结构化查询语言(Structured Query Language)简称SQL，是一种特殊目的的编程语言，

是一种数据库查询和程序设计语言，用于存取数据以及查询、更新和管理关系数据库系统。

分四大类：

DDL:数据定义语言（数据库及数据库对象的创建、修改、删除）**create alter drop**

DCL:数据控制语言（权限控制）授权和收回权限 grant revoke

DML:数据操作语言 （对数据表中的数据进行增删改操作） **insert delete update**

DQL:数据查询语言（对数据表中的数据进行查询操作）**select**

DDL:

create:创建

1、创建数据库

语法结构：create database [if not exists] 数据库名;

2、创建表

语法结构：

CREATE TABLE 表名

(

字段名 数据类型 约束说明,

字段名 数据类型 约束说明,

......

字段名 数据类型 约束说明

);

alter：修改

1、向表中添加新字段

alter table student add sage int after sname; 在sname字段后面添加sage字段

alter table student add saddr varchar(100) after semail;-- 在指定的列后面加新列

alter table student add id int first;-- 在第一个字段的前面加新列

2、删除字段

alter table student drop id; 删除id字段

alter table student drop sage; 删除sage字段

1. 修改字段

modify只能修改字段的属性信息，不能改名

alter table student modify sgender char(20);

change既可以修改字段名，还可以改属性信息

alter table student change sgender ssex CHAR(2);

4、修改表名

alter table stu rename student;

5、修改表的编码格式

alter table student convert to character set 'utf8';

drop：删除

-- 删除表

drop table 表名;

-- 删除数据库

drop database 库名;

约束

约束：为了保证表中数据的完整性，给表添加的一些限制。

完整性：准确性 正确性

数据库完整性类型：

实体完整性：行方向记录与记录之间不能出现重复数据，冗余：重复。

域完整性：列方向符合一些规范。数据类型约束、默认约束

引用完整性：限制两个表之间关系，通过外键约束来完成

自定义完整性：存储过程 和 触发器来完成。

约束类型：

1.主键约束：primary key （实体完整性）

一个表中最多有一个主键，作为主键的列不允许出现重复数据，同时该列自动非空。

2.唯一约束：unique（实体完整性）

一个表中可以有多个，设置唯一约束的列不允许出现重复数据，但是允许为空，空值最多只有一个。

3.非空约束：not null（域完整性），创建表的同时添加

4.默认值约束：default（域完整性）

设置了默认约束的列，当不给值是就用默认值填充。

1. 外键约束：foreign key （引用完整性）

-- 创建带有约束的表

create table student(

sid int primary key,-- 主键约束 要求学号唯一且不允许为空

gid int,-- 关联的班级编号，与grade表中的gid建立外键关系

sname varchar(20) unique not null,-- 唯一约束 要求姓名唯一 非空约束，姓名不允许为空

sgender CHAR(2) default '男' NOT NULL,-- 性别的默认值是男 且不允许为空

sage INT default 20 NOT NULL,-- 年龄默认20 且不允许为空

saddr VARCHAR(100),

foreign key(gid) references grade(gid)-- 外键约束 学生表的gid与班级表gid产生关联

);

-- 给表添加约束

-- 添加主键约束

alter table student add primary key(sid);

-- 删除主键约束

alter table student drop primary key;

-- 添加唯一约束

alter table student add unique(sname);

-- 删除唯一约束

alter table student drop key sname;

-- 添加默认约束

alter table student alter sgender set default '男';

-- 删除默认约束

alter table student alter sgender drop default;

-- 添加外键约束

alter table student add foreign key (gid) references grade(gid);

-- 删除外键约束

alter table student drop foreign key student\_ibfk\_1;

alter table student drop key gid;

自增列

1.自动增长，默认从1开始

2.用户插入数据时可以不给值

3.必须是整型才可以设置自增列

4.设置自增列的字段必须是主键，自增列的编号不能重复

格式：

将学号sid设置为自增列

alter table student modify sid int auto\_increment;

设置自增列的初始值

alter table student auto\_increment=1000;

DML

1. 添加数据insert

单行插入

1、insert into 表名(字段名1，字段名2，........)

values(值1，值2，......);

举例1：

insert into student

values(1001,1,'张三','男',23,'北京海淀区');

举例2：

insert into student(gid,sname,sgender,sage,saddr)

values(2,'李四',DEFAULT,24,'河北唐山');

2、insert into 表名 set 字段名=值, 字段名=值,........

insert into student set gid=1,sname='王五',sage=25;

3、多行插入

insert into student(gid,sname,sgender,sage,saddr)

values

(1,'翠花','女',21,'河北保定'),

(3,'武松',DEFAULT,45,'山东梁山');

1. 修改数据update

update 表名 set 字段名=新值,字段名=新值..... where 修改条件；

举例1：

update student set saddr='北京朝阳区' where sname='王五';

举例2：

update student set sname='李四四',sgender='女' where sid=1002;

1. 删除数据delete

delete from student 删除表的全部数据

delete from student where 删除条件；

**带有主外键关系的表在添加和删除数据时需要注意的问题：**

1. **添加数据时，要先向主表中添加数据，再向子表中添加数据**
2. **删除数据时，要先删除子表的数据，再删除主表的数据**

**truncate 删除表的全部数据**

**truncate table 表名；**

**delete 与 truncate区别：**

1. delete既可以根据条件删除，也可以删除全部数据，而truncate只能全删，truncate后面不能跟where 条件。
2. delete删除数据后，表中的自增列不会从头开始，truncate删除后自增列会从头开始。
3. delete在删除带有主外键关系的表数据时，遵循先删除子表再删除主表的原则；truncate不删除带有主外键关系的主表中的数据。
4. 当删除全部数据时，truncate的效率会比delete高。