**DDL – триггеры**

**Назначение DDL-триггеров:**

* Запрет изменений схемы БД
* Выполнение автоматической операции-ответ на изменение БД
* Журнализирование схемы БД и ведение лога

**Отличие от DML-триггеров:**

* Нет таблиц inserted, deleted
* Бывает только FOR (AFTER)
* Привязываются к определенной DDL-команде
* Область действия – БД / сервер

DDL-триггер области БД реагирует на DDL-команды БД, в которой он хранится

DDL-триггеры области сервера хранятся в БД master и реагируют на DDL-команды уровня сервера

**Пример:**

1. Создание триггера уровня БД запрещающего удалять и изменять таблицы

CREATE TRIGGER TrDenyTableChanges

ON DATABASE

FOR ALTER\_TABLE, DROP\_TABLE

AS

BEGIN

PRINT 'Запрет на изменение/удаление'

ROLLBACK --откат к исход. сост.

END

1. Создание триггера уровня сервера запрещающего вход определённому логину

use master

CREATE TRIGGER TrDenyEnterIspp

ON ALL SERVER

WITH EXECUTE AS 'ispp1119'

FOR LOGON

AS

IF (ORIGINAL\_LOGIN() = 'ispp1119')

ROLLBACK

DISABLE TRIGGER TrDenyEnterIspp ON ALL SERVER

ENABLE TRIGGER

CREATE TRIGGER TrLog

ON DATABASE

FOR CREATE\_TABLE, ALTER\_TABLE, DROP\_TABLE

AS

INSERT INTO QueriesLog

SELECT EVENTDATA() --какая то команда сработала

value ('(/EVENT\_INSTANCE / TSQL Command / CommandText)[1]','char(max)')

DDL\_DATABASE\_LEVEL\_EVENTS --перехват всех DDL-команд

**Транзакции**

Транзакции обеспечивают выполнение всех операций, / откат изменений в случае неудачи выполнения какой-либо операции

TCL – Transaction Control Language

BEGIN TRAN ИмяТранзакции

--выполнение SQL-команд

BEGIN TRAN ИмяМетки WITH MARK --указание метки

--выполнение SQL-команд

ROLLBACK TRAN --откат изменений внесённых в БД

SAVE TRAN ИмяТочкиСохранения --создание точки сохранения, к которой можно откатиться

COMMIT TRAN --сохранение изменений

уровни изоляции транзакции: SET TRANSACTION ISOLATION LEVEL