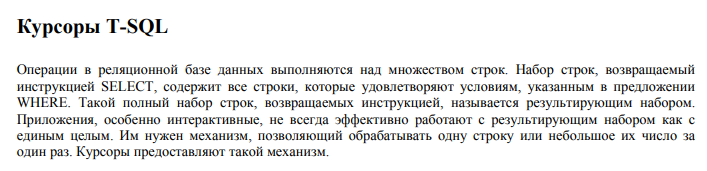


Курсоры



**Курсор – указатель к множеству значений строк.**

**Курсор – результат sql-запроса и указатель.**

**Область видимости:**

* **Локальная**
* **Глобальная**

**Типы:**

* **Static**

*Статический курсор* не изменяется после создания и всегда отображает тот набор данных, который существовал на момент его *открытия*.

* **Dynamic**

На время выборки сервер блокирует строки, а все *изменения*, вносимые пользователем в полный результирующий набор *курсора*, будут видны в *курсоре*. Однако если другой пользователь внес *изменения* уже после выборки данных *курсором*, то они не отразятся в *курсоре* .

* **Key set**
* **Fast forward**

**По способу перемещения:**

* **Forward-only**
* **Scroll**

**По параллельному доступу:**

* **Read-only**
* **Optimistic**
* **Scroll-locks**

**Курсор используется, когда нужно сделать что-то в цикле (н-р удалить объекты)**

*Управление курсором* реализуется путем выполнения следующих команд:

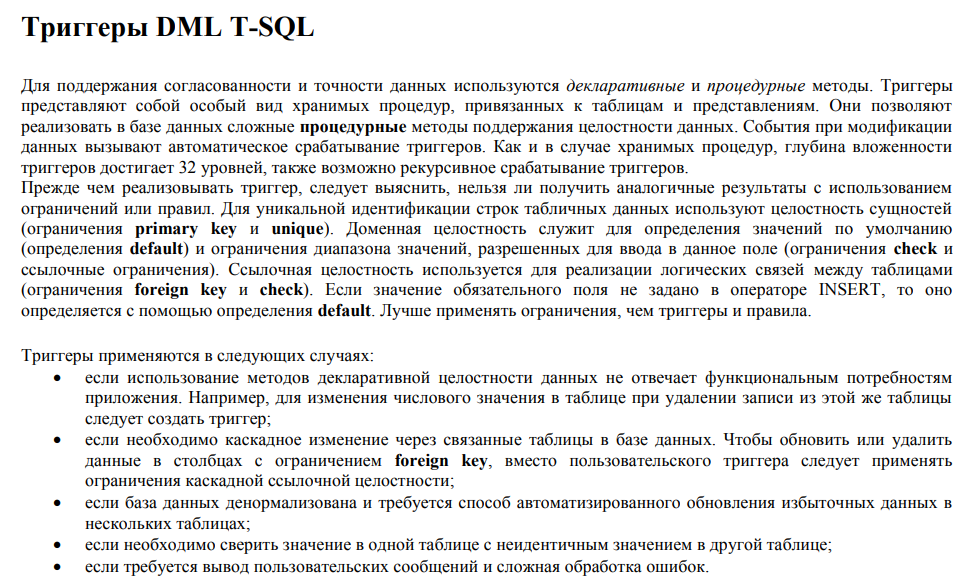
* DECLARE – создание или *объявление курсора* ;
* OPEN – *открытие курсора*, т.е. наполнение его данными;
* FETCH – *выборка из курсора* и *изменение* строк данных с помощью курсора;
* CLOSE – *закрытие курсора* ;
* *DEALLOCATE* – *освобождение курсора*, т.е. удаление курсора как объекта.

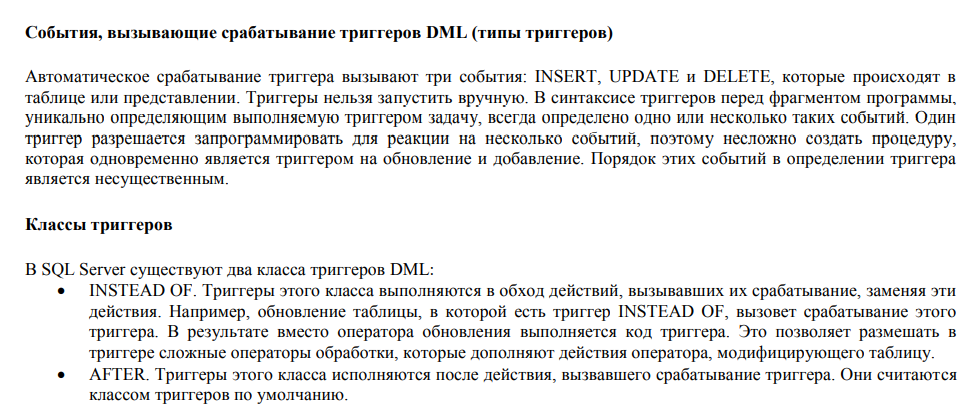
Курсор также отслеживает текущую позицию в результирующем наборе. Можно рассматривать положение курсора как указатель на текущую запись, аналогично тому, как индекс массива указывает на значение в определенном месте в массиве.

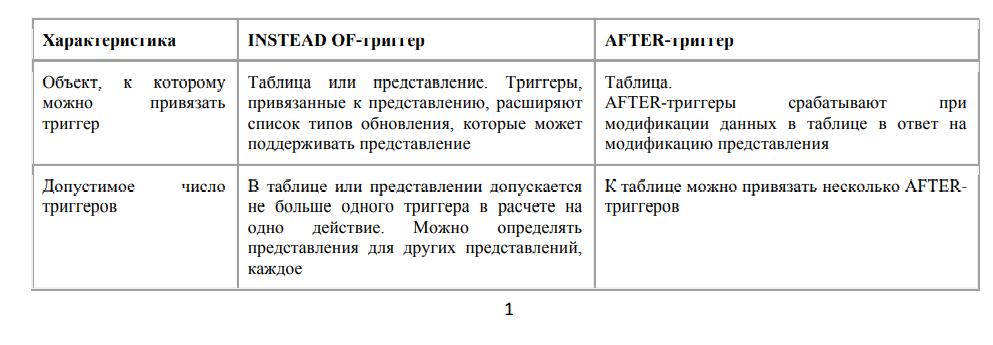
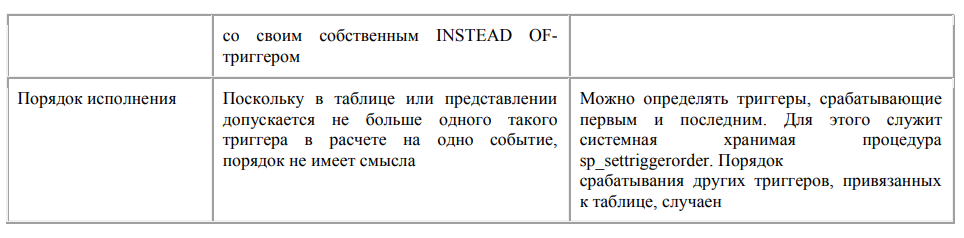
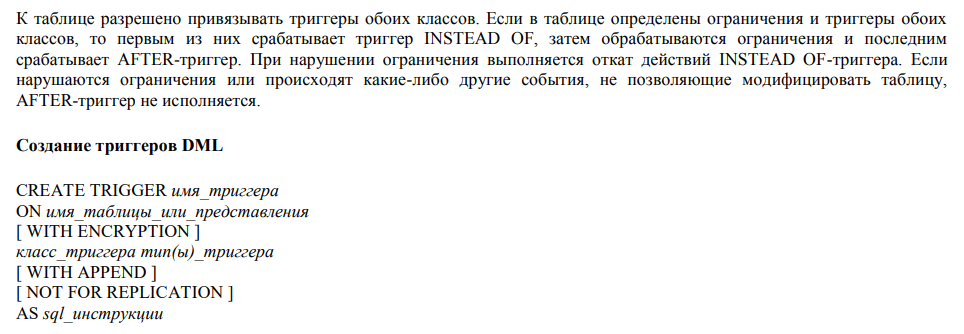
Обычно *курсоры* используются для выбора из *базы данных* некоторого подмножества хранимой в ней информации. В каждый момент времени прикладной программой может быть проверена одна строка *курсора*.

Скорость выполнения операций обработки данных посредством *курсора* заметно ниже, чем у стандартных средств *SQL*.

Курсор — ссылка на контекстную область памяти. В некоторых реализациях языка программирования SQL (Oracle, Microsoft SQL Server) — получаемый при выполнении запроса результирующий набор и связанный с ним указатель текущей записи. бы сказал, что курсор — это виртуальная таблица которая представляет собой альтернативное хранилище данных. При этом курсор, позволяет обращаться к своим данным, как к данным обычного массива.





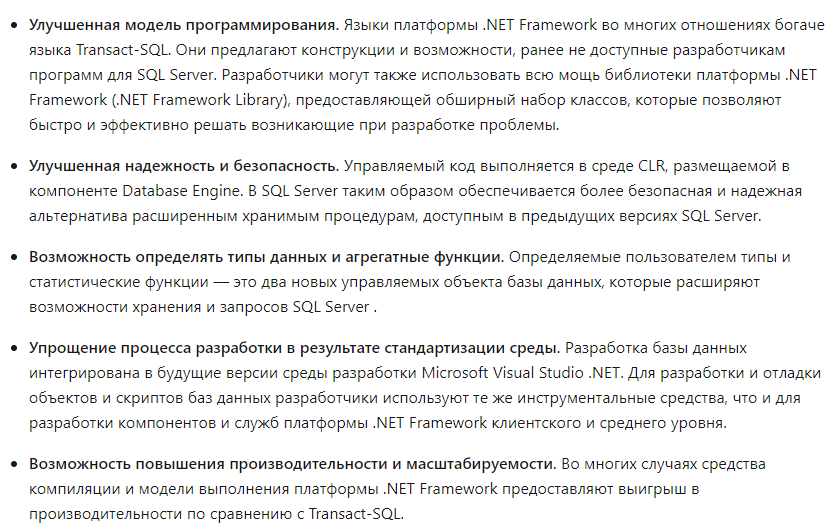
  

**CLR**

Можно использовать языки платформы .NET Framework для определения объектов базы данных, таких как хранимые процедуры и триггеры, и извлечения и обновления данных для баз данных SQL Server. Эти объекты часто называются объектами SQL Server со средой CLR или объектами SQL CLR.

================================================================

Среда CLR предоставляет управляемому коду такие услуги, как межъязыковая интеграция, управление доступом для кода, управление временем существования объекта, а также поддержку отладки и профилирования. Для пользователей и разработчиков SQL Server интеграция со средой CLR означает, что теперь можно писать хранимые процедуры, триггеры, определяемые пользователем типы и функции (скалярные и возвращающие табличное значение), а также определяемые пользователем агрегатные функции на любом языке среды .NET, включая Microsoft Visual Basic и Microsoft Visual C#.



В SQL CLR, как обычно, действуют основные принципы ADO.NET. Для подключения к исходной базе данных нужен экземпляр класса SqlConnection. Однако, хранимую процедуру, которую нужно создать, уже выполняется в базе данных.

в Мiсrоsоft разработана концепция контекстного подключенця (context connection). Контекстное подключение можно создать с помощью такой строки подключения:

"context connection = true"

Тогда следующий экземпляр SqlConnection содержит контекстное подключение – то есть то же подключение, по которому вызвана хранимая процедура:

SqlConnection contextConnection = new SqlConnection("context connection = true") ;