# 实验一：建立数据库

#### 姓名：谢宝玛 学号：1120233506

## **（一）**实验目的

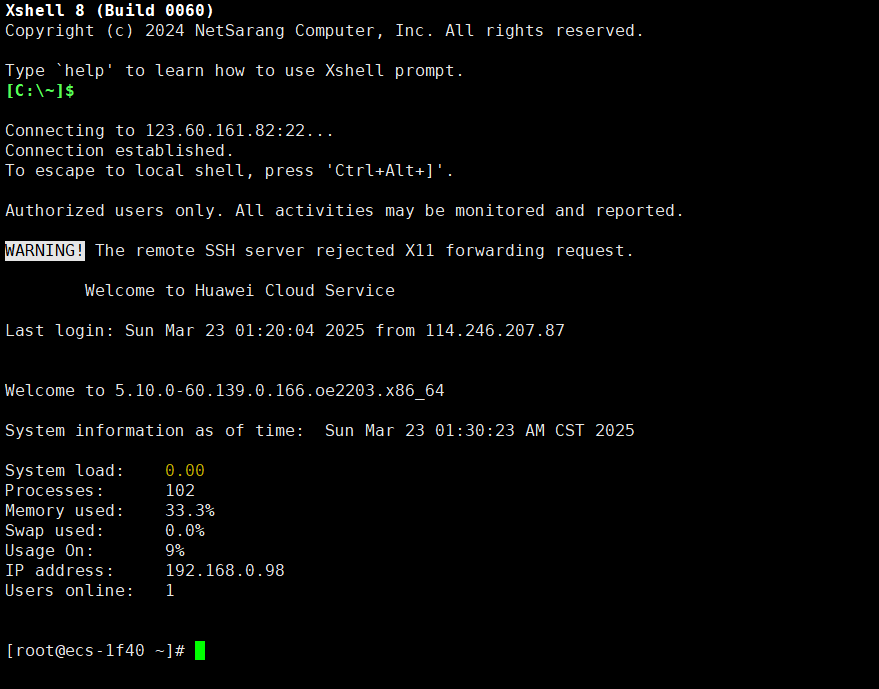
建立“学籍与成绩管理系统”表格；

• 建立表之间的参照关系

• 建立适当的索引

## （二）实验过程：

### 1，购买华为ECS服务器，选择openEuler操作系统，远程登录



### 购买Gauss数据库服务器，远程登录



### 建表

#### （1）学生表



##### SQL

CREATE TABLE xs (

xh CHAR(10) PRIMARY KEY, -- 学号，字符类型，长度10，主键

xm CHAR(8) NOT NULL, -- 姓名，字符类型，长度8，不能为空

ydh CHAR(2) NOT NULL, -- 所属学院代号，字符类型，长度2，不能为空

bj CHAR(8) NOT NULL, -- 班级，字符类型，长度8，不能为空

chrq DATE NOT NULL, -- 出生日期，日期类型，不能为空

xb CHAR(2) NOT NULL, -- 性别，字符类型，长度2，不能为空

FOREIGN KEY (ydh) REFERENCES xyb(ydh) -- 外键，关联学院表的ydh字段

);

INSERT INTO xs (xh, xm, ydh, bj, chrq, xb)

VALUES ('20230001', '张三', '01', '01', '2000-05-15', 'M');

##### 结果

屏幕截图 2025-03-23 160620

#### （2）课程表



##### SQL

CREATE TABLE kc (

kcbh CHAR(3) PRIMARY KEY, -- 课程编号，字符类型，长度3，主键

kc CHAR(20) NOT NULL, -- 课程名称，字符类型，长度20，不能为空

lx CHAR(10), -- 课程类型，字符类型，长度10，允许为空

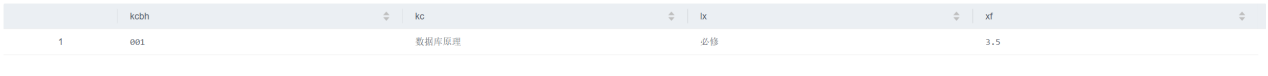
xf NUMERIC(5,1) -- 学分，数值类型，总长度5，小数位1，允许为空

);

INSERT INTO kc (kcbh, kc, lx, xf)

VALUES ('001', '数据库原理', '必修', 3.5);

##### 结果



#### （3）学院表



##### SQL

CREATE TABLE xyb (

ydh CHAR(2) PRIMARY KEY, -- 学院代号，字符类型，长度2，主键

ymc CHAR(30) NOT NULL -- 学院名称，字符类型，长度30，不能为空

);

INSERT INTO xyb (ydh, ymc)

VALUES ('01', '计算机学院');

##### 结果



#### （4）教师表



##### SQL

CREATE TABLE js (

xm CHAR(8) NOT NULL, -- 姓名，字符类型，长度8，不能为空

jsbh CHAR(10) PRIMARY KEY, -- 教师编号，字符类型，长度10，主键

zc CHAR(6) NOT NULL, -- 职称，字符类型，长度6，不能为空

ydh CHAR(2), -- 所属学院代号，字符类型，长度2，允许为空

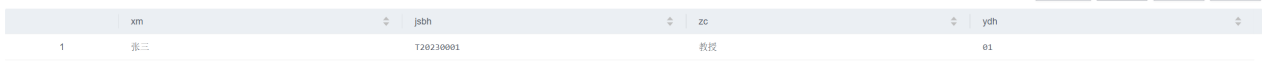
FOREIGN KEY (ydh) REFERENCES xyb(ydh) -- 外键，关联学院表的ydh字段

);

INSERT INTO js (xm, jsbh, zc, ydh)

VALUES ('张三', 'T20230001', '教授', '01');

##### 结果



#### （5）授课表



##### SQL

CREATE TABLE sk (

kcbh CHAR(3) NOT NULL, -- 课程编号，字符类型，长度3，不能为空

bh CHAR(10) NOT NULL, -- 教师编号，字符类型，长度10，不能为空

PRIMARY KEY (kcbh, bh), -- 联合主键，由课程编号和教师编号组成

FOREIGN KEY (kcbh) REFERENCES kc(kcbh), -- 外键，关联课程表的kcbh字段

FOREIGN KEY (bh) REFERENCES js(jsbh) -- 外键，关联教师表的jsbh字段

);

INSERT INTO sk (kcbh, bh)

VALUES ('001', 'T20230001');

##### 结果



#### （6）学生选课表



##### SQL

CREATE TABLE xk (

xh CHAR(10) NOT NULL, -- 学号，字符类型，长度10，不能为空

kcbh CHAR(3) NOT NULL, -- 课程编号，字符类型，长度3，不能为空

jsbh CHAR(10) NOT NULL, -- 教师编号，字符类型，长度10，不能为空

cj NUMERIC(5,1), -- 成绩，数值类型，总长度5，小数位1，允许为空

PRIMARY KEY (xh, kcbh, jsbh), -- 联合主键，由学号、课程编号和教师编号组成

FOREIGN KEY (xh) REFERENCES xs(xh), -- 外键，关联学生表的xh字段

FOREIGN KEY (kcbh) REFERENCES kc(kcbh), -- 外键，关联课程表的kcbh字段

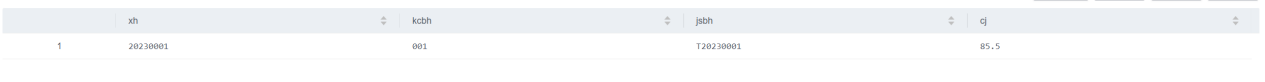
FOREIGN KEY (jsbh) REFERENCES js(jsbh) -- 外键，关联教师表的jsbh字段

);

INSERT INTO xk (xh, kcbh, jsbh, cj)

VALUES ('20230001', '001', 'T20230001', 85.5);

##### 结果



#### （7）结果



### 建立索引

#### 在xs表中将xh作为唯一索引

##### SQL

CREATE INDEX idx\_xh ON xs (xh);

#### 在js表中将jsbh作为唯一索引

##### SQL

CREATE INDEX idx\_jsbh ON js (jsbh);

#### 在js表中将jsbh作为唯一索引

##### SQL

CREATE INDEX idx\_xyh ON xyb (ydh);

## **（三）**实验结论

### 1，问题

#### （1）openGauss数据库难以安装（未解决）

本人已经在openEuler系统和CentOS 系统中尝试多次，都会遇到各种报错，最终放弃在服务器中安装openGauss数据库。

解决：购买了华为的GaussDB 数据库服务器。

#### （2）存在编码的问题

在Gauss数据库中，我设置的编码方式是UTF-8，由于中文和英文的编码不同，占据的位数也不同，于是会出现插入中文数据报错的情况，比如：

value too long for type character(8) Where: referenced column: xm

需要注意中文的长度。

## 实验体会

从安装开始openGauss数据库就透露出一股半成品的味道。

安装的过程及其繁琐，下载软件包，更改用户，更改系统配置，设置用户权限，使用脚本进行安装，只能使用single-node，这些东西都让openGauss不像是一个成熟的数据库软件。

连接数据库的客户端更是离谱，data studio根本不是一个发行的软件，更像一个开源的工具，需要自己配置Java环境，下载部署。最终我也没有成功使用data studio，而是用了华为提供的DAS。

个人感觉实验一的内容并不难，但是花了很多时间在安装，部署，启动这些琐碎的事情上，实在让人恼火，而且本人最终并没有成功，而是直接去买了台数据库服务器。与优秀，开源，成熟的关系型数据库相比（比如MySQL），openGauss明显还不完善。但在使用上感觉还可以，与传统的关系型数据库没有什么区别。