

数据表

概念

- 表和表结构：每个数据库包含了若干个表。表是SQL Server中最主要的数据库对象，它是用来**存储数据**的一种**逻辑结构**。**表由行和列组成**，因此也称为**二维表**。表是在日常工作和生活中经常使用的一种**表示数据及其关系**的形式
- 表结构：组成表的各列的名称及数据类型， 统称为表结构。
- 记录：每个表包含了若干行数据， 它们是表的“值” ， 表中的一行称为一个记录。
- 字段：表中的一列称为字段。
例如，表3.1中表结构为（学号， 姓名， 性别， 出生时间， 专业， 总学分， 备注），包含7个字段， 由5个记录组成。
- 空值：空值(NULL)通常表示未知、不可用或将在以后添加的数据。若一个列允许为空值， 则向表中输入记录值时可不为该列给出具体值； 而一个列若不允许为空值， 则在输入时必须给出具体值
- 关键字：若表中记录的某一字段或字段组合能唯一标识记录， 则称该字段或字段组合为候选关键字(Candidate key)。若一个表有多个候选关键字， 则选定其中一个为主关键字 （Primary key),也称为主键。

```
create table 表名(  
    字段1 字段1类型 列级完整性约束条件,  
    字段2 字段2类型 列级完整性约束条件,  
    字段3 字段3类型 列级完整性约束条件,  
    ...  
    字段n 字段n类型 列级完整性约束条件  
)
```

语法

创建

注：列级完整性约束条件如下

- PRIMARY KEY:指定该字段为主键（不为空且唯一可以为多个关键字设置主键）
- NULL/NOT NULL:指定的字段允许/不允许为空， 如果没有约束条件， 则默认为NULL(列级约束)
- UNIQUE:指定字段取值唯一， 即每条记录的指定字段值不能重复 (允许列中有一个空值)
- DEFAULT <默认值>： 指定设置字段的默认值。（列级约束）
- CHECK <条件表达式>： 对输入值检验， 拒绝接受不满足条件的值。只看非空约束（不是null 的）
- identity(1,1)自增长（第一个值,步长） 不能直接添加数据自动增长 —— 如果想强制执行 set identity_insert 表名 on
- foreign key 外键 注意字段类型必须一样 —— foreign key reference 表名(字段)
- 注意： 第二条和第四条是列级约束其他的是列级约束和表级约束都可以 列级约束是 只能针对当列约束（表级约束可以多列）

```
create table student(  
    sno char(8) not null primary key,  
    same char(10) not null,  
    grender char(2) null default '男' check(grender='女' or  
    grender='男'),  
    sbirth date,  
    email char(30) unique, -- 唯一约束  
    major char(20),  
    chedit int check(chedit>=0 and chedit<120),  
    remark varchar(500)  
)
```

例子

```
create table student(  
    sno char(8) ,  
    same char(10),  
    grender char(2) null default '男' check(grender='女' or  
    grender='男'),  
    sbirth date,  
    email char(30) unique, -- 唯一约束  
    major char(20),  
    chedit int check(chedit>=0 and chedit<120),  
    remark varchar(500),  
    primary key(sno,same) -- 联合主键  
)
```

注意：一个主键可以由多个字段构成

操作

修改数据表

- 增加字段
 - 语法：alter table 表名 add 新字段名称 数据类型 列级完整性约束条件
 - 例子：alter table student add sql char(30) null
- 修改字段的数据类型
 - 语法：alter table 表名 alter column 字段名称 新数据类型
 - 例子：alter table student alter column sql char(20)
- 删除字段
 - 语法：alter table 表名 drop column 字段名称
 - 例子：alter table student drop sql
- 添加约束
 - 语法：alter table 表名 add constraint 约束名 约束条件 (字段名称)
 - 例子：alter table student add constraint uq_stu_uq unique (sql)
- 修改约束
 - 语法：alter table 表名 alter constraint 约束名 约束条件 (字段名称)
 - 例子：alter table student alter constraint uq_stu_uq null
- 删除约束
 - 语法：alter table 表名 drop constraint 约束名
 - 例子：alter table student drop uq_stu_uq

删除表

- 语法：drop table 表名
- 例子：drop table student

关系的完整性

- 关系的完整性（最大限度地保证数据的**正确性**）
- 关系模型的完整性规则是对关系的某种约束条件。
- 关系模型中允许定义3类完整性约束
 - 实体完整性 —— 实体完整性规则若属性A是基本关系R的主属性， 则属性A不能取空值。
 - 参照完整性 —— 学生、课程、学生与课程之间的多对多联系选修可以用如下3个关系表示。
√学生 (**学号**, 姓名, 性别, 专业号, 年龄)
√课程 (**课程号**, 课程名, 学分)
√选修 (**学号**, **课程号**, 成绩)
 - 用户自定义定义完整性 —— 用户自定义的完整性就是针对某一具体关系数据库的约束条件， 它反映某一具体应用所涉及的数据必须满足语义要求。例如某个属性必须取唯一值、属性值之间应满足一定的函数关系、某属性的取值范围在0~100之间等