

# 上机实验报告

课程名： 数据库安全

指导教师： 熊光彩

姓名	周诚远	班级	SC011801	实验 得分	
		学号	2018302191		
实验 时间	2021. 3. 24	实验 地点	启翔楼 305		
实验 主题	实验二：数据约束				
实验 目的	<p>(1) 掌握使用规则实施数据完整性 a、规则的创建 b、规则的绑定 c、规则绑定的松绑 d、规则的查看与删除</p> <p>(2) 掌握使用默认值实施数据完整性 a、默认值的创建 b、默认值的绑定 c、默认值绑定的松绑 d、默认值的查看与删除</p> <p>(3) 掌握使用约束强制列数据的完整性。a、主键约束；b、外键约束；c、UNIQUE 约束，它强制表的非主键列的唯一性；d、CHECK 约束，用于定义列允许的数据格式和允许的值。e、DEFAULT 约束，当将某行首次插入表中但不知道某列的值时可以使用它，但是仍然希望使用期望的值来填充列；f、非空值约束</p>				
实验 内容	<p>(1) 使用约束强制列数据的完整性（唯一性约束、CHECK 约束、DEFAULT 约束）。a、创建学生关系表 S，学号 Sno 为主键，同时要求为姓名 Sname 字段建立唯一性约束。b、创建好 S 后，修改 S 表，取消姓名 Sname 的唯一性约束。c、再次修改 S 表，增加姓名 Sname 的唯一性约束。d、创建课程关系表 C，课程号 Cno 为主键，同时要求为学分 Ccredit 字段创建 CHECK 约束，使 <math>15 \geq Ccredit &gt; 0</math>。e、创建好 C 后，修改 C 表，禁止学分 Ccredit 的 CHECK 约束。f、再次修改 C 表，启用学分 Ccredit 的 CHECK 约束。g、创建选修关系表 SC，课程号 Cno 和学分数 Cno 共同构成主键，同时要求为成绩 Grade 字段创建 Default 约束，使成绩 Grade 的缺省默认值为 60 分。h、创建好 SC 后，修改 SC 表，取消成绩 Grade 的 Default 约束。i、再次修改 SC 表，增加成绩 Grade 的 Default 约束，使成绩 Grade 的缺省默认值为 0 分。</p> <p>(2) 使用规则，实施数据完整性 a、创建规则 Grade_Rule：限制取值为 0 到 100；创建规则 Ssex_Rule：限制取值为('M', 'F')；创建规则 Sno_Rule：总共 10 位字符，限制第 1~2 位为字母(a~z)，第 3 位取字母-，其余为数字(0~9)；b、将 Grade_Rule 规则绑定到 SC 表 Grade 列；将 Ssex_Rule 规则绑定到 S 表的 Ssex 列；将 Sno_Rule 分别绑定到 S</p>				

	<p>表的 Sno 列和 C 表的 Cno 列。c、以上绑定规则的各列，如果在表定义时，就已定义了 CHECK/DEFAULT 等约束，绑定规则会是怎样的情况？请列举各种可能的情况，并加以验证，根据实验结果，总结相应的实验结论。d、绑定规则后，分别进行数据的插入，验证绑定规则的有效性。e、将绑定到 SC 表 grade 列的规则 Grade_Rule，予以松绑。f、规则松绑后，再次进行数据的插入予以验证。g、删除 Grade_Rule 规则。</p> <p>(3) 使用默认值，实施数据完整性 a、创建默认值 Sage_Default，取值为 20 创建默认值 Ssex_Default_1，取值为 'M' 创建默认值 Ssex_Default_2，取值为 '男' b、将 Sage_Default 绑定到 S 表 Sage 列；将 Ssex_Default_1 绑定到 S 表 Ssex 列。c、绑定默认值之后，分别进行数据的插入，验证绑定默认值的有效性。d、将绑定到 S 表 Ssex 列的默认值 Ssex_Default_1，予以松绑。e、删除 Ssex_Default_1 默认值。f、再试着将 Ssex_Default_2 默认值绑定到 S 表 Ssex 列（该列同时存在 Ssex_Rule 绑定），观察有什么结果？g、以上绑定默认的各列，如果在表定义时，就已定义了 CHECK/DEFAULT 等约束，绑定默认值会是怎样的情况？请列举各种可能的情况，并加以验证，根据实验结果，总结相应的实验结论。</p> <p>(4) 思考总结 a、如果：表中的列，在表定义时，先不建立约束，而是绑定默认值/规则来实施数据完整性。b、如果：再修改表，对表的各列进行主键、外键、UNIQUE、CHECK、DEFAULT、NOT NULL 等约束的定义，会出现怎样的情况？请列举各种可能的情况，并加以验证，根据实验结果，总结相应的实验结论。</p>
<p>实验 结果 / 实验 结论</p>	<p>1. 创建表</p> <pre> USE SchoolManagement; CREATE TABLE S(Sno NVARCHAR(15) NOT NULL PRIMARY KEY,                 Sname NVARCHAR(10) UNIQUE, Ssex nvarchar(1),                 Sage SMALLINT, Sdept NVARCHAR(50), BirthPlace nvarchar(100)); CREATE TABLE C(Cno nvarchar(15) NOT NULL PRIMARY KEY,                 Cname nvarchar(50), Cpno nvarchar(15), Ccredit decimal(3,1) CHECK (15&gt;=Ccredit and Ccredit&gt;0)); CREATE TABLE SC(Sno nvarchar(15) not null foreign key references S(Sno),                   Cno nvarchar(15) not null foreign key references C(Cno) Primary Key(Sno,Cno),                   Grade decimal(4,1) default 60); </pre> <p>2. 取消禁止和重新启用约束</p> <pre> ALTER TABLE C DROP CONSTRAINT CK_C_Ccredit__07F6335A; ALTER TABLE SC DROP DF__SC_Grade__0EA330E9; ALTER TABLE S DROP UQ__S__52723D27023D5A04; ALTER TABLE C ADD CHECK (Ccredit&gt;0 and Ccredit&lt;=15); ALTER TABLE SC add default 60 for grade; ALTER TABLE S ADD UNIQUE (Sname) </pre>

### (3) 添加规则

```
3 create rule Grade_Rule
  as
  -@value between 0 and 100

  sp_bindrule Grade_Rule, 'SC.Grade'

3 create rule Ssex_Rule
  as
  -@value = 'M' or @value='F';

  sp_bindrule Ssex_Rule, 'S.Ssex'

3 create rule Sno_Rule
  as
  -len(@value)=10 and @value like '[a-z][a-z]-[0-9][0-9][0-9][0-9][0-9][0-9][0-9]';

3 sp_bindrule Sno_Rule, 'S.Sno'
  -sp_bindrule Sno_Rule, 'S.Sno'
```

### 测试规则

```
insert into S values(2,2,2,2,2,2);
INSERT INTO C VALUES (1,1,1,1);

--DROP RULE
--INSERT INTO SC VALUES(1,1,111)
```

#### 消息

消息 513, 级别 16, 状态 0, 第 81 行  
的插入或更新与先前的 CREATE RULE 语句所指定的规则发生冲突。该语句已终止。冲突发生于数据库 'SchoolManagement', 表 'dbo.S', 列 'Sno'。  
句已终止。

消息 513, 级别 16, 状态 0, 第 82 行  
的插入或更新与先前的 CREATE RULE 语句所指定的规则发生冲突。该语句已终止。冲突发生于数据库 'SchoolManagement', 表 'dbo.C', 列 'Cno'。  
句已终止。

```
--insert into S values('aa-1234567','F','F',1,'F','F');
--insert into C values('aa-1234567','F',1,1);
3 INSERT INTO SC VALUES ('aa-1234567','aa-1234567',101);
--DROP RULE Ssex_Rule
--INSERT INTO SC VALUES(1,1,111)
```

#### 消息

消息 513, 级别 16, 状态 0, 第 84 行  
的插入或更新与先前的 CREATE RULE 语句所指定的规则发生冲突。该语句已终止。冲突发生于数据库 'SchoolManagement', 表 'dbo.SC', 列 'Grade'。  
句已终止。

### 规则解绑和删除

```
3 EXEC sp_unbindrule 'SC.Grade'
EXEC sp_unbindrule 'S.Ssex'
EXEC sp_unbindrule 'S.Sno'
EXEC sp_unbindrule 'C.Cno'

drop rule Grade_Rule
```

在表已经有定义 check 和 default 的时候且表空时, 仍然可以定义规则, 不管冲突与已经存在的是否冲突。但是当数据表中有数据时, 增加约束会出现问题。规则则不会。

```

exec sp_bindrule Grade_Rule, 'C.Ccredit'
go
--drop rule Grade_Rule

insert into C values('aa-2234567', 'F', 1, 21);

```

#### 消息

消息 547, 级别 16, 状态 0, 第 96 行  
 INSERT 语句与 CHECK 约束"CK\_C\_Ccredit\_\_0F975522"冲突。该冲突发生于数据库"SchoolManagement", 表"dbo.C", column 'Ccredit'。  
 语句已终止。

## (4) 添加默认值

```

insert into C values('aa-2234567', 'F', 1, 21);
create default Sage_Default
as 20
exec sp_bindefault Sage_Default, 'S.Sage'
go
insert into S (Sno, Sname, Sdept, BirthPlace) values('bb-1234567', '2', '1', '1')
select * from S ;

```

#### 结果 消息

Sno	Sname	Ssex	Sage	Sdept	BirthPlace
bb-1234567	2	男	20	1	1

## 重绑默认值后，会覆盖原有默认值

```

exec sp_bindefault Ssex_Default_1, 'S.Ssex'
go
insert into S (Sno, Sname, Sdept, BirthPlace) values('bc-1234567', '3', '1', '1')
select * from S ;

```

#### 结果 消息

Sno	Sname	Ssex	Sage	Sdept	BirthPlace
bb-1234567	2	男	20	1	1
bc-1234567	3	M	20	1	1

与规则冲突，可以绑定但是无法 insert

```
exec sp_bindefault Ssex_Default,'S.Ssex'
go
insert into S (Sno,Sname,Sdept,BirthPlace) values('bc-1234562','5','1','1')
--select *from S ;
```

消息

已将默认值绑定到列。

消息 513，级别 16，状态 0，第 1 行

列的插入或更新与先前的 CREATE RULE 语句所指定的规则发生冲突。该语句已终止。冲突发生于数据库 'SchoolManagement'，表 'dbo.S'，列 'Ssex' 语句已终止。

解绑并删除

```
EXEC sp_unbindefault 'S.Ssex'
go
drop default Ssex_Default_1
--insert into S (Sno,Sname,Sdept,Birtl
--select *from S ;
```

消息

已解除了表列与其默认值之间的绑定。

通过上述验证，在已存在 check 和 default 的情况下不会对绑定默认值产生影响，只会在插入时失败。Default 的值可以被覆盖。

(5) 思考总结

可以在表定义时，先不建立约束，而是在后来绑定默认值/规则来实施数据完整性。

```
alter table SC
add constraint FK_ePosition
foreign key(grade) references grade(grade)
```

消息

消息 1776，级别 16，状态 0，第 114 行

主被引用表 'grade' 中没有与外键 'FK\_ePosition' 中的引用列表匹配的主键或候选键。

消息 1750，级别 16，状态 0，第 114 行

无法创建约束。请参阅前面的错误消息。

	<p>外键相关联的必须具有 unique 属性，不然无法添加主键唯一无法增加，删改必须清除现有关系</p> <pre> alter table S add constraint Sname primary key(sname); --alter table SC --add constraint FK_ePosition --foreign key(grade) references grade(grade) </pre> <p>消息</p> <p>消息 1779, 级别 16, 状态 0, 第 113 行 表 'S' 已经有针对它定义的主键。 消息 1750, 级别 16, 状态 0, 第 113 行 无法创建约束。请参阅前面的错误消息。</p> <p>UNIQUE、CHECK、DEFAULT、NOT NULL 等约束的定义存在时，可以修改，添加，但是若与已有数据发生冲突，则无法完成添加约束。</p> <pre> ALTER TABLE S ADD UNIQUE (Sname) </pre> <p>消息</p> <p>消息 1505, 级别 16, 状态 1, 第 117 行 因为发现对象名称 'dbo.S' 和索引名称 'UQ__S__52723D271CF15040' 有重复的键, 所以 CREATE UNIQUE INDEX 语句终止。重复的键值为 (2)。 消息 1750, 级别 16, 状态 0, 第 117 行 无法创建约束。请参阅前面的错误消息。 语句已终止。</p>
实验心得	<p>通过本次实验，我了解了数据库的约束和规则以及默认值的定义，这使得我对数据库的数据完整性有了更加深刻的认识。</p>