

实验二报告

一、 观察并回答问题

1. 关于视图

(1) sakila.mwb 模型图中共有几个 View?

答: 共有 7 个 View。

(2) 分析以下 3 个视图, 回答以下问题:

视图名	关联表	作用
actor_info	actor,film_actor,film,film_category,category	提供了所有演员的列表及所演的电影, 电影按类别细分
film_list	actor,film_actor,film,film_category,category	电影表视图, 包含了所有电影的信息, 包括电影名, 描述, 类别, 价格, 时长, 演员等
sales_by_store	payment,rental,inventory,store,staff	商店销售视图, 包含了每个商店的经理及销售额

(3) 分别执行以下 2 句 SQL 语句:

```
update staff_list set `zip code` = '518055' where ID = '1';
```

```
update film_list set price = 1.99 where FID = '1';
```

截图执行结果, 并分析一下视图在什么情况下可以进行 update 操作, 什么情况下不能?

Output				
Action Output				
#	Time	Action	Message	Duration / Fetch
1	14:02:51	update staff_list set `zip code` = '518005' where ID = '1'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
2	14:02:51	update film_list set price = 1.99 where FID = '1'	Error Code: 1288. The target table film_list of the UPDATE is not updatable	0.000 sec

视图只有满足下列条件才可更新:

1、select 语句在选择列表中没有聚合函数, 也不包含 TOP, GROUP BY, UNION(除非视图是分区视图)或 DISTINCT 子句。聚合函数可以用在 FROM 子句的子查询中, 只要不修改函数返回的值。

2、select 语句的选择列表中没有派生列。派生列是由任何非简单列表达式(使用函数、加法或减法运算符等)所构成的结果集列。

3、select 语句中的 FROM 子句至少引用一个表。select 语句不能只包含非表格格式的表达式(即不是从表派生出的表达式)。

4、INSERT, UPDATE 和 DELETE 语句在引用可更新视图之前, 也必须如上述条件指定的那样满足某些限制条件。只有当视图可更新, 并且所编写的 UPDATE 或 INSERT 语句只修改视

图的 FROM 子句引用的一个基表中的数据时，UPDATE 和 INSERT 语句才能引用视图。只有当视图在其 FROM 子句中只引用一个表时，DELETE 语句才能引用可更新的视图。

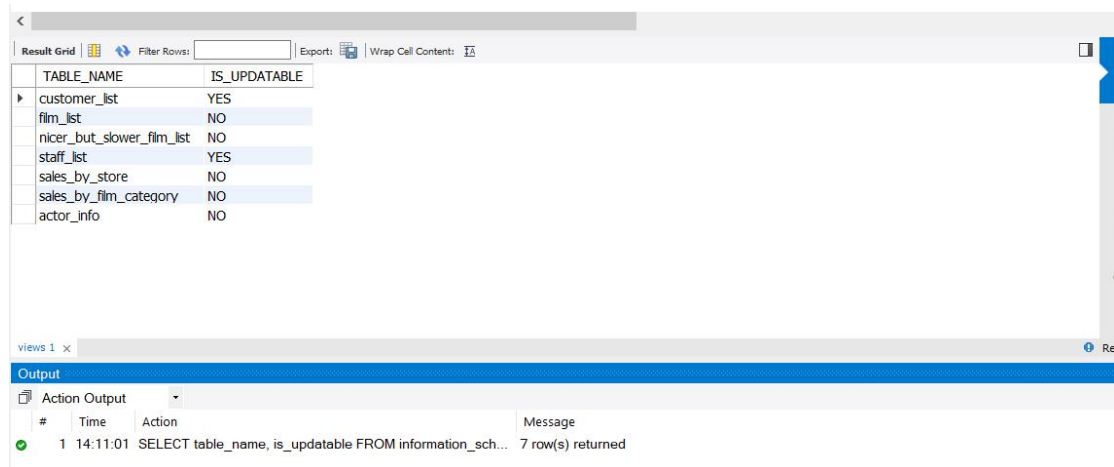
视图在包含以下结构时不可以更新

- 1、含有集合运算符：union，union all,intersect
- 2、distinct 关键字
- 3、group by,order by,connect by 和 start with
- 4、子查询
- 5、分组函数
- 6、需要更新的列不是视图定义的
- 7、有连接查询
- 8、违反基本表的约束

(4) 执行以下命令查询 sakila 数据库中的视图是否可更新，截图执行结果：

```
SELECT table_name, is_updatable FROM information_schema.views  
WHERE table_schema = 'sakila';
```

执行结果：



TABLE_NAME	IS_UPDATABLE
customer_list	YES
film_list	NO
nicer_but_slower_film_list	NO
staff_list	YES
sales_by_store	NO
sales_by_film_category	NO
actor_info	NO

#	Time	Action	Message
1	14:11:01	SELECT table_name, is_updatable FROM information_sch...	7 row(s) returned

2. 关于触发器

(1) 触发器 customer_create_date 建在哪个表上？这个触发器实现什么功能？

答：建在 customer 表上。

实现功能：当 customer 插入新行时，customer_create_date 触发器将 customer 表的 create_date 字段设置为当前时间和日期。

(2) 在这个表上新增一条数据，验证一下触发器是否生效。（截图语句和执行结果）

语句：

```
INSERT INTO customer
```

VALUES (600, 1, 'Allen', 'Iverson', 'Allen.Iverson@sakilacustomer.org', 1, 1, '2000-1-1 00:00:00', '2000-01-01 00:00:00');

select * from customer;

执行结果:

customer_id	store_id	first_name	last_name	email	address_id	active	create_date	last_update
591	1	KENT	ARSENAU...	KENT.ARSENAULT@sakilacustomer.org	597	1	2006-02-14 22:04:37	2006-02-15 04:57:20
592	1	TERRANCE	ROUSH	TERRANCE.ROUSH@sakilacustomer....	598	0	2006-02-14 22:04:37	2006-02-15 04:57:20
593	2	RENE	MCALISTER	RENE.MCALISTER@sakilacustomer.org	599	1	2006-02-14 22:04:37	2006-02-15 04:57:20
594	1	EDUARDO	HIATT	EDUARDO.HIATT@sakilacustomer.org	600	1	2006-02-14 22:04:37	2006-02-15 04:57:20
595	1	TERRENCE	GUNDER...	TERRENCE.GUNDERSON@sakilacust...	601	1	2006-02-14 22:04:37	2006-02-15 04:57:20
596	1	ENRIQUE	FORSYTHE	ENRIQUE.FORSYTHE@sakilacustome...	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	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	Allen	Iverson	Allen.Iverson@sakilacustomer.org	1	1	2000-01-01 00:00:00	2000-01-01 00:00:00

- (3) 我们可以看到 sakila-schema.sql 里的语句是用于创建数据库的结构, 包括表、视图、触发器等, 而 sakila-data.sql 主要是用于往表写入数据。但 sakila-data.sql 里有这样一个建立触发器 payment_date 的语句, 这个触发器是否可以移到 sakila-schema.sql 里去执行? 为什么?

```

sakila-schema.sql sakila-data.sql x
0 10 20 30 40 50 60 70 80 90
0341 (16037,599,1,5843,'2.99','2005-07-10 17:14:27','2006-02-15 22:24:10'),
0342 (16038,599,2,6800,'9.99','2005-07-12 17:03:56','2006-02-15 22:24:10'),
0343 (16039,599,2,6895,'2.99','2005-07-12 21:23:59','2006-02-15 22:24:10'),
0344 (16040,599,1,8965,'6.99','2005-07-30 03:52:37','2006-02-15 22:24:11'),
0345 (16041,599,2,9630,'2.99','2005-07-31 04:57:07','2006-02-15 22:24:11'),
0346 (16042,599,2,9679,'2.99','2005-07-31 06:41:19','2006-02-15 22:24:11'),
0347 (16043,599,2,11522,'3.99','2005-08-17 00:05:05','2006-02-15 22:24:11'),
0348 (16044,599,1,14233,'1.99','2005-08-21 05:07:08','2006-02-15 22:24:12'),
0349 (16045,599,1,14599,'4.99','2005-08-21 17:43:42','2006-02-15 22:24:12'),
0350 (16046,599,1,14719,'1.99','2005-08-21 21:41:57','2006-02-15 22:24:12'),
0351 (16047,599,2,15590,'8.99','2005-08-23 06:09:44','2006-02-15 22:24:12'),
0352 (16048,599,2,15719,'2.99','2005-08-23 11:08:46','2006-02-15 22:24:13'),
0353 (16049,599,2,15725,'2.99','2005-08-23 11:25:00','2006-02-15 22:24:13');
0354 COMMIT;
0355
0356 --
0357 -- Trigger to enforce payment_date during INSERT
0358 --
0359
0360 CREATE TRIGGER payment_date BEFORE INSERT ON payment
0361 FOR EACH ROW SET NEW.payment_date = NOW();
0362
0363 --
0364 -- Dumping data for table rental
0365 --
0366
0367 SET AUTOCOMMIT=0;
0368 INSERT INTO rental VALUES (1,'2005-05-24 22:53:30',367,130,'2005-05-26 22:04:30',1,'2006-02-1

```

答: 不可以。如果移到 sakila-schema.sql 里执行, 那么在执行 sakila-data.sql 导入数据时会 将导入 payment 中的数据 payment_date 都修改为导入数据的时间。

3. 关于约束

- (1) store 表上建了哪几种约束? 这些约束分别实现什么功能? (至少写 3 个)

约束类型	功能
主键约束 PRIMARY KEY (`store_id`)	指定表的一列或几列的组 合的值在表中具有唯一 性, 能唯一地指定一行 记录。
惟一性约束 UNIQUE KEY `idx_unique_manager` (`manager_staff_id`)	指定一个或多个列的组 合的值具有惟一性, 以 防止在列中输入重复的 值。

外键约束 CONSTRAINT `fk_store_address` FOREIGN KEY(`address_id`) REFERENCES `address` (`address_id`) ON DELETE RESTRICT ON UPDATE CASCADE	当在定义主关键字约束的表中更新列值，时其它表中有与之相关的外关键字约束的表中的外关键字列也将被相应地做相同的更新。
--	---

(2) 图中第 343 行的 ON DELETE RESTRICT 和 ON UPDATE CASCADE 是什么意思？

答：

ON DELETE RESTRICT: 当在父表(即外键的来源表)中删除对应记录时，首先检查该记录是否有对应外键，如果有则不允许删除。

ON UPDATE CASCADE: 当在父表(即外键的来源表)中更新对应记录时，首先检查该记录是否有对应外键，如果有则也更新外键在子表(即包含外键的表)中的记录。

二、 创建新用户并分配权限

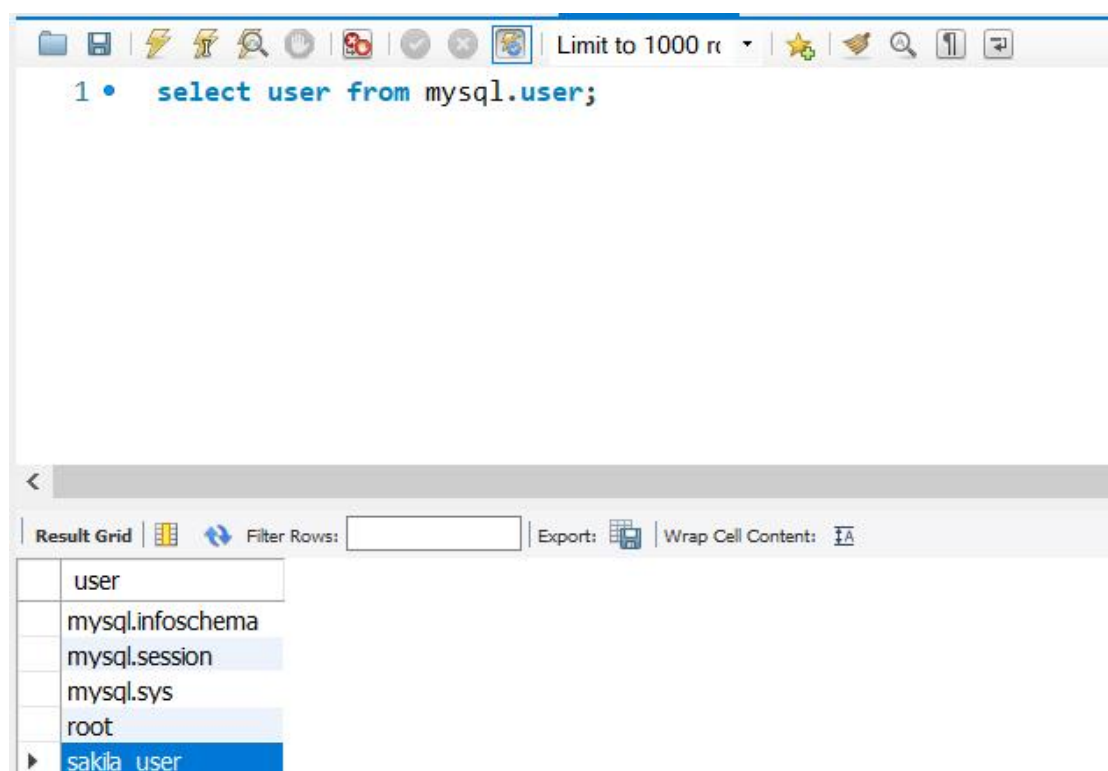
(截图语句和执行结果)

(1) 执行命令新建 sakila_user 用户（密码 123456）：



```
1 • create user 'sakila_user'@'localhost' identified by '123456';
```

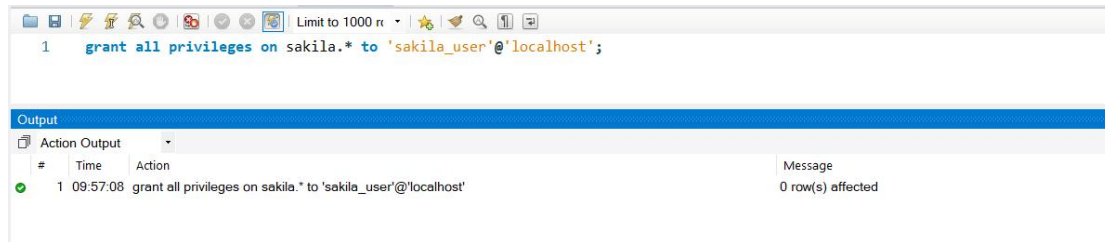
(2) 执行命令查看当前已有用户；



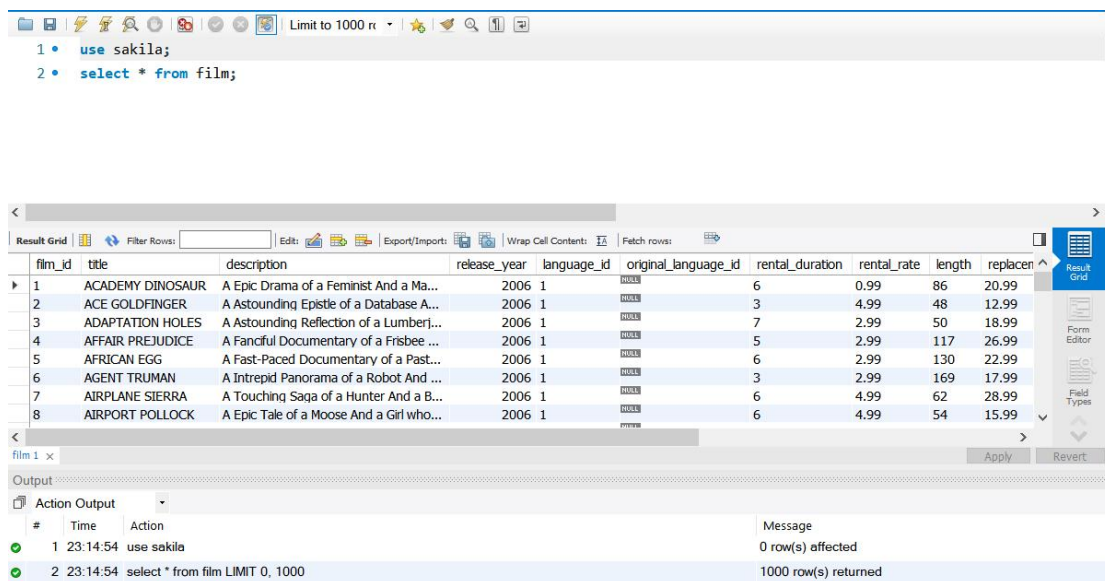
```
1 • select user from mysql.user;
```

user
mysql.infoschema
mysql.session
mysql.sys
root
sakila_user

(3) 执行命令把 sakila 数据库的访问权限赋予 sakila_user 用户；



(4) 切换到 sakila_user 用户，执行 `select * from film` 操作。



三、设计并实现

根据应用场景，为 Sakila 数据库合理地设计并实现：

(截图语句和执行结果)

1. 设计 1 个视图，至少关联 2 个表；

(1) 执行新建视图的语句，并截图 SQL 和执行结果：

```
create view `customer_address` as
select customer_id, first_name, last_name, address, city, country
from customer, address, city, country
where customer.address_id = address.address_id and address.city_id = city.city_id
and city.country_id = country.country_id
order by customer_id;
```

执行结果:

```
1 create view `customer_address` as
2 select customer_id, first_name, last_name, address, city, country
3 from customer, address, city, country
4 where customer.address_id = address.address_id and address.city_id = city.city_id
5 and city.country_id = country.country_id
6 order by customer_id;
```

Output:

#	Time	Action	Message
1	10:37:27	create view `customer_address` as select custom...	0 row(s) affected

(2) 执行 select * from [视图名], 截图执行结果:

1 • `SELECT * FROM sakila.customer_address;`

Result Grid

	customer_id	first_name	last_name	address	city	country
1	1	MARY	SMITH	1913 Hanoi Way	Sasebo	Japan
2	2	PATRICIA	JOHNSON	1121 Loja Avenue	San Bernardino	United States
3	3	LINDA	WILLIAMS	692 Joliet Street	Athenai	Greece
4	4	BARBARA	JONES	1566 Inegl Manor	Myingyan	Myanmar
5	5	ELIZABETH	BROWN	53 Idfu Parkway	Nantou	Taiwan
6	6	JENNIFER	DAVIS	1795 Santiago de Compostela Way	Laredo	United States
7	7	MARIA	MILLER	900 Santiago de Compostela Parkway	Kraqujevac	Yugoslavia
8	8	SUSAN	WILSON	478 Joliet Way	Hamilton	New Zealand

Output:

#	Time	Action	Message
1	10:37:27	create view `customer_address` as select custom...	0 row(s) affected
2	10:38:13	SELECT * FROM sakila.customer_address LIMIT ...	599 row(s) returned

2. 设计 1 个触发器, 需要体现触发器生效。

(1) 执行新建触发器的语句, 并截图 SQL 和执行结果:

```
create
trigger customer_update_date
before update ON customer
for each row
set new.last_update = NOW();
```


执行结果：

```
1 • create
2   trigger customer_update_date
3   before update ON customer
4   for each row
5   set new.last_update = NOW();
```

Output			
Action Output			
#	Time	Action	Message
✓ 1	10:54:06	create trigger customer_update_date before updat...	0 row(s) affected

(2) 验证触发器是否生效，截图验证过程：

1 • update customer			
2 set `customer`.`active` = 0			
3 where customer_id = 1;			
4 • select * from customer;			

Result Grid									
	customer_id	store_id	first_name	last_name	email	address_id	active	create_date	last_update
1	1	1	MARY	SMITH	MARY.SMITH@sakilacustomer.org	5	0	2006-02-14 22:04:36	2022-12-03 10:58:15
2	1	1	PATRICIA	JOHNSON	PATRICIA.JOHNSON@sakilacustomer....	6	1	2006-02-14 22:04:36	2006-02-15 04:57:20
3	1	1	LINDA	WILLIAMS	LINDA.WILLIAMS@sakilacustomer.org	7	1	2006-02-14 22:04:36	2006-02-15 04:57:20
4	2	2	BARBARA	JONES	BARBARA.JONES@sakilacustomer.org	8	1	2006-02-14 22:04:36	2006-02-15 04:57:20
5	1	1	ELIZABETH	BROWN	ELIZABETH.BROWN@sakilacustomer....	9	1	2006-02-14 22:04:36	2006-02-15 04:57:20
6	2	2	JENNIFER	DAVIS	JENNIFER.DAVIS@sakilacustomer.org	10	1	2006-02-14 22:04:36	2006-02-15 04:57:20
7	1	1	MARIA	MILLER	MARIA.MILLER@sakilacustomer.org	11	1	2006-02-14 22:04:36	2006-02-15 04:57:20
8	2	2	SUSAN	WILSON	SUSAN.WILSON@sakilacustomer.org	12	1	2006-02-14 22:04:36	2006-02-15 04:57:20

Output			
Action Output			
#	Time	Action	Message
✓ 1	10:58:15	update customer set `customer`.`active` = 0 where...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
✓ 2	10:58:15	select * from customer LIMIT 0, 1000	599 row(s) returned

四、思考题

(这部分不是必做题，供有兴趣的同学思考)

在阿里开发规范里有一条“【强制】不得使用外键与级联，一切外键概念必须在应用层解决。”请分析一下原因。你认为外键是否没有存在的必要？

答：我认为外键没有存在的必要，外键的存在导致在修改一个表的数据时需要对通过外键关联的表进行修改，影响修改的速度，并且维护的成本也比较高。