) insert 语句还用了一个函数 NOW()，是做什么的呢？
用于更新 update_date，NOW() 函数返回当前时间