

数据库系统课程实验报告

|  |  |
| --- | --- |
| 实验名称： | 实验七：数据库的完整性 |
| 实验日期： | 2024.5.13-- |
| 实验地点： | 文宣楼B313 |
| 提交日期： | 2024.5.18 |
|  | |
| 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
   1. 理解并掌握关系数据库完整性的运行机制
   2. 完整性约束定义>完整性约束检查>违约处理
   3. 理解并掌握关系数据库完整性主要约束类型及其含义和作用
   4. PRIMARY KEY，FOREIGN KEY，NOT NULL，UNIQUE，CHECK
   5. 理解并掌握关系数据库完整性定义、修改、删除和重命名的方法
   6. CREATE TABLE，ALTER TABLE
   7. 熟练掌握 openGauss 下通过系统表 pg\_constraint 查看完整性信息的方法
   8. 熟练掌握 openGauss 下通过查看表结构来查看主外码信息的方法
   9. 熟练掌握 openGauss 下通过查看完整性约束定义的方法
2. 实验内容和步骤
3. 创建两张表：雇员表 Emp 和工作表 Work，它们的表结构如下：



步骤如下：

1.

cd /opt/software/openGauss/script/

//步骤 1 在数据库主节点服务器上，切换至omm操作系统用户环境

su - omm

gs\_om -t status //步骤 2 查看服务是否启动

gs\_om -t start //步骤 3 启动数据库服务

gsql -d sales -p 26000 -U wyn -W Bigdata@123 -r //连接到数据库

4.创建表

Create table Work(WorkID CHAR(3),LowerSalary DECIMAL (8,2),UpperSalary DECIMAL (8,2) );

Create table Emp(Eid CHAR(5) NOT NULL,Ename varchar(10),WorkID CHAR(3),Salary DECIMAL(8,2),Phone varchar(11) NOT NULL);;





（2）分别为两张表插入如下数据，查看插入操作是否成功。

雇员表数据：{(‘10001‘, ’Smith‘, ’001‘, 2000, ’13800010001‘),(’10001‘, ’Jonny‘, ’001‘, 3000,’13600010002‘)，

(’10002‘, ’Mary‘, ’002‘, 2500, ’13800020002‘) }

工作表数据：{ (‘001‘, 1000, 5000)，(’002‘, 2000, 8000) }

步骤如下：

INSERT INTO Emp (Eid, Ename, WorkID, Salary, Phone)

VALUES

('10001', 'Smith', '001', 2000, '13800010001'),

('10001', 'Jonny', '001', 3000, '13600010002'),

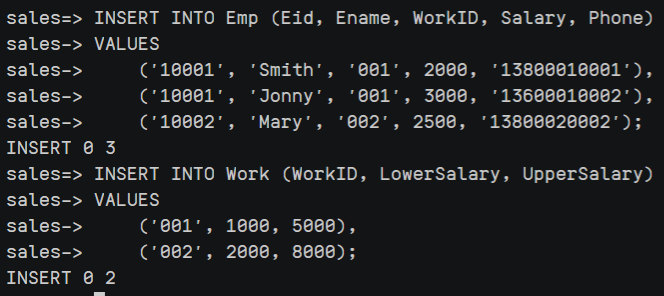
('10002', 'Mary', '002', 2500, '13800020002');

INSERT INTO Work (WorkID, LowerSalary, UpperSalary)

VALUES

('001', 1000, 5000),

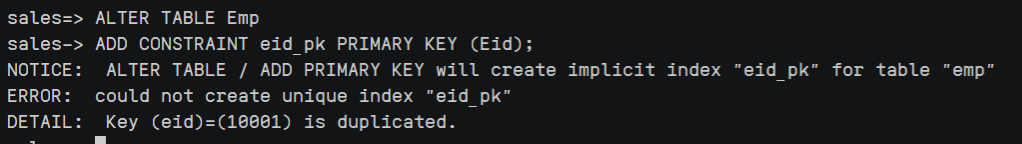
('002', 2000, 8000);



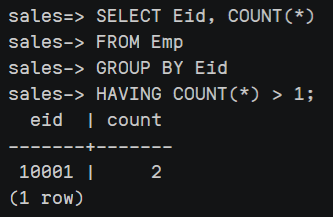
（3）修改雇员表的结构，设置 Eid 为主码，主码名称为 eid\_pk，查看该操作是否成功。若不成功，请说明原因并思考如何处理才能成功添加约束。要求：所有约束都要显式给出约束名，不可由系统默认，因为删除约束时需要用到约束名。

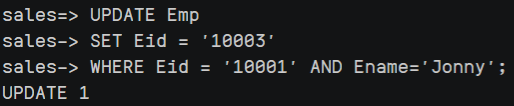
ALTER TABLE Emp

ADD CONSTRAINT eid\_pk PRIMARY KEY (Eid);

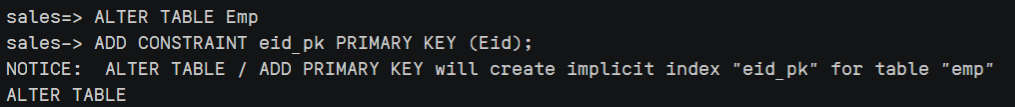


在尝试创建主键约束时，发现了重复的键值（在列 "Eid" 中）。主键约束要求所有值都是唯一的，因此不能有重复值,创建失败

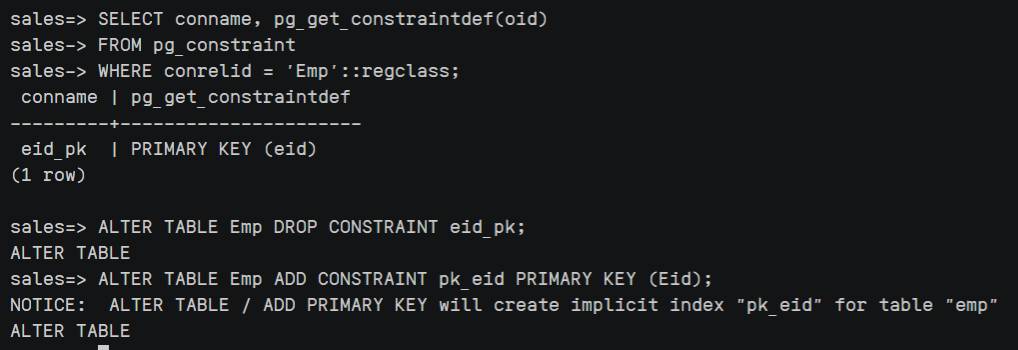
查找到了重复的eid

进行更新：

创建成功：



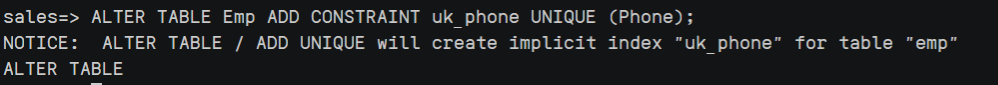
（4）将 eid 为主码的约束名 eid\_pk 改为 pk\_eid。



（5）设置雇员表中的 phone 字段取唯一值，查看该操作是否成功？若不成功说明原因。

-- 添加唯一性约束

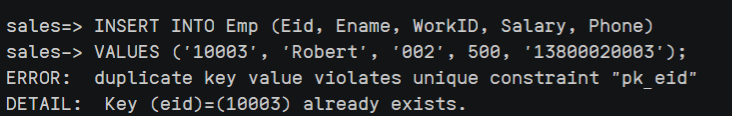
ALTER TABLE Emp ADD CONSTRAINT uk\_phone UNIQUE (Phone);

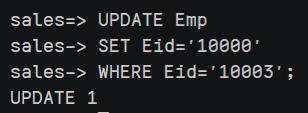


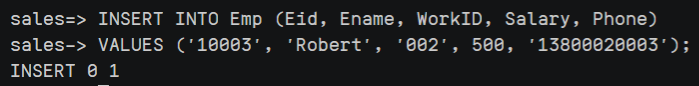
（6）给雇员表添加一条新记录(‘10003’,’Amy’,’002’, 3000,’13800020003’)，查看执行结果。

INSERT INTO Emp (Eid, Ename, WorkID, Salary, Phone)

VALUES ('10003', 'Robert', '002', 500, '13800020003');



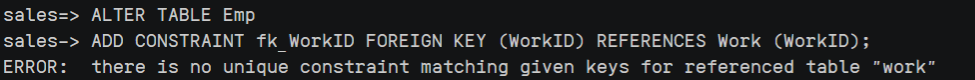
更新一下：

插入成功：

（7）设置工作表的 WorkID 为主码。

ALTER TABLE Work

ADD CONSTRAINT pk\_WorkID PRIMARY KEY (WorkID);

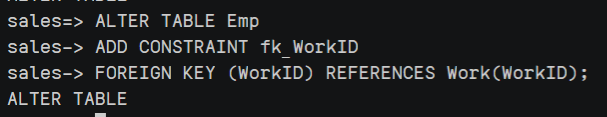


（8）修改雇员表，设置雇员表的 WorkID 字段为外码，它引用工作表中的 WorkID 字段，查看操作是否成功？若不成功说明原因。

ALTER TABLE Emp

ADD CONSTRAINT fk\_WorkID

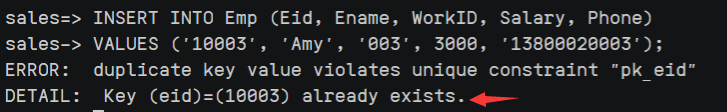
FOREIGN KEY (WorkID) REFERENCES Work(WorkID);



（9）给雇员表添加一条新记录(‘10003’,’Amy’, ‘003’, 3000, ‘13800020003’)，查看操作是否成功？若不成功说明原因。

INSERT INTO Emp (Eid, Ename, WorkID, Salary, Phone)

VALUES ('10003', 'Amy', '003', 3000, '13800020003');

不成功：

因为Eid是主键不能重复，已存在Eid为10003的了

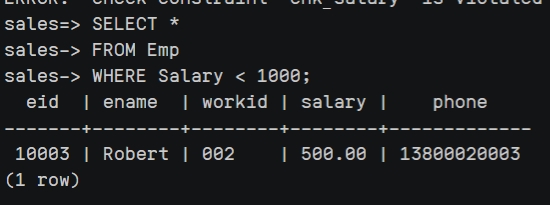
（10）在雇员表中，设置雇员工资必须大于或等于 1000。查看操作是否成功？若不成功说明原因。

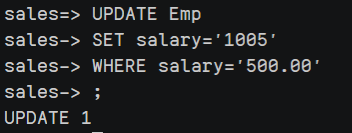
ALTER TABLE Emp

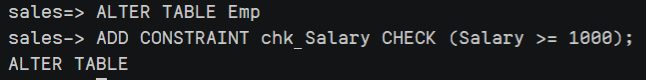
ADD CONSTRAINT chk\_Salary CHECK (Salary >= 1000);

不成功：

因为有现有的行不符合这个约束：



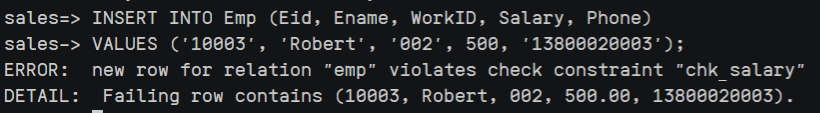
更新一下：

添加成功：

（11）给雇员表添加一条新记录(‘10003’,’Robert’,‘002’,500,‘13800020003’)，查看执行操作是否成功？若不成功说明原因。

INSERT INTO Emp (Eid, Ename, WorkID, Salary, Phone)

VALUES ('10003', 'Robert', '002', 500, '13800020003');



不满足工资要高于1000的约束

（12）在工作表中，设置其最低工资不超过最高工资。

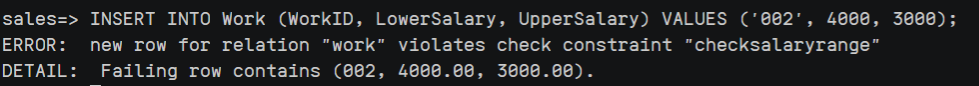
ALTER TABLE Work

ADD CONSTRAINT CheckSalaryRange CHECK (LowerSalary <= UpperSalary);



（13）给工作表添加一条新记录(‘002’,4000,3000)，查看操作是否成功？若不成功说明原因。

INSERT INTO Work (WorkID, LowerSalary, UpperSalary) VALUES ('002', 4000, 3000);



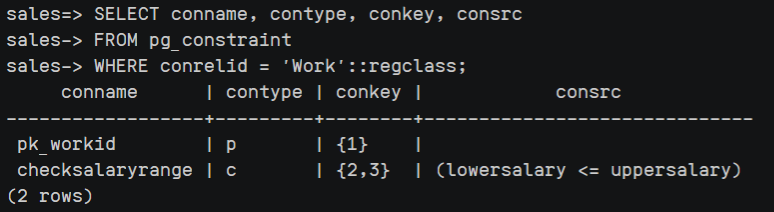
试图向表中插入的记录的 LowerSalary 值（4000）大于 UpperSalary 值（3000），这违反了在表定义中设置的约束，因此插入操作会失败。

（14）通过查看 openGauss 的系统表 pg\_constraint 了解表上的约束。

SELECT conname, contype, conkey, consrc

FROM pg\_constraint

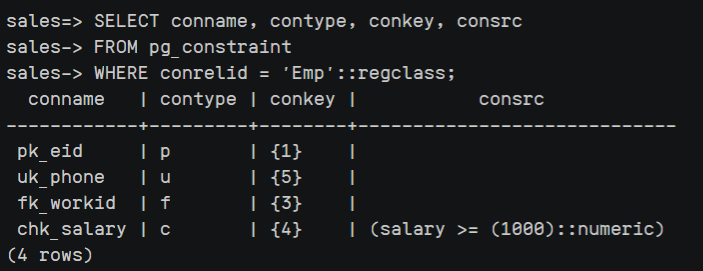
WHERE conrelid = 'Work'::regclass;



SELECT conname, contype, conkey, consrc

FROM pg\_constraint

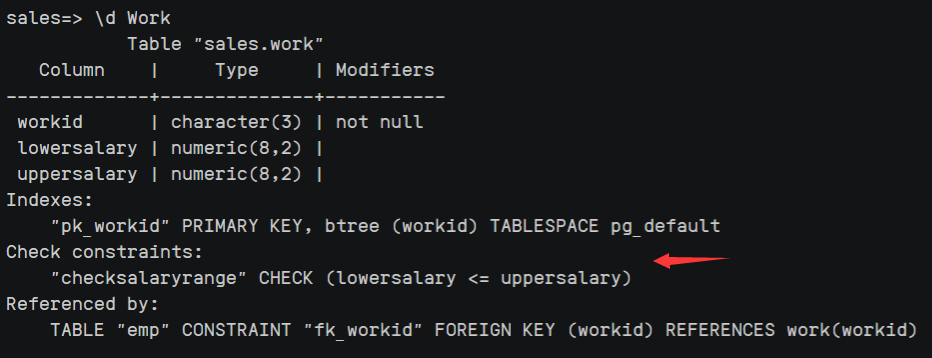
WHERE conrelid = 'Emp'::regclass;

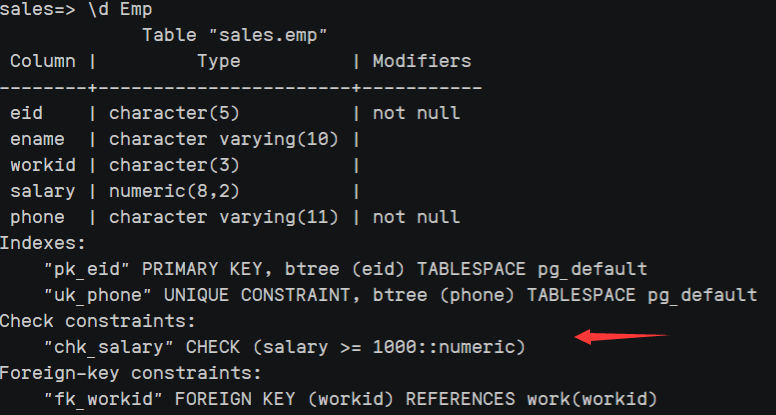


（15）通过 gsql 命令\d+ table\_name 查看该表上的约束定义。

\d Work

\d Emp





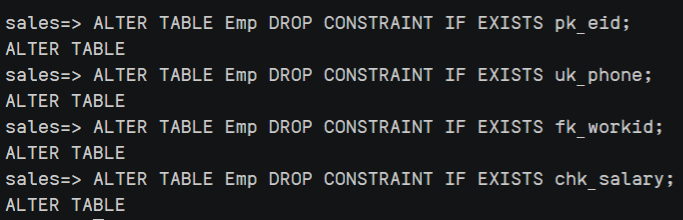
（16）删除雇员表的所有约束，包括主码约束、外码约束和其他约束。

ALTER TABLE Emp DROP CONSTRAINT IF EXISTS pk\_eid;

ALTER TABLE Emp DROP CONSTRAINT IF EXISTS uk\_phone;

ALTER TABLE Emp DROP CONSTRAINT IF EXISTS fk\_workid;

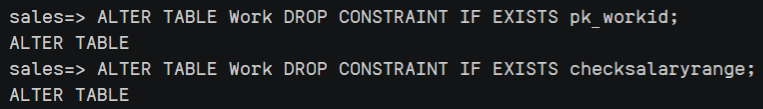
ALTER TABLE Emp DROP CONSTRAINT IF EXISTS chk\_salary;



（17）删除工作表所有约束，包括主码约束。

ALTER TABLE Work DROP CONSTRAINT IF EXISTS pk\_workid;

ALTER TABLE Work DROP CONSTRAINT IF EXISTS checksalaryrange;



1. 实验总结

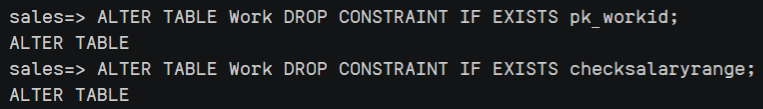
3.1 完成的工作

创建了两张表：雇员表和工作表，并在表上进行了一系列添加主键约束、外键约束、值不能高于某数的约束等等

3.2 对实验的认识

通过实验增进了对各类约束的认识，也更加熟悉添加约束的防方法。

3.3 遇到的困难及解决方法

没遇到什么太大的困难，小困难及其解决记录在上边的步骤里了。