

Тригери

Тригери

- ❑ Програма, която се активира при настъпване на някакво събитие. Тригерите, които ние ще разглеждаме са свързани с DML заявки (INSERT, UPDATE, DELETE, изпълнявани върху таблици и изгледи) и ще са написани на Transact-SQL (скриптовия език, който поддържа MSSQL Server; MSSQL тригери могат да се пишат и на езици поддържани от CLR).
- ❑ При създаването се задава името на таблицата/изгледа, за която е тригера, действието при което се активира, типа и тялото на тригера.
- ❑ За дадена таблица/изглед може да има много AFTER тригери за едно и също действие. Може да се указва кой да е тригера, който да се изпълни пръв и кой да се изпълни последен. Останалите се изпълняват в случаен ред.
- ❑ MSSQL поддържа два типа тригери
 - AFTER тригери - изпълняват се след като действието, което ги активира се е случило
 - INSTEAD OF тригери - изпълняват се вместо действието, което ги активира. Това е удобен начин да направим обновяеми, изгледи, които не са, като императивно (чрез скрипт) укажем какво да се промени в базата

AFTER TRIGGER

- ❑ Стартира се след изпълнение на заявката, която го активира
- ❑ Част е от същата транзакция, в която участва заявката, която го активира. Ако няма експлицитно създадена транзакция - то тогава е част от имплицитно създадената такава от сървъра само за заявката, която го активира.
- ❑ В тялото на тригера може да се използва ROLLBACK и това ще отмени както модификациите направени в тялото на тригера, така и тези от заявката, която го е задействала.
- ❑ В тялото на тригера са достъпни две специални таблици с имена `inserted` и `deleted`. Те имат същата структура, както таблицата, за която е създаден тригера. `Inserted` съдържа новодобавените редове, когато тригера е задействан от `INSERT` заявка или новата версия, на редове обновени от `UPDATE` заявка (за тригери задействани от `UPDATE`). Аналогично `deleted` съдържа изтритите редове (в случай на `DELETE`) или старата версия на обновените редове (в случай на `UPDATE`).

INSTEAD OF TRIGGER

- ❑ Действията описани в тялото на тези тригери се изпълняват вместо действието, което ги задейства.
- ❑ Таблиците `inserted` и `deleted` са достъпни и тук, но сървъра не е извършил реално модификациите в базата (в таблицата, за която е дефиниран тригера). Ако имаме `IDENTITY` колона - няма да имаме генериран нов идентификатор (за разлика от случая с `AFTER` тригер).
- ❑ За дадена таблица и дадено DML действие (`INSERT/UPDATE/DELETE`) може да имаме максимум един `INSTEAD OF` тригер.

Създаване на TRIGGER

```
CREATE TRIGGER trigger_name
ON { table | view }
{ FOR | AFTER | INSTEAD OF }
{ [ INSERT ] [ , ] [ UPDATE ] [ , ] [ DELETE ] }
AS { sql_statement [ ; ] [ ,...n ]
```

- ❑ FOR и AFTER са синоними (и в двата случая става въпрос за AFTER тригери)
- ❑ С ALTER TRIGGER може да променяме дефиницията на тригера
- ❑ С DROP TRIGGER може да изтрием тригер
- ❑ С DISABLE TRIGGER trigger_name ON table_name може временно да деактивираме тригер. След това с ENABLE TRIGGER може да го активираме наново.