|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ | | | | | | | | |
|  | | |  | | |  | | |
| ФГБОУ ВО «Пермский государственный  национальный исследовательский университет» | | | | | | | | |
|  | | |  | | |  | | |
|  | | ОТЧЕТ  по лабораторной работе № 3 «Триггеры» по дисциплине «Базы данных и СУБД» | | | | |  | |
|  | | |  | | |  | | |
|  | Работу выполнил  студент гр. \_\_\_\_\_\_\_\_\_\_\_\_\_\_  Зимин И. В.\_\_\_\_\_\_\_\_\_\_\_\_  (подпись)  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2018 | | |  | Проверил  \_\_\_\_\_\_\_\_\_\_\_ кафедры МОВС  (доц., ст. преп., асс.)  Постаногов И. С.\_\_\_\_\_\_\_  (подпись)  «\_\_\_» \_\_\_\_\_\_\_\_ 2018 | | |  |
|  |  | | |  |  | | |  |
| Пермь 2018 | | | | | | | | |

1 Постановка задачи

Решить проблему рассогласованности спец. таблицы и отслеживаемой таблицы в случае, если в последнюю была проведена вставка в обход разработанной ранее хранимой процедуры (ХП) новой записи с идентификатором, превышающим предыдущий максимум, или было проведено обновление идентификаторов, приведшее к превышению максимума. Для этого автоматически создавать триггер для таблицы, имя которой передаётся в ХП в качестве параметра для каждой новой уникальной пары 'имя таблицы' + 'имя столбца', который только в описанных выше случаях превышения обновляет соответствующее текущее максимальное значение в спец. таблице. Имя триггера, помимо осмысленной части, должно содержать GUID.

Провести тестирование корректности работы программы.

Написать отчёт о проделанной работе.

2 Описание алгоритма работы программы

Триггер – [хранимая процедура](https://ru.wikipedia.org/wiki/%D0%A5%D1%80%D0%B0%D0%BD%D0%B8%D0%BC%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D0%B4%D1%83%D1%80%D0%B0) особого типа, которую пользователь не вызывает непосредственно, а исполнение которой обусловлено действием по модификации данных: добавлением INSERT, удалением DELETE строки в заданной таблице, или изменением UPDATE данных в определённом столбце заданной таблицы [реляционной базы данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D1%8B%D0%B5_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85). Триггеры применяются для обеспечения целостности данных и реализации сложной [бизнес-логики](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B7%D0%BD%D0%B5%D1%81-%D0%BB%D0%BE%D0%B3%D0%B8%D0%BA%D0%B0). Триггер запускается сервером автоматически при попытке изменения данных в таблице, с которой он связан.

Напомню, спец таблица, значения в которой будет обновлять создаваемый триггер, имеет 4 столбца: «Идентификатор», «Имя таблицы», «Имя столбца», «Текущее макс. значение»

Таблица 2.1 spec таблица

|  |  |
| --- | --- |
| Имя столбца | Тип данных |
| id | int |
| tableName | nvarchar(50) |
| columnName | nvarchar(50) |
| curMax | int |

При вставке и обновлении значений в таблицах, содержащихся в этой таблице, ранее разработанная ХП не изменяла при надобности текущих значений в спец таблице.

Создадим такой триггер, который после вставки или изменения ячейки в таблице, к которой этот триггер относится, обновляет текущее максимальное значение в таблице spec в записи с соответствующими именем и столбцом таблицы для которой мы создаём триггер.

В уже созданной ранее ХП, которая возвращает следующее за максимальное целое число в столбце определённой таблицы, произведем следующие изменения:

1. Триггер будем создавать только в том случае если на вход нашей ХП были поданы данные, еще не содержащиеся в спец таблице.
2. Название триггера будет содержать в себе «Имя таблицы» и «Имя столбца», а также GUID, который представляет из себя уникальный идентификатор.
3. Осуществим поиск максимального значения из всех вставленных или измененных значений в данном столбце данной таблицы.
4. Если полученное в предыдущем пункте значение превосходит текущее максимальное в соответствующей записи спец таблице, меняем его, иначе оставляем прежним.
5. Так как имена таблиц и столбцов могут быть различными и имена триггеров для каждой таблицы и определённого столбца должно быть различны, мы соберем его по частям и запишем в строку. После чего запустим на выполнение.

3 Особенности реализации на SQL

1. Триггер создается при помощи сочетания ключевых слов CREATE TRIGGER, после которого следует имя триггера, затем ключевое слово ON за которым требуется написать название таблицы, для которой будет создаваться триггер. Далее после ключевого слова AFTER указываем список действий по модификации данных через запятую, после которых будет запускаться тело триггера. После того как все нужные операции перечислены, пишется ключевое слово AS после которого с ключевого слова BEGIN начинается тело триггера, по его окончании пишется ключевое слово END.

CREATE TRIGGER <имя триггера>

AFTER название\_операции\_1, …

AS

BEGIN

<Тело триггера>

END

1. Создаем переменную типа nvarchar(max) с помощью оператора DECLARE, где и будет храниться сам триггер.

DECLARE @trigger nvarchar(MAX)

GUID для названия триггера будем получать при помощи функции NEWID() которая возвращает уникальную последовательность символов

1. Максимальное значение из вставленных или измененных будем искать при помощи системной таблицы inserted. В таблице inserted находятся копии строк, с которыми работали действия по модификации данных INSERT или UPDATE.

DECLARE @maxIns int

SELECT @maxIns = MAX(inserted.' + QUOTENAME(@columnName) + ') FROM inserted

1. Значение в спец таблице будем обновлять если найденное максимальное больше чем текущее.

UPDATE spec SET curMax = @maxIns WHERE id = ' + CAST(@nextId AS nvarchar(50)) + ' AND @maxIns > curMax

1. Когда триггер полностью записан в строку, его можно запустить будем это осуществлять с помощью системной функции EXEC*,* данная функция позволяет запустить на выполнение триггер, который хранится в переменной типа *nvarchar(max).*

DECLARE @trigger nvarchar(MAX) = '

CREATE TRIGGER ' + QUOTENAME('UpdateCurMax' + @tableName + '\_' + @columnName + '\_' + CAST(NEWID() AS nvarchar(36))) +

' ON ' + QUOTENAME(@tableName) + '

AFTER INSERT, UPDATE AS

BEGIN

DECLARE @maxIns int

SELECT @maxIns = MAX(inserted.' + QUOTENAME(@columnName) + ') FROM inserted

UPDATE spec SET curMax = @maxIns WHERE id = ' + CAST(@nextId AS nvarchar(50)) + ' AND @maxIns > curMax

END'

EXEC(@trigger)

1. Тестирование программы

Проверим следующий набор тестов:

1. Для таблицы с одним столбцом (Предварительно вызвав ХП)
   1. Вставка новых значений
      * Больше текущего максимального
      * Меньше текущего максимального
      * Сразу несколько различных значений
   2. Обновление значений в столбце
      * Чтобы стало больше текущего максимального
      * Текущее максимальное изменяем на меньшее
2. Для таблицы с несколькими столбцами (Предварительно вызвав ХП)
3. Вставка новых значений
   * Вставка только в один столбец
   * Вставка в оба столбца
   * Вставка сразу несколько различных значений
4. Обновление значений в столбце
   * Обновление только одного столбца
   * Обновление обоих столбцов

Тест №1.1. Вставим значение 12 в таблицу test в столбец num

|  |  |  |  |
| --- | --- | --- | --- |
| id | nameTable | columnName | currentMax |
| 1 | spec | id | 1 |
| 2 | test | num | 12 |

Вставим значение 8 в таблицу test в столбец num

|  |  |  |  |
| --- | --- | --- | --- |
| id | nameTable | columnName | currentMax |
| 1 | spec | id | 1 |
| 2 | test | num | 12 |

Вставим значение 8 и 16 в таблицу test в столбец num

|  |  |  |  |
| --- | --- | --- | --- |
| id | nameTable | columnName | currentMax |
| 1 | spec | id | 1 |
| 2 | test | num | 16 |

Тест №1.2. Изменим значение 16 на 18 в таблицу test в столбец num

|  |  |  |  |
| --- | --- | --- | --- |
| id | nameTable | columnName | currentMax |
| 1 | spec | id | 1 |
| 2 | test | num | 18 |

Изменим значение 18 на 9 в таблицу test в столбец num

Состояние таблицы не поменяется.

Тесты были пройдены успешно.