

Частное учреждение образования  
«Колледж бизнеса и права»

УТВЕРЖДАЮ

Заместитель директора  
по учебной работе

\_\_\_\_\_Голубков В.К.

«\_\_\_»\_\_\_\_\_2012

Специальность 2-40 01 01: «Программное обеспечение ин- формационных технологий»	Дисциплина: «Базы данных и системы управления базами данных»
Составлена на основании учебной программы, утвержденной директором Колледжа бизнеса и права 25.11.2011	

**Лабораторная работа № 23**

**Инструкционно-технологическая карта**

Тема: Создание, управление и программирование триггеров

Цель работы: научиться проектировать, кодировать и отлаживать триггеры с применением хранимых процедур для получения информации о триггерах базы данных

Время выполнения: 2 часа

**Краткие теоретические сведения**

**Триггер** – это специальный тип хранимой процедуры, которая автоматически выполняется при возникновении некоторого события (попытке выполнить операции удаления, добавления, редактирования). Как и любой объект базы данных триггер создается с помощью команды CREATE.

**Синтаксис команды создания триггера**

```
CREATE TRIGGER <имя триггера>  
ON <имя представления или таблицы>  
[WITH INCRYPTION]  
{ { {FOR|AFTER}<[DELETE][,][INSERT] [,][UPDATE]> }|INSTEAD OF}  
[WITH APPEND]  
[NOT FOR REPLICATION]  
AS  
<SQL –ОПЕРАТОР  
....  
....  
....>
```

где ON – имя объекта, для которого триггер используется;

WITH INCRYPTION – кодирует текст представления;

WITH APPEND – используется для совместимости с версией 6.5;

NOT FOR REPLICATION – меняет правила запуска триггера. Такой триггер не будет стартовать при выполнении над таблицей операций связанных с репликацией данных.

При срабатывании триггера создаются таблицы **INSERTED**, **DELETED**:

**INSERTED** – для хранения добавленных записей;

**DELETED** – для хранения удаленных записей.

Таблицы видимы только для триггера и существуют только при выполнении триггера.

Таблица может иметь произвольное количество триггеров любых типов (**INSERT**, **UPDATE**, **DELETE**). По умолчанию триггер выполняется, когда изменение данных завершено; если же указать опцию **INSTEAD OF**, то создаётся триггер, выполняющийся вместо изменения данных.

Изменить триггер можно с помощью оператора **ALTER TRIGGER**.

В Microsoft SQL Server используются два типа триггеров **AFTER (после)** и **INSTEAD OF (вместо)**, а также три их типа: **INSERT**, **UPDATE**, **DELETE**. Триггеры вставки **INSERT** – стартуют каждый раз при добавлении в таблицу новой записи при этом создается таблица **INSERTED**. Триггеры удаления **DELETE** – стартуют каждый раз при удалении из таблицы записи и, как следствие этого, создается таблица **DELETED**. Триггер правки **UPDATE** – стартует при внесении изменений в существующие записи таблицы, так как при правке выполняются две операции удаления и вставки, то и служебных таблиц создается две **INSERTED** и **DELETED**.

Триггеры не имеют параметров и не выполняются явно, это значит, что триггер запускается только при попытке изменения данных, событие, заставляющее триггер выполнять свои действия, называется разрешающим событием и обычно говорят, что оно запускает триггер.

По умолчанию все триггеры (**INSERT**, **UPDATE**, **DELETE**) срабатывают после выполнения оператора изменения данных – это триггеры **AFTER**. Триггеры **AFTER** не используются для представлений. Кроме того, в SQL Server используются триггеры **INSTEAD OF**, которые выполняются вместо оператора предполагаемого изменения данных. Триггеры всегда составляют часть транзакции. Если триггер (или другая часть транзакции) терпит неудачу, то транзакция отменяется.

Существует как минимум три причины, по которым используются триггеры.

Во-первых, триггер может быть запущен независимо от разрешившего его приложения для отмены подозрительных действий.

Во-вторых, триггер может контролировать ограничения целостности. В прошлом триггеры были единственным средством обеспечения ссылочной целостности.

В-третьих, триггеры, позволяют поддерживать некоторую избыточность данных в базе данных. Считается, что избыточность в базе данных значительно снижает ее надежность и эффективность, и, тем не менее, все-таки с ней приходится сталкиваться.

Триггеры можно включать и выключать с помощью команды ALTER.

#### Синтаксис команды

ALTER TABLE <имя таблицы>

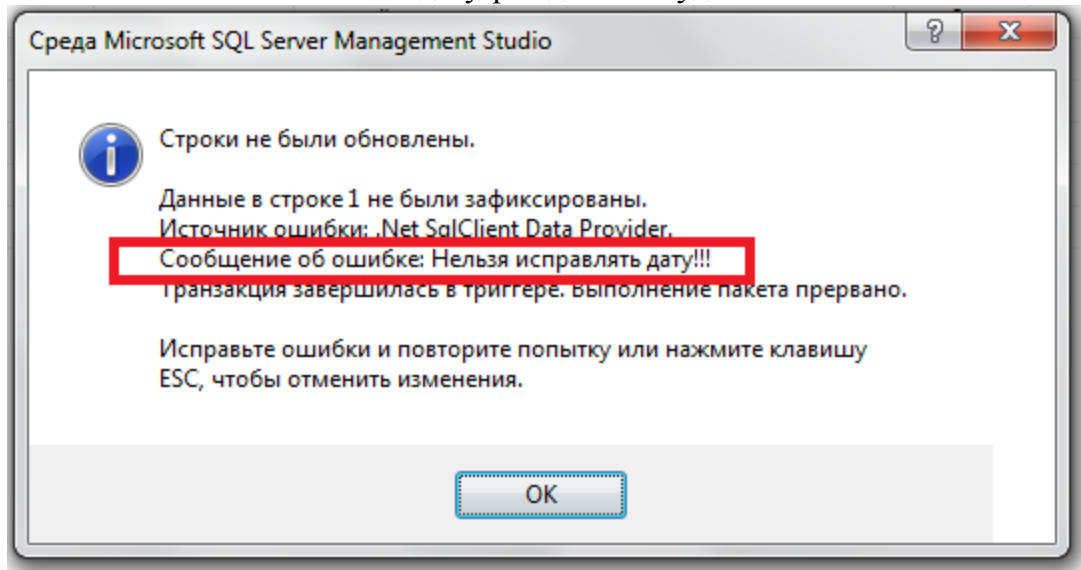
<ENABLE|DISABLE> TRIGGER <ALL|<имя триггера>>

**Пример №1.** Написать триггер, запрещающий корректировать данные о дате рождения в таблице «Студенты».

Решение.

```
create trigger Запрет_на_изменение_даты_рождения on Студенты
for update
as
if EXISTS
(select 'true'
from inserted I, deleted D
where I.[Код студента]=D.[Код студента] and (I.[Дата рождения]<D.[Дата рождения]
| or I.[Дата рождения]>D.[Дата рождения]))
begin
RAISERROR('Нельзя исправлять дату!!!',16,1)
-- Откат транзакции в случае возникновения ошибки
ROLLBACK TRAN
END
```

Теперь любая попытка изменить дату рождения студентов вызовет сообщение:



**Пример №2.** Написать триггер, запрещающий изменять оценку в таблице «Оценки» на оценку выше, при этом учесть, что применяется пятибалльная система оценки.

Решение.

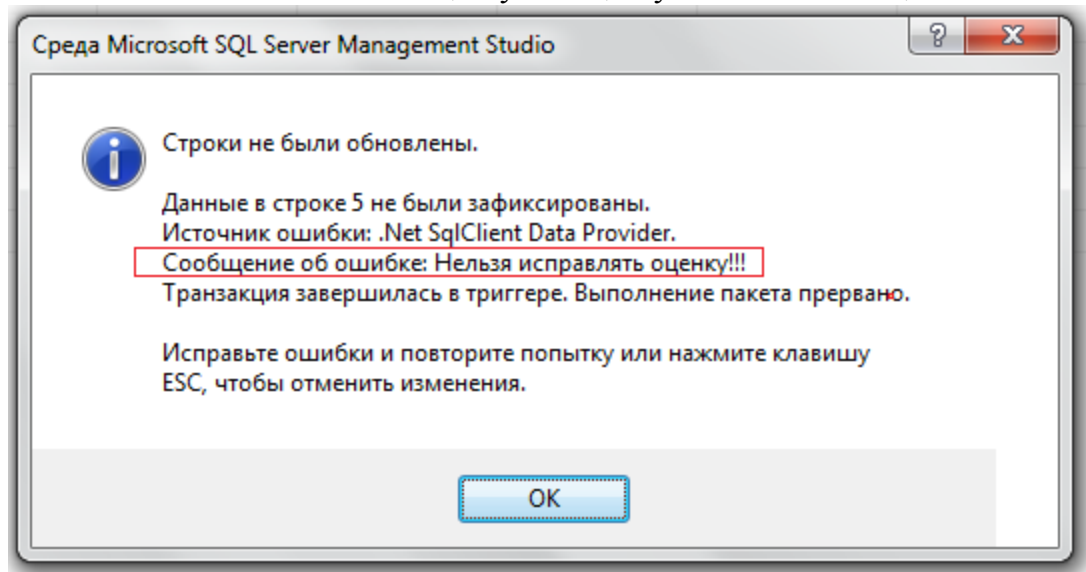
```

create trigger Запрет_на_изменение_оценки on Оценки
for update,delete,insert
as
if exists

(select 'true'
 from inserted I, deleted D
 where I.[Оценка 1]>D.[Оценка 1] or I.[Оценка 1]>5)
begin
    RAISERROR('Нельзя исправлять оценку!!!',16,1)
-- Откат транзакции в случае возникновения ошибки
    ROLLBACK TRAN
END

```

Теперь любая попытка изменить оценку на оценку вызовет сообщение об ошибке



**Пример №3.** Написать триггер, запрещающий изменять курс на значение меньшее существующего.

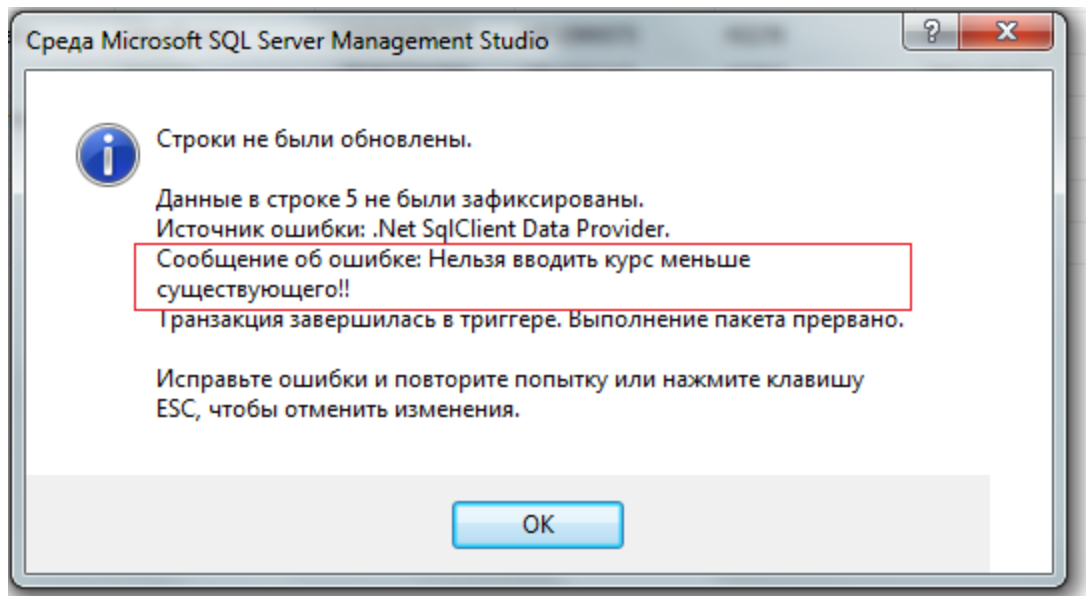
Решение.

```

create trigger Запрет_на_изменение_курса on Студенты
for update
as
if exists
(select 'true'
 from inserted I, deleted D
 where I.[Код студента]=D.[Код студента] and I.Курс<D.Курс)
begin
    RAISERROR('Нельзя вводить курс меньше существующего!!!',16,1)
-- Откат транзакции в случае возникновения ошибки
    ROLLBACK TRAN
end

```

Теперь любая попытка изменить курс на значение меньшее за существующее вызовет сообщение об ошибке:



### Порядок выполнения работы

1. Изучить теоретический материала.
2. Выполнить все примеры и проверить результаты запросов.
3. Получить у преподавателя задание для индивидуальной работы.
4. Ответить на контрольные вопросы.

### Контрольные вопросы

1. Дайте определение понятию триггер.
2. Опишите синтаксис объявления триггера.
3. Назовите и опишите таблицы, которые создаются при срабатывании триггера.
4. Назовите команду создания и изменения триггера.
5. Назовите команду отключения и включения триггера.
6. Назовите и опишите два типа триггеров в MS SQL Server.
7. Перечислите и опишите основные свойства триггера.
8. Назовите три причины, по которым используются триггеры.

Преподаватель

С.В. Бурко

Рассмотрено на заседании цикловой  
комиссии программного обеспечения  
информационных технологий №10  
Протокол № от « » \_\_\_\_\_ 201\_  
Председатель ЦК К.О. Якимович