

ЛАБОРАТОРНАЯ РАБОТА № 4.

Триггеры.

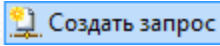
Цель: научиться создавать триггеры в среде SQL Server Management Studio Express.

Программное обеспечение: SQL Server Management Studio Express

Триггеры предназначены для реализации бизнес-логики и журналирования работы пользователей с данными.

Задание 1: Создать триггер “Заявление_OnInsert” на событие INSERT для таблицы “Заявления”. Триггер должен проверять, что абитуриент не может подать более трех заявлений на разные специальности. Если абитуриент уже подал три заявления, то вызывается откат транзакции по добавлению заявления в таблицу “Заявления”.

Порядок выполнения:

- 1) Откройте SQL Server Management Studio и подключитесь к серверу баз данных.
- 2) Щелкните на панели инструментов кнопку 
- 3) В открывшемся окне введите следующий код на языке SQL:

```
create trigger Заявление_OnInsert on Заявления
for insert
as
-- Код абитуриента, подавшего заявление
declare @код_абитуриента int
select @код_абитуриента=код_абитуриента from inserted
-- Количество заявлений абитуриента
declare @заявлений int
select @заявлений=count(*)
from заявления
where код_абитуриента=@код_абитуриента
if (@заявлений > 3)
begin
raiserror ('Абитуриент [%d] не может подать более трех заявлений',
16,1,@код_абитуриента)
rollback tran
end
```

Примечание: в тексте триггера используется системная команда raiserror, вызывающая показ сообщения об ошибке. Более подробную информацию о команде raiserror можно найти в справке SQL Server Books Online.

- 4) Исполните введенный код SQL. На вкладке “Сообщения” выводится сообщение об успешном завершении команды.

- 5) Для проверки работы созданного триггера щелкните на панели инструментов кнопку “Создать запрос”.
- 6) В открывшемся окне введите следующий код на языке SQL (код абитуриента и код специальности посмотрите в таблице “Заявления”):

```
insert into заявления values (8,4,'не зачислен')
```

- 7) Исполните введенный код SQL.
- 8) На вкладке “Сообщения” выводится сообщение об ошибке, если для указанного абитуриента в таблице “Заявления” уже введено три заявления.

Задание 2: Создать триггер “Абитуриент_OnUpdate”, который вызывается при обновлении таблицы “Абитуриенты”. Триггер используется для записи в таблицу “Журнал” сообщения об изменении статуса абитуриента.

Порядок выполнения:

- 1) Откройте SQL Server Management Studio и раскройте дерево в окне “Обозреватель объектов” до узла “Таблицы” вашей базы данных.
- 2) Раскройте узел “Абитуриенты”.
- 3) Щелкните правой кнопкой узел “Триггеры” и выберите пункт меню “Создать триггер...”. Появится окно с текстом шаблона запроса для создания триггера.
- 4) Замените исходный код следующим кодом:

```
create trigger Абитуриент_OnUpdate on Абитуриенты
for update
as
    -- Проверка того, что изменился статус абитуриента
    if update(статус)
    begin
        -- Код абитуриента, для которого был изменен статус
        declare @код_абитуриента int
        -- Старый статус абитуриента
        declare @статус_был varchar(16)
        -- Новый статус абитуриента
        declare @статус_стал varchar(16)

        -- Таблица inserted содержит обновленные записи после изменения
        select @код_абитуриента=Код_абитуриента,@статус_стал=Статус
            from inserted
        -- Таблица deleted содержит обновленные записи до изменения
        select @статус_был=Статус
            from deleted
        insert into Журнал values
            ('Статус абитуриент ['+cast(@код_абитуриента as varchar(16))+
            '] был изменен с ['+@статус_был+
            '] на ['+@статус_стал+
            '] пользователем ['+current_user+']',
            getdate())
    end
```

Примечание: в тексте триггера используется системная команда CAST({переменная} as {тип данных}) для приведения типа переменной и системная функция GETDATE() для получения текущей даты и времени.

- 5) Исполните введенный код SQL. На вкладке “Сообщения” выводится сообщение об успешном завершении команды.
- 6) Для проверки работы созданного триггера щелкните правой кнопкой таблицу “Абитуриенты”, затем щелкните пункт меню “Изменить первые 200 строк”. Появится закладка “Table - dbo.Абитуриенты”.
- 7) Измените значения поля “Статус” какого-либо абитуриента, после чего закройте закладку “Table - dbo.Абитуриенты”.
- 8) Щелкните правой кнопкой таблицу “Журнал”, затем щелкните пункт меню “Изменить первые 200 строк”. Появится закладка “Table - dbo.Журнал”.
- 9) Найдите запись с созданным сообщением об изменении статуса абитуриента. Закройте закладку “Table - dbo.Журнал”.

Задание 3: Создайте триггер “Экзамен_OnInsert” на событие INSERT для таблицы “Экзамены”. Триггер должен проверять, что для специальности в один день проводится лишь один экзамен. Если на дату, на которую добавляется экзамен, уже назначен для этой специальности экзамен, то должен вызываться откат транзакции по добавлению экзамена в таблицу “Экзамены”.

Задание 4: Создайте триггер “Специальность_OnUpdate” на событие UPDATE для таблицы “Специальности”. Триггер должен записывать в таблицу “Журнал” сообщение об изменении количества вакансий по специальности.

Задание 5: Создайте триггер “Оценка_OnInsert” на событие INSERT для таблицы “Оценки”. Триггер должен проверять, что у абитуриента, оценка которого добавляется, есть заявление на специальность, для которой проводился экзамен. Если добавляемая оценка абитуриента связана с экзаменом по специальности, на которую абитуриент не подавал заявления, то должен вызываться откат транзакции по добавлению оценки в таблицу “Оценки”.

Алгоритм создания триггера "Экзамен_OnInsert":

- 1) Объявить переменные: код_специальности, дата_экзамена, количество_экзаменов.
- 2) Записать в объявленные переменные код_специальности и дата_экзамена из вставляемой строки inserted.
- 3) С помощью функции count() записать в переменную количество экзаменов для выбранных кода специальности и даты экзамена.
- 4) Если количество_экзаменов > 1, то вывести сообщение об ошибке и выполнить откат транзакции.

Алгоритм создания триггера "Оценка_OnInsert":

- 1) Объявить переменные: код_абитуриента, код_экзамена, код_ специальности, количество_заявлений
- 2) Записать в объявленные переменные Код_абитуриента и Код_экзамена из вставляемой строки inserted.
- 3) Записать в переменную Код_специальности из таблицы экзамены для экзамена, записанного в переменную во 2-ом пункте.
- 4) Записать в переменную количество заявлений для объявленного абитуриента по объявленной специальности.
- 5) Если количество_заявлений = 0, то вывести сообщение об ошибке и выполнить откат транзакции.