# 实验报告

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 《数据库系统》——实验报告 | | | | | |
| 姓名： | 张家健 | 班级： | 软件工程二班 | 学号： | 202030482478 |
| 题目： | 交互式SQL 1（DDL语言实践） | | | 日期： | 2022/3/27 |
| 实验环境：  Windows 10  8.0.28 MySQL Community Server - GPL | | | | | |
| 实验内容与完成情况：  (1)设有如下关系表S：  S(NO, NAME, SEX, AGE, CLASS)，主关键字是NO。  其中NO为学号，char(2)，学号不能为空，值是唯一的；NAME为姓名，char(10)，姓名的取值也唯一；SEX为性别，char(2)；AGE为年龄，int； CLASS为班号，char(5)。写出实现下列功能的SQL语句。   1. 创建数据库test，在test中创建表S；      1. 插入一个记录（25，“李明”，“男”，21，“95031”）；     再插入一个记录（10，“王丽”，“女”，20，“95101”）；     1. 插入“95031”班学号为30，姓名为“郑和”的学生记录；      1. 对表S，按学号升序建唯一索引（索引名为sno）；   对表S，按年龄降序建索引（索引名为sage）；     1. 向S表添加“入学时间（comedate）”列，其数据类型为日期型（datetime）；      1. 删除S表的sage索引；      1. 将年龄的数据类型改为smallint；      1. 删除学生姓名必须取唯一值的约束；（注意：MySQL与标准SQL语言有区别）      1. 删除S表      1. 创建数据库db\_SPJ。 2. 确保当前操作的数据库为db\_SPJ      1. 在db\_SPJ中创建以下四个关系（表）   **供应商表S(SNO,SNAME,STATUS,CITY)**  **零件表P(PNO,PNAME,COLOR,WEIGHT)**  **工程项目表J(JNO,JNAME,CITY)**  **供应情况表SPJ(S NO,PNO,JNO,QTY)**   1. 插入各表数据   采用 insert into 表名  values  (field1,field2,...) 的命令格式向各表插入数据  结果如下：           1. 导出所有的数据（供下一实验用）      1. 约束检查示例    * + - 1. 创建表S，SNO字段采用PRIMARY KEY和NOT NULL约束，SNAME字段采用UNIQUE约束；   Create table S(SNO char(3) primary key not null,  SNAME char(10) unique,  STATUS int,  CITY char(3));     * + - * 1. 创建表P，WEIGHT字段采用CHECK约束，CHECK条件为WEIGHT>0；   Create table P(PNO char(3) primary key,  PNAME char(10) ,  COLOR char(1),  WEIGHT int,check(WEIGHT>0));     * + - * 1. 创建表J，JNAME字段采用DEFAULT约束，DEFAULT默认值为“工厂”；   Create table J(JNO char(3) primary key,  JNAME char(10) not null default “工厂” ,  CITY char(3));     * + - * 1. 创建表SPJ，SNO、PNO和JNO字段采用主键和外键约束；      * + - * 1. 检查主键唯一约束；         2. 检查主键不能为空约束；         3. 检查UNIQUE约束，表S中的SNAME字段采用了UNIQUE约束，因此SNAME的值必须唯一，当发生重复时报错；         4. 检查DEFAULT约束，表J中的JNAME字段采用了DEFAULT约束，默认值为“工厂”，若插入时采用默认值，JNAME字段会自动填入默认值“工厂”，      * + - * 1. 检查外键级联操作CASCADE，当表P中主键“P2”更新为“P200”时，SPJ表中相应位置的外键“P2”同时更新为“P200”；      * + - * 1. 检查外键级联操作NO ACTION，当被参照表中发生删除或更新时将会报错；   Alter table SPJ add constraint outside\_link foreign key(SNO)  References S(SNO) on update no action on delete no action;  再用下图所示命令检测。    经操作发现在创建外键时不指定级联，则默认为NO ACTION。   * + - * 1. 检查外键级联操作SET DEFAULT，此时外键必须事先设定默认值，表SPJ中将SNO默认值设为“S1”，PNO默认值设为“P1”，JNO默认值设为“J1”，当被参照表S中“S2”所在行被删除后，SPJ表中对应的“S2”更新为默认值“S1”；   1648354440(1)  可如上图对各表进行设置默认值，操作类似。  考核内容：   1. 创建一个临时表test，包含至少4个不同类型的属性，并指定主码。   Create table test(  NO int unique,  NAME varchar(10) primary key,  SEX char(2),  AGE Smallint);   1. 修改test表中一个已有属性为另一类型。   Alter table test modify AGE varchar(2);   1. 修改test表，插入一个新的列“new\_attr”，数据类型为日期型      1. 向test表中插入2条数据   Insert into test  values  (1,"张三","男",19,"2020-09-06"),  (2,"李四","男",20,"2020-09-07");   1. 更新test表中某一行数据   update test set AGE=19 where NAME="李四";   1. 按test表中某一属性升序建立索引   create index sno on test(new\_attr asc);   1. 删除test表任一索引   alter table test drop index sno;   1. 删除test表中所有数据（但保留表）   truncate table test;（删除整张表后，再重新建立相同表结构的表）  或 delete from test;（仅删除表数据，性能比前者慢）   1. 删除test表   drop table test; | | | | | |