

数据库实验lab10

曹广杰

15352015 数据科学与计算机

授课教师：刘玉葆

Content

数据库实验lab10

Content

实验目的

句法汇总

实验内容

为table添加约束

检验约束信息

为table添加约束冲突

为table添加规则

检验规则信息

基于约束添加规则

实验目的

1. 学习用户自定义约束，并实践用户完整性；
2. 利用短语NOT NULL, UNIQUE, CHECK保证用户定义完整性。

句法汇总

1. 为table设置主键以及设置主键的要求。
2. varchar与char的不同。
3. 设置主键的语法格式。
4. 创建事务与设置回滚选项。
5. 外键的建立条件。

实验内容

为table添加约束

在worker中加入约束U3，令sage值大于等于0。

```
1 alter table worker
2 add constraint rule_age check (sage >= 0)
```

句法总结：

添加约束: `add constraint` 约束名 `check` 约束条件

检验约束信息

为了检验之前的约束信息是否有效, 这里添加两个数据元组, 其中一个不符合要求:

```
1 insert into worker
2 values('00004', 'ZONE', 'm', '-1', 'magic')
3
4 insert into worker
5 values('00005', 'Flash', 'm', '23', 'trans')
```

这里插入闪电侠的数据和他的死对头zone的数据。由于zone的年龄不符合条件, 故而添加操作不能成功:

```
1 消息 513, 级别 16, 状态 0, 第 19 行
2 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
   'School', 表 'dbo.worker', 列 'sage'。
```

而年轻的闪电侠flash的数据可以插入到当前的table中。

为table添加约束冲突

加入约束U4, 令sage值<0, 观察执行是否成功, 分析原因。

```
1 alter table worker
2 add constraint U8 check (sage < 0)
```

此处的约束添加操作与之前的约束添加操作冲突——之前的操作已经限定了sage的信息, 这里的添加操作只允许对这个部分进行强化。

添加操作之后会报错:

```
1 消息 547, 级别 16, 状态 0, 第 10 行
2 ALTER TABLE 语句与 CHECK 约束"U8"冲突。该冲突发生于数据库"School", 表"dbo.worker", column
   'sage'。
```

为table添加规则

加入规则R2, 确保插入的记录的sage值在1到100之间, 并绑定到sage属性。

```
1 go
2 create rule R2
3 as @value in (0, 100)
4 go
5 exec sp_bindrule R2, 'worker.[sage]'
```

需要先创建rule, 才可以将其添加到worker表中。

句法:

- 创建规则: `create rule` 规则名 `as` "对@value的操作";
- 绑定到属性: `exec sp_bindarule` 规则名, `'table名.[规则约束的属性]';`

- 区间操作: `@value in (下限, 上限);`

检验规则信息

在前文中我们添加对于年龄的限定信息, 超过100岁的不能入档。所以此处笔者添加一个违反当前要求的元组。尝试将太乙真人纳入魔法部:

```
1 insert into worker
2 values('00006', 'Taiyi', 'm', '1000', 'magic')
```

操作结果显示失败, 因为太乙真人存在的时间太长了:

```
1 消息 547, 级别 16, 状态 0, 第 10 行
2 ALTER TABLE 语句与 CHECK 约束"R2"冲突。该冲突发生于数据库"School", 表"dbo.worker", column
  'sage'。
```

解除rule之后进行操作就可以了:

```
1 exec sp_unbindrule 'worker.[sage]'
2 drop rule R2
```

句法总结:

- 添加规则: `go create rule 规则名`
- 移除规则: `drop rule 规则名` (需要解绑之后才能移除)
- 绑定规则: `exec sp_bindrule 规则名, 对应的table的属性上`
- 解绑规则: `exec sp_unbindrule 'table名.[对应的属性]'`

基于约束添加规则

已知示例三中已插入sage为38的记录, 那么加入规则R3, 令sage大于50。观察加入规则R3的操作是否能成功。

```
1 go
2 create rule R3
3 as @value > 50
4 go
5 exec sp_bindrule R3, 'worker.[sage]'
```

可以成功, 该操作与之前的限定年龄信息大于等于0并不冲突, 是强化操作。