**19. Управление транзакциями.**

Существует 2 способа управления транзакциями.

* API баз данных (ODBC, OLE DB, ADO)
* Инструкции языка T-SQL (BEGIN TRANSACTION, COMMIT TRANSACTION, COMMIT WORK, ROLLBACK TRANSACTION, ROLLBACK WORK и SET IMPLICIT\_TRANSACTIONS.)

*Для каждой транзакции следует использовать только* ***один*** *из методов, скажем не следует начинать через API а заканчивать через T-SQL.*

**Режимы транзакций**

[Явные транзакции](https://technet.microsoft.com/ru-ru/library/ms175127(v=sql.105).aspx)

**Запуск:**

* Функция API или инструкция BEGIN TRANSACTION (она же BEGIN TRAN).

**Окончание:**

* Инструкция COMMIT гарантирует, что все изменения в пределах данной транзакции стали постоянной частью базы данных.
* Инструкция ROLLBACK отменяет все изменения, сделанные в пределах транзакции, возвращая данные в то состояние, в котором они находились на начало транзакции.

[Автоматическая фиксация транзакций](https://technet.microsoft.com/ru-ru/library/ms187878(v=sql.105).aspx)

Режим по умолчанию для Database Engine. Каждая отдельная инструкция языка Transact-SQL фиксируется после завершения.

[Неявные транзакции](https://technet.microsoft.com/ru-ru/library/ms188317(v=sql.105).aspx)

Функция API, либо инструкция SET IMPLICIT\_TRANSACTIONS ON. Следующая инструкция автоматически запускает новую транзакцию. После завершения этой транзакции следующая инструкция языка Transact-SQL запускает новую транзакцию.

Транзакции контекста пакета

Явная или неявная транзакция, которая запускается в сеансе MARS, становится транзакцией контекста пакета. SQL Server автоматически выполняет откат для транзакции контекста пакета, которая не фиксируется и откат которой не был выполнен после завершения пакета.

*Любое окончание транзакции ведет к освобождению используемых ею ресурсов (например блокировок).*

Рассмотрим пример с переводом суммы со счета покупателя на счет продавца. Пусть для простоты сумма будет фиксированная – 100 рублей, данные о счетах находятся в таблице Accounts, у продавца и покупателя уникальные в рамках таблицы имена («Покупатель1» и «Продавец1») и только по одному счету. Тогда транзакция может выглядеть следующим образом:

BEGIN TRAN

UPDATE Accounts SET AccountSum=AccountSum-100 WHERE Name='Покупатель1';

UPDATE Accounts SET AccountSum=AccountSum+100 WHERE Name='Продавец1';

COMMIT TRAN

Если в процессе выполнения подобной транзакции до выполнения оператора COMMIT произойдет сбой электропитания, после процедуры восстановления БД будет возвращена в состояние, предшествующее началу транзакции (подробнее об этом далее в данной главе). Некоторые дополнительные особенности обработки СУБД SQL Server такой последовательности операций будут рассмотрены в лабораторных.