当用户中添加时候，作为表也会添加并绑定角色

CREATE TRIGGER trg\_用户\_插入\_作为

ON [education].[dbo].[用户]

AFTER INSERT

AS

BEGIN

DECLARE @用户名 VARCHAR(50), @角色ID INT, @用户ID INT;

-- 遍历所有新插入的用户记录

DECLARE inserted\_cursor CURSOR FOR

SELECT 用户ID, 用户名

FROM INSERTED;

OPEN inserted\_cursor;

FETCH NEXT FROM inserted\_cursor INTO @用户ID, @用户名;

-- 遍历每一条插入的记录

WHILE @@FETCH\_STATUS = 0

BEGIN

-- 根据用户名或其他标识来判断角色ID

IF @用户名 = '管理员'

BEGIN

SET @角色ID = 1; -- 管理员

END

ELSE IF @用户名 = '导师'

BEGIN

SET @角色ID = 2; -- 导师

END

ELSE IF @用户名 = '考生'

BEGIN

SET @角色ID = 3; -- 考生

END

ELSE IF @用户名 = '学科负责人'

BEGIN

SET @角色ID = 4; -- 学科负责人

END

ELSE IF @用户名 = '学科秘书'

BEGIN

SET @角色ID = 5; -- 学科秘书

END

ELSE IF @用户名 = '监督成员'

BEGIN

SET @角色ID = 6; -- 监督成员

END

ELSE IF @用户名 = '研究生管理秘书'

BEGIN

SET @角色ID = 7; -- 研究生管理秘书

END

ELSE IF @用户名 = '外部用户'

BEGIN

SET @角色ID = 8; -- 外部用户

END

ELSE IF @用户名 = '访客'

BEGIN

SET @角色ID = 9; -- 访客

END

ELSE IF @用户名 = '研究生主管领导'

BEGIN

SET @角色ID = 10; -- 研究生主管领导

END

ELSE

BEGIN

-- 如果无法匹配角色，设置默认角色ID

SET @角色ID = 3; -- 默认角色为考生

END

-- 检查"作为"表中是否已有该用户和角色的绑定记录，避免重复插入

IF NOT EXISTS (SELECT 1 FROM [education].[dbo].[作为] WHERE 用户ID = @用户ID AND 角色ID = @角色ID)

BEGIN

-- 在"作为"表中插入一条记录，绑定用户和角色

INSERT INTO [education].[dbo].[作为] (用户ID, 角色ID)

VALUES (@用户ID, @角色ID);

END

-- 继续处理下一个用户

FETCH NEXT FROM inserted\_cursor INTO @用户ID, @用户名;

END

CLOSE inserted\_cursor;

DEALLOCATE inserted\_cursor;

END;

测试：

INSERT INTO [education].[dbo].[用户] (用户ID,用户名, 更新时间, 创建时间, 角色ID, 邮箱, 手机号, 姓名, 密码)

VALUES (42,'导师', CURRENT\_TIMESTAMP, CURRENT\_TIMESTAMP, NULL, 'admin@example.com', '1234567890', '管理员姓名', 'admin\_password');



导师如果增添或者导师的学院ID更该 属于表也会发生相应的变化

CREATE TRIGGER trg\_导师\_插入\_更新\_属于

ON [education].[dbo].[导师]

AFTER INSERT, UPDATE

AS

BEGIN

DECLARE @导师ID INT, @学院ID INT;

-- 获取插入或更新的导师ID和学院ID

SELECT @导师ID = 导师ID, @学院ID = 学院ID

FROM INSERTED;

-- 如果是插入操作

IF EXISTS (SELECT 1 FROM INSERTED WHERE 导师ID = @导师ID)

BEGIN

-- 如果属于表中没有该导师的记录，则插入新记录

IF NOT EXISTS (SELECT 1 FROM [education].[dbo].[属于] WHERE 导师ID = @导师ID)

BEGIN

INSERT INTO [education].[dbo].[属于] (导师ID, 学院ID)

VALUES (@导师ID, @学院ID);

END

ELSE

BEGIN

-- 如果属于表中已有该导师的记录，更新学院ID

UPDATE [education].[dbo].[属于]

SET 学院ID = @学院ID

WHERE 导师ID = @导师ID;

END

END

END;

测试：

-- 插入新的导师记录

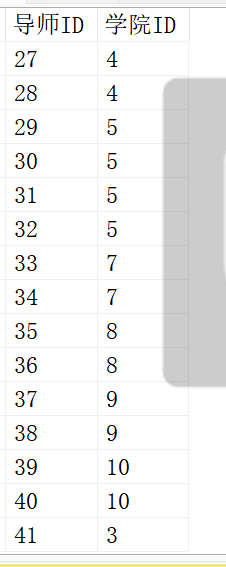
INSERT INTO [education].[dbo].[导师]

(导师ID, 照片路径, 用户ID, 更新时间, 创建时间, 是否具有招生资格, 简介, 职称, 导师类别, 审核状态, 学院ID)

VALUES

(41, 'path/to/photo01.jpg', 41, GETDATE(), GETDATE(), 1, '导师简介', '副教授', '全职', '已通过', 3);

导师ID 学院ID



-- 更新导师记录的学院ID

UPDATE [education].[dbo].[导师]

SET 学院ID = 5

WHERE 导师ID = 41;



招生条目的增改删在编制表中均有体现

-- 创建触发器 trg\_招生目录条目\_增删改\_编制

-- 创建触发器 trg\_招生目录条目\_增删改\_编制

CREATE TRIGGER trg\_招生目录条目\_增删改\_编制

ON [education].[dbo].[招生目录条目]

AFTER INSERT, DELETE, UPDATE

AS

BEGIN

-- 插入操作：新增的招生目录条目会在编制表中增加相应的记录

IF EXISTS (SELECT \* FROM INSERTED)

BEGIN

INSERT INTO [education].[dbo].[编制] (学院ID, 招生目录条目ID)

SELECT 学院ID, 招生目录条目ID

FROM INSERTED;

END

-- 删除操作：删除的招生目录条目会在编制表中移除相应的记录

IF EXISTS (SELECT \* FROM DELETED)

BEGIN

DELETE FROM [education].[dbo].[编制]

WHERE 招生目录条目ID IN (SELECT 招生目录条目ID FROM DELETED);

END

-- 更新操作：更新招生目录条目的学院ID时，编制表中的记录也会被更新

IF EXISTS (SELECT \* FROM INSERTED) AND EXISTS (SELECT \* FROM DELETED)

BEGIN

-- 只删除 DELETED 中的记录

DELETE FROM [education].[dbo].[编制]

WHERE 招生目录条目ID IN (SELECT 招生目录条目ID FROM DELETED);

-- 插入新的记录

INSERT INTO [education].[dbo].[编制] (学院ID, 招生目录条目ID)

SELECT 学院ID, 招生目录条目ID

FROM INSERTED;

END

END;

测试：

-- 插入招生目录条目

INSERT INTO [education].[dbo].[招生目录条目]

(招生目录条目ID,年份, 学院ID, 创建时间, 更新时间, 是否招收同等学力, 学科ID)

VALUES

(32,2024, 3, GETDATE(), GETDATE(), 1, '070503');

学院ID，招生目录条目ID

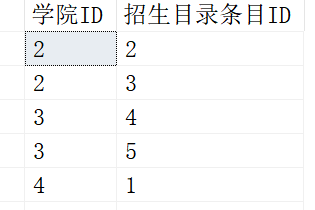


-- 更新招生目录条目

UPDATE [education].[dbo].[招生目录条目]

SET 学院ID = 4

WHERE 招生目录条目ID = 1;



删除目前实现不了，因为招生目录条目有好几个表关联，其他表中有它的外键，不可以随意删除。

总成绩增删改在拥有表中体现

CREATE TRIGGER trg\_总成绩\_增删改\_拥有

ON [education].[dbo].[总成绩]

AFTER INSERT, DELETE, UPDATE

AS

BEGIN

-- 插入操作：当 `总成绩` 表新增记录时，向 `拥有` 表插入对应的关联记录

IF EXISTS (SELECT \* FROM INSERTED)

BEGIN

INSERT INTO [education].[dbo].[拥有] (考生ID, 总成绩ID, 类型)

SELECT 考生ID, 总成绩ID, 类别

FROM INSERTED

WHERE NOT EXISTS (

SELECT 1

FROM [education].[dbo].[拥有]

WHERE 总成绩ID = INSERTED.总成绩ID

);

END

-- 删除操作：当 `总成绩` 表删除记录时，从 `拥有` 表中删除相应记录

IF EXISTS (SELECT \* FROM DELETED)

BEGIN

DELETE FROM [education].[dbo].[拥有]

WHERE 总成绩ID IN (SELECT 总成绩ID FROM DELETED);

END

-- 更新操作：当 `总成绩` 表更新记录时，同步更新 `拥有` 表中相应记录

IF EXISTS (SELECT \* FROM INSERTED) AND EXISTS (SELECT \* FROM DELETED)

BEGIN

-- 删除 DELETED 中对应的旧记录

DELETE FROM [education].[dbo].[拥有]

WHERE 总成绩ID IN (SELECT 总成绩ID FROM DELETED);

-- 插入新的记录

INSERT INTO [education].[dbo].[拥有] (考生ID, 总成绩ID, 类型)

SELECT 考生ID, 总成绩ID, 类别

FROM INSERTED;

END

END;

测试：

-- 插入新记录到总成绩表

INSERT INTO [education].[dbo].[总成绩] (总成绩ID,类别, 总分数值, 考生ID, 更新时间, 创建时间)

VALUES (100,'初试', 85, 1, GETDATE(), GETDATE());



这个还是无法做到删除 因为拥有表有它的外键，拥有表应先删除.