|  |
| --- |
| **TRƯỜNG ĐẠI HỌC LÂM NGHIỆP**  **BỘ MÔN TIN HỌC**  **BÀI BÁO CÁO TÌM HIỂU CHỦ ĐỀ**  **HỌC PHẦN: HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**  **Sinh viên thực hiện : Lê Văn Lộc**  **Mã sinh viên : 204 107 0220**  **Lớp : K65 – HTTT (CNTT)**  **Năm học : 2021 – 2022**  ***Hà Nội, tháng 4 năm 2022*** |

**MỤC LỤC**

[I) Tìm hiểu chung về Trigger 3](#_Toc100870637)

[1. Khái niệm: 3](#_Toc100870638)

[2. Các loại Trigger 3](#_Toc100870639)

[3. Cú pháp tạo một Trigger: 4](#_Toc100870640)

[4. Ưu điểm và nhược điểm của Trgger: 4](#_Toc100870641)

[Ưu điểm 4](#_Toc100870642)

[Nhược điểm 4](#_Toc100870643)

[II) Ví dụ: 5](#_Toc100870644)

**Chủ đề:** Tìm hiểu về Trigger trong SQL Sever

1. **Tìm hiểu chung về Trigger**
2. **Khái niệm:**

* Trigger là một loại stored procedure đặc biệt được thực thi (execute) một cách tự động khi có một sự kiện thay đổi dữ liệu (data modification) xảy ra như Update, Insert hoặc Delete. Trigger được dùng để đảm bảo tính toàn vẹn dữ liệu (Data Integrity) hoặc thực hiện các quy tắc nghiệp vụ (business rules) nào đó.
* Mỗi bảng thường sẽ có 3 thao tác làm thay đổi dữ liệu đó là: Update (Cập nhật), Insert (Chèn), Delete (Xóa). Và đôi khi mỗi hành động như vậy ta sẽ có những ràng buộc trên bảng để giúp bảo toàn dữ liệu, lúc này sử dụng trigger là một giải pháp tốt.

1. **Các loại Trigger**

* **Có ba loại trigger như sau:**
* **Update:** Loại trigger sẽ được kích hoạt khi có hành động cập nhật dữ liệu.
* **Insert:** Loại trigger sẽ được kích hoạt khi có hành động thêm dữ liệu.
* **Delete:** Loại trigger sẽ được kích hoạt khi có hành động xóa dữ liệu.
* **Lưu ý:** SQL Server cung cấp 2 bảng ảo dành riêng cho trigger tên là INSERTED vaf DELETED, hai bảng này sẽ lưu trữ dữ liệu của các row trước hoặc sau khi hành động xảy ra.

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Inserted** | **Deleted** |
| **Insert** | Dữ liệu của hàng vừa chèn | Rỗng |
| **Update** | Dữ liệu mới của hàng vừa cập nhật | Dữ liệu cũ của hàng vừa cập nhật |
| **Delete** | Rỗng | Dữ liệu của hàng bị xóa |

* **Trigger dùng để làm gì?**
* Trigger SQL Server được sử dụng để kiểm tra ràng buộc (check constraints) trên nhiều quan hệ (nhiều bảng/table) hoặc trên nhiều dòng (nhiều record) của bảng.
* Bên cạnh đó, việc sử dụng Trigger để chương trình có những hàm chạy ngầm nhằm phục vụ những trường hợp hữu hạn và thường không sử dụng cho mục đích kinh doanh hoặc giao dịch.

1. **Cú pháp tạo một Trigger:**

**CREATE TRIGGER <** tên\_trigger **>**

**ON <** tên\_bảng **>**

**AFTER {**DELETE, INSERT, UPDATE**}**

**AS**

**Begin**

**<** câu\_lệnh\_SQL **>**

**End**

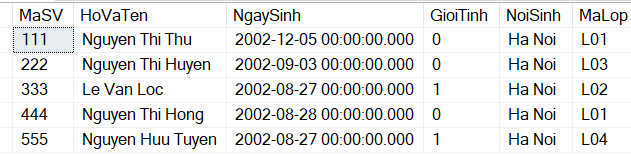
* + - **Chú ý:** Tên Trigger tự đặt phải dễ hiểu và tránh trùng với tên các hàm; SQL Sever không phân biệt chữ in hoa hay in thường, chỉ cần viết đúng theo cú pháp.

1. **Ưu điểm và nhược điểm của Trgger:**

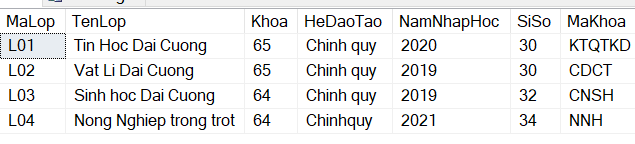
|  |  |
| --- | --- |
| **Ưu điểm** | **Nhược điểm** |
| * Trigger có thể bắt lỗi logic ở mức cơ sở dữ liệu. * Có thể dùng trigger là một cách khác để thay thế việc thực hiện những công việc hẹn giờ theo lịch. * Trigger rất hiệu quả khi được sử dụng để kiểm soát những thay đổi của dữ liệu trong bảng. | * Trigger chỉ là một phần mở rộng của việc kiểm tra tính hợp lệ của dữ liệu chứ không thay thế được hoàn toàn công việc này. * Trigger hoạt động ngầm ở trong csdl, không hiển thị ở tầng giao diện. Do đó, khó chỉ ra được điều gì xảy ra ở tầng csdl. * Trigger thực hiện các update lên bảng dữ liệu vì thế nó làm gia tăng lượng công việc lên cơ sở dữ liệu và làm cho hệ thống chạy chậm lại. * Nếu có nhiều trigger lồng nhau: Có thể rất khó gỡ lỗi và khắc phục sự cố. Điều này làm tiêu tốn thời gian và tài nguyên để phát triển. |

1. **Ví dụ:**

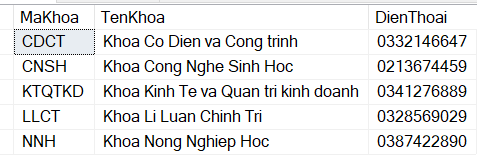
* **Cho database sau:**
* **SV**(***MaSV***, HoVaTen, NgaySinh, GTinh, NoiSinh, MaLop)
* **Lop** (***MaLop***, TenLop, Khoa, HeDaoTao, NamNhapHoc, SiSo, MaKhoa)
* **Khoa** (***MaKhoa***, TenKhoa, DienThoai)
* **MonHoc** (***MaMonHoc***, TenMonHoc, SoDVHT)
* **DiemThi** (***MaSV***, ***MaMonHoc***, DiemLan1, DiemLan2)
* **Dữ liệu nhập vào:**
* **SV**



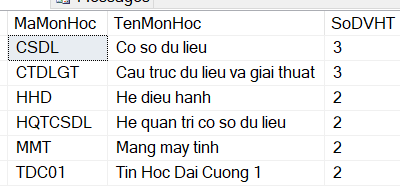
* Lop



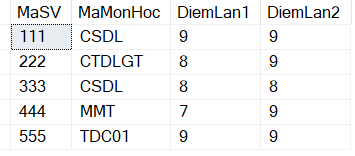
* Khoa



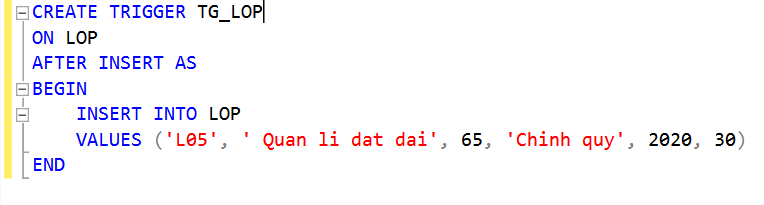
* MonHoc



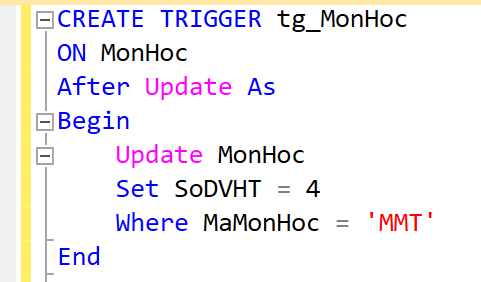
* DiemThi



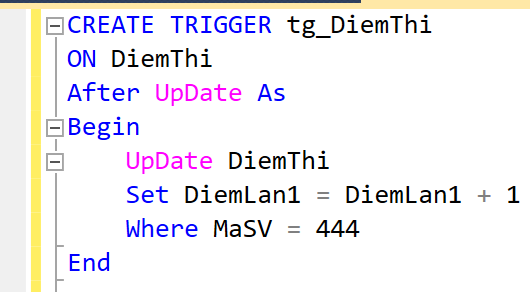
1. **Tạo Trigger chèn thêm dữ liệu vào bảng lớp với dữ liệu chèn thêm như sau: (‘L05’, ‘Quan li dat dai’, 65, ‘ Chinh quy’, 2020, 32)**

****

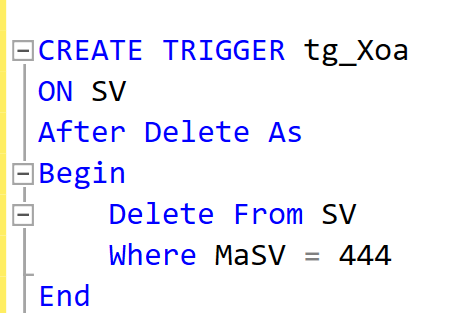
* Thực thi: Select \* From Lop­\_UpDate
  1. **Tạo Trigger cập nhật dữ liệu cho môn học có mã là MMT lên thành 4 đơn vị học trình?**

****

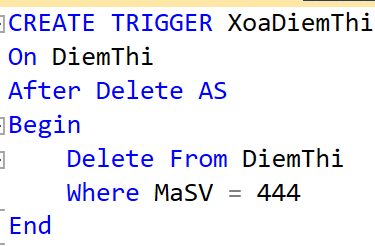
* Thực thi Trigger: Select \* From MonHoc\_UpDate
  1. **Tạo Trigger cập nhật điểm lần 1 của sinh viên có mã là 444 tăng thêm 1?**

****

* Thực thi Trigger: Select \* From DiemThi
  1. **Tạo Trigger xóa bỏ thông tin của sinh viên có mã là 444**

****

* Thực thi Trigger: Select \* From SV
  1. **Tạo Trigger xóa bỏ điểm thi của sinh viên có mã là 444**

****

* Kết quả sau khi thực hiện kiểm tra Trigger: Điểm thi của sinh viên có mã sinh viên là 444 đã bị xóa khỏi bảng