# ЛАБОРАТОРНАЯ РАБОТА №12. СОЗДАНИЕ, ПРОГРАММИРОВАНИЕ И УПРАВЛЕНИЕ ТРИГГЕРАМИ

Изучение назначения и типов триггеров, условий их активации, синтаксиса и семантики команд языка Transact – SQL для их создания, модификации, переименования, программирования и удаления, а также приобретение навыков их проектирования, кодирования и отладки с применением хранимых процедур для получения информации о триггерах базы данных.

**ЗАДАНИЕ**

**Задание 1**. Создать таблицу authsmall из таблицы authors базы данных Pubs и для новой таблицы запрограммировать триггер auth\_del, который будет выводить информацию о попытках удаления и количестве удаляемых строк, выполнив действия:

1. Создание таблицы authsmall c колонками au\_id, au\_fname, au\_lname, phone и копирование в нее данных из таблицы authors:

**SELECT au\_id, au\_fname, au\_lname, phone INTO authsmoll**

**FROM authors**

**PRINT ‘Содержимое таблицы authsmoll:’ SELECT \* FROM authsmall**

**Изображение выглядит как текст

Автоматически созданное описание**

1. Создание и программирование триггера:

**CREATE TRIGGER auth\_del ON authsmall**

**FOR DELETE AS**

**PRINT ‘Попытка удаления’ + STR (@@ POWCOUNT)+**

**‘строк в таблице authsmall’**

**PRINT ‘Пользователь’ + CURRENT\_USER IF CURRENT\_USER <> ‘dbo’**

**BEGIN**

**PRINT ‘Удаление запрещено’ ROLLBACK TRANSACTION END**

**ELSE**

**PRINT ‘Удаление разрешено’**

**Изображение выглядит как текст

Автоматически созданное описание**

1. Тестирование триггера :

**DELETE FROM authsmall WHERE au\_fname = ‘Johnson’ DELETE FROM authsmall WHERE 2\*2=5**

**Задание 2**. Создать триггер auth\_upd для таблицы authsmall, построенный в первом задании, который будет разрешать изменение столбца au\_id этой таблицы всем, кроме владельца dbo, выполнив следующие действия:

1. Создание и программирование триггера:

**CREATE TRIGGER auth\_upd ON authsmall**

**FOR UPDATE AS**

**SET NOCOUNT ON -- не сообщать о завершении команд; PRINT ‘Попытка изменения данных в таблице authsmall’ IF (COLUMNS\_UPDATE () &1)! = 0 -- 1-й столбец;**

**PRINT ‘Изменение столбца au\_id’**

**IF (COLUMNS\_UPDATE () &2)! = 0 -- 2-й столбец;**

**PRINT ‘Изменение столбца au\_fname’**

**IF (COLUMNS\_UPDATE () &4)! = 0 -- 3-й столбец;**

**PRINT ‘Изменение столбца au\_lname’ IF UPDATE (Phone)**

**PRINT ‘Изменение столбца phone’**

**IF ((CURRENT\_USER = ‘dbo’) AND (COLUMNS\_UPDATED()&1)! = 0 -- 1-ый стлбец; BEGIN**

**PRINT ‘Пользователь dbo не может изменять’ + ‘идентификационный номер автора’**

**ROLLBACK TRANSACTION END**

**Изображение выглядит как текст

Автоматически созданное описание**

1. Тестирование триггера:

**UPDATED authsmall SET phone =‘415 986 - 7020’, au\_fname = ‘John’ WHERE au\_lname = ‘Green’**

**UPDATED authsmall SET phone =‘913 843 - 7302’, au\_id = ‘748-126859’ WHERE au\_lname = ‘Smith’**

**Задание 3**. Создать триггер для команд INSERT и UPDATE, запрещающий производить изменения для автора Billy Geitsi, выполнив действия:

1. Создание и программирование триггера:

**CREATE TRIGGER auth\_ ins\_upd ON authsmall**

**FOR INSERT, UPDATE AS**

**IF EXISTS (SELECT \* FROM authsmall -- inserted; WHERE au\_lname = ‘Geitsi’ -- фамилия;**

**au\_fname = ‘Billy’) -- имя; BEGIN**

**PRINT ‘Недопустимо написание кнги’+ ‘автором Billy Geitsi’**

**ROLLBACK TRANSACTION END**

**Изображение выглядит как текст

Автоматически созданное описание**

1. Тестирование триггера:

**UPDATE authsmall SET au\_lname = ‘Geitsi’, au\_fname = ‘Billy’ WHERE au\_lname = ‘Smith’.**