



数据库系统概论

An Introduction to Database System

第三章 关系数据库标准语言SQL



内蒙古农业大学计算机与信息工程学院



3.4 数据更新

- 学习目标

- 掌握SQL语言**insert** 插入数据语句格式并能够在实践案例中应用
- 掌握SQL语言**update** 修改数据语句格式并能够在实践案例中应用
- 掌握SQL语言**delete**删除数据语句格式并能够在实践案例中应用



3.4 数据更新

学生选课数据库

- Student(sno,sname,ssex,birthdate,cno,adress)
- course(cno,cname,cpno,credit)
- sc(sno,cno,Grade)



3.4 数据更新

1. 插入数据
2. 修改数据
3. 删除数据



1. 插入数据

- 两种插入数据方式
 - 插入单个元组
 - 插入子查询结果



1. 插入数据

- 插入单个元组
 - 语句格式

INSERT

INTO <表名> [(<属性列1>[, <属性列2 >...])]

VALUES (<常量1> [, <常量2>] ...)



1. 插入数据

■ INTO子句

- 指定要插入数据的表名及属性列名
- 指定**全部**属性列：属性列的顺序可与表定义中的顺序**不一致**
- **没有**指定任何属性列：表示要插入的是一条完整的元组，新插入的元组必须在每个属性列上有值，且插入数据顺序与表定义中属性列的顺序**一致**
- 指定**部分**属性列：新插入的元组在**没有指定**的属性列上取空值



1. 插入数据

- VALUES子句
 - 提供的值必须与**INTO**子句匹配
 - 值的个数
 - 值的类型



任务6：学生管理系统数据的更新

[1] 将一个新学生元组(2016010104, 陈冬, 男, 1999-04-12, 20160101, 内蒙古呼和浩特市)插入到Student表中

```
insert  
into student  
values ('2016010104', '陈冬', '男', '1999-04-12', '20160101', '内蒙古呼和浩特市')
```

dbo.student
列
sno (char(11), not null)
sname (varchar(50), not null)
ssex (char(2), not null)
birthdate (date, not null)
clno (char(8), null)
adress (varchar(50), null)

```
insert  
into student(sno, sname, ssex, birthdate, clno, adress)  
values ('2016010104', '陈冬', '男', '1999-04-12', '20160101', '内蒙古呼和浩特市')
```



任务6：学生管理系统数据的更新

[2]插入一条选课记录（'2016030101','004'）

```
insert  
into sc(sno, cno)  
values('2016030101', '004')
```

```
insert  
into sc  
values('2016030101', '004', NULL)
```

```
dbo.sc  
列  
sno (char(11), not null)  
cno (char(3), null)  
grade (numeric(5,0), null)
```



1. 插入数据

- 插入子查询结果
- 语句格式

INSERT

INTO <表名> [(<属性列1> [, <属性列2>...)]

子查询;

- 说明
 - 子查询**SELECT**子句中目标列必须与**INTO**子句属性列匹配
 - 值的个数
 - 值的类型



1. 插入数据

[例3]对每一个班级，求学生的平均年龄，并把结果存入数据库表class-avgage中

第一步：建表

```
create table class_avgage  
(clno char(8),  
 avg_age int)
```

第二步：插入数据

```
insert  
into class_avgage  
select clno, avg(year(getdate())-year(birthdate))  
from student  
group by clno
```



1. 插入数据

- **RDBMS**在执行插入语句时会检查所插元组是否破坏表上已定义的完整性规则
 - 实体完整性
 - 参照完整性
 - 用户自定义的完整性
 - **NOT NULL**约束
 - **UNIQUE**约束
 - **check**约束



2.修改数据

- 修改数据与修改基本表的区别

update语句与alter语句



2.修改数据

- 三种修改方式
 - 修改某一个元组的值
 - 修改多个元组的值
 - 带子查询的修改语句



2.修改数据

- 语句格式

UPDATE <表名>

SET <列名>=<表达式>[, <列名>=<表达式>]...

[**WHERE** <条件>];

- 功能

- **SET**子句把<表达式>的值赋值给相应的属性列，可以同时多个属性列值进行修改
- 修改指定表中满足**WHERE**子句条件的元组
- 如果省略**WHERE**子句，表示要修改表中的所有元组



任务6：学生管理系统数据的更新

◆ 修改某一个元组的值

[4]将学生2016010101的出生日期改为1994-05-26

```
update student  
set birthdate='1994-05-26'  
where sno='2016010101'
```

	sno	sname	ssex	birthdate	clno	adress
1	2016010101	李勇	男	2000-01-05	20160101	内蒙古呼和浩特市
2	2016010102	刘诗晨	女	2001-08-15	20160101	内蒙古呼和浩特市
3	2016010201	王义明	男	2000-04-17	20160102	内蒙古鄂尔多斯市
4	2016010202	张婷婷	女	2000-10-12	20160102	内蒙古鄂尔多斯市
5	2016010203	李丽	女	1999-12-01	20160102	内蒙古鄂尔多斯市
6	2016020101	贾向东	男	2001-08-18	20160201	内蒙古包头市
7	2016020102	陈红	女	2001-11-15	20160201	内蒙古包头市
8	2016030101	张伊凡	女	2001-01-25	20160301	内蒙古呼和浩特市



任务6：学生管理系统数据的更新

◆ 修改多个元组的值

[5]将所有选修001号课程的学生们的选课成绩提高3分

```
update sc  
set grade=grade+3  
where cno='001'
```

	sno	cno	grade
1	2016010101	001	85
2	2016010101	003	80
3	2016010101	002	90
4	2016010102	001	85
5	2016010102	002	80
6	2016010102	003	82
7	2016010102	005	89
8	2016020101	001	72
9	2016020101	002	59
10	2016020101	004	80

任务6：学生管理系统数据的更新

◆ 带子查询的修改语句

[6]将20160102班全体学生的成绩置零

```
update sc
set grade=0
where sno in
(select sno
from student
where cno='20160102')
```

	sno	cno	grade
1	2016010101	001	85
2	2016010101	003	80
3	2016010101	002	90
4	2016010102	001	85
5	2016010102	002	80
6	2016010102	003	82
7	2016010102	005	89
8	2016020101	001	72
9	2016020101	002	59
10	2016020101	004	80

```
update sc, student
set grade=0
where student.sno=sc.sno and cno='20160102'
```





2. 修改数据

- **RDBMS**在执行修改语句时会修改操作是否破坏表上已定义的完整性规则
 - 实体完整性
 - 参照完整性
 - 用户自定义的完整性
 - **NOT NULL**约束
 - **UNIQUE**约束
 - **check**约束



2. 修改数据

- 练习1：将学号为2016010101的学生年龄减小1岁

```
update student  
set birthdate=dateadd(year, 1, birthdate)  
where sno='2016010101'
```

- 练习2：将所有学生的年龄增加1岁

```
update student  
set birthdate=dateadd(year, -1, birthdate)  
where sno='2016010101'
```



3. 删除数据

- 删除数据与删除基本表的区别

delete语句与drop语句



3. 删除数据

- 三种删除方式
 - 删除某一个元组的值
 - 删除多个元组的值
 - 带子查询的删除语句



3. 删除数据

- 语句格式

DELETE

FROM <表名>

[**WHERE** <条件>];

- 功能

- ◆ 删除指定表中满足**WHERE**子句条件的元组
- ◆ **WHERE**子句缺省表示要删除表中的所有元组，但表的定义仍在字典中



任务6：学生管理系统数据的更新

◆ 删除某一个元组的值

[7]删除学号为2016020101学生的基本信息

```
delete  
from student  
where sno='2016020101'
```

	sno	sname	ssex	birthdate	clno	adress
1	2016010101	李勇	男	2000-01-05	20160101	内蒙古呼和浩特市
2	2016010102	刘诗晨	女	2001-08-15	20160101	内蒙古呼和浩特市
3	2016010201	王义明	男	2000-04-17	20160102	内蒙古鄂尔多斯市
4	2016010202	张婷婷	女	2000-10-12	20160102	内蒙古鄂尔多斯市
5	2016010203	李丽	女	1999-12-01	20160102	内蒙古鄂尔多斯市
6	2016020101	贾向东	男	2001-08-18	20160201	内蒙古包头市
7	2016020102	陈红	女	2001-11-15	20160201	内蒙古包头市
8	2016030101	张伊凡	女	2001-01-25	20160301	内蒙古呼和浩特市



任务6：学生管理系统数据的更新

◆ 删除多个元组的值

[8]删除学号为2016020101学生的选课记录

```
delete  
from sc  
where sno='2016020101'
```

	sno	cno	grade
1	2016010101	001	85
2	2016010101	003	80
3	2016010101	002	90
4	2016010102	001	85
5	2016010102	002	80
6	2016010102	003	82
7	2016010102	005	89
8	2016020101	001	72
9	2016020101	002	59
10	2016020101	004	80

任务6：学生管理系统数据的更新

◆ 带子查询的删除语句

[9]删除20160102班所有学生的选课记录

```
delete
from sc
where sno in
      (select sno
       from student
       where cno='20160102')
```

	sno	cno	grade
1	2016010101	001	85
2	2016010101	003	80
3	2016010101	002	90
4	2016010102	001	85
5	2016010102	002	80
6	2016010102	003	82
7	2016010102	005	89
8	2016020101	001	72
9	2016020101	002	59
10	2016020101	004	80

```
delete
from sc, student
where cno='20160102' and student.sno=sc.sno
```





下课了。。。

追
求



休息一会儿。。。