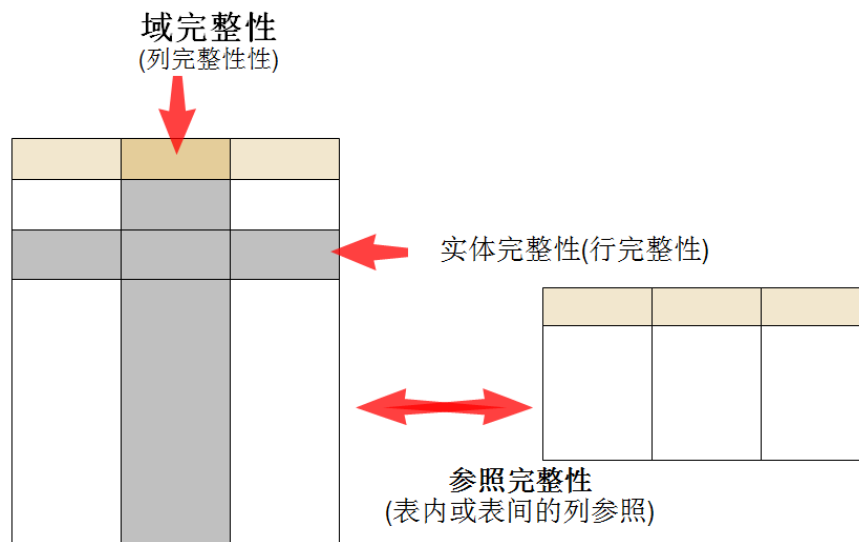


第 1 章 实现数据完整性

数据完整性类型



数据完整性实现方式

完整性类型	约束类型
域	DEFAULT (默认值约束)
	CHECK (检查约束)
	FOREIGN KEY (外键约束)
实体	PRIMARY KEY (主键约束)
	UNIQUE (唯一值约束)
参照	FOREIGN KEY (外键约束)

实现实体完整性

Student 表 学生必须有学号 不能有重复的学号 身份证号不能重

1. 给学生表 学号添加主键

```
insert dbo.TStudent values ('0000000002', '韩立刚', '男', '132302197506055634', '1975-06-5', 'hanligang@onest.net', '网络班', getdate())
```

主键唯一 并且不允许为空 clustered 记录存放按顺序

```
alter table TStudent
add
constraint PK_StudentID
primary key clustered (StudentID)
```

删除主键的命令

```
ALTER TABLE [dbo].[TStudent] DROP CONSTRAINT [PK_StudentID]
```

主键唯一 并且不允许为空 nonclustered 记录存放按录入顺序

```
alter table TStudent
add
constraint PK_StudentID
primary key nonclustered (StudentID)
```

2.给身份证号那列添加唯一约束 允许为空 只允许一条记录为空

```
alter table TStudent
add
constraint U_cardID
unique nonclustered (cardID)
```

```
ALTER TABLE [dbo].[TStudent] DROP CONSTRAINT [U_cardID]
```

3.设置自增列 作为主键

```
create table ts
(
studentID int,
studengName nvarchar(10),
sex nchar(1)
)
insert ts (studengName,sex ) values ('韩立军','男')
```

以下命令允许ts表插入自增列

```
SET IDENTITY_INSERT ts ON
```

插入式可以为自增列赋值

```
insert ts (studentID,studengName,sex ) values (10,'韩立刚','男')
```

产生最大编号

```
declare @studentN int
```

```
select @studentN=max(studentID)+1 from dbo.ts
insert dbo.ts (studentID,studengName,sex) values (@studentN,'韩立军','男')
```

```
select * from dbo.ts
```

实现域完整性

学生成绩的范围在 0-150 之间,Email 的字段中必须出现@, 出生日期在应该在 1950 到今天,cardID 长度必须在 15 位以上,性别只能输入“男”和“女”,否则不认为合法,EnterTime 默认为当前时间。

1. 学生成绩的范围在 0-150 之间

检查现在的数据是否满足 check 约束

```
ALTER TABLE [dbo].[TScore] WITH CHECK ADD CONSTRAINT [CK_mark] CHECK
(( [mark]>=(0) AND [mark]<=(150)))
GO
```

不检查现有数据是否满足check要求

```
ALTER TABLE [dbo].[TScore] WITH NOCHECK ADD CONSTRAINT [CK_mark] CHECK
(( [mark]>=(0) AND [mark]<=(150)))
ALTER TABLE [dbo].[TScore] CHECK CONSTRAINT [CK_mark]
```

2.Email 的字段中必须出现@

```
ALTER TABLE [dbo].[TStudent] WITH CHECK ADD CONSTRAINT [CK_Email] CHECK
(( [Email] like '%@%.%' ) )
```

3.出生日期在应该在 1950 到今天

```
ALTER TABLE [dbo].[TStudent] WITH CHECK ADD CONSTRAINT [CK_Birthday]
CHECK (( [Birthday]>'1950-1-1' AND [Birthday]<getdate() ) )
GO
```

4.cardID 长度必须在 15 位以上 一列不允许有多个约束

```
ALTER TABLE [dbo].[TStudent] WITH NOCHECK ADD CONSTRAINT [CK_cardID]
CHECK (( len([cardID])>=(15) ) )
```

5.性别只能输入“男”和“女”

```
ALTER TABLE [dbo].[TStudent] WITH CHECK ADD CONSTRAINT [CK_sex] CHECK
(( [sex]='女' OR [sex]='男' ) )
GO
```

6.默认值约束

在班级列上添加默认值约束

```
alter table dbo.TStudent
add
constraint DF_Class default '不知道'
for Class
```

实现参照完整性

在插入和删除数据时，引用完整性用于维持参照表和被参照表之间的数据一致性。

给成绩表插入记录，studentID 必须在学生表中存在，成绩表 studentID 列参照学生表的 StudentID,如果该学生在学生表中删除,则自动在成绩表中删除,如果该学生 StudentID 变化,则在成绩表中也自动更改.

```
ALTER TABLE [dbo].[TScore] WITH CHECK ADD CONSTRAINT
[FK_TScore_TStudent] FOREIGN KEY ([StudentID])
REFERENCES [dbo].[TStudent] ([StudentID])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
级联更改
级联删除
```

使约束失效

添加约束时对现有数据的约束检查失效，该功能仅适用于 check 约束和 foreign key 约束。

添加 check 约束 不对现有数据检查

```
ALTER TABLE [dbo].[TStudent] WITH NOCHECK ADD CONSTRAINT [CK_Sex] CHECK
(( [sex]='女' OR [sex]='男' ))
```

添加新数据时，使约束失效

```
ALTER TABLE [dbo].[TStudent] NOCHECK CONSTRAINT [CK_Sex]
ALTER TABLE [dbo].[TScore] NOCHECK CONSTRAINT [FK_TScore_TStudent]
```

使外键约束生效

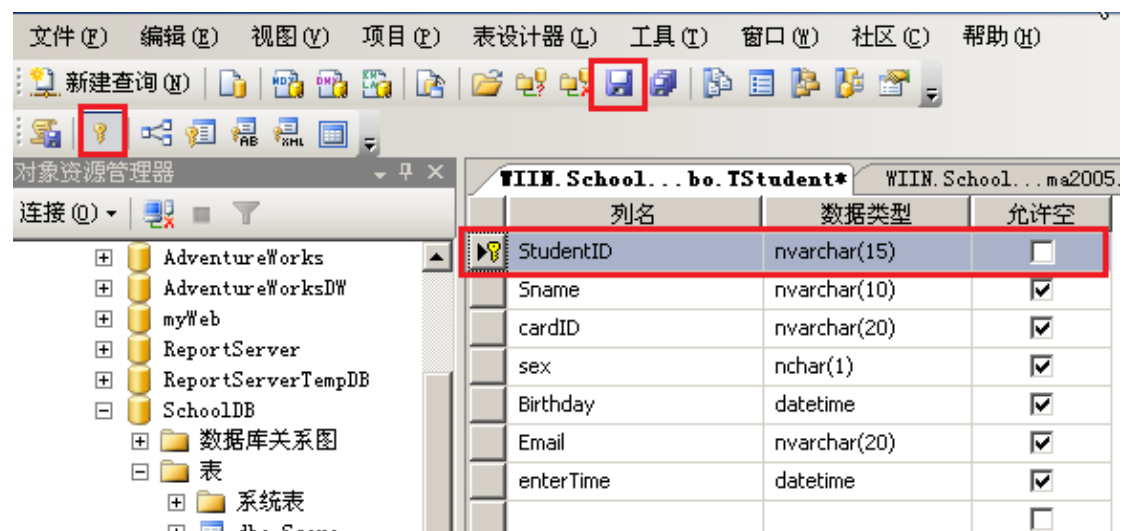
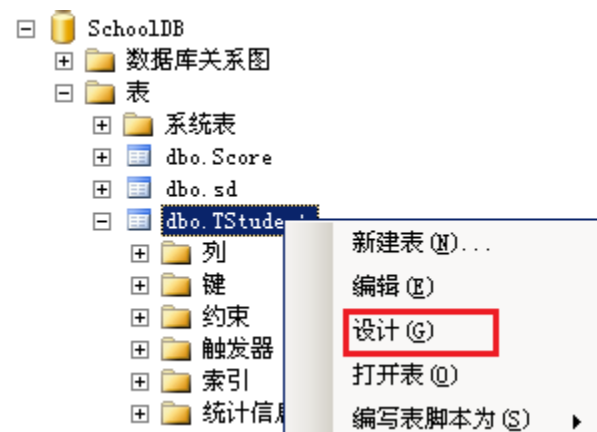
```
ALTER TABLE [dbo].[TScore] CHECK CONSTRAINT [FK_TScore_TStudent]
```

主键约束和唯一性约束不能够失效

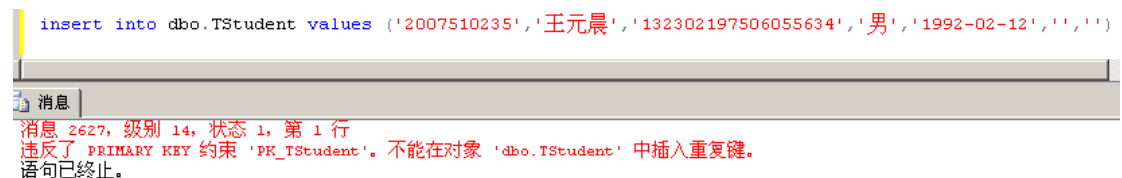
Check 约束 默认值约束 外键约束

实现数据完整性的参考步骤

创建主键约束

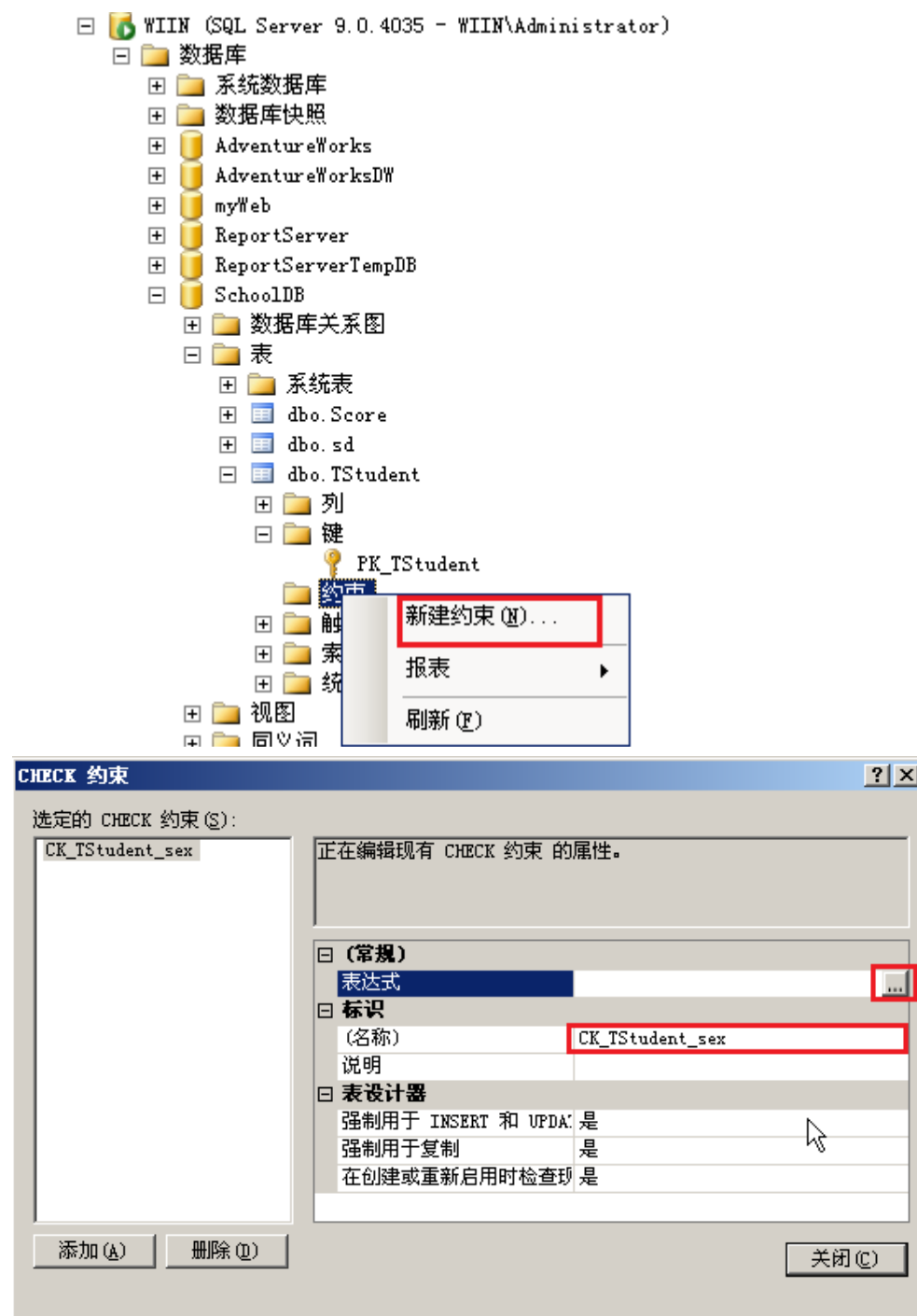


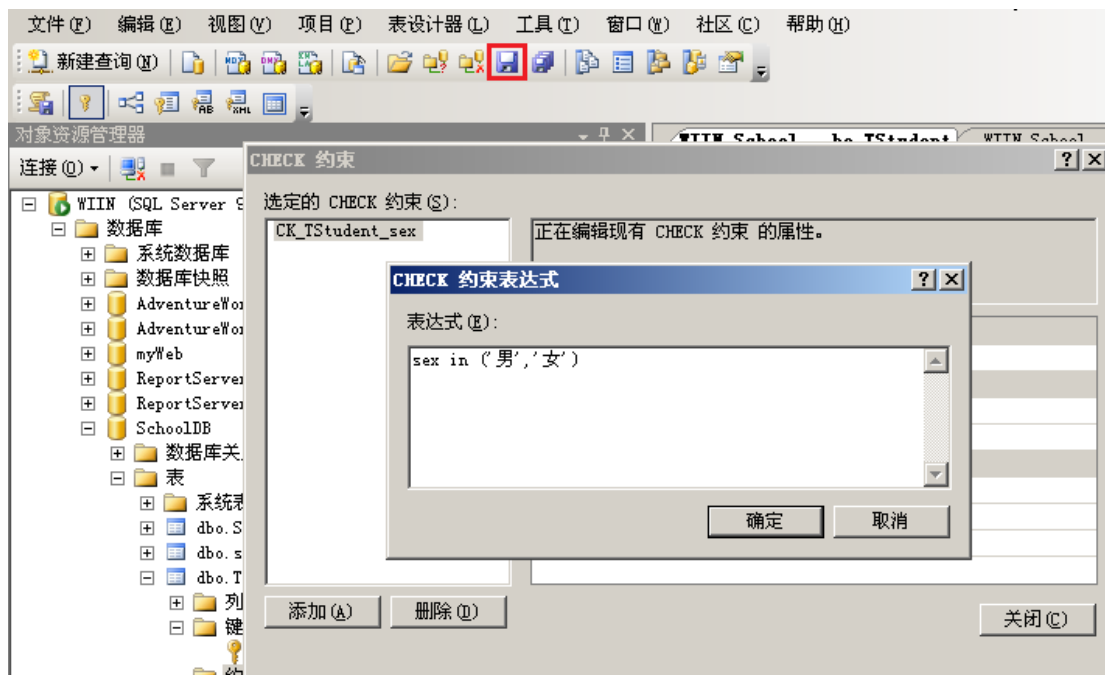
再次插入学号相同的记录失败



创建 check 约束

规定列能够接受的值，性别列只能接受“男”和“女”

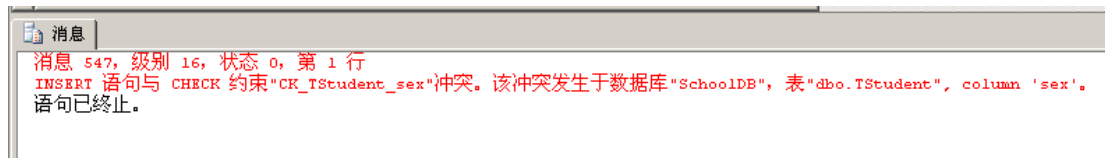




点击保存

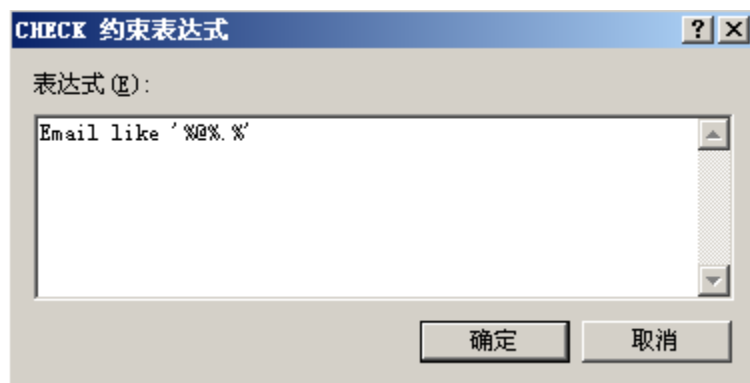
插入一条记录，性别为“南”

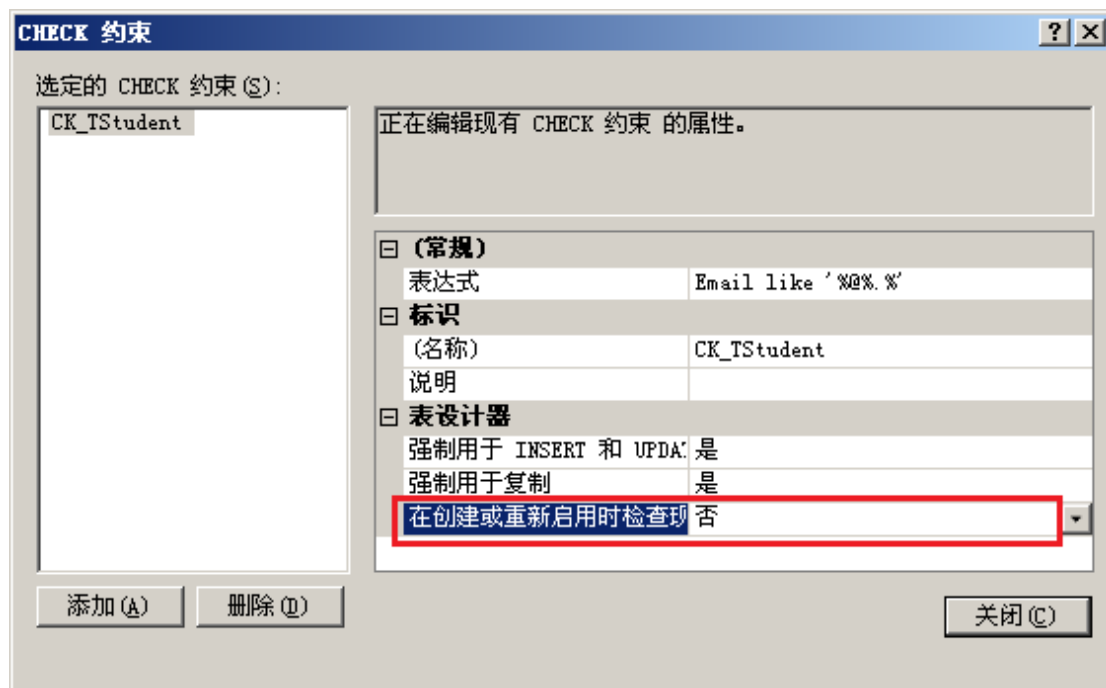
```
insert into dbo.TStudent values ('2007510236','王元晨','南','132302197506055634','1992-02-12','','')
```



输入表达式 Email like '%@%.%'

不检查现有记录是否满足约束





% 0 个或多个字符串

_ 任何单个的字符

[] 在指定区域或集合内的任何单个字符

[^] 不在指定区域或集合内的任何单个字符

下面语句能够给成功插入一条记录

```
insert into dbo.TStudent values ('2007510326','王元晨','男',  
'132302197506055634','1992-02-12','wangyuanc@sohu.com','','')
```

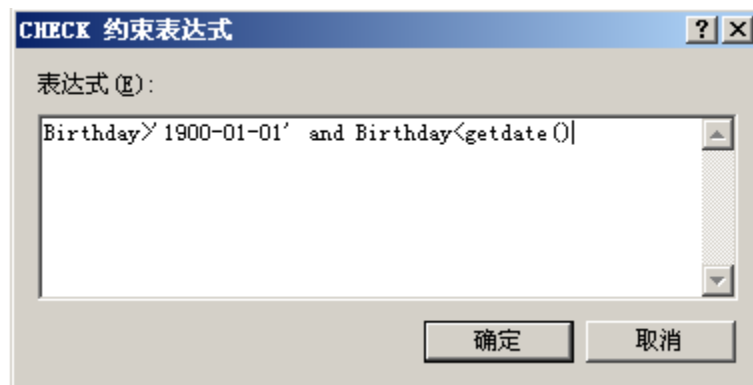
下面语句将会插入失败

```
insert into dbo.TStudent values ('2007510326','王元晨','男',  
'132302197506055634','1992-02-12','wangyuancsohu.com','','')
```

以下语句也会失败

```
insert into dbo.TStudent values ('2007510326','王元晨','男',  
'132302197506055634','1992-02-12','wangyuanc@sohucom','','')
```

创建约束 生日必须在 1900-01-01 至今



插入以下记录都会失败

```
insert into dbo.TStudent values ('2007510326', '王元晨', '男',  
    '132302197506055634', '2022-02-12', 'wangyuanc@sohu.com', '', '')  
insert into dbo.TStudent values ('2007510326', '王元晨', '男',  
    '132302197506055634', '1982-02-12', 'wangyuanc@sohu.com', '', '')
```

默认值

WIIN.School...bo.TStudent		WIIN.School...ma2005.sql*	
	列名	数据类型	允许空
🔑	StudentID	nvarchar(15)	<input type="checkbox"/>
	Sname	nvarchar(10)	<input checked="" type="checkbox"/>
	cardID	nvarchar(20)	<input checked="" type="checkbox"/>
	sex	nchar(1)	<input checked="" type="checkbox"/>
	Birthday	datetime	<input checked="" type="checkbox"/>
	Email	nvarchar(20)	<input checked="" type="checkbox"/>
▶	enterTime	datetime	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

列属性	
📁 (常规)	
(名称)	enterTime
默认值或绑定	getdate()
数据类型	datetime
允许空	是
📁 表设计器	
RowGuid	否
📁 标识规范	
标识规范	否
不用于复制	否

点击“保存”

```
insert into dbo.TStudent  
(StudentID, Sname, sex, cardID, Birthday, Email)  
values ('2007510276', '王 晨', '男', '132302197506055634',
```

'2011-02-12', 'wang@edu2act.org')

查看记录

```
select * from dbo.TStudent where Sname='王晨'
```

发现有时插入时间

课上练习

在 Score 表上创建约束，分数在 0-150 之间，并验证。

实现参照完整性

要求 Score 表中的学号必须在 TStudent 有

WIIN (SQL Server 9.0.4035 - WIIN\Administrator)

- 数据库
 - 系统数据库
 - 数据库快照
 - AdventureWorks
 - AdventureWorksDW
 - myWeb
 - ReportServer
 - ReportServerTempDB
 - SchoolDB
 - 数据库
 - dbo. (右键菜单: 新建表(N)..., 编辑(E), 设计(G), 打开表(O), 编写表脚本为(S), 查看依赖关系(V), 全文索引(I), 报表)
 - 表
 - 系统
 - dbo.
 - 视图
 - 同义词

Microsoft SQL Server Management Studio

文件(F) 编辑(E) 视图(V) 项目(P) 表设计器(D) 工具(T) 窗口(W) 社区(C) 帮助(H)

对象资源管理器

连接(C) WIIN (SQL Server 9.0.4035 - WIIN\Administrator)

WIIN.SchoolDB - dbo.Score

列名	数据类型	允许空
StudentID	nvarchar(10)	<input checked="" type="checkbox"/>
math	int	<input checked="" type="checkbox"/>
english	int	<input checked="" type="checkbox"/>
history	int	<input checked="" type="checkbox"/>
		<input type="checkbox"/>


外键关系 [?] [X]

选定的 关系 (S):

FK_Score_Score*

正在编辑新的 关系 的属性。 需要先填充“表和列规范”属性，
然后才能接受新的 关系。

☐ (常规)

☒ 表和列规范 

在创建或重新启用时检查 是

☐ 标识

(名称) FK_Score_Score

说明

☐ 表设计器

☒ INSERT 和 UPDATE 规范

强制外键约束 是

强制用于复制 是

添加 (A) 删除 (D) 关闭 (C)

表和列

关系名 (N):
FK_Score_TStudent

主键表 (P): TStudent 外键表: Score

StudentID	StudentID
-----------	-----------

确定 取消

表和列

关系名 (N):
FK_Score_TStudent

主键表 (P): TStudent 外键表: Score

StudentID	StudentID
-----------	-----------

确定 取消

设置级联删除和级联更新

外键关系

选定的 关系 (S):
FK_Score_TStudent

正在编辑现有 关系 的属性。

在创建或重新启用时检查约束 是

☒ 标识
(名称) FK_Score_TStudent
说明

☒ 表设计器

☒ INSERT 和 UPDATE 规范

更新规则	层叠
删除规则	层叠
强制外键约束	是
强制用于复制	是

添加 (A) 删除 (D) 关闭 (C)

点击保存,

删除 Score 表中的记录

```
delete dbo.TScore where StudentID=2007510235
```

在 Score 表中插入一条记录

```
insert into dbo.TScore values ('2007510235','0001',78)
```

--能够插入成功，因为 TStudent 表中有该学号

```
insert into dbo.TScore values ('2007510245','0001',78)
```

--不能够插入成功，因为 TStudent 表中没有该学号

```
select * from dbo.TScore where studentID='2007510235'
```

--查看插入的记录

```
update      dbo.TStudent      set      StudentID='2009223400'      where  
StudentID='2007510235'
```

--更新主表的记录

```
select * from dbo.TScore where studentID='2009223400'
```

--再次查看插入的记录 发现学号已经变化

删除主表记录

```
delete dbo.TStudent where StudentID='2009223400'
```

--再次查看 Score 表中记录发现已经被级联删除

```
select * from dbo.TScore where StudentID='2009223400'
```