表与表的关系

> - 一对一关系：人和身份证，一对一关系，
> - 一对多关系：班级和学生，一个班级可以对应多个学生,数据库通过设置主外键关联关系，来维护两张表的一对多的关系，主键和外键都是天加到字段上的属性。
> - 多对多关系：学生和选课，商品和顾客，通过建立第三张表专门存放外加字段，从而实现多对多的关系,实际上是通过设置两个一对多关系来间接表示多对多关系。
> - 删除表：删除表要==先删除从表，再删除主表==，

表操作

- 创建表：==CREATE TABLE 表名（字段名 字段类型 约束...）==

> - MySQL字符类型
>   - 数值类型
>   - tinyint(1一个字节)
>   - smallint(2 个字节)
>   - mediumint(3个字节)
>   - int(4个字节) 一个字节包括8位（0、1）二进制数。
>   - 日期和时间
>   - year（1个字节） 范围：1901--2155
>   - time（3个字节） 显示范围："—838:59:59"~"838:59:59",负数是因为可以表示时间间隔。
>   - date（3个字节）日期范围：“1000-01-01 ”~“9999-12-31 ”
>   - datetime（8个字节）日期的范围：“1000-01-01 00：00：00”~“9999-12-31 23：59：59”
>   - timestamp（4个字节）时间戳：表示1970，1，1 00：00：00到现在的毫秒数（1000=1毫秒）。可以用于导入相同的内容而不重复
>   - 字符串类型
>   - char(M)个字节
>   - varchar(M) 长度+1个字节--->0--65535
>   - text 长度+1个字节 ---->0---65535
>   - 浮点型：（常用double）
>   - float （4个字节）单精度浮点型范围小
>   - double（8个字节）范围大
>   - decimal（M,D）占M+2个字节，decimal（5,2），M是数值得最大数（精度）D小数点后数值的个数（标度）
>   - 二进制类型：（常用blob ）
>   - 保存视频，音频，图片格式的数据，
>   - 保存：缩成二进制流保存到数据库
>   - 使用：从数据库将二进制流读出，OutputStream还原到本地硬盘。
>   - bit (M) M位的二进制数据M/8个字节
>   - binary （M）M个字节
>   - varbinary （M）长度+1个字节

    - tinyblob 255个字节
    - blob 2的16次方—1个字节
    - longblob 2的32次方—1个字节

- （==SQLserver==）字符类型

```
- character（字符串）
  - char（size）保存固定长度的字符串
  - varchar(n)可变长度的字符串，最多8000个字符
  - text可变长度的字符串，最多2GB字符数据。
- Unicode字符串
  - nchar(n)固定长度的Unicode数据，最多4000字符。
  - nvarchar(n)可变长度Unicode数据，最多4000字符。
  - ntext 可变长度的Unicode数据，最多2GB。
- Binary类型
  - bit允许0、1、或null
  - binary（n）固定长度的二进制，最多8000字节
  - varbinary(n) 可变长度的二进制，最多8000字节
  - image可变长度的二进制，最多2GB。
- number类型
  - tinyint：允许0--255的所有数字
  - int：4字节
  - bagint:8字节
  - float：
  - real
  - money：十进制货币数字
- date类型：
  - datetime:(从1733年1月1日--9999年12月31日，精度3.33毫秒),8 bytes
  - date:仅存储日期，0001年1月1日到9999年12月31日，3bytes
- 其他数据类型
  - uniqueldentifier 存取全局标识符（GUID）
  - xml存取XML格式化数据。最多2GB。
  - cursor存储对用于数据库操作的指针应用。
- 常见的字段类型选择
  - 字符类建议用varchar、nvarchar
  - 金额货币采用money
  - 自增长，采用bigint数据类型，（数据量大int装不下，修改麻烦）
  - 时间类型，datetime类型
```

什么是约束，有哪些?

- 创建表的时候，对表进行限定，保证之后插入表的数据完整性和准确性
- 约束种类
    - primary 主键约束，非空即唯一
    - no null 非空
    - unique 唯一
    - autoincrement 主键自增长，当主键integer，可以自增长。
    - foreign key外键 一张表的外键可以关联另外一张表的主键，而保证数据的完整性。

- check 约束用来限制列中值的范围
- default 约束用于向列中插入默认值

创建表实例

```
```
  create table student(
  id integer primary key,主键
  name text not null ,文本最大长度65535, 非空
  age integer unique,唯一
  gender text,
  emall text,
 check(age>0) 约束用于限制列中默认值的范围
 );
```
```

更新表: ALTER+TABLE+表名

- 增加列 **add**

```
给student增加一classname列.
alter table student add calssname text;
```

- 删除列 **drop**

```
删除表student的一个列（column）yy.
alter table student drop column yy
```

- 修改列字段类型 **modify**

```
修改表lals的Sname的数据类型，varchar--text
alter table lals modify  Sname text
```

- 修改列名 **chanage** （同时可以修改字段类型）

```
将数表student的gender改变成sex。
alter table student change gender sex text
```

- 修改表名 **rename**

```
将表名student 修改为stu
alter table student rename to stu
```

查询表

- show tables://查询当前数据库下面的所有表
- desc + 表名://查询表的详细信息。

> - 删除表：
> - drop table

对数据的操作（重要）

- 新增：

```
insert into   表名（字段列表 可以省略，如果省略则表示每个字段都需要添加）
values(值列表);
insert  into  user  values（1，“张三”，“男”）；//varchar类型     单双引号均可
```

- 删除：

```
delete  from  表名;   //谨慎使用，一旦执行，会清空整张表。造成开发事故。
delete from  表名  where  条件;  //通过id删除，通过name 去删除,建议使用，加上
where  条件。
delete from  user  where  id= 1;   //删除id为 1的数据
```

- 修改：

```
update 表名  set 字段名 = 字段值 ;   //同样谨慎使用，一旦使用将更改全部数据
update 表名  set 字段名 = 字段值   where  条件；//必须使用该表示方法  加上 where
条件。
```

- 查询：

```
select * from 表名;
select * from 表名 where 条件;
```