**Курсор** является программной конструкцией, которая дает возможность пользователю обрабатывать строки результирующего набора запись за записью. Курсоры бывают *локальные* и *глобальные* (по умолчанию), *статические* и *динамические* (по умолчанию).

Курсор (cursor) в MS SQL Server — это объект базы данных, который позволяет выполнять итерацию (проход) по результатам выполнения запроса и обрабатывать их по одной записи за раз. Курсоры используются для обработки строк таблицы одну за другой, причем на каждом шаге выполняется определенный набор действий.

Курсор состоит из трех основных элементов:

- Определение и инициализация курсора. В этом элементе определяется запрос для получения данных, которые будут обрабатываться курсором.

- Обработка данных курсором. В этом элементе определяется цикл, который перебирает данные, полученные курсором, и обрабатывает их по одной записи за раз.

- Закрытие курсора. В этом элементе осуществляется закрытие курсора и освобождение всех ресурсов, занятых им.

Курсоры могут быть использованы в случаях, когда нужно обработать данные внутри хранимой процедуры или триггера, а также в приложениях, написанных на языках программирования, которые не могут выполнить итерацию по строкам таблицы. Однако, использование курсоров может привести к снижению производительности при работе с большими объемами данных, поэтому рекомендуется использовать их с осторожностью и только в тех случаях, когда это необходимо.

Курсор объявляется в операторе DECLARE

Оператор FETCH считывает одну строку из результирующего набора и продвигает указатель на следующую строку. Количество переменных в списке после ключевого слова INTO должно быть равно количеству столбцов результирующего набора, а порядок их должен соответствовать порядку перечисления столбцов в SELECT-списке.

После выполнения FETCH проверяется значение функции **@@fetch\_status**, которая возвращает значение 0, если оператор FETCH выполнен успешно; −1, если достигнут конец результирующего набора и строка не считывается; −2, если выбранная строка отсутствует в БД. В зависимости от полученного результата цикл продолжается и считывается следующая строка, или цикл заканчивается.

Курсор закрывается с помощью оператора CLOSE

Оператор FETCH используется в MS SQL для извлечения следующей строки результата запроса, связанного с курсором. Он предназначен для получения данных из курсора по одной строке за раз.

Когда вы создаете курсор в MS SQL, вы можете переместить его на любую строку в наборе результатов. Оператор FETCH позволяет переместить курсор на следующую строку, получить значения столбцов этой строки и продолжить обработку. Таким образом, оператор FETCH позволяет обрабатывать результаты запроса построчно.

FETCH можно использовать с различными параметрами, например, NEXT, PRIOR, FIRST, LAST, ABSOLUTE, RELATIVE, чтобы перемещаться по строкам набора результатов в различных направлениях и на разные расстояния.

В целом, оператор FETCH предоставляет гибкость и контроль над обработкой результатов запроса в курсорах.

Функция @@FETCH\_STATUS используется в SQL Server для проверки статуса последней операции извлечения данных (fetch) в курсоре. Она возвращает целочисленное значение, которое указывает на результат последней операции fetch.

Значение @@FETCH\_STATUS будет равно 0, если операция fetch была успешно выполнена, и -1, если операция fetch не вернула ни одной строки, либо была выполнена неправильно. Оно также может принимать значения 1, 2 или 3, которые указывают на различные виды ошибок, связанных с операцией fetch.

Функция @@FETCH\_STATUS может быть полезна при написании скриптов, использующих курсоры, для проверки наличия и обработки ошибок при извлечении данных из курсора. Например, можно использовать @@FETCH\_STATUS в цикле while, чтобы проверять, были ли извлечены все строки из курсора, и прерывать цикл, если возникла ошибка.

**Cursor local**

Локальный курсор (cursor local) в SQL Server - это тип курсора, который используется для доступа к результатам запроса в пределах одного соединения с базой данных.

Он является наиболее распространенным типом курсора в SQL Server и используется для обхода набора строк в результатах запроса одну за другой.

Локальный курсор может быть установлен только в пределах текущей сессии, и доступ к его результатам невозможен из других соединений с базой данных. Он может быть использован для выполнения операций на результатами запроса в процедурах хранения или скриптах, где необходимо выполнить набор операций на результатах запроса по мере их поступления, одну за другой.

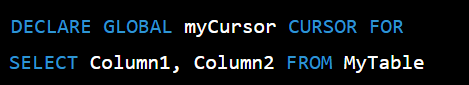
*Локальный* курсор может применяться в рамках одного пакета и ресурсы, выделенные ему при объявлении, освобождаются сразу после завершения работы пакета.

**Cursor global**

*Глобальный* курсор может быть объявлен, открыт и использован в разных пакетах. Выделенные ему при объявлении ресурсы освобождаются только после выполнения оператора DEALLOCATE или при завершении сеанса пользователя.

В Microsoft SQL Server глобальный курсор (cursor global) - это курсор, который может быть использован в любой точке соединения с базой данных, а не только в том соединении, в котором он был создан. Таким образом, глобальный курсор может быть использован в разных соединениях к базе данных.

Для создания глобального курсора необходимо использовать ключевое слово GLOBAL после ключевого слова DECLARE CURSOR, а затем определить имя курсора. Пример:



После создания глобального курсора он может быть использован в любом соединении с базой данных, используя его имя (myCursor в примере выше). Для закрытия глобального курсора используется команда DEALLOCATE GLOBAL:



Глобальные курсоры могут быть полезны в некоторых сценариях, когда необходимо использовать один и тот же курсор в разных частях приложения или в разных соединениях к базе данных. Однако следует помнить, что глобальные курсоры могут потреблять больше ресурсов и могут повлиять на производительность базы данных. Поэтому их следует использовать осторожно и только в случае необходимости.

**Cursor static**

Открытие *статического* курсора приводит к выгрузке результирующего набора во временную таблицу системной БД **TEMPDB**, и все дальнейшие операции осуществляются с этой таблицей.

После открытия курсора все текущие изменения в исходных таблицах не будут отражаться в результирующем наборе.

Статический курсор (static cursor) в MS SQL Server - это тип курсора, который создается на основе статического (заранее определенного) набора данных, и не позволяет изменять данные в наборе данных во время работы курсора.

При создании статического курсора, данные из набора результатов кэшируются на сервере и используются для навигации курсора. Поскольку данные уже находятся в кэше, статический курсор позволяет выполнять быстрый доступ к данным, в том числе по случайным запросам.

Однако статические курсоры требуют больше ресурсов сервера для создания и поддержания, так как данные хранятся в кэше, даже если они больше не используются в запросе. Кроме того, если данные в таблице изменились, то статический курсор не отражает эти изменения, пока он не будет закрыт и снова открыт.

Функция @@cursor\_rows используется для получения количества строк, которые были извлечены или обработаны курсором в текущий момент. Она возвращает количество строк, обработанных последней операцией курсора, независимо от того, были ли они выбраны для возврата.

Функция @@cursor\_rows может быть полезна, если требуется проверить, сколько строк было обработано или извлечено в процессе работы с курсором. Она также может использоваться для оптимизации работы с курсором и повышения производительности, например, путем настройки размера блока данных, которые выбираются курсором за один раз.

**Cursor dynamic**

Если заменить LOCAL STATICна LOCAL DYNAMIC, то изменения будут отражаться в результирующем наборе.

Динамический курсор (cursor dynamic) в MS SQL Server - это курсор, который не хранит данные в оперативной памяти и не блокирует таблицу в момент открытия. Вместо этого он выполняет запрос и возвращает набор результатов. Курсор динамический, так как он может изменяться в зависимости от изменения данных в таблице, которую он обрабатывает.

Динамический курсор создается с помощью оператора DECLARE CURSOR, который задает запрос для выборки данных. После открытия курсора с помощью оператора OPEN, данные могут быть получены с помощью функции FETCH.

Один из основных преимуществ динамического курсора состоит в том, что он позволяет изменять запрос, который используется для выборки данных, после открытия курсора. Это может быть полезно, если требуется изменить условия выборки данных. Кроме того, динамические курсоры обычно более эффективны, чем статические курсоры, поскольку они не хранят данные в оперативной памяти, что уменьшает использование ресурсов сервера.

**Scroll**

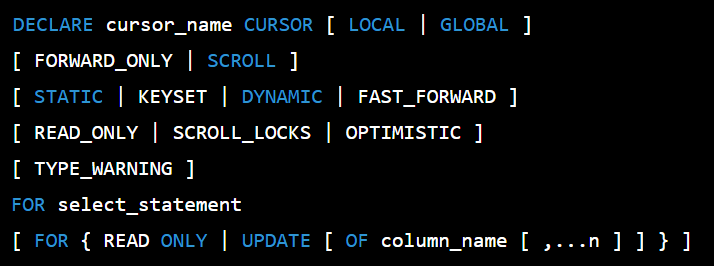
По умолчанию для курсора установлен атрибут SCROLL, позволяющий применять оператор FETCH с дополнительными опциями позиционирования.

Атрибут **scroll** у курсора в Microsoft SQL Server позволяет выполнять прокрутку вперед и назад по строкам результирующего набора данных курсора.

Без этого атрибута курсор можно двигать только в одном направлении - от начала до конца набора данных. Например, если курсор перемещается по таблице, он может только двигаться от первой записи до последней.

Но с атрибутом **scroll**, курсор может перемещаться в любом направлении, итерируя по всем записям результирующего набора несколько раз и выбирая строки в разных порядках.

Кроме того, использование атрибута **scroll** в сочетании с другими опциями курсора может помочь ускорить выполнение операций, связанных с чтением данных.



Ключевые слова, которые могут использоваться после оператора FETCH, включают в себя следующие:

- NEXT - используется для явного указания, что следует извлечь следующую строку из курсора. Если ключевое слово NEXT опущено, то по умолчанию будет выбрана следующая строка.

- FROM cursor\_name - имя курсора, из которого необходимо извлечь строки.

- INTO @variable1 [,...n] - необязательное ключевое слово INTO, которое указывает переменные, в которые должны быть сохранены значения столбцов строки курсора, которая была извлечена. Можно указать одну или несколько переменных через запятую.

- ABSOLUTE n - извлекает строку с указанным номером n.

- RELATIVE n - извлекает строку относительно текущей позиции курсора на указанное количество строк n.

- FIRST - извлекает первую строку в курсоре.

- LAST - извлекает последнюю строку в курсоре.

- PRIOR - извлекает предыдущую строку относительно текущей позиции курсора.

- CURRENT - извлекает текущую строку.

- ALL - извлекает все строки из курсора.

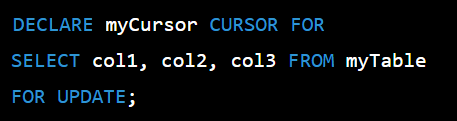
Ключевые слова ABSOLUTE, RELATIVE, FIRST, LAST, PRIOR, CURRENT и ALL могут использоваться только в курсорах, которые имеют атрибут SCROLL. Если курсор не имеет атрибута SCROLL, то FETCH может использоваться только с ключевым словом NEXT.

**For update current of**

Курсоры с установленным свойством FOR UPDATE помимо чтения данных из строк с помощью оператора FETCH, могут эти строки изменять или удалять с помощью операторов UPDATE и DELETE, если в секции WHERE эти операторы используют операцию CURRENT OF, для которой указывается имя курсора (в примере ниже Primer2).

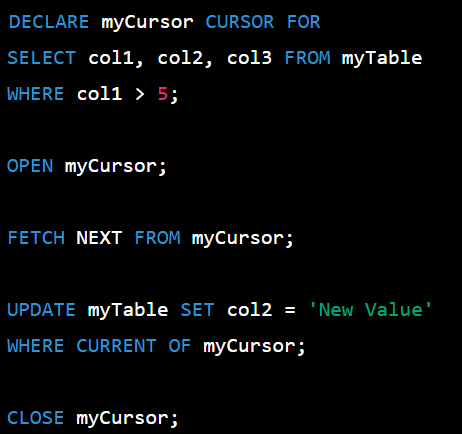
Оператор FOR UPDATE в курсорах указывает, что данные, на которые ссылается текущая запись курсора, могут быть обновлены. Это означает, что другие пользователи не смогут изменять эти данные до тех пор, пока они заблокированы для обновления в текущем курсоре.

Синтаксис оператора FOR UPDATE:



Оператор CURRENT OF используется для обновления или удаления текущей строки, на которую ссылается курсор. Он позволяет выполнить операции обновления и удаления без явного указания ключевых полей, что удобно в некоторых сценариях.

Пример использования CURRENT OF:



В этом примере, после открытия курсора, оператор FETCH NEXT перемещает курсор на первую строку, которая соответствует условию. Затем оператор UPDATE обновляет значение в колонке col2 для текущей строки, на которую ссылается курсор с помощью оператора CURRENT OF.