**分类：**

**小型：**access

**中型or大型：**oracle,sql server, sybase, informix(安全),mysql

**海量：**db2

**数据库选择：**

1:成本

2:功能多少

3:并发性

4:安全性

**SQL语言：**

1:DDL(数据定义语言) create drop alter

2:DML(数据操作语言) insert update delete

3:DQL(数据查询语句) select

4:DCL(数据控制语句) grant revoke commit rollback

create database LiangshanHeros

use LiangshanHeros --使用某个数据

go

create table hero --create一张表

drop table hero --删除一张表

(heroId int,heroName varchar(50),heroNickname varchar(50),salary int)

**插入：**

**insert** **into** hero values(1,’宋江’，’及时雨’,2000) --insert表值

如果插入部分字段，指定要插入的字段列表:

**insert into** hero(heroId, heroName) values(2,’宋江’)

**查询：**

**select** \* from hero --简单查询数据

**select** \* from hero where salary<200 --查询表中salary<200的数据

查询指定列：select 字段1,字段2… from 表名 where 条件

取消重复行：select distinct 字段 from 表名 where 条件

算数表达式：select 字段，算术式 “列名”from 表名

like 的模糊查询：1:%代表0 或多个字符,2:\_代表一个字符，

相当于linux 通配符下的\*和?,例如：

查询首字符为s的员工姓名和工资：

select ename,sal, from emp where ename like ‘s%’

查询显示第三个字符为0 的所有员工的姓名和工资

select ename,sal from emp where ename like ‘\_\_0%’

where 中 in:批量查询，

select \* from emo where empno in(123,456,789)

is null :

select \* from emp where 变量名is null

逻辑操作：and , or,

order by: 按顺序排,默认是升序

工资按顺序排：select \* from emp order by sal asc(desc 降序)

分页查询：字查询

复杂查询：min,max,sum,count,avg

显示最低工资和该员工的其他信息：

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

如果含有group by,having,order by ,那么顺序肯定是group by,having,order by

**修改：**

**update** hero **set** salary=salary\*1.1 where salary <100 --把工资低于100的数据工资提高10%。若要修改null的数据，注意是where a is null 不是a=null

**删除：**

**delete** **from** hero where salary=200 --删除salary为200的数据

表的管理：表命名：

1:必须以字母或者\_开头

2:长度不超过128字符

3:不要使用sql server中关键字命名

4:使用如下字符：a-z, A-Z,$,#\_命名

数据类型：

**字符**

char：定长字符,最大8000字符(非unicode –unicode 用2bytes 表示一个字符，即一个unicode 表示一个汉字)，char(10) 内添加’小韩’，则前四个字节是小韩，后六个字节用空格补全，没有被回收.

varchar: 变长字符，自动回收不用的字符,一般比char慢。

ntext:可变长度unicode,最大长度是2^30-1.

text:可变长度非unicode，最大长度为2^31-1

nchar: 定长，unicode编码，最大4000字符。即char(n) n最大为4000.

nvarchar: 变长,最大4000字符

数字：

bit: 1 bit

smallint: 2bytes

int: 4bytes

bigint: 8bytes

numeric & float :推荐numeric 表达小数。

日期：

datetime: 日期

timestamp:时间戳

图片：

image:

视频：

binary:

一般图片和视频都另存在一个服务器上，存取效率高，一般不直接存在数据库上.

primary key:任何类型都能做主健，类似于dict中的key，主健必须给值，不能为空。

foreign key: 外健，相当于两个表里的数据共享

如: deptno int foreign key references dept(deptno)

注意，外健只能指向另一个表的主健，并且数据类型要一致。