**湘潭大学**

**实 验 报 告**

**课 程： 数据库原理**

**实验题目： 数据库设计与实施**

**学 院： 信息工程学院**

**班 级： 2017软件工程1班**

**学 号： 2016550813**

**姓 名： 朱琦**

**完成日期： 2019年5月27日**

### 一．实验目标

通过理论学习与实验设计，使自己具备如下能力：

1. 将数学、自然科学、工程基础和专业知识用于解决软件工程相关领域的复杂工程问题的能力。
2. 针对数据库应用领域的具体情况，提出数据库的设计方案，并对其进行规范的能力。
3. 选用合适的数据库管理工具承担数据库系统的实施与维护工作的能力。

### 二．实验项目

### 本组成员：朱琦、黄家文、彭祯源、陶重月

### 1．图书采购与库存

### ① E-R模型

### D:\软件\1473066484\FileRecv\MobileFile\Image\~QPLLSDGA26L7T@GYG}PXRF.png

### ②关系模型

根据E-R模型转换成关系数据模型，如下：

**⑴ press(press\_id,name,address,phone)**

**主键：press\_id**

**⑵ buyer(id,name,phone)**

**主键：id**

**⑶book(ISBN,book\_publisher,name,author,unit\_price,count,order\_id,stock)**

**主键:ISBN**

**外键: order\_id**

**⑷purchase\_order(order\_id,total\_money,order\_date,consignee\_address, press\_id, id)**

**主键：order\_id**

**外键1：press\_id**

**外键2: id**

**(5) include(order\_id, ISBN)**

**主键： order\_id, ISBN**

**外键1：order\_id**

**外键2：ISBN**

### ③关系模型的说明

press

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| press\_id | varchar(20) | 主键 | 出版社编号 |
| name | varchar(20) | not null | 出版社名字 |
| address | varchar(30) | not null | 出版社地址 |
| phone | varchar(11) | not null | 电话号码 |

buyer

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| id | varchar(18) | 主键 | 采购员身份证号 |
| name | varchar(20) | not null | 采购员姓名 |
| phone | varchar(11) | not null | 电话号码 |

book

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| ISBN | varchar(13) | 主键 | 书号 |
| name | varchar(20) | not null | 书名 |
| author | varchar(20) | not null | 作者 |
| unit\_price | number(\*,1) | not null | 单价 |
| count | int | not null | 数量 |
| stock | int | Not null | 书店库存量 |

purchase\_order

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性 | 类型 | 约束 | | 说明 |
| order\_id | char(15) | 主键 | | 订单号 |
| total\_money | number(\*,1) | not null | | 总金额 |
| order\_date | varchar(15) | not null | | 订购日期 |
| consignee\_address | varchar(30) | not null | 收货地址 | |
| press\_id | varchar(20) | 外键 | 出版社编号 | |
| id | varchar(18) | 外键 | 采购员身份证号 | |

include

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| order\_id | varchar(15) | 主键（外键） | 订购单号 |
| ISBN | varchar(13) | 主键（外键） | 书号 |

### ④创建基本表

### 用CREATE TABLE命令在Oracle中实现该模型。

### create table T\_press(

### press\_id varchar(20),

### name varchar(20) not null,

### address varchar(30) not null,

### phone varchar(11) not null,

### primary key (press\_id)

### );

### create table T\_buyer(

### id varchar(18),

### name varchar(20) not null,

### phone varchar(11) not null,

### primary key (id)

### );

### create table T\_book (

### ISBN varchar(13),

### book\_publisher VARCHAR2(30) not null,

### name varchar(20) not null,

### author varchar(20) not null,

### unit\_price number(\*,1) not null,

### count int not null,

### stock int not null,

### primary key (ISBN)

### );

### create table T\_purchase\_order

### (order\_id varchar(15),

### order\_date varchar(15) not null,

### total\_money number(\*,1) not null,

### consignee\_address varchar(30) not null,

### id varchar(18) not null,

### press\_id varchar(20) not null,

### primary key (order\_id),

### foreign key (press\_id) references T\_press on delete cascade,

### foreign key (id) references T\_buyer on delete cascade

### );

### create table T\_include

### (order\_id varchar(15),

### ISBN varchar(13),

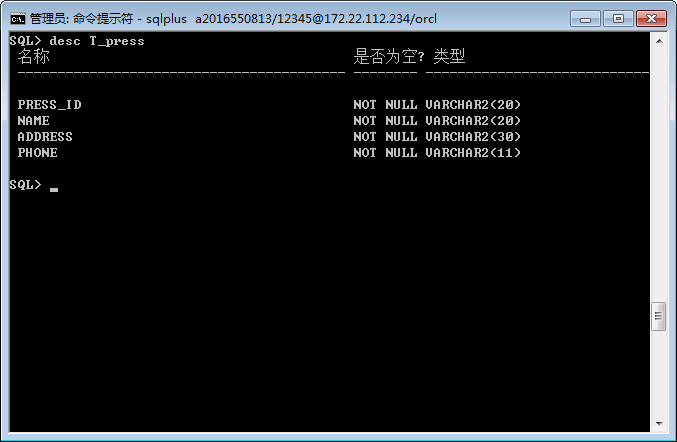
### primary key(order\_id,ISBN),

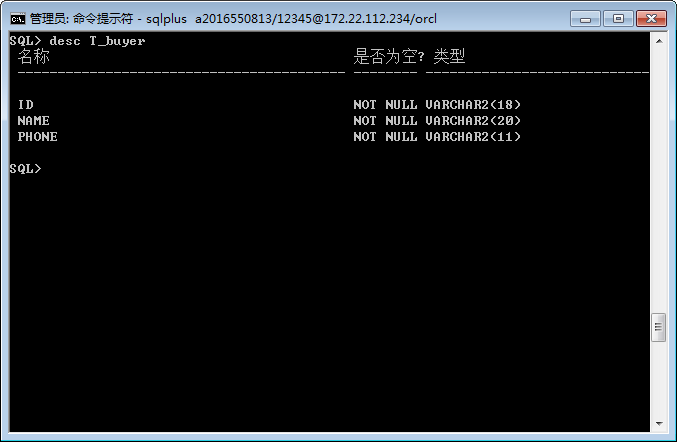
### foreign key (order\_id) references T\_purchase\_order on delete cascade,

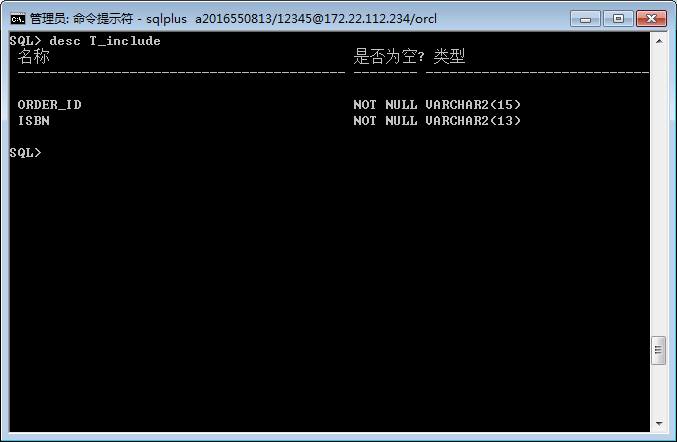
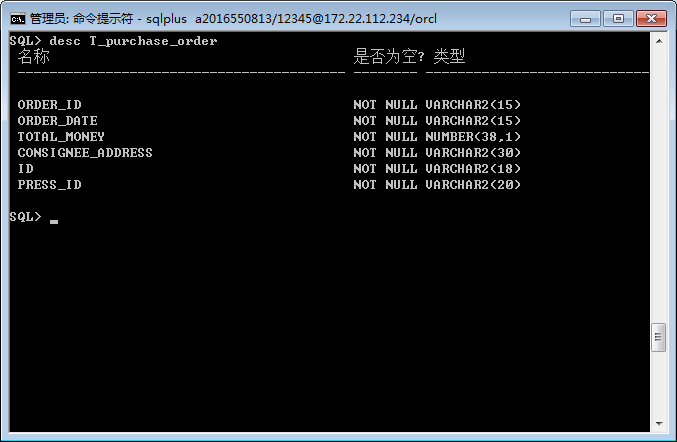
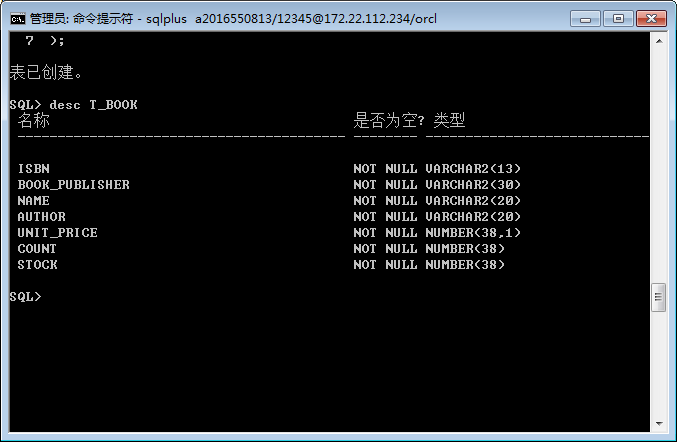
### foreign key (ISBN) references T\_book on delete cascade

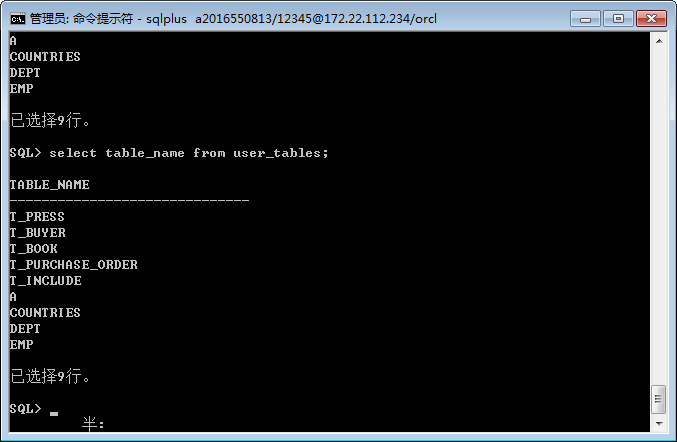
### );

### 各表结构展示：









### 2．图书销售

### E-R模型

### I4%S%AZ(JL6I`2VS5E5J[9N

### ②关系模型

根据E-R模型转换成关系数据模型，如下：

**⑴orderandbooks (link\_id, order\_number, book\_number，number)**

**主键：link\_id**

**外键: order\_number, ISBN**

**⑵Order (order\_number, vip\_id, charge\_id, order\_make, send\_date, order\_price)**

**主键: order\_number**

**外键：vip\_id，charge\_id, charge\_id**

**⑶User (vip\_id, name，address, number, password)**

**主键：vip\_id**

**⑷chargeman(charge\_id, name，address, number, password)**

**主键：charge\_id**

### ③关系模型的说明

User

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| vip\_id | NUMBER(10) | 主键，not null | 会员号 |
| name | VARCHAR(10) | not null | 姓名 |
| address | VARCHAR(20) | not null | 地址 |
| phone | VARCHAR(10) | not null | 电话 |
| password | VARCHAR(20) | not null | 密码 |

Book

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| ISBN | VARCHAR(13) | 主键，not null | 书号 |
| book\_publisher | VARCHAR(30) | not null | 出版社 |
| name | VARCHAR(20) | not null | 书名 |
| author | VARCHAR(20) | not null | 作者 |
| unit\_price | NUMBER(\*,1) | not null | 单价 |
| count | INT | not null | 数量 |
| stock | INT | not null | 库存 |

Chargeman

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| charge\_id | varchar(15) | 主键，not null | 负责人号 |
| name | varchar(13) | not null | 名字 |
| address | varchar(15) | not null | 地址 |
| password | varchar(10) | not null | 密码 |
| Phone | varchar（10） | not null | 电话 |

Order

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| order\_number | varchar(15) | 主键,not null | 订单号 |
| vip\_id | varchar(13) | 外键,not null | 会员号 |
| charge\_id | varchar(15) | 外键,not null | 负责人号 |
| order\_make | DATE | not null | 订单创建时间 |
| send\_date | DATE | not null | 送货时间 |
| order\_price | int | not null | 订单价格 |

orderandbooks

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| link\_id | int | not null,主键 | 关系号 |
| order\_number | int | not null，外键 | 订单号 |
| ISBN | varchar(13) | not null | 书号 |

### ④创建基本表

### 用CREATE TABLE命令在Oracle中实现该模型。

### create table T\_User

### (

### vip\_id NUMBER(10) not null,

### name VARCHAR(10) not null,

### address VARCHAR2(100 CHAR) not null,

### phone VARCHAR(10) not null,

### password VARCHAR(20) not null,

### primary key (vip\_id)

### );

### create table T\_chargeman

### (

### charge\_id VARCHAR(10) not null,

### name VARCHAR(10) not null,

### address VARCHAR2(100 CHAR) not null,

### phone VARCHAR(10) not null,

### password VARCHAR(20) not null,

### primary key (charge\_id)

### );

### create table T\_Order

### (

### order\_number NUMBER(20) not null,

### vip\_id NUMBER(10) not null,

### charge\_id VARCHAR(10) not null,

### order\_make DATE not null,

### send\_date DATE not null,

### order\_price NUMBER(10,2) not null,

### primary key (order\_number),

### foreign key (charge\_id) references T\_chargeman on delete cascade,

### foreign key (vip\_id) references T\_User on delete cascade

### );

### create table T\_orderandbooks

### (

### link\_id int not null,

### order\_number int not null,

### ISBN VARCHAR(13) not null,

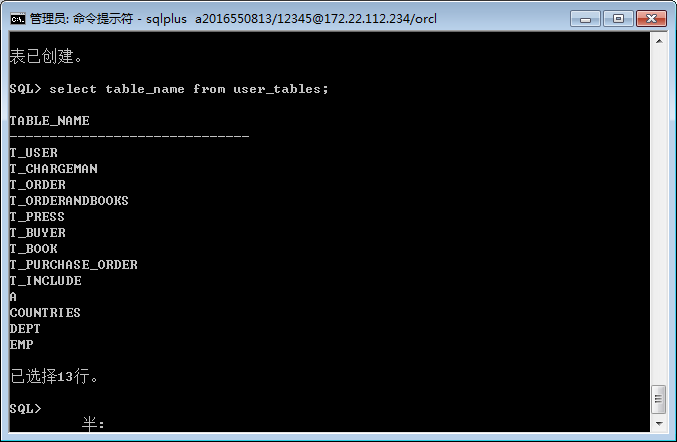
### primary key (link\_id),

### foreign key (ISBN) references T\_book on delete cascade,

### foreign key (order\_number) references T\_Order on delete cascade

### );

### 已建表展示：



### 3．修改大学数据库中的表

### a.增加“专业”表。T\_专业 专业有属性：专业名、所属系、专业必修学分数、专业选修学分数等。

### create table T\_专业(

### 专业名 varchar(15),

### 所属系 varchar(15),

### 专业必修学分数 number(3),

### 专业选修学分数 number(3)

### );

### 

### 增加“班级”表。 T\_班级 班级有属性：班级名、所属专业、学生数等。

### create table T\_班级(

### 班级名 varchar(15),

### 所属专业 varchar(15),

### 学生数 number(3)

### );

### 

### b.在department中增加两个属性，分别表示该系任课教师人数与学生人数

### alter table department add 任课教师人数 number(3);

### alter table department add 学生人数 number(3);

### 

### 删除属性budget。

### alter table department drop column budget;

### c.在student中增加三个属性，第1个表示学生状态取值0(默认)、1、2、3，分别表示在校学习、毕业、结业、退学。

### alter table student add 学生状态 number(1) default 0 check(学生状态 in(0,1,2,3));

### 第2个属性，4位整数，记录学生状态获得的时间：入学时间、毕业时间、退学时间等。

### alter table student add 学生状态获得时间 number(4);

### 第3个属性，表示学生所在班级。

### alter table student add 所在班级 varchar(15);

### 

### 删除属性dept\_name。

### alter table student drop column dept\_name;

### 将tot\_cred分解为两个属性：已修必修学分数、已修选修学分数。

### alter table student drop column tot\_cred;

### alter table student add 已修必修学分数 number(3) ;

### alter table student add 已修选修学分数 number(3) ;

### 

### d.教师有可能由于各种原因不再任课，为此在instructor中增加一个属性，类型为char(1)，取值1或0，分别表示任课或不任课，默认值为1。

### alter table instructor add 任课状态 char(1) default 1 check(任课状态 in('1','0'));

### 

### e.每一个开课在选课前都有人数限制，选课后有选课人数统计，在section中用属性students表示。再增一个属性memo表示该开课的对象。

### alter table section add students number;

### alter table section add memo varchar(15);

### 

### f.课程类型有公共基础课(1,2)、专业必修课(3)、专业选修课(4)与通识课(5)。课程在第几学期开设都有规定。在course中增加两个属性，course\_type：用1-5表示课程类型；term：用1-8与0表示学期，0表示该课程不再开设。

### alter table course add course\_type number(1) check(course\_type in (1,2,3,4,5));

### alter table course add term number(1) check(term in (0,1,2,3,4,5,6,7,8));

### 

### 4．自拟合作项目

### —-住院信息管理系统

### E-R模型

### 

### ②关系模型

根据E-R模型转换成关系数据模型，如下：

**patient(name,id\_number,sex,address,phone,old);**

**主键：id\_number**

**sickbed(bed\_number,bed\_kind,patient\_name)；**

**主键：bed\_number**

**doctor（doctor\_name,doctor\_id,sex,department,job\_title）**

**主键：doctor\_id**

**nurse (nurse\_name,nurse\_id,sex,system\_name)**

**主键：nurse\_id**

**drug(drug\_id,drug\_name,price,kind,area\_name);**

### ③关系模型的说明

patient（病人）

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| name | varchar(10) | not null | 姓名 |
| id\_number | varchar(20) | not null,主键 | 身份证号码 |
| sex | varchar(3) | not null | 性别 |
| address | varchar(30) |  | 家庭住址 |
| phone | varchar(12) | not null | 电话号码 |
| old | int | 大于0，not null | 年龄 |

sickbed（病床）

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| bed\_number | varchar(10) | not null,主键 | 病床号 |
| bed\_kind, | varchar（10） | not null | 病房类型 |
| patient\_name | varchar(5) |  | 入住病人名字 |

doctor（医生）

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| doctor\_name | varchar(10) | not null | 医生姓名 |
| doctor\_id | varchar(10) | not null，主键 | 工作编号 |
| sex | varchar(3) |  | 性别 |
| department | varchar(10) | not null | 所属科室 |
| job\_title | varchar(10) | not null | 职称 |

nurse (护士)

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| nurse\_name | varchar(10) | not null | 护士姓名 |
| nurse\_id | varchar(10) | not null，主键 | 护士编号 |
| sex | varchar(5) | not null | 护士性别 |
| system\_name | varchar(10) | not null | 所属科室名 |

drug

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 类型 | 约束 | 说明 |
| drug\_id | varchar(10) | not null，主键 | 药品编号 |
| drug\_name | varchar(10) | not null | 药品名 |
| price | numeric(10,1) | not null，且大于0 | 价格 |
| kind | varchar(5) | not null | 药品类别 |
| area\_name | varchar(10) | not null | 药品所在区域 |

### ④创建基本表

### 用CREATE TABLE命令在Oracle中实现该模型。

### create table T\_patient(

### name varchar(10) not null,

### id\_number varchar(20) not null,

### sex varchar(3) not null,

### address varchar(30) ,

### phone varchar(12) not null,

### old int check(old>0) ,

### primary key (id\_number)

### );

### create table T\_sickbed(

### bed\_number varchar(10) not null,

### bed\_kind varchar(10) not null ,

### patient\_name varchar(5) ,

### primary key (bed\_number)

### );

### create table T\_doctor(

### doctor\_name varchar(10) not null,

### doctor\_id varchar(10) not null,

### sex varchar(3),

### department varchar(10) not null,

### job\_title varchar(10) not null,

### primary key (doctor\_id)

### );

### create table T\_nurse (

### nurse\_name varchar(10) not null,

### nurse\_id varchar(10) not null,

### sex varchar(5) not null,

### system\_name varchar(10) not null,

### primary key (nurse\_id )

### )；

### create table T\_drug(

### drug\_id varchar(10) not null,

### drug\_name varchar(10) not null,

### price numeric(10,1) not null check(price>0),

### kind varchar(5) not null,

### area\_name varchar(10) not null,

### primary key (drug\_id)

### );

### 

### 三．实验错误解决方案

问题一： SQL语句无法结束

1、问题的出现

在实验项目4中创建“T\_nurse”表，出现语句未结束的情况。

### create table T\_nurse (

### nurse\_name varchar(10) not null,

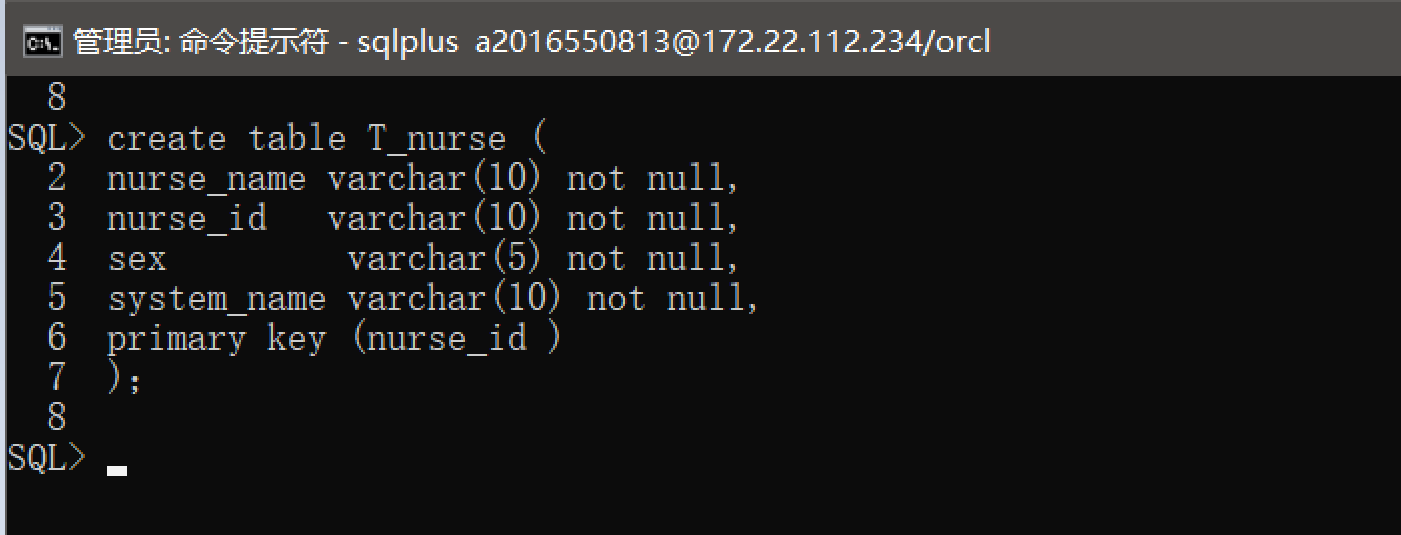
### nurse\_id varchar(10) not null,

### sex varchar(5) not null,

### system\_name varchar(10) not null,

### primary key (nurse\_id )

### )；



2、问题分析

错误出现在第7行，语句结束的符号为英文分号，而本次输入的为中文分号。

3、解决方案

### 解决方案如下：

### 修改符号

### create table T\_nurse (

### nurse\_name varchar(10) not null,

### nurse\_id varchar(10) not null,

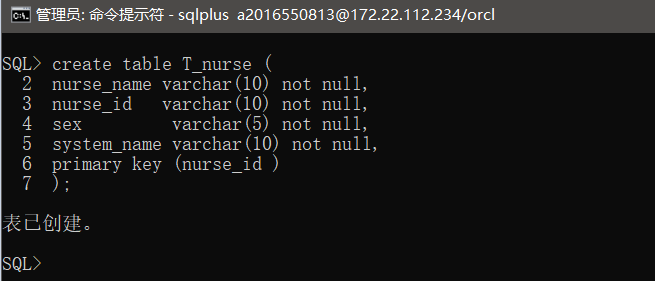
### sex varchar(5) not null,

### system\_name varchar(10) not null,

### primary key (nurse\_id )

### );

执行该语句后，表已创建，问题解决。

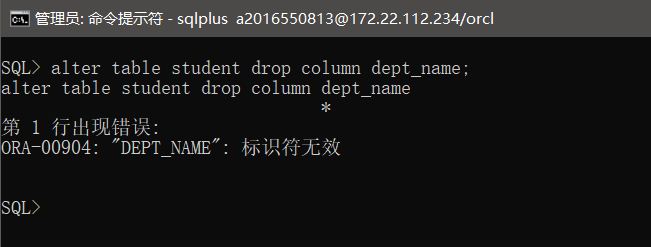


问题二： ORA-00904问题

1. 问题的出现

### 删除属性dept\_name。

### alter table student drop column dept\_name;



1. 问题分析

dept\_name无法找到，desc查看该表，发现不存在该属性，查看记录，发现上一步已经执行完毕，该列已经删除。

1. 问题解决

创建该列，再执行删除操作，即删除成功。