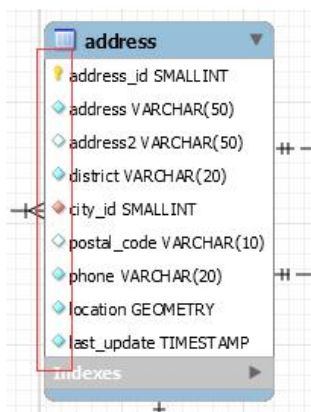


实验一报告

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？
(观察字段的属性)



标记	意义
💡	主键
💎	非空属性
◇	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？

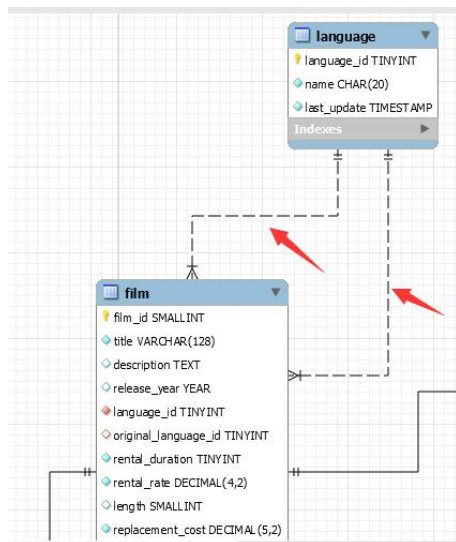
film 表、film_actor 表、actor 表这三张表

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 表、rental 表、inventory 表、film 表

4. film 和 language 表间的 2 条虚线表示什么意思？

film 中同一部电影可能存在不同语言版本：
language_id, original_language_id。因此用两条线分别连接 language 表的主键和 film 表中的两个外键。



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；

```
SELECT
    store_id, address, district, city, country
FROM
    store,
    address,
    city,
    country
WHERE
    store.address_id = address.address_id
    AND address.city_id = city.city_id
    AND city.country_id = country.country_id
```

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

Output				
Action Output				
#	Time	Action	Message	Duration / Fetch
1	17:44:56	SELECT store_id,address,district,city,country FROM store, address, city, country WHERE store.ad...	2 row(s) returned	0.000 sec / 0.000 sec

- 2、 哪些演员出演过影片《ROCKY WAR》？请列出他的 first_name, last_name;

```
SELECT
    first_name, last_name
FROM
    film,
    film_actor,
    actor
WHERE
    film.film_id = film_actor.film_id
    AND film_actor.actor_id = actor.actor_id
    AND film.title = 'ROCKY WAR'
```

	first_name	last_name
▶	SISSY	SOBIESKI
	PENELOPE	CRONYN
	RENEE	TRACY

1 00:03:13 SELECT first_name,last_name FROM film, film_actor, actor WHERE film.film_id = film_actor.film_id ... 3 row(s) returned

0.000 sec / 0.000 sec

- 3、 找出租 DVD 花费最高的前 5 名，请列出他们的 first_name, last_name 和每个人花费的金额;

```
SELECT
    first_name, last_name, SUM(amount)
FROM
    customer,
    rental,
    payment
WHERE
    customer.customer_id = rental.customer_id
    AND rental.rental_id = payment.rental_id
GROUP BY customer.customer_id
ORDER BY SUM(amount) DESC
LIMIT 5
```

	first_name	last_name	sum(amount)
▶	KARL	SEAL	221.55
	ELEANOR	HUNT	216.54
	CLARA	SHAW	195.58
	MARION	SNYDER	194.61
	RHONDA	KENNEDY	194.61

4 11:58:04 SELECT first_name,last_name,sum(amount) FROM customer, rental, payment WHERE customer.custo... 5 row(s) returned

0.078 sec / 0.000 sec

- 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；

```
SELECT
    film.film_id, title, SUM(amount)
FROM
    film,
    inventory,
    rental,
    payment
WHERE
    film.film_id = inventory.film_id
    AND inventory.inventory_id = rental.inventory_id
    AND rental.rental_id = payment.rental_id
GROUP BY film.film_id
ORDER BY SUM(amount) DESC
LIMIT 1
```

	film_id	title	SUM(amount)
▶	879	TELEGRAPH VOYAGE	231.73

#	Time	Action	Message	Duration / Fetch
1	13:50:36	SELECT	film.film_id,title,SUM(amount)FROM film, inventory, rental, payment WHERE film.film_id=i... 1 row(s) returned	0.062 sec / 0.000 sec

- 5、 哪个演员出演的电影超过 35 部？ 请列出演员 id、演员名、出演的电影数；

```
SELECT
    actor.actor_id,first_name,last_name,count(film.film_id)
FROM
    film,
    film_actor,
    actor
WHERE
    film.film_id = film_actor.film_id
    AND film_actor.actor_id = actor.actor_id
GROUP BY actor.actor_id
having count(film.film_id)>35
```

	actor_id	first_name	last_name	count(film.film_id)
▶	23	SANDRA	KILMER	37
	81	SCARLETT	DAMON	36
	102	WALTER	TORN	41
	107	GINA	DEGENERES	42
	181	MATTHEW	CARREY	39
	198	MARY	KEITEL	40

#	Time	Action	Message	Duration / Fetch
1	14:04:12	SELECT	actor.actor_id,first_name,last_name,count(film.film_id)FROM film, film_actor, actor WHERE fil... 6 row(s) returned	0.015 sec / 0.000 sec

6、 请找出没有租借过电影《TELEGRAPH VOYAGE》的顾客姓名；

```
SELECT
    first_name, last_name
FROM
    customer
WHERE
    NOT EXISTS( SELECT
        *
        FROM
            rental,
            inventory,
            film
        WHERE
            customer.customer_id = rental.customer_id
            AND rental.inventory_id = inventory.inventory_id
            AND inventory.film_id = film.film_id
            AND film.title = 'TELEGRAPH VOYAGE')
```

共 572 条，截图截不全

Result Grid			Filter Rows: <input type="text"/>	
	first_name	last_name		
▶	MARY	SMITH		
	LINDA	WILLIAMS		
	BARBARA	JONES		
	ELIZABETH	BROWN		
	JENNIFER	DAVIS		
	MARIA	MILLER		
	SUSAN	WILSON		
	MARGARET	MOORE		
	DOROTHY	TAYLOR		
	LISA	ANDERSON		
	NANCY	THOMAS		
	KAREN	JACKSON		

Action Output						
#	Time	Action	Message		Duration / Fetch	
1	14:31:49	SELECT first_name, last_name FROM customer WHERE NOT EXISTS(SELECT * FROM ...	572 row(s) returned		0.000 sec / 0.000 sec	

7、 查询演过《ELEPHANT TROJAN》和《SPLASH GUMP》这两部电影的演员，列出其姓名；

```

SELECT
    first_name, last_name
FROM
    actor, film_actor, film
WHERE
    actor.actor_id = film_actor.actor_id
    AND film.film_id = film_actor.film_id
    AND film.title = 'ELEPHANT TROJAN'
    AND actor.actor_id IN (SELECT
        actor.actor_id
    FROM
        actor, film_actor, film
    WHERE
        actor.actor_id = film_actor.actor_id
        AND film.film_id = film_actor.film_id
        AND film.title = 'SPLASH GUMP')

```

	first_name	last_name
▶	PENELOPE	GUINESS
	CAMERON	STREEP

#	Time	Action	Message	Duration / Fetch
1	15:01:40	SELECT first_name, last_name FROM actor, film_actor, film WHERE actor.actor_id = film_actor.actor_id ...	2 row(s) returned	0.000 sec / 0.000 sec

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；

```

SELECT
    category.category_id, category.name, COUNT(film_category.film_id)
FROM
    category, film_category
WHERE
    category.category_id = film_category.category_id
GROUP BY category.category_id

```


	category_id	name	COUNT(film_category.film_id)
▶	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

#	Time	Action	Message	Duration / Fetch
1	15:14:02	SELECT category.category_id,category.name,COUNT(film_category.film_id)FROM category,film_category WHERE...	16 row(s) returned	0.000 sec / 0.000 sec

9、 有哪些影片是 2 个商店都有库存的？

```

SELECT
    film.film_id,title
FROM
    film,inventory,store
WHERE
    film.film_id = inventory.film_id
    and inventory.store_id = store.store_id
GROUP BY film.film_id
having max(store.store_id)>min(store.store_id)

```

共 563 条，截图截取一部分：

	film_id	title
▶	1	ACADEMY DINOSAUR
	4	AFFAIR PREJUDICE
	6	AGENT TRUMAN
	7	AIRPLANE SIERRA
	9	ALABAMA DEVIL
	10	ALADDIN CALENDAR
	11	ALAMO VIDEOTAPE
	12	ALASKA PHANTOM
	15	ALIEN CENTER
	16	ALLEY EVOLUTION
	17	ALONE TRIP
	18	ALTER VICTORY
	19	AMADEUS HOLY
	21	AMERICAN CIRCUS
	22	AMISTAD MIDSUMMER
	23	ANACONDA CONFES...
	25	ANGELS LIFE

#	Time	Action	Message	Duration / Fetch
1	15:29:16	SELECT film.film_id,title FROM film.inventory,store WHERE film.film_id = inventory.film_id and inventory.store...	563 row(s) returned	0.015 sec / 0.000 sec

- 10、 查询单次租借影片时间最长的 6 位客户，列出其 first_name、last_name 和当次租借时长；

```
SELECT
    first_name,last_name,timestampdiff(second,rental_date,return_date) as rental_time
FROM
    rental,customer
WHERE
    rental.customer_id = customer.customer_id
order by rental_time desc
limit 6
```

	first_name	last_name	rental_time
▶	MARTIN	BALES	799140
	ELAINE	STEVENS	799140
	JACQUELINE	LONG	799080
	JAMES	GANNON	799080
	VERA	MCCOY	799080
	PEARL	GARZA	799080

#	Time	Action	Message	Duration / Fetch
1	16:02:42	SELECT first_name,last_name,timestampdiff(second,rental_date,return_date) as rental_time FROM rental,customer...	6 row(s) returned	0.016 sec / 0.000 sec

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

```
insert into customer
values (600,1,'Xinhao','Liu','Xinhao.Liu@sakilacustomer.org',1,1,now(),now())
```

customer_id	store_id	first_name	last_name	email	address_id	active	create_date	last_update
597	1	FREDDIE	DUGGAN	FREDDIE.DUGGAN@sakilacustomer.org	603	1	2006-02-14 22:04:37	2006-02-15 04:57:20
598	1	WADE	DELVALLE	WADE.DELVALLE@sakilacustomer.org	604	1	2006-02-14 22:04:37	2006-02-15 04:57:20
599	2	AUSTIN	CINTRON	AUSTIN.CINTRON@sakilacustomer.org	605	1	2006-02-14 22:04:37	2006-02-15 04:57:20
600	1	Xinhao	Liu	Xinhao.Liu@sakilacustomer.org	1	1	2022-11-26 16:22:16	2022-11-26 16:22:16

5	16:22:16	insert into customer values (600,1,'Xinhao','Liu','Xinhao.Liu@sakilacustomer.org',1,1,now(),now())	1 row(s) affected	0.016 sec
---	----------	--	-------------------	-----------

值得注意：由于 cunsumer 表中，store_id 和 address_id 为外键，添加时应该使用 store 表和 address 表中已有的 store_id 和 address_id 值。

- 12、 修改刚才在 customer 表中新增的那条数据；

```
update customer set last_update = now() where customer_id = 600
```

customer_id	store_id	first_name	last_name	email	address_id	active	create_date	last_update
597	1	FREDDIE	DUGGAN	FREDDIE.DUGGAN@sakilacustomer.org	603	1	2006-02-14 22:04:37	2006-02-15 04:57:20
598	1	WADE	DELVALLE	WADE.DELVALLE@sakilacustomer.org	604	1	2006-02-14 22:04:37	2006-02-15 04:57:20
599	2	AUSTIN	CINTRON	AUSTIN.CINTRON@sakilacustomer.org	605	1	2006-02-14 22:04:37	2006-02-15 04:57:20
600	1	Xinhao	Liu	Xinhao.Liu@sakilacustomer.org	1	1	2022-11-26 16:22:16	2022-11-26 16:41:35

#	Time	Action	Message	Duration / Fetch
1	16:41:35	update customer set last_update = now() where customer_id = 600	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec

13、 删除第 11 步新增的那条数据。

```
delete from customer where customer_id = 600
```

customer_id	store_id	first_name	last_name	email	address_id	active	create_date	last_update
596	1	ENRIQUE	FORSYTHE	ENRIQUE.FORSYTHE@sakilacustomer.org	602	1	2006-02-14 22:04:37	2006-02-15 04:57:20
597	1	FREDDIE	DUGGAN	FREDDIE.DUGGAN@sakilacustomer.org	603	1	2006-02-14 22:04:37	2006-02-15 04:57:20
598	1	WADE	DEL VALLE	WADE.DEL VALLE@sakilacustomer.org	604	1	2006-02-14 22:04:37	2006-02-15 04:57:20
599	2	AUSTIN	CINTRON	AUSTIN.CINTRON@sakilacustomer.org	605	1	2006-02-14 22:04:37	2006-02-15 04:57:20

4	17:01:18	delete from customer where customer_id = 600	1 row(s) affected	0.000 sec
---	----------	--	-------------------	-----------

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，截图并分析原因。

发生如下报错：

```
Message
Error Code: 1062. Duplicate entry '1' for key 'actor.PRIMARY'
```

原因：actor_id 是 actor 表的主键，主键需要满足唯一性，若不满足则会报错。

- 2) insert 语句还用了一个函数 NOW()，是做什么的呢？

Now() 函数用于返回当前时间，用该函数实现对表中的时间属性进行更新。