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？

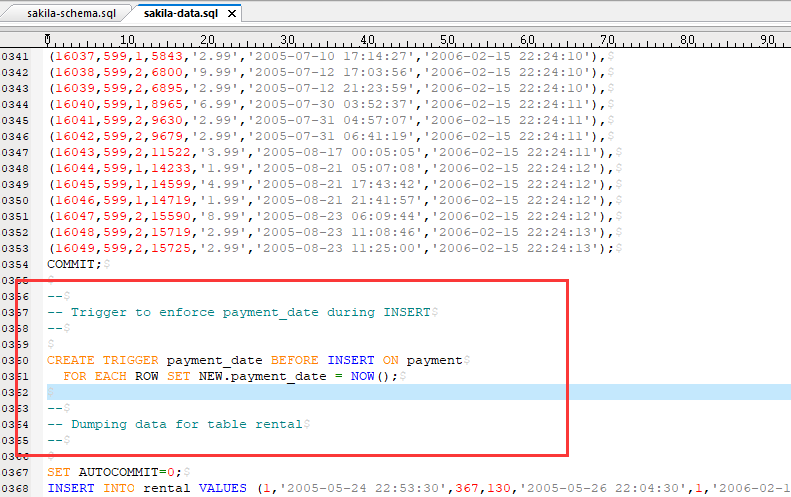
7 个 View.

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、actor、category | 展示演员的编号、姓名及其所出演的按照电影类别分类的所有电影名 |
| film\_list | film、film\_category、category、film\_actor、actor | 展示电影的详细信息，包括电影编号、名称、简介、分类、分级、租金、片长、以及出演演员列表 |
| sales\_by\_store | payment、rental、staff、inventory、store、address、city、country | 展示各租赁店的地址、经理及总销售额 |

### 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？

**不可以**。

该触发器涉及到 NOW() 函数，如果移到sakila-schema.sql 去执行，那么在执行sakila-data.sql文件时，该触发器涉及到的表中关于 date 的字段会因 NOW() 函数而变为现在的时间。因此这类触发器应该在表初始化完成后定义。

### 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？

customer 表。

1. 这个触发器实现什么功能？

在 customer 表中有新数据插入前，先为 create\_time 字段预设为当前时间。

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）

SQL 语句：

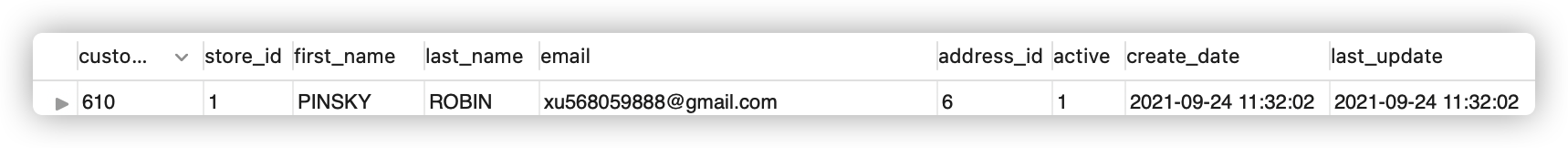
INSERT INTO customer (store\_id, first\_name, last\_name, email, address\_id)

VALUES (1, 'PINSKY', 'ROBIN', 'xu568059888@gmail.com', 6);

图形用户界面, 文本, 应用程序

描述已自动生成我们发现，在 DataGrip 中，该条语句会报错，信息如下：

根据提示信息，create\_date 没有 default value，在定义时其被设置为 not null，故在插入语句中应该对其进行赋值。

我们忽略这个错误，继续执行，发现执行成功，其结果如下图所示：

发现 create\_date 字段非空，其值就是执行插入操作时的时间，因此可以说明触发器生效。

## 设计并实现

根据应用场景，为Sakila数据库合理地设计并实现：

1. 设计1个视图，至少关联3个表；

**视图名**：order\_list

**关联表**：payment、customer、staff、rental、inventory、film

**SQL语句**：

CREATE VIEW `order` AS

SELECT `payment\_id` AS `OID`,

CONCAT(`c`.`first\_name`, ' ', `c`.`last\_name`) AS `customer\_name`,

CONCAT(`s`.`first\_name`, ' ', `s`.`last\_name`) AS `staff\_name`,

`f`.`title`,

`p`.`amount` AS `cost`,

`p`.`payment\_date` AS `create\_date`

FROM

(((((`payment` `p`

JOIN `customer` `c` ON ((`p`.`customer\_id` = `c`.`customer\_id`)))

JOIN `staff` `s` ON ((`p`.`staff\_id` = `s`.`staff\_id`)))

JOIN `rental` `r` ON ((`p`.`rental\_id` = `r`.`rental\_id`)))

JOIN `inventory` `i` ON ((`r`.`inventory\_id`=`i`.`inventory\_id`)))

JOIN `film` `f` ON ((`i`.`film\_id` = `f`.`film\_id`)))

ORDER BY `create\_date`;

表格

描述已自动生成**视图展示**：

**使用展示**：

① 编写SQL 语句：

SELECT customer\_name, SUM(cost) AS amount

FROM `order`

WHERE create\_date between '2005-05-24 00:00:00' and '2005-05-30 00:00:00'

GROUP BY customer\_name

ORDER BY amount DESC

LIMIT 10;

②图形用户界面, 表格

描述已自动生成 结果：

1. 设计1个触发器，需要在报告里体现触发器生效。

**触发器功能**：在 payment 表插入数据时，对 amount 字段的值进行合法性检查。

**触发器 SQL 语句**：

CREATE TRIGGER `ist\_amount\_check` BEFORE INSERT ON `payment`

FOR EACH ROW

BEGIN

DECLARE err\_msg varchar(16);

IF NEW.amount < 0

THEN

SET err\_msg = 'invalid amount!';

SIGNAL SQLSTATE 'XJW00' SET MESSAGE\_TEXT = err\_msg;

END IF;

END;

**功能展示**：

① 首先创建两条 SQL 语句：

INSERT INTO rental (rental\_date, inventory\_id, customer\_id, return\_date, staff\_id) VALUES ('2021-09-24 16:42:22',601,162,'2021-09-24 16:42:34', 1)

本条语句用于构造 payment 表中数据所需的 rental\_id.

INSERT INTO payment (customer\_id, staff\_id, rental\_id, amount, payment\_date)

VALUES (162, 1, 16050, -2.99, '2021-09-24 16:45:37')

本条语句用于构造不合法的 payment 插入语句（amount 小于 0）

②图形用户界面, 应用程序

描述已自动生成 结果：

说明触发器生效图形用户界面, 文本

描述已自动生成。修改上述插入语句，将 -2.99 更改为 2.99 ，得到如下结果：

说明插入成功。