SQL学习：

一、对数据库的操作命令

1. 启动数据库 net start mysql
2. 登陆数据库 mysql -uroot -p密码
3. 查看有哪些数据库 show databases
4. 创建数据库 create database demo
5. 删除数据库 drop database demo
6. 进入指定数据库 use demo
7. 查看数据库中有哪些表 show tables;

二、对表的操作命令 SQL语言 (ctrl+回车)

1、创建表 注：首选要选中所使用的数据库

Create table emp(

eid int primary key, --主键，唯一，类型可更改

ename nvarchar(45),

eage int,

address nvarchar(45)

);

2、添加一行数据

a、添加全部字段

insert into emp values(1,"小明",23);

b、添加部分字段

insert into emp (eid,ename) values (2,"小红");

注意：主键必须给值，且不能重复，不能为null

Inser into 表名 (字段名) values (值)

3、修改一条数据

1、只修改一个字段

update emp set ename = '小灰' where eid = 2;

2、修改多个字段

update emp set ename = '小红',age = 22 where eid = 2;

3、如果条件为null值 ，则使用is null判断

注：主键也可以修改但不能重复

Update 表名 set 字段名 = 值 where 条件 and(or) 条件 and(or) 条件

4、删除数据

1、删除一条数据

delete from emp where ename = '小红' ;

2、删除全部数据

Delete from emp;

Delete form 表名 where 条件 and(or) 条件 and(or) 条件

注： 删除数据库 删除表

Drop database 数据库名 Drop table 表名

查询语句

以案例说明：

新建两张表

1、部门表

create table dept(

deptno int primary key,

dname nvarchar(45),

address nvarchar(45)

);

2、emp 员工表

create table emp(

empno int primary key,

ename nvarchar(45),

job nvarchar(45),

mgr int,

hiredate datetime,

sal numeric(8,2),

comm numeric(8,2),

deptno int

);

这两张表就有主外键的关系，创建外键约束语句：

alter table emp add constraint emp\_dept foreign key (deptno) references dept(deptno);

注意：a、外键只能指向主键

b、外键和主键类型必须一致

数据：

部门表数据

insert into dept values (10,'accounting','new york');

insert into dept values (20,'research','dallas');

insert into dept values (30,'sales','chicago');

insert into dept values (40,'operations','boston');

职员表数据

insert into emp (empno,ename,job,mgr,hiredate,sal,deptno) values (7369,'smith','clerk',7902,'1980-12-17',800.00,20);

insert into emp values(7499,'allen','salesman',7698,'1981-2-20',1600.00,300.00,30);

insert into emp values(7521,'ward','salesman',7698,'1981-2-22',1250.00,500.00,30);

insert into emp (empno,ename,job,mgr,hiredate,sal,deptno) values (7566,'jones','manager',7839,'1981-4-2',2975.00,20);

insert into emp values(7654,'martin','salesman',7698,'1981-9-28',1250.00,1400.00,30);

insert into emp (empno,ename,job,mgr,hiredate,sal,deptno) values (7698,'blake','manager',7839,'1981-5-1',2850.00,30);

insert into emp (empno,ename,job,mgr,hiredate,sal,deptno) values (7782,'clark','manager',7839,'1981-6-9',2450.00,10);

insert into emp (empno,ename,job,mgr,hiredate,sal,deptno) values (7788,'scott','analyst',7566,'1987-4-19',3000.00,20);

insert into emp (empno,ename,job,hiredate,sal,deptno) values (7839,'king','president','1981-11-17',5000.00,10);

insert into emp values (7844,'turner','salesman',7698,'1981-9-8',1500.00,0.00,30);

insert into emp (empno,ename,job,mgr,hiredate,sal,deptno) values (7876,'adams','clerk',7788,'1987-5-23',1100.00,20);

insert into emp (empno,ename,job,mgr,hiredate,sal,deptno) values (7900,'james','clerk',7698,'1981-12-3',950.00,30);

insert into emp (empno,ename,job,mgr,hiredate,sal,deptno) values (7902,'ford','analyst',7566,'1981-12-3',3000.00,20);

insert into emp (empno,ename,job,mgr,hiredate,sal,deptno) values (7934,'miller','clerk',7782,'1982-1-23',1300.00,10);

简单查询

1、查询所有列

Select \* from emp(表名) where 条件 （\*能不用则不用，它查询所有列）

2、查询指定列

Select 字段1，字段2 from 表名 where 条件

Eg:查询smith的薪水、工作，所在部门

select job ,sal,deptno from emp where ename = 'smith';

3、取消重复的行 distinct(消除完全一样的行，保留一行)

Eg:统计共有多少个部门编号

Select distinct depton from emp;

查询多个字段时，只有当多个字段都一样时才会被取消重复行

4、算术表达式的使用

Eg:显示每个员工的年工资

select ename,sal\*12 '年工资' from emp; 指定查询后结果的列名

如果把年工资改为sal\*12+comm\*12 工资加奖金，查看查询结果

如何处理：

select ename,sal\*12+ifnull(comm,0) '年工资' from emp;

eg:显示工资大于2000的员工

显示工资在2000到3000之间的员工

between 2000 and 3000（>=2000 <=3000）

显示入职时间在1981-05-01之后的员工

常用sql函数

AVG(col)返回指定列的平均值  
COUNT(col)返回指定列中非NULL值的个数  
MIN(col)返回指定列的最小值  
MAX(col)返回指定列的最大值  
SUM(col)返回指定列的所有值之和

IFNULL(arg1,arg2) 如果arg1不是空，返回arg1，否则返回arg2

5、模糊查询（关键字like）

%:表示0到多个字符 \_：表示单个字符

eg:如何显示首字母为S的员工姓名和工资

select ename,sal from emp where ename like 's%';

查询第三个字母为o的员工姓名和工资

select ename,sal from emp where ename like '\_\_o%';

6、在where条件中使用in

Eg：如何显示员工编号为：7499，7782，7839的员工薪水

select ename ,sal from emp where empno in(7499,7782,7839);

Is null使用：

Eg:显示没有上级的员工姓名

select ename from emp where mgr is null;

7、使用逻辑操作符

Eg:查询工资高于800或岗位为manager的雇员，同时还要满足他们的姓名首写字母为j

select ename,sal,job from emp where (sal>500 or job ='manager') and ename like 'j%';

注：括号表示一个条件，不能去掉

8、使用order by 进行排序 asc 升序排列(默认) desc降序排列

Eg:如何按照工资从低到高的的顺序显示员工的信息

select ename ,sal from emp order by sal asc;

eg:按照部门升序而员工工资降序排列

select ename ,sal,deptno from emp order by deptno desc,sal asc ;

使用别名进行排序(别名不要为中文)

eg:统计每个员工的年薪，并按照年薪排序

select ename ,(sal+ ifnull(comm,0))\*12 nx from emp order by nx;

9、where条件子查询

* 1. 如何显示所有员工中的最高工资和最低工资

Eg:最低工资：select min(sal) from emp

最低工资和员工姓名：

select min(sal),ename from emp where sal = (select min(sal) from emp);

最高工资和员工姓名：

select min(sal),ename from emp where sal = (select min(sal) from emp);

注：sql语句执行顺序从右向左

10、group by 和 having 子句

Group by用于对查询的结果进行分组统计，having用于限制分组显示的结果

Eg:如何显示每个部门的平均工资和最高工资

select max(sal),avg(sal) from emp group by deptno;

eg:显示平均工资低于2000的部门编号和它的平均工资

select avg(sal) a , deptno from emp group by deptno having a<2000;

表的复杂查询

1、多表查询是指基于两个以上的表的查询，在实际应用中，单表查询不能满足我们的需求

Eg：显示sales部门的位置及员工的信息

笛卡尔集：select \* from emp, dept;

通过条件使之关联起来

select \* from emp e, dept d where e.deptno = d.deptno and d.dname = ‘sales’;

注：别名的使用,

加深练习

eg:如何显示部门号为10的部门名，员工名和工资

select d.dname,e.ename,e.sal from emp e,dept d

where d.deptno = e.deptno and d.deptno = 10;

eg:显示员工名，工资，部门名称，并按照部门号降序排序

select d.deptno, d.dname,e.ename,e.sal from emp e,dept d

where d.deptno = e.deptno order by d.deptno desc;

1. 自连接/内连接：在同一张表上的连接查询

Eg:显示某个员工的上级姓名，比如显示ford的上级姓名

select ename from emp where empno = (select mgr from emp where ename='ford');

eg:显示每个员工的姓名和他的上级姓名

select e.ename '员工',b.ename '上级' from emp e,emp b where e.mgr = b.empno;

1. 子查询

子查询是指嵌入在其它查询语句中的查询，也叫嵌套查询

单行子查询

单行子查询是指只返回一行数据的子查询语句

Eg:显示与smith同一部门的所有员工

select ename from emp where deptno =

(select deptno from emp where ename = 'smith');

多行子查询

多行子查询是指返回多行数据的子查询

Eg:查询和部门10相同的工作岗位的员工姓名，工资，部门编号

select job, ename,sal,deptno from emp where job in

(select distinct job from emp where deptno = 10);

在from中使用子查询

Eg:查询高于部门平均工资的员工姓名、薪水

select ename,sal,t.v from emp e, (select avg(sal) v ,deptno from emp group by deptno) t where e.sal > t.v and e.deptno = t.deptno;

注：当在from子句使用子查询时，该子查询会被当作一个临时表来对待，当在from子句中使用子查询时，必须给子查询指定别名

4、分页查询

按入职时间顺序显示第1到5个入职的员工

select \* from emp order by emp.hiredate limit 0,5;

关键字: limit start , total;

注：第一个数字start代表从哪一个开始取值，不包括当前值，如从第5条记录开始取值，则start应设置为 5-1=4 第二个数字total代表取多条记录

1. 左外连接、右外连接

左外连接：指 join关键词左边的表的记录全部都要显示，如果没有匹配的记录，则显示null

右外连接：指join关键词右边的表的记录全部都要显示，如果没有匹配的记录，则显示null

Eg:显示所有员工的上级名称，没有上级的显示null1

1、显示所有员工的上级名称：

select worker.ename click,boss.ename boss from emp worker ,emp boss where worker.mgr = boss.empno;

2、没有上级的显示null

select worker.ename click,boss.ename boss from emp worker left join emp boss on worker.mgr = boss.empno;

补充：

约束：

auto\_increment 自增

Not null 非空

Unique 唯一约束，不能重复，可以放null值

primary key 主键，非空，唯一 /表可以有复合主键，即可以有多列联合作一个主键

foreign key 外键/不一定指向另一表的主键，但必须是唯一索引

check (age>7) 检测/mysql暂不支持该约束

unsigned、 无符号(不区分正负)

default 默认值

eg:

create table stu(

id int primary key auto\_increment,

stuno varchar(45) unique,

age int unsigned default 0,

sname varchar(45) default 'aa',

cid int not null,

constraint stu\_fk foreign key (cid) references class(cid)

);

练习：

商店售货系统表设计案例：

现有一个商店的数据库，记录客户及其购物情况，由下面三个表组成：

商品表goods(商品号goodsid ,商品名 goodsName,单价price,类别 goodsType)

客户表customer(客户号 cid, 姓名 cname,住址：address，邮箱:email,姓别:sex)

销售表sales(客户号 cid ,商品号 goodsid,购买数量 nums)

用sql语句建表要求：

1. 每个表的主外键
2. 客户姓名不能为null
3. 邮箱不能重复
4. 客户姓别默认是男

表的修改，修改表的字段

添加一个字段

alter table class add column cno int default 1;

修改字段的类型

alter table class modify column cno varchar(20);

修改字段名称（可同时修改字段类型）：

alter table class change cno classno int(10) not null;

删除一个字段

alter table class drop column cno;

修改表的名字

alter table class rename to clas;

删除表：(存在主外键时要先删除从表再删除主表)

drop table class;

myslq数据库的备份和恢复：

1、备份指定表

mysqldump -u username -p dbname table1 table2 ... >d:\BackupName.sql

其中：

a、dbname参数表示数据库的名称；

b、table1和table2参数表示需要备份的表的名称，为空则整个数据库备份；

c、BackupName.sql参数表设计备份文件的名称，文件名前面可以加上一个绝对路径。通常将数据库备份成一个后缀名为sql的文件；

2、备份多个数据库

Mysqldump -u username -p --databases dbname2 dbname2 > d:\backup.sql

3、备份所有数据库

mysqldump -u root -p --all-databases > d:\all\_backup.sql

还原数据库

mysql -u root -p 【dbname】 < backup.sq