实验说明及举例

第一步，以root用户登录mysql系统。例如：

。

第二步，创建一个以自己**学号**为用户名的用户，并赋予该用户对所有数据库的所有操作权限。例如，有位同学的学号为“42004040224”，则使用下面命令创建用户名为41604040224, 密码为：pwd123 的新用户。



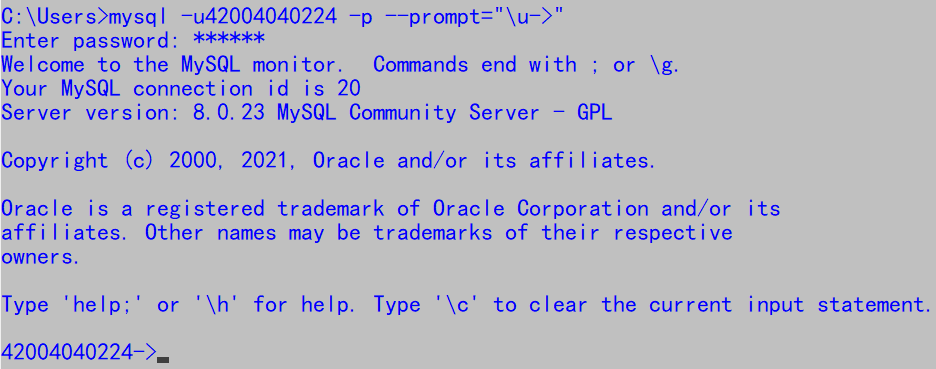
然后为该用户分配使用数据库的权限，如下图：



第三步，用exit命令退出MySQL，用下面的命令登录刚才创建的用户：



您将会看到如图所示的界面：



第四步，开始做实验和作业，并按要求将结果截屏粘贴到“答案粘贴处”。例如：

题目：请用show databases;显示所有数据库。

|  |
| --- |
| 答案粘贴处： |

**注意：如果截图中用户名没有显示学号或者显示的不是自己学号，将被认为是抄袭，不会得到任何分数。截图工具可以使用电脑端qq中的截图功能**。

班级： 工商管理一班 ；姓名： 王安琪 ；学号： 42005010120 ；

实验一、数据库和数据表的创建与管理

目的：

1. 熟练掌握创建数据库、数据表结构定义的方法。
2. 熟练掌握查看数据库、表信息的方法。
3. 熟练掌握修改表结构的方法。

内容：

第一部分 数据库的创建、修改、删除

1. 在MySQL中创建一个名为db\_school的数据库，要求设置默认字符集为GB2312，字符集的校对规则为gb2312\_chinese\_ci。

|  |
| --- |
| 答案粘贴处： |

1. 查看当前用户可查看的数据库列表。

|  |
| --- |
| 答案粘贴处： |

1. 修改已有数据库db\_school的默认字符集和校对规则分别为GBK, GBK\_chinese\_ci。

|  |
| --- |
| 答案粘贴处： |

1. 删除数据库db\_school，并使用show databases查看。

|  |
| --- |
| 答案粘贴处： |

第二部分 数据表的创建、修改、删除

1. 首先按照第1题的要求重新创建db\_school，然后在数据库db\_school中定义学生表tb\_student，其表结构如表1所示，并要求使用InnoDB引擎存储表数据。

表1 学生表tb\_student表结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 含义/列名 | 数据类型 | 允许NULL值 | 约束 |
| 1 | 学号/studentNo | CHAR(10) | 不能为空 | 主键 |
| 2 | 姓名/studentName | VARCHAR(10) | 不能为空 |  |
| 3 | 性别/sex | CHAR(2) |  |  |
| 4 | 出生日期/birthday | DATE |  |  |
| 5 | 籍贯/native | VARCHAR(20) |  |  |
| 6 | 民族/nation | VARCHAR(10) |  | 默认值为‘汉’ |
| 7 | 所属班级/classNo | CHAR(6) |  |  |

|  |
| --- |
| 答案粘贴处： |

1. 用命令show tables查看当前数据库中的所有表。

|  |
| --- |
| 答案粘贴处： |

1. 用命令desc tb\_student或者show columns from tb\_student命令查看tb\_student的表结构。

|  |
| --- |
| 答案粘贴处： |

1. 用命令show create table tb\_student查看tb\_student表的详细表结构。（注意：SHOW CREATE TABLE语句不仅可以查看创建表时的详细语句，而且还可以查看存储引擎和字符编码）。

|  |
| --- |
| 答案粘贴处： |

1. 将tb\_student的表结构复制到tb\_student2，并向tb\_student2中添加一个INT型字段id,要求其不能为NULL，取值唯一且自动增加，并将该字段添加到表的第一个字段。

|  |
| --- |
| 答案粘贴处： |

1. 向tb\_student表中添加一个varchar(16)类型的字段department，用于描述学生所在院系，要求设置其默认值为“城市学院”，并将该字段添加到原表nation之后。

|  |
| --- |
| 答案粘贴处： |

1. 将tb\_student中的字段birthday重命名为age，并将其数据类型更改为TINYINT，允许其为NULL，默认值为18。用DESC 查看tb\_student。

|  |
| --- |
| 答案粘贴处： |

1. 将tb\_student表中的字段department的默认值删除。

|  |
| --- |
| 答案粘贴处： |

1. 将tb\_student表中的字段department的默认值改为’环化学院’。

|  |
| --- |
| 答案粘贴处： |

1. 将tb\_student表中的字段department的数据类型更改为varchar(20)，取值不允许为空，并将此字段移至字段studentName之后。用desc tb\_student命令查看结果。

|  |
| --- |
| 答案粘贴处： |

1. 删除数据表tb\_student2中的字段id。

|  |
| --- |
| 答案粘贴处： |

1. 使用RENAME [TO]子句将数据库db\_school中的数据表tb\_student2重新命名为backup\_tb\_student。

|  |
| --- |
| 答案粘贴处： |

1. 使用RENAME TABLE语句将数据库db\_school中的表backup\_tb\_student再重新命名为tb\_student2。

|  |
| --- |
| 答案粘贴处： |

1. 删除数据库db\_school中的数据表tb\_student,tb\_student2。

|  |
| --- |
| 答案粘贴处： |

第三部分 数据完整性约束实验

1. 重新按照表1创建tb\_student数据表，要求以表级完整性约束方式定义主键，并指定主键约束名为pk\_student。

CREATE TABLE tb\_student (

studentNo CHAR(10) NOT NULL,

studentName VARCHAR(10) NOT NULL,

sex CHAR(2),

birthday DATE,

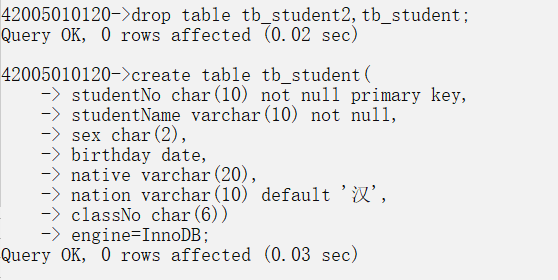
native VARCHAR(20),

nation VARCHAR(20) default ‘汉’,

classNo CHAR(6),

constraint pk\_student primary key(studentNo)

) engine=InnoDB default charset=gb2312;



1. 在数据库db\_school中，按照表2的结构创建tb\_class。要求：使用InnoDB存储引擎，gb2312字符集，主键约束为列级完整性约束，唯一约束为表级完整性约束其约束名为uq\_class。

表2 数据表tb\_class的表结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 含义/列名 | 数据类型 | 允许NULL值 | 约束 |
| 1 | 班级编号/classNo | CHAR(6) | 不能为空 | 主键 |
| 2 | 班级名称/className | VARCHAR(20) | 不能为空 | 唯一 |
| 3 | 所属院系/department | VARCHAR(20) | 不能为空 |  |
| 4 | 年级/grade | ENUM(‘1’,’2’,’3’,’4’) |  |  |
| 5 | 班级最大人数/ classNum | TINYINT |  |  |

CREATE TABLE tb\_class (

classNo CHAR(6) PRIMARY KEY NOT NULL,

className VARCHAR(20) NOT NULL,

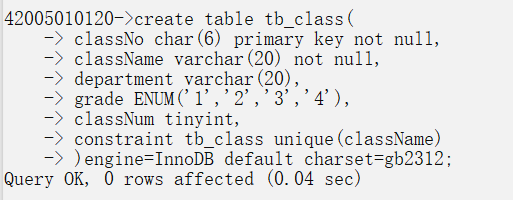
department VARCHAR(20),

grade ENUM(‘1’,’2’,’3’,’4’),

classNum TINYINT,

constraint uq\_class unique(className)

) engine=InnoDB default charset=gb2312;



1. 首先删除数据表tb\_student，按照表1重新创建tb\_student，在创建的同时建立tb\_student到tb\_class的外键约束（两个表相同含义的属性是classNo，因此classNo是tb\_student的外键），约束名为fk\_student，并定义相应的参照动作，更新操作为级联（cascade），删除操作为限制(restrict)，数据表引擎为InnoDB，字符集为gb2312。

|  |
| --- |
| 答案粘贴处： |

1. 在数据库db\_school中按照表3创建tb\_course表，要求：外键名字为fk\_course，引擎为InnoDB，默认字符集为gb2312。

表3 tb\_course表的结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 含义/列名 | 数据类型 | 允许NULL值 | 约束 |
| 1 | 课程号/courseNo | CHAR(6) | 不能为空 | 主键 |
| 2 | 课程名/courseName | VARCHAR(20) | 不能为空 | 唯一 |
| 3 | 学分/credit | DECIMAL(3,1) | 不能为空 |  |
| 4 | 课时数/courseHour | TINYINT(2) | 不能为空 |  |
| 5 | 开课学期/ term | TINYINT(1) |  |  |
| 6 | 先修课程/priorCourse | CHAR(6) |  | 外键，对应主键为tb\_course的courseNo |

|  |
| --- |
| 答案粘贴处： |

1. 在数据库db\_school中定义数据表tb\_score，表结构如表4所示, 引擎为InnoDB，默认字符集为gb2312。

表4. tb\_score的表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 含义/列名 | 数据类型 | 允许NULL值 | 约束 | |
| 1 | 学号/studentNo | CHAR(10) | 不能为空 | 学号和课程号为联合主键，主键名为pk\_score | 外键，对应tb\_student的studentNo,外键名fk\_score\_stuNo |
| 2 | 课程号/courseNo | CHAR(6) | 不能为空 | 外键，对应tb\_course的courseNo,外键名fk\_score\_courNo |
| 3 | 成绩/credit | DECIMAL(4,1) | 不能为空 |  | |

|  |
| --- |
| 答案粘贴处： |

第四部分 更新完整性约束条件

1. 删除在表tb\_score中定义的外键约束fk\_score\_stuNo。

|  |
| --- |
| 答案粘贴处： |

1. 删除在表tb\_student中定义的主键约束。

|  |
| --- |
| 答案粘贴处： |

1. 添加主键约束，用alter table语句在tb\_student对studentNo重新添加主键。

|  |
| --- |
| 答案粘贴处： |

1. 添加外键约束，用alter table语句在tb\_score表对studentNo重新添加外键，对应的主键为tb\_student表的studentNo，外键名称为fk\_score\_stuNo。

|  |
| --- |
| 答案粘贴处： |