实验二报告

## 观察并回答问题

### 观察sakila.mwb并回答问题

1. 图中共有几个View？

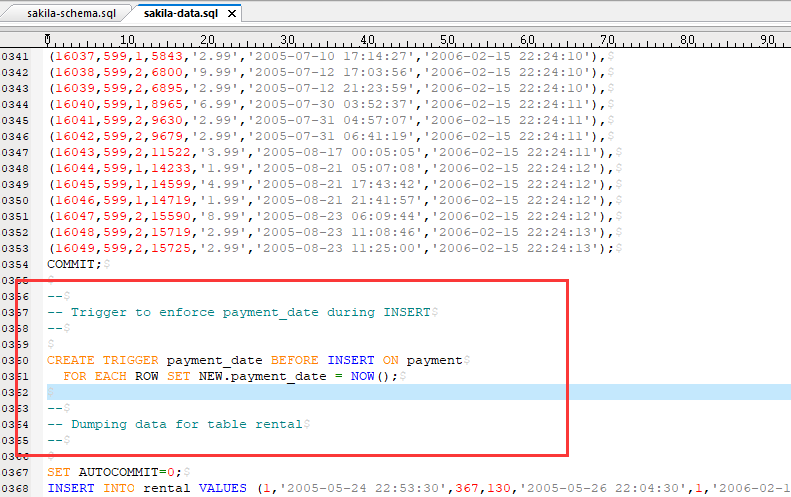
图中共有7个视图

1. 分析以下3个视图，回答以下问题：

|  |  |  |
| --- | --- | --- |
| 视图名 | 关联表 | 作用 |
| actor\_info | film, film\_category, film\_actor | 显示演员及其参演电影信息 |
| film\_list | film, film\_category, film\_actor,, film\_category, film\_actor, actor, category | 显示每一部电影的片名、描述、分类、评分、演员等具体信息 |
| sales\_by\_store | payment, rental, inventory, store, address, city, country, staff | 显示商店名，位置，经理和销售额 |

### 观察sakila-schema.sql和sakila-data.sql并回答问题

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的语句：



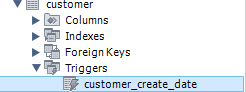
请同学们思考，这个触发器是否可以移到sakila-schema.sql里去执行？为什么？

不行，因为sakila-schema.sql只负责建立表的结构，数据的导入与它无关，如果触发器移到sakila-schema.sql里面，触发器不会触发

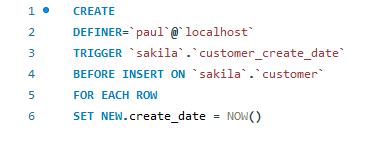
### 观察数据库的触发器customer\_create\_date并回答问题

1. customer\_create\_date触发器建在哪个表上？

在customer这个表上



1. 这个触发器实现什么功能？



这个触发器的功能是在插入一条记录时，将创建的日期自动置为当前执行操作时的日期。

1. 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）

insert into customer

values(601,1,'Alisan','Lisa','Alisan.Lisa@sakilacustomer.org',71,1,null,null);



## 设计并实现

根据应用场景，为Sakila数据库合理地设计并实现：

（注意：请将创建语句、调用结果截图记录到实验报告里）

1. 设计1个视图，至少关联3个表；

视图功能：显示每个客户每次的租借信息，包括用户名，电影片名，租借时间，归还时间和借还间隔。

创建语句：

CREATE

ALGORITHM=UNDEFINED

DEFINER=`root`@`localhost`

SQL SECURITY DEFINER

VIEW `customer\_rental` As

select

`c`.`customer\_id` As `ID`,

concat(`c`.`first\_name`,\_utf8mb4' ',`c`.`last\_name`) AS `name`,

`film`.`title` As `title`,

`r`.`rental\_date` As `租借日期`,

`r`.`return\_date` As `归还日期`,

(timestampdiff(day,`r`.`rental\_date`,`r`.`return\_date`)) As `租借时间`

from

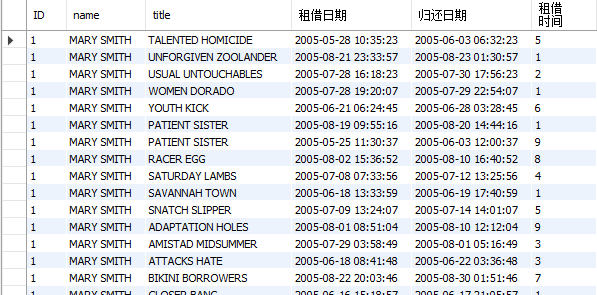
((`customer` `c`

join `rental` `r` on((`c`.`customer\_id` = `r`.`customer\_id` ))

join `inventory` on ((`r`.`inventory\_id` = `inventory`.`inventory\_id` ))

join film on ((`film`.`film\_id` = `inventory`.`film\_id`)) ))

视图：



1. 设计1个触发器，需要在报告里体现触发器生效。

新建一张表customer\_history，用来存储从customer中删掉的用户信息，触发器customer\_delete在customer表上。当customer表中删除一条记录时，触发器自动向customer\_history中插入相同的记录，删除的时间为操作执行时间。

customer\_history表创建语句：

CREATE TABLE `customer\_history` (

`customer\_id` smallint unsigned NOT NULL ,

`store\_id` tinyint unsigned NOT NULL,

`first\_name` varchar(45) NOT NULL,

`last\_name` varchar(45) NOT NULL,

`email` varchar(50) DEFAULT NULL,

`address\_id` smallint unsigned NOT NULL,

`active` tinyint(1) NOT NULL DEFAULT '1',

`delete\_date` datetime NOT NULL,

`last\_update` timestamp NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`customer\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

customer\_delete触发器创建语句：

DELIMITER ;;

CREATE TRIGGER

`delete\_customer` AFTER DELETE ON `customer` FOR EACH ROW BEGIN

INSERT INTO customer\_history(customer\_id, store\_id, first\_name, last\_name, email, address\_id,active,delete\_date,

last\_update)

VALUES(old.customer\_id, old.store\_id, old.first\_name, old.last\_name, old.email, old.address\_id,old.active,now(),

old.last\_update);

END ;;

DELIMITER ;

用实验1中的第11问和第13问：

向customer中插入第600号客户

insert into address

values(606,'1329 Fukuyama Street',null,'Heilongjiang',537,27107,288241215399,st\_geomfromtext('POINT (128.0449753 46.9804391)'),now());

insert into customer

values(600,1,"Luis","Sword",'Sword.Luis@sakilacustomer.org',606,1,now(),now());

然后删除此用户

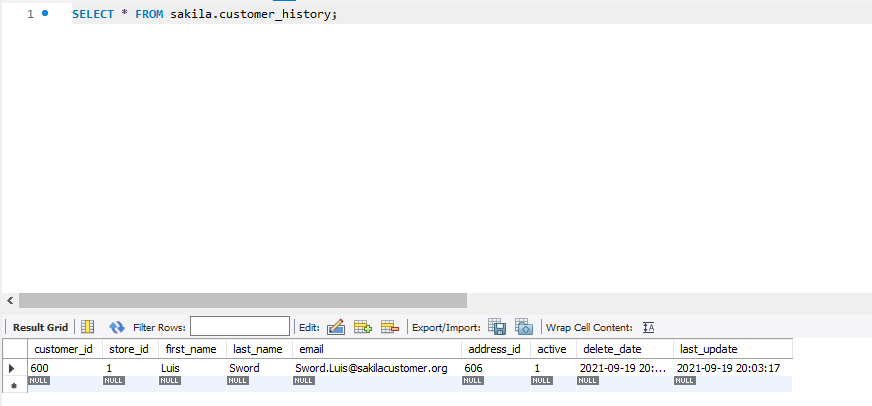
delete from customer

where customer\_id = 600;

delete from address

where address\_id = 606;

customer\_history中自动插入了此用户信息



## 思考题

*（这部分不是必做题，供有兴趣的同学思考）*

如果可以给这个数据库做修改，你想修改哪个地方？指出并说明原因。

MySQL新手，还不了解数据库，用着挺顺手，功能还没探索完全，是在没有资格谈论这个数据库有什么不足。