

BÀI THỰC HÀNH SỐ 3

1 TRIGGER

Câu 1: Cho bảng NHANVIEN (MaNV, Hoten, Ngaysinh,...).Viết một trigger để đảm bảo rằng khi thêm một nhân viên mới vào thì tuổi của nhân viên không được ≥ 45

Câu 2: Tạo trigger để tránh xoá 2 bản ghi trong bảng Nhanvien đồng thời.

Câu 3: Tạo UPDATE trigger đảm bảo rằng cột lương không được lớn hơn 1000000 và ngày sinh không lớn hơn ngày hiện tại.

Câu 4: Tạo một trigger không cho phép cập nhật cột Ngaysinh trong bảng Nhanvien.

Câu 5: Hiện thị các trigger trong bảng Nhanvien

Câu 6: tạo trigger để kiểm tra dữ liệu nhập vào cột MaP trong bảng Nhanvien phải là dữ liệu đã tồn tại trong cột MaP của bảng Phong

Câu 7: Tạo một cascade trigger để khi xóa một phòng trong bảng Phong thì toàn bộ nhân viên của phòng đó cũng bị xóa. Sau đó trigger này sẽ kích hoạt hành động hiển thị thông tin của những nhân viên còn lại.

Câu 8: Tạo một view View_Cau8 chứa MaNV, Hoten, TenP. Thử thực hiện việc thêm/xóa/sửa trên trigger đó. Từ đó viết instead of trigger để thay thế cho những tác insert, update, delete nguyên thủy.

2 TRANSACTION

Câu 1: Bảng NHANVIEN có các cột (MaNV, ...,MaP,...)

Bảng Phong có các cột (MaP, TenP, Soluong)

Viết một giao dịch để thực hiện công việc sau: Nhân viên có mã nhân viên là 'NV0005' chuyển sang làm ở phòng 'KT'.

BEGIN TRAN ThemNV

Declare @pos char(10)

Select @pos=MaP from NHANVIEN where MaNV='NV0005'

UPDATE NHANVIEN Set MaP='KT' where MaNV='NV0005'

Update Phong set soluong=soluong+1 where maP='KT'

Update Phong set soluong=soluong-1 where map=@pos

Commit tran ThemNV

Câu 2: Viết một giao dịch để thêm nhân viên vào phòng 'QT'.

Câu 3: So sánh mức cô lập của READ COMMITTED và READ UNCOMMITTED. Tạo một bảng DOCGIA (MaDG, Hoten). Giả sử ban đầu trong bảng DOCGIA chưa có độc giả nào có tên 'AAA'

a. Trường hợp 1

T1	T2
BEGIN TRAN Update DOCGIA Set ten='AAA' Where MaDG<11 Waitfor delay '00:00:15' Rollback tran	Begin tran SET TRAN ISOLATION LEVEL READ COMMITTED Select * from DOCGIA Where ten = 'AAA' Commit tran

Table 1

b. Trường hợp 2

T1	T2
BEGIN TRAN Update DOCGIA Set ten='AAA' Where MaDG<11 Waitfor delay '00:00:15' Rollback tran	Begin tran SET TRAN ISOLATION LEVEL READ UNCOMMITTED Select * from DOCGIA Where ten = 'AAA' Commit tran

Table 2

Bài 4: So sánh mức cô lập READ COMMITTED và REPEATABLE READ. Giả sử ban đầu bản ghi có madg='1' có tên là 'XXX'

a. Trường hợp 1

T1	T2
----	----

BEGIN TRAN SET TRAN ISOLATION READ COMMITTED SELECT ten from docgia Where madg='1' Waitfor delay '00:00:15' commit tran	Begin tran Update docgia Set ten='AAA' where madg='1' Commit tran
---	--

Table 3

b. Trường hợp 2

T1	T2
BEGIN TRAN SET TRAN ISOLATION READ COMMITTED SELECT ten from docgia Where madg='1' Waitfor delay '00:00:15' Select ten from docgia where madg='1' commit tran	Begin tran Update docgia Set ten='AAA' where madg='1' Commit tran

Table 4

c. Trường hợp 3

T1	T2
BEGIN TRAN SET TRAN ISOLATION REPEATABLE READ SELECT ten from docgia Where madg=1 Waitfor delay '00:00:15' Select ten from docgia where madg=1 commit tran	Begin tran Update docgia Set ten='AAA' where madg='1' Commit tran Commit tran

Table 5

Nhận xét: T1 sẽ nhận được 2 bản ghi giống nhau. T2 phải chờ T1 kết thúc mới update được

Câu 5: So sánh mức cô lập REPEATABLE READ và SERIALIZABLE

a. Trường hợp 1

T1	T2
BEGIN TRAN SET TRAN ISOLATION REPETABLE READ SELECT ten from docgia Where madg>90 Waitfor delay '00:00:15' Select ten from docgia where madg='1' commit tran	Begin tran Insert docgia values (111, 'tuyet',.. Commit tran

Table 6

b. Trường hợp 2

T1	T2
BEGIN TRAN SET TRAN ISOLATION SERIALIZABLE SELECT ten from docgia Where madg>90 Waitfor delay '00:00:15' Select ten from docgia where madg='1' commit tran	Begin tran Insert docgia values (111, 'tuyet',.. Commit tran

Table 7

Nhận xét: Hai câu truy vấn của T1 nhận được kết quả giống nhau.