## 一、为什么要学习数据库

1.持久化数据

2.方便检索

3.存储大量数据

4.共享、安全

5.通过组合分析，获取新的数据

## 二、数据库的相关概念

​ \*\*DBMS、DB、SQL\*\*

DB：database数据库，存储一系列有组织数据的容器

DBMS：Database Management System数据库管理系统，使用DBMS管理和维护DB

SQL：StructureQueryLanguage结构化查询语言，程序员用于和DBMS通信的语言

## 三、数据库存储数据的特点

1、数据先放在表中，表再放在库中

2、一个库可以有多张表，每张表都有自己的唯一标识名

3、一张表的设计，类似于java中”类"的设计

​ 表中的字段的设计，类似于属性的设计

​ 表中的单条记录，类似于一个对象

​ 表中的所有的记录，类似于对象的集合

​ orm ：object relation mapping 对象关系映射

## 四、初始MySQL

### 1、MySQL产品的介绍

MySQL 前身属于瑞典的一家公司AB，2008年被sun公司收购，2009年sun被oracle公司收购。

特点：

① 体积小、安装较方便

②开源、免费

③性能高、稳定性好

④兼容性好

### 2、MySQL产品的安装 ★

基于C/S架构的DBMS，需要安装服务端和客户端

www.oracle.com

MySQL5.5

### 3、MySQL服务的启动和停止 ★

方式一：图形化

右击——计算机管理——服务——MySQL服务

方式二：通过管理员身份运行dos

net start 服务名 net stop 服务名

### 4、MySQL服务的登录和退出 ★

\*\*方式一：通过dos命令\*\*

mysql -h主机名 -P端口号 -u用户名 -p密码

\*\*注意：\*\*

如果是本机，则-h主机名 可以省略

如果端口号是3306，则-P端口号可以省略

\*\*方式二：通过图形化界面客户端\*\*

通过sqlyog，直接输入用户名、密码等连接进去即可

### 5、MySQL的常见命令和语法规范

\*\*①常见命令\*\*

show databases 显示当前连接下所有数据库

show tables 显示当前库中所有表

show tables from 库名 显示指定库中所有表

show columns from 表名 显示指定表中所有列

use 库名 打开/使用指定库

\*\*②语法规范\*\*

不区分大小写

每条命令结尾建议用分号

注释：

#单行注释

-- 单行注释

/\*多行注释\*/

## 五、DQL语言的学习 ★

DQL: Data Query Language 数据查询语言

### 1、基础查询 ★

语法：

select 查询列表 from 表名;

特点：

①查询结果集是一个虚拟表

②查询列表可以是单个字段、多个字段、常量、表达式、函数，可以是以上的组合

引申1：起别名

select 字段名 as "别名" from 表名;

select 字段名 " 别名" from 表名;

引申2：+的作用

作用：加法运算

- 如果两个操作数都是数值型，则直接做加法运算

- 如果其中一个为非数值型，则将强值转换成数值型，如果转换失败，则当做0

'123'+4====》127

‘abc’+4====》 4

- 如果其中一个为null，则结果直接为null

引申3：去重

select distinct department\_id from employees;

引申4：补充函数

select version();

select database();

select user();

select ifnull(字段名,表达式);

select concat(字符1，字符2,字符3);

select length(字符/字段);获取字节长度

###

### 2、条件查询 ★

\*\*语法：\*\*

select 查询列表

from 表名

where 筛选条件;

\*\*特点：\*\*

筛选条件的分类：

①按条件表达式筛选

关系运算符：>= < <= > <> =

②按逻辑表达式筛选

逻辑运算符：and or not

③模糊查询

like:一般和通配符搭配使用

​ \_ 任意单个字符

​ % 任意多个字符

between and：一般用于判断某字段是否在指定的区间

​ a between 10 and 100

in：一般用于判断某字段是否在指定的列表

​ a in(10,30,50)

is null:判断null值

is not null 判断非空值

​

### 3、排序查询 ★

\*\*语法：\*\*

select 查询列表

from 表名

where 筛选条件

order by 排序列表 ;

\*\*特点：\*\*

①排序列表可以是 单个字段、多个字段、函数、表达式、别名、列的索引，以及以上的组合

②升序，通过asc 降序，通过desc

### 4、常见函数 ★

说明：sql中的函数分为单行函数和分组函数

调用语法：select 函数名(实参列表);

1》字符函数

concat(str1,str2,..):拼接字符

substr(str,pos):截取从pos开始的所有字符， 起始索引从1开始

substr(str,pos,len):截取len个从pos开始的字符，起始索引从1开始

length(str):获取字节个数

char\_length(str):获取字符个数

upper(str):变大写

lower(str):变小写

trim(【substr from】str):去前后指定字符，默认是去空格

left(str,len):从左边截取指定len个数的 字符

right(str,len):从右边截取指定len个数的 字符

lpad(str,substr,len):左填充

rpad(str,substr,len)：右填充

strcmp(str1,str2):比较两个字符的大小

instr(str,substr):获取substr在str中第一次出现的索引

2》数学函数

ceil(x):向上取整

floor(x):向下取整

round(x,d):四舍五入

mod(x,y):取模/取余

truncate(x,d):截断，保留小数点后d位

abs(x):求绝对值

3》日期函数

now():获取当前日期——时间

curtime():只有时间

curdate():只有日期

date\_format(date,格式):格式日期为字符

str\_to\_date(str,格式):将字符转换成日期

datediff(date1,date2):获取两个日期之间的天数差

year(date)

month(date) 。。。。。。

\*\*4》流程控制函数\*\*

①if(条件，表达式1，表达式2)：如果条件成立，返回表达式1，否则返回表达式2

②case 表达式

when 值1 then 结果1

when 值2 then 结果2

...

else 结果n

end

\*\*③\*\*

\*\*case\*\*

when 条件1 then 结果1

when 条件2 then 结果2

...

else 结果n

end

### 5、分组函数 ★

sum求和

avg平均

max最大

min最小

count个数

特点：

1》实参的字段的类型，sum和avg只支持数值型，其他三个可以支持任意类型

2》这五个函数都忽略null值

3》count可以支持以下参数

count(字段)：查询该字段非空值 的个数

count(\*) ：查询结果集的行数

count(1)：查询结果集的行数

4》分组函数可以和distinct搭配使用，实现去重的统计

select count(distinct 字段) from 表;

###

### 6、分组查询 ★

\*\*语法：\*\*

select 分组函数,分组的字段

from 表名

where 分组前的筛选条件

group by 分组列表

having 分组后的筛选条件

order by 排序列表;

\*\*特点：\*\*

1》分组列表可以是单个字段、多个字段

2》筛选条件分为两类

\*\*筛选的基表 使用的关键字 位置\*\*

分组前筛选 原始表 where group by前面

分组后筛选 分组后的结果集 having group by后面

### 7、连接查询 ★

说明：当查询中涉及到了多个字段，则需要通过多表连接

笛卡尔乘积：

​ 出现原因：没有有效的连接条件

​ 解决办法：添加有效的连接条件

--------------------------SQL92语法------------------------

语法：

\*\*select 查询列表\*\* ①

\*\*from 表1 别名,表2 别名,... \*\*②

\*\*where 连接条件\*\* ③

\*\*and 筛选条件\*\*④

group by 分组列表⑤

having 分组后筛选⑥

order by 排序列表;⑦

\*\*执行顺序：\*\* ②③④⑤⑥①⑦

------------------

分类：等值连接 非等值连接 内连接

--------------------------SQL99语法------------------------

语法：

select 查询列表 ①

from 表1 别名②

【连接类型】【inner】 join 表2 别名 on 连接条件 ③

【inner】 join 表3 别名 on 连接条件

where 筛选条件 ④

group by 分组列表⑤

having 分组后的筛选⑥

order by 排序列表⑦

\*\*执行顺序：\*\* ②③④⑤⑥①⑦

---------------

分类： 内连接（inner）

外连接：左外（left outer）右外（right outer）全外（full outer）

交叉连接（cross）

### 8、子查询 √

当一个查询语句中又嵌套了另一个完整的select语句，则被嵌套的select语句称为子查询或内查询

外面的select语句称为主查询或外查询。

分类： .按子查询出现的位置进行分类：

1、select后面

要求：子查询的结果为单行单列（标量子查询）

2、from后面

要求：子查询的结果可以为多行多列

3、where或having后面 ★

要求：子查询的结果必须为单列

单行子查询

多行子查询

4、exists后面

要求：子查询结果必须为单列（相关子查询）

特点：

1、子查询放在条件中，要求必须放在条件的右侧

2、子查询一般放在小括号中

3、子查询的执行优先于主查询

4、单行子查询对应了 单行操作符：> < >= <= = <>

多行子查询对应了 多行操作符：any/ some all in

----------------------------------------------------

### 9、分页查询 ★

应用场景：当页面上的数据，一页显示不全，则需要分页显示

分页查询的sql命令请求数据库服务器-服务器响应查询到的多条数据-前台页面

语法：

select 查询列表--------------------------7

from 表1 别名-----------------------①

join 表2 别名---------------------------2

on 连接条件---------------------------------3

where 筛选条件----------------------------4

group by 分组-------------------------------5

having 分组后筛选---------------------------6

order by 排序列表-----------------------------8

limit 起始条目索引,显示的条目数----------------9

特点：

①起始条目索引如果不写，默认是0

②limit后面支持两个参数

参数1：显示的起始条目索引

参数2：条目数

公式：

假如要显示的页数是page，每页显示的条目数为size

select \*

from employees

limit (page-1)\*size,size;

---------------------------------------------

### 10、union联合查询 √

说明：当查询结果来自于多张表，但多张表之间没有关联，这个时候往往使用联合查询，也称为union查询

语法：

select 查询列表 from 表1 where 筛选条件

union

select 查询列表 from 表2 where 筛选条件

特点：

1、多条待联合的查询语句的查询列数必须一致，查询类型、字段意义最好一致

2、union实现去重查询

union all 实现全部查询，包含重复项

**-------------------------------------------------------**

## 六、DML语言的学习 ★

DML(Data Manipulation Language)数据操纵语言：insert update delete

对表中的数据的增删改

### 1、插入语句

语法：

插入单行：

insert into 表名(字段名1,字段名2 ,...) values (值1，值2,...);

插入多行：

insert into 表名(字段名1,字段名2 ,...) values

(值1，值2,...),(值1，值2,...),(值1，值2,...);

特点：

①字段和值列表一一对应，包含类型、约束等必须匹配

②数值型的值，不用单引号，非数值型的值，必须使用单引号

③字段顺序无要求

### 2、修改语句

语法：

update 表名 set 字段名 = 新值,字段名=新值,...

where 筛选条件;

### 3、删除语句

方式1：delete语句

语法：delete from 表名 where 筛选条件;

方式2：truncate语句

语法：truncate table 表名;

【面试题】delete和truncate的区别

1.delete可以添加where条件

truncate不能添加where条件，一次性清除所有数据

2.truncate的效率较高

3.如果删除带自增长列的表，

使用delete删除后，重新插入数据，记录从断点处开始

使用truncate删除后，重新插入数据，记录从1开始

4.delete 删除数据，会返回受影响的行数

truncate删除数据，不返回受影响的行数

5.delete删除数据，可以支持事务回滚

truncate删除数据，不支持事务回滚

**-----------------------------------------------------**

## 七、DDL语言的学习

DDL语言:Data Define Language数据定义语言,用于对数据库和表的管理和操作

### 1、库和表的管理 √

-----------库的管理-------

一、创建数据库

create database【if not exists】 库名；

二、删除数据库

Drop database【if exists】库名；

---------表的管理-------

一、创建表 ★

语法：

Create table【if not exists】表名 (

字段名 字段类型 【（长度）字段约束】,

字段名 字段类型 【（长度）字段约束】,

………

);

案例：添加约束

CREATE TABLE IF NOT EXISTS stuinfo(

stuid INT PRIMARY KEY,#添加了主键约束

stuname VARCHAR(20) UNIQUE NOT NULL,#添加了唯一约束+非空

stugender CHAR(1) DEFAULT '男',#添加了默认约束

email VARCHAR(20) NOT NULL,

age INT CHECK(age BETWEEN 0 AND 100),#添加了检查约束，mysql不支持

majorid int,

CONSTRAINT fk\_stuinfo\_major FOREIGN KEY (majorid) REFERENCES major(id)#添加了外键约束（表约束）

）；

二、修改表[了解]

语法：ALTER TABLE 表名 ADD|MODIFY|CHANGE|DROP COLUMN 字段名 字段类型 【字段约束】;

1.修改表名

alter table 表名rename to 新名；

2.添加字段

alter table 表名 add column 字段名 TIMESTAMP NOT NULL;

DESC students;

3.修改字段名

alter table 表名 change column 字段名 新字段名；

4.修改字段类型或约束

alter table 表名 modify column 字段名 类型或约束;

5.删除字段

alter table 表名 drop column 字段名

DESC students;

三、删除表 √

Drop table 【if exists】表名；

四、复制表√

仅仅复制表的结构

CREATE TABLE 新表名 LIKE 旧表名;

复制表的结构+数据

CREATE TABLE 新表名 SELECT \* FROM 表名 where 条件;

案例：复制某几个字段到新表中，但不复制数据：

CREATE TABLE 新表名

SELECT 字段名

FROM 表名

WHERE 0;

### 2、常见数据类型介绍 √

1、整型

Tinynt smallint mediumint int/integer bigint

1 2 3 4 8

特点：

都可设置无符号和有符号，默认有符号，通过unsigned设置无符号

若超出范围，则报out or range异常，插入临界值

长度可不指定，默认有一个长度。长度代表显示的最大宽度，若不够，左边用0填充，搭配zerofull，默认变为无符号整型。

2、小数

浮点型float（M，D） double （M，D）

定点型 dec（M，D） decimal（M，D）

特点：1）。M:整数部位+小数部位 D:小数部位

2）。M　D都可省略。Decimal默认M为１０，D为０。

Float和double会根据插入的数据的精度来决定。

３）。定点数的精度较高，如果要求插入数值的精度较高则考虑使用，如货币运算等。

3、字符型

较短文本：char（M） varchar（Ｍ）

　　　 M 特点　　　　　空间的耗费 效率

char（Ｍ）　可省略，默认为１ 固定长度的字符　 比较耗费 高

varchar（Ｍ）　不可省略 　　可变长度的字符　　比较节省 　　低

其他：binary和varbinary用于保存较短的二进制

Enum 用于保存枚举

Set 用于保存集合。

较长文本：

Text blob 存储图片数据

n表示最多字符个数

4、日期型



### 3、常见约束 √

说明：用于限制表中字段的数据的，从而进一步保证数据表的数据是一致的、准确的、可靠的！

介绍：

Not null 非空：用于限制该字段为必填项

default 默认：用于限制该字段没有显式插入值，则直接显式默认值

primary key 主键：用于限制该字段值不能重复，设置为主键列的字段默认不能为空。一个表只能有一个主键，当然可以是组合主键。

unique 唯一：用于限制该字段值不能重复

字段是否可以为空 一个表可以有几个

主 键 × 1个

唯 一 √ n个

check检查：用于限制该字段值必须满足指定条件。

foreign key 外键:用于限制两个表的关系,要求外键列的值必须来自于主表的关联列

要求：

①主表的关联列和从表的关联列的类型必须一致，意思一样，名称无要求

②主表的关联列要求必须是主键

列级约束：六大约束语法上都支持，但外键约束没有效果。

表级约束：除非空、默认，其他都支持。

【constraint 约束名】约束类型（字段名）

外键：

1、要求在从表设置外键关系

2.从表的外键列的类型和主表的关联列类型要求一致或兼容，名称无要求

3.主表的关联列必须是一个key，一般是主键或唯一。

4.插入数据时，先插入主表，在插从表；删除时，先删从表，再删主表。

添加约束：创建表时添加约束：

通用写法：

Create table if not exists stuinfo(

id INT PRIMARY KEY,

stuname VARCHAR(20),

sex CHAR(1)

age INT DEFAULT 18,

seat INT UNIQUE,

majored INT,

CONSTRAINT fk\_stuinfo\_major FOREIGN KEY(majored) REFERENCE major(id)

);

修改表时添加约束：

1、添加列级约束：alter table 表名 modify column 字段名 字段类型 新约束；

2、添加表级约束：

alter table 表名 add【constraint 约束名】约束类型（字段名）【外键的引用】；

删除约束：

自增长列（标识列）：

1、自增长列要求必须设置在一个键上，比如主键或唯一键

2、自增长列要求数据类型为数值型

3、一个表至多有一个自增长列

4、可设置步长：set auto\_increment=3；

**-----------------------------------------------------**

## 八、TCL语言的学习

###事务：

查询存储引擎：show engines

常用存储引擎：innodb(支持事务), myisam , memory。

概念：由一条或多条sql语句组成，要么都成功，要么都失败

特性：ACID属性

原子性：事务时一个不可分割的工作单位

一致性 隔离性 持久性

###事务处理

分类：

隐式事务：没有明显的开启和结束标记

比如dml语句的insert、update、delete语句本身就是一条事务

insert into stuinfo values(1,'john','男','ert@dd.com',12);

显式事务：具有明显的开启和结束标记

前提：必须先设置自动提交功能为禁用【set autocommit=0;】

一般由多条sql语句组成，必须具有明显的开启和结束标记

SHOW VARIABLES LIKE '%auto%'

演示事务的使用步骤

1、取消事务自动开启

SET autocommit = 0;

2、开启事务

START TRANSACTION;

3、编写事务的sql语句

#将张三丰的钱-5000

UPDATE stuinfo SET balance=balance-5000 WHERE stuid = 1;

#将灭绝的钱+5000

UPDATE stuinfo SET balance=balance+5000 WHERE stuid = 2;

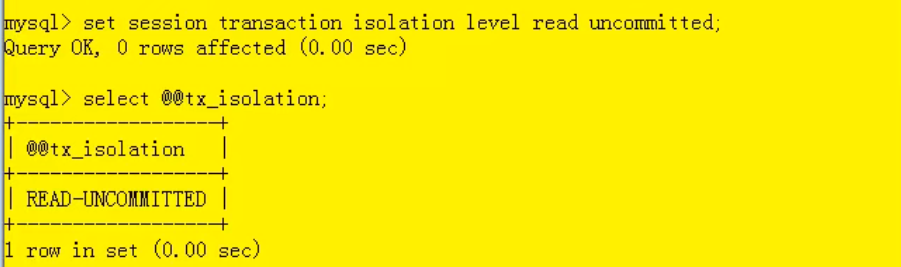
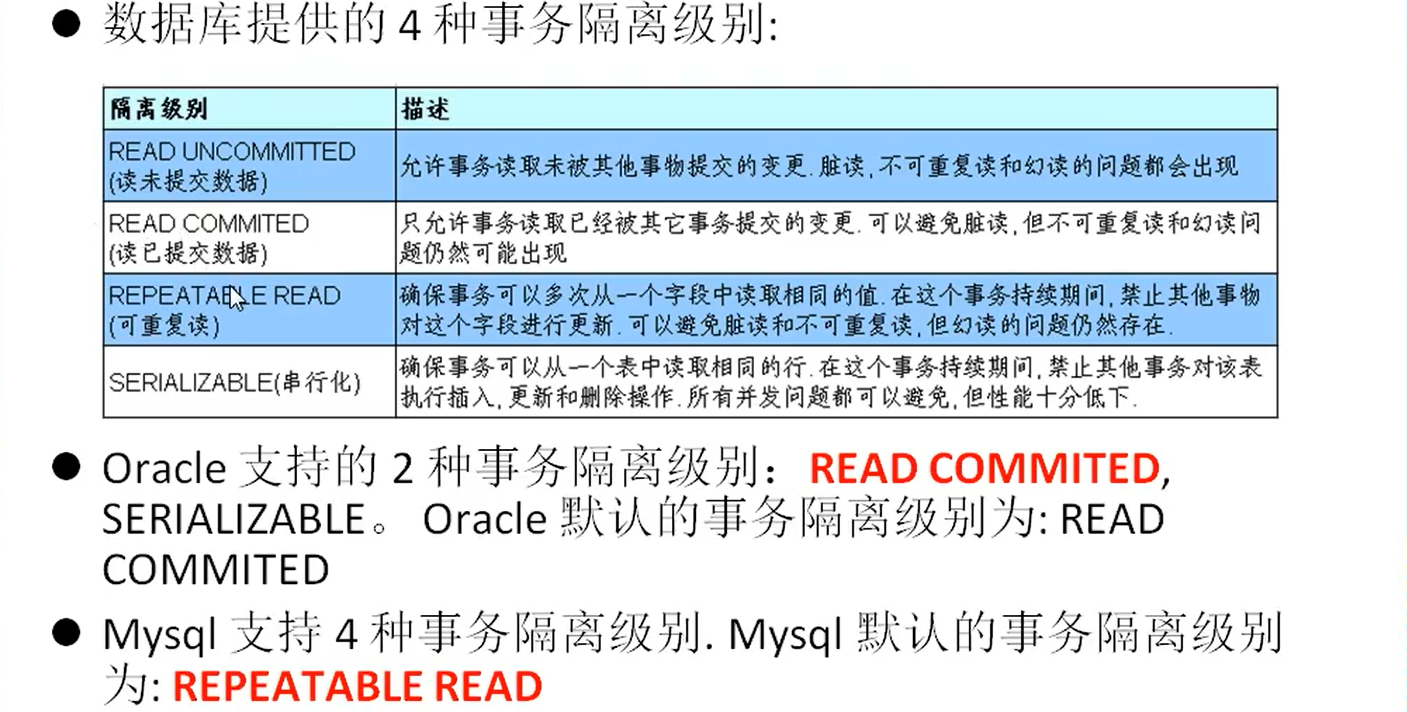
#4、结束事务

#提交

#commit;

#回滚rollback;

SELECT \* FROM stuinfo;



事务的隔离级别：

脏读 不可重复读 幻读

Read uncommitted √ √ √

Read committed × √ √

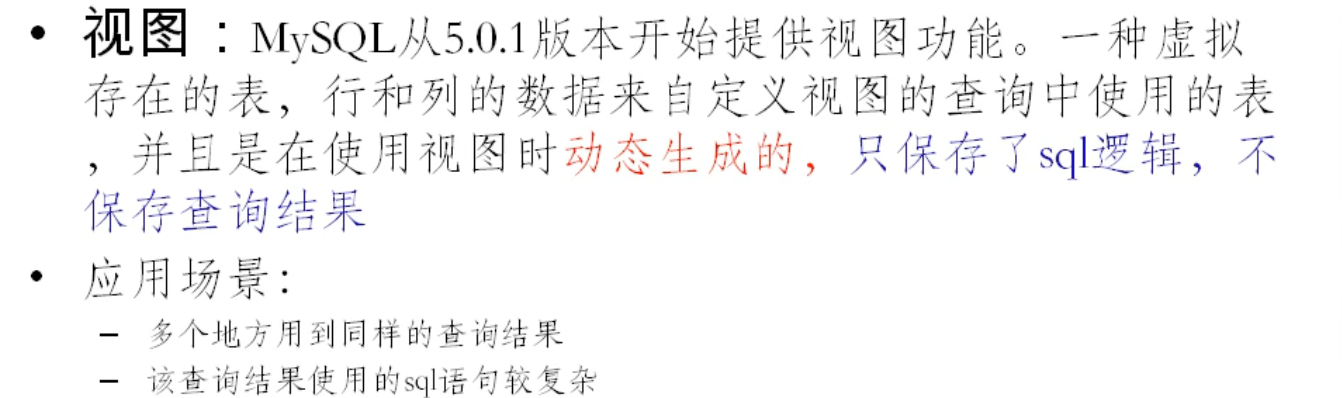
Repeatable read × × √

Serializable × × ×

事务保存点：savepoint;

Savepoint a; rollback to a;

###视图：



语法：创建：create view 视图名 as 查询语句；

使用：select 查询字段 from 视图名 where 条件；

修改：create or replace 视图名 as 查询语句；

Alter view 视图名 as 查询语句；

删除：drop view 视图名1，视图名2，……；

查看：desc 视图名； show create view 视图名【\G（格式化）】；

更新：插入（insert）、删除(delete)、修改(update) (原图数据也变化)

具备以下特点的视图不允许更新：

1、包含以下关键字：分组函数、distinct、group、by、having、union、union all

2、常量视图

3、select 中包含子查询

4、join

5、from一个不能更新的视图

6、where子句的子查询引用了from子句中的表

视图和表的比较：

创建关键字 是否实际占用物理空间 使用

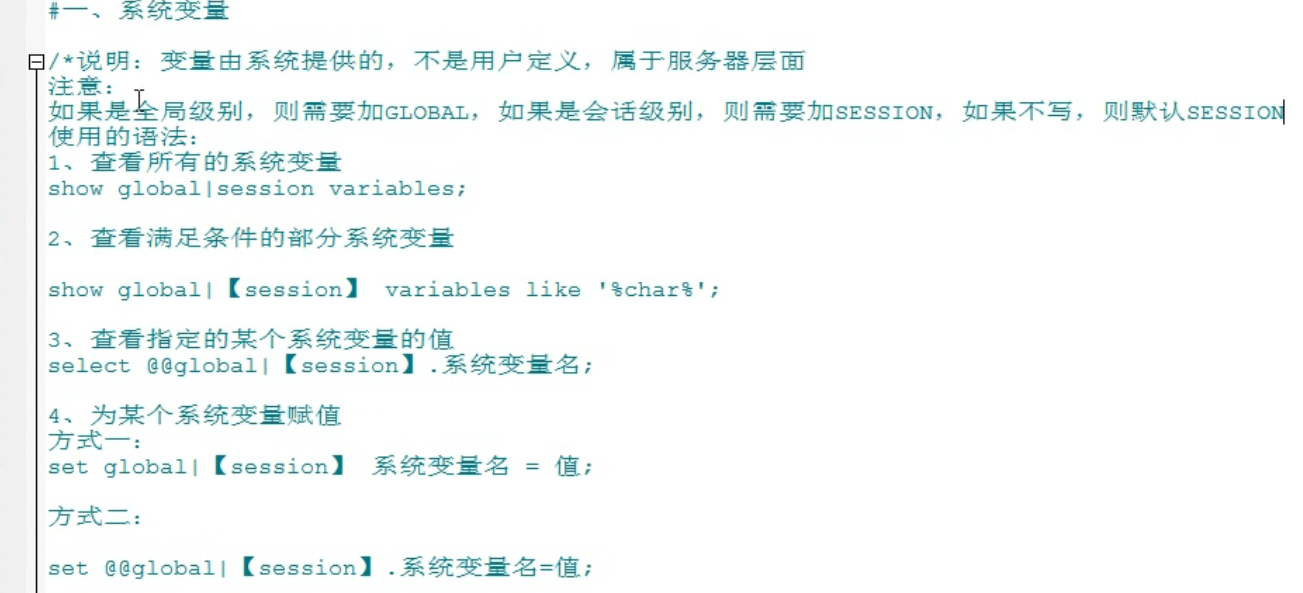
视图 create view 只保存sql语句 增删改查，一般不能增删改

表 create table 保存数据 增删改查

#九、变量

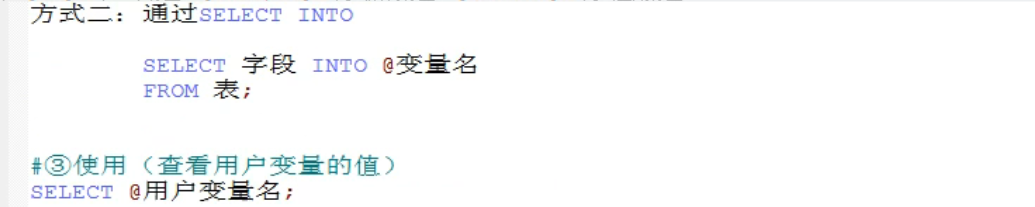
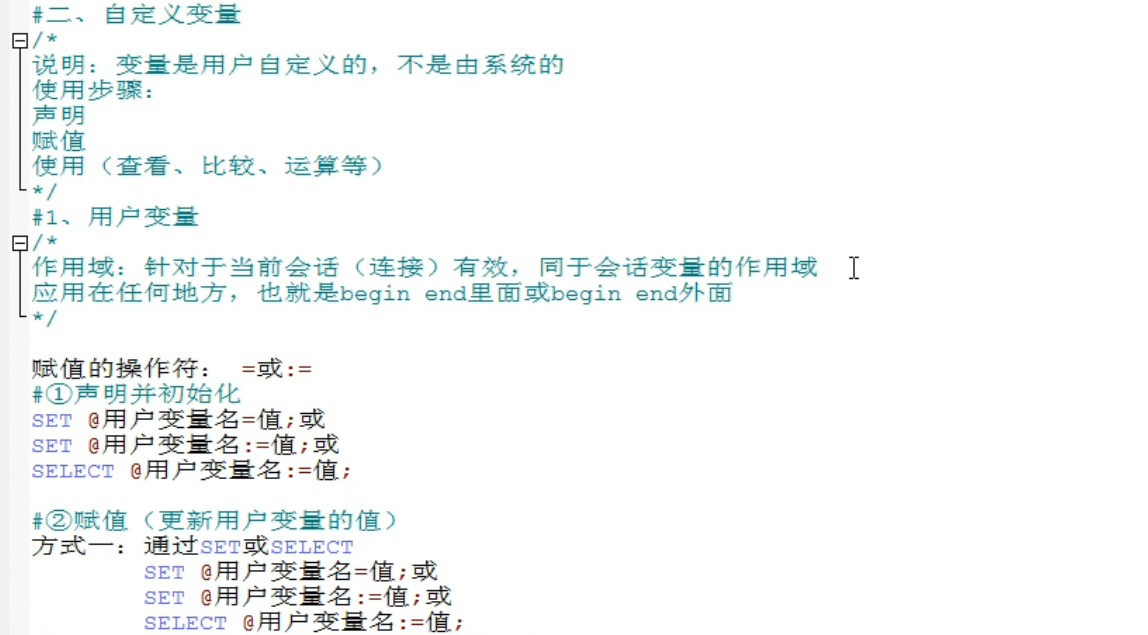
系统变量：全局变量、绘画变量

自定义变量： 用户变量、局部变量



2》、 会话变量

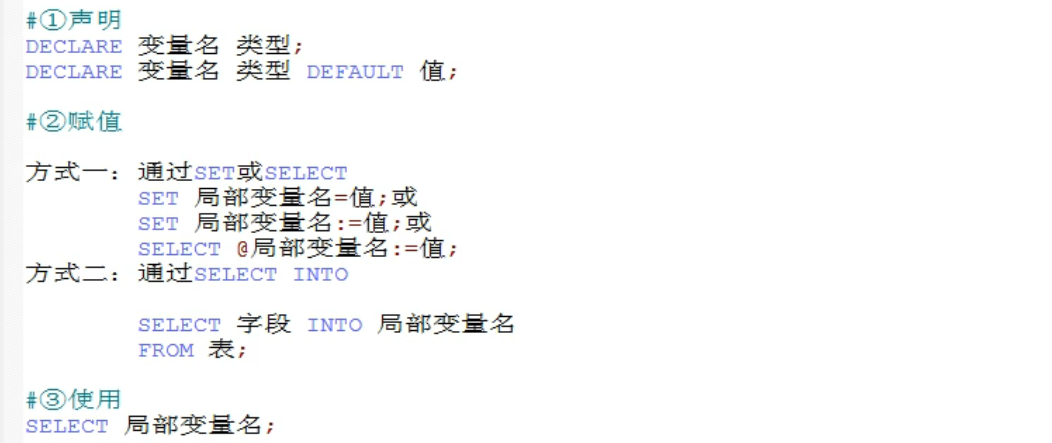
作用域：仅仅针对于当前会话（链接）有效



#2、局部变量

作用域：仅仅在定义它的begin end 中有效

应有在begin end 中的第一句！！！



**------------------------------------------------------------------------**

#九、存储过程和函数

存储过程：

含义：一组预先编译好的SQL语句的集合（类似批处理语句）

1、提高代码的重用性

2、简化操作

3、减少编译次数并减少了与数据库服务器的链接次数，提高效率。

创建语法：

create procedure 存储过程名（参数列表）

begin

存储过程体（一组合法的SQL语句）

End

注意：

1、参数列表包含3个部分：参数模式 参数名 参数类型

参数模式：

In：该参数可作为输入，传入值

Out：该参数可作为输出，返回值

Inout：该参数即可作为输入，也可输出，传入值和返回值。

2、若存储过程体中只有一条语句，begin end可省略，每条SQL语句必须加分号。

存储过程结尾·可使用DELIMITER 重新设置，DELIMITER 结束标志

调用语法：call 存储过程名（实参列表）；

删除存储过程：drop procedure 存储过程名；（一次一条）

查看存储过程的信息：show create procedure 存储过程名；