

数据库系统课程实验报告

|  |  |
| --- | --- |
| 实验名称： | 实验五：数据更新 |
| 实验日期： | 2024.4.16- |
| 实验地点： | 文宣楼B313&课后 |
| 提交日期： | 2024.4. |
|  | |
| 学号： | 37220222203791 |
| 姓名： | 魏一诺 |
| 专业年级： | 软工2022级 |
| 学年学期： | 2023-2024学年第二学期 |

1. 实验目的

• 熟练掌握单条记录和小批量数据插入的方法（INSERT）

• 熟练掌握使用子查询实现数据插入的方法(INSERT INTO…SUBQUERY)

• 熟练掌握数据修改和删除的方法（UPDATE,DELETE,TRUNCATE）

• 理解约束对数据更新的影响

1. 实验内容和步骤
2. 为地区表 regions 新增一条记录：(‘5’,’Oceania’)。

INSERT INTO regions (region\_id, region\_name) VALUES ('5', 'Oceania');



1. 将 countries 表中的国家名为 Austrialia 的 region\_id 改为 5。

UPDATE countries SET region\_id = 5 WHERE country\_name = 'Australia';



（3）使用一条批量插入数据语句为 countries 表新增 5 条记录：('NO','Norway','1'), ('ES','Spain','1'),

('SE','Sweden','1'), ('PT','Portugal','1'), ('NZ','New Zealand','5')。

INSERT INTO countries (country\_id, country\_name, region\_id)

VALUES

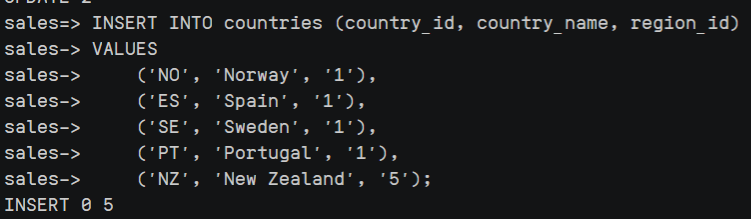
('NO', 'Norway', '1'),

('ES', 'Spain', '1'),

('SE', 'Sweden', '1'),

('PT', 'Portugal', '1'),

('NZ', 'New Zealand', '5');



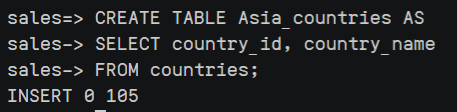
（4）创建一张名为 Asia\_countries(country\_id,country\_name)的新表，其中字段为 countries 表

中的同名字段。

CREATE TABLE Asia\_countries AS

SELECT country\_id, country\_name

FROM countries;



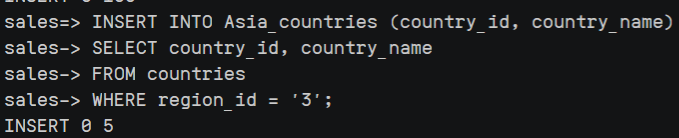
（5）将 countries 表中所有亚洲国家的数据插入到该表中。（要求使用插入子查询结果的方法实现）

INSERT INTO Asia\_countries (country\_id, country\_name)

SELECT country\_id, country\_name

FROM countries

WHERE region\_id = '3';



（6）创建一张名为 order\_total(order\_id,total\_price)的视图，该视图存放每个订单号及其总价，

其中 total\_price 为总价，其值为数量 quantity 与单价 unit\_price 乘积之和，order\_id，quantity

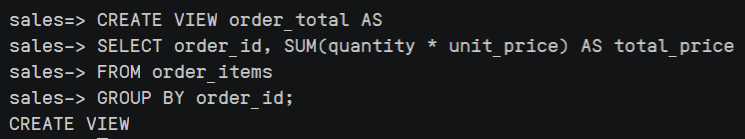
和 unit\_price 为 order\_items 表中的同名字段。

CREATE VIEW order\_total AS

SELECT order\_id, SUM(quantity \* unit\_price) AS total\_price

FROM order\_items

GROUP BY order\_id;

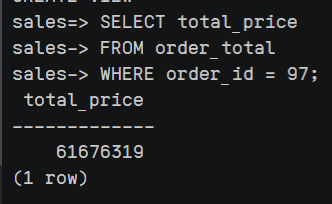


（7）查询 order\_total 视图中订单号 order\_id 为 97 的总价并记录该结果。

SELECT total\_price

FROM order\_total

WHERE order\_id = 97;

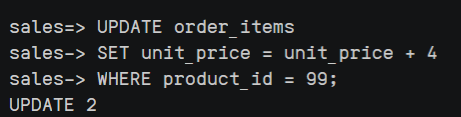


（8）将 order\_items 表中 product\_id 为 99 的单价 unit\_price 增加 4 元。

UPDATE order\_items

SET unit\_price = unit\_price + 4

WHERE product\_id = 99;

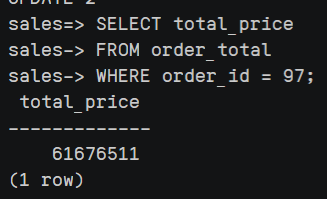


（9）查询视图 order\_total 中订单号 order\_id 为 97 的总价，将其与第（7）步的结果进行比较，观察其异同。

SELECT total\_price

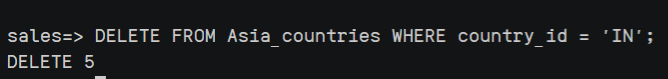
FROM order\_total

WHERE order\_id = 97;



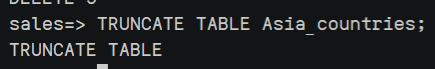
（10）使用 delete 命令删除 Asia\_countries 表中 country\_id 为 IN 的记录。

DELETE FROM Asia\_countries WHERE country\_id = 'IN';



1（11）使用 truncate 命令清空 Asia\_countries 表的所有记录。

TRUNCATE TABLE Asia\_countries;



（12）删除 Asia\_countries 表和视图 order\_total。

-- 删除表

DROP TABLE Asia\_countries;

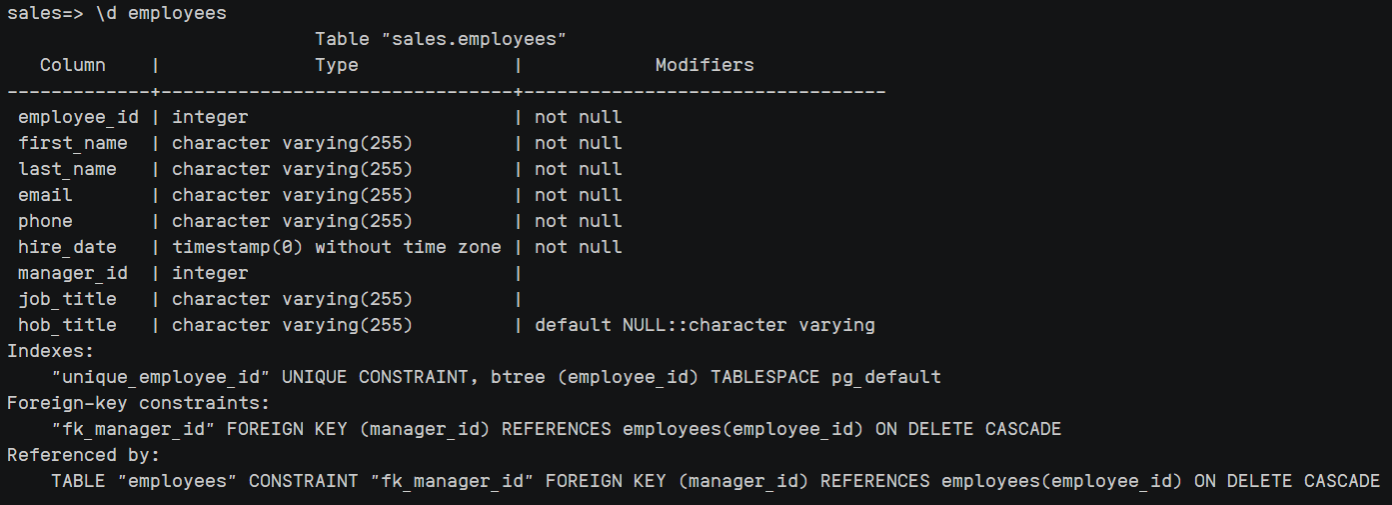
-- 删除视图

DROP VIEW order\_total;



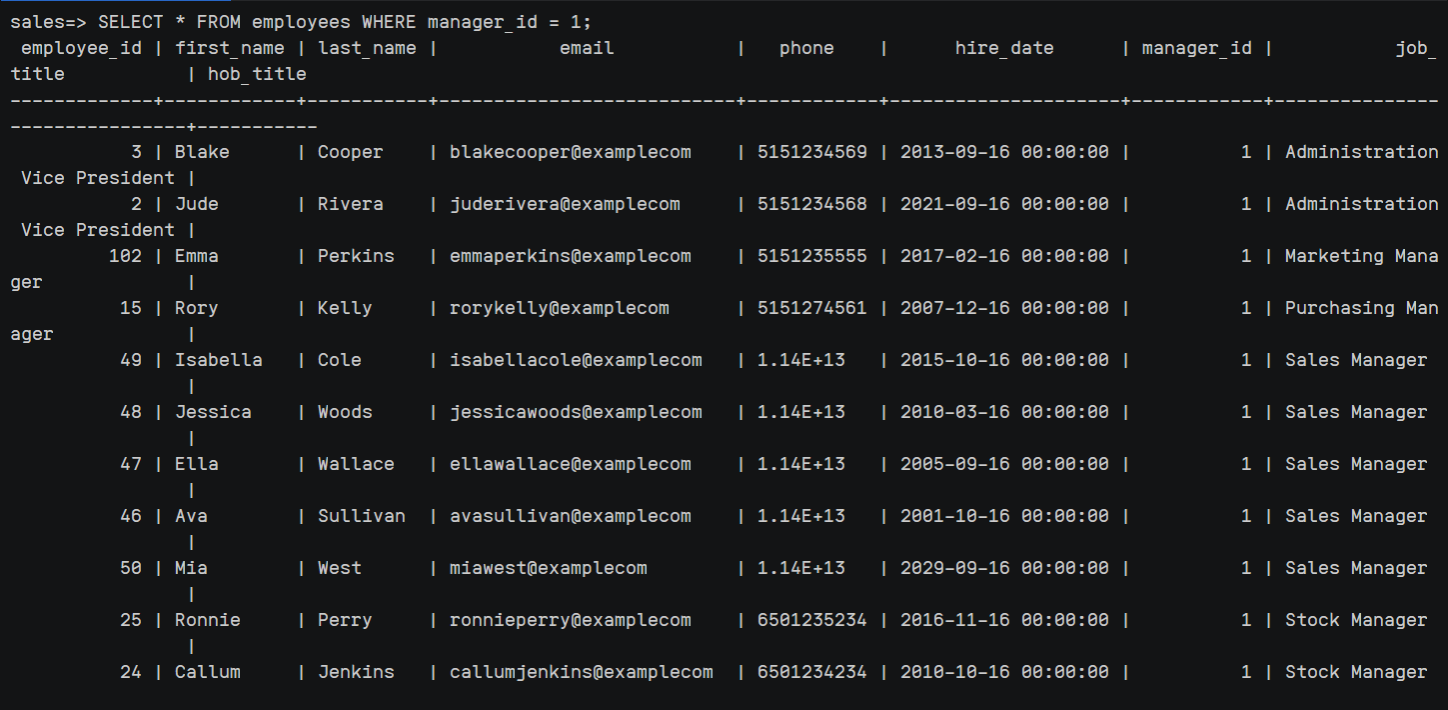


（13）使用命令\d employees 查看 employees 表的外码约束语句，包括 on delete cascade 选项。



（14）查询 employees 表中 manager\_id 为 1 的记录。

SELECT \* FROM employees WHERE manager\_id=1

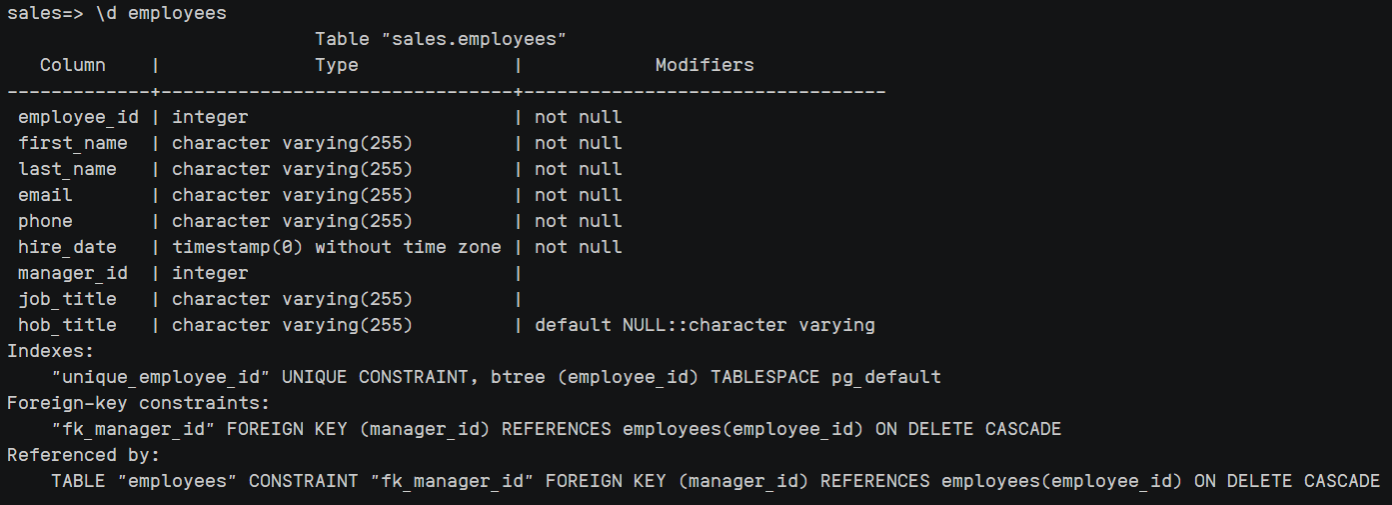


（15）修改 employees 表的外码约束，去掉外码约束中的 on delete cascade 选项，但保留原有的外码引用，即 manager\_id 引用本表上的 employee\_id。（可通过先删后建实现）

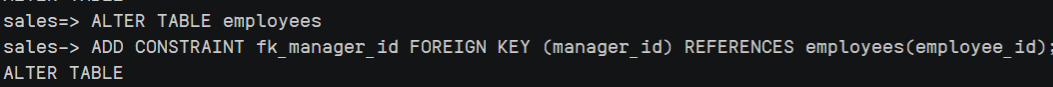
ALTER TABLE employees

ADD CONSTRAINT new\_fk\_manager\_id FOREIGN KEY (manager\_id) REFERENCES employees(employee\_id);

ALTER TABLE employees DROP CONSTRAINT fk\_manager\_id;

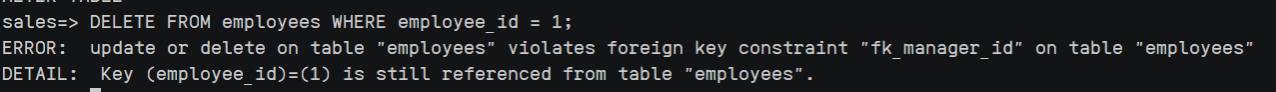




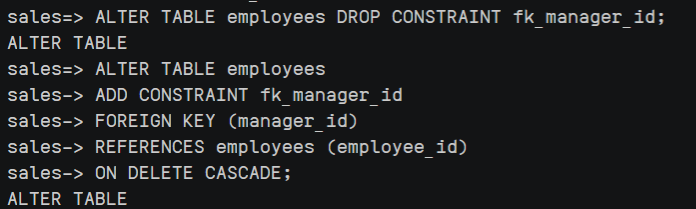


（16）删除 employees 表中 employee\_id 为 1 的记录，观察操作结果。

DELETE FROM employees WHERE employee\_id=1



（17）修改 employees 表的外码约束，增加 on delete cascade 选项，即回到最初的外码约束状态。



（18）再次执行第（16）步，观察操作结果



1. 实验总结

3.1 完成的工作

（将创建表的时候没有添加的约束补充添加上了）修改数据、新增数据、批量新增数据、使用表中数据新建表、使用表中数据和总和创建视图、查询特定数据、修改指定数据，体会了外键约束的含义

3.2 对实验的认识

练习了更多丰富的指令用法，意识到它可以完成更多的任务

3.3 遇到的困难及解决方法

困难1：“使用命令\d employees 查看 employees 表的外码约束语句，包括 on delete cascade 选项。”完成这个任务时，\d employees后出现的表中并没有任何外码约束语句的信息。

解决：原来是我没有添加外码约束语句，创建表时只添加了非空约束。添加上就好啦

ALTER TABLE employees --添加外键约束

ADD CONSTRAINT fk\_manager\_id

FOREIGN KEY (manager\_id)

REFERENCES employees (employee\_id)

ON DELETE CASCADE;

ALTER TABLE employees --添加唯一约束

ADD CONSTRAINT unique\_employee\_id UNIQUE (employee\_id);