

BÁO CÁO ĐỒ ÁN THỰC HÀNH
MÔN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU
GVHD:Tiết Gia Hồng

THÔNG TIN NHÓM

CÀI ĐẶT TÌNH HUỐNG TRANH CHẤP

I. Sinh viên thực hiện: Trần Văn Khuê, Trần Minh Bảo

1. Tình huống 1: Ảnh hưởng chức năng Xem nhà

ERR01: Phantom T1 : thực hiện xem thông tin nhà. T2 : thực hiện xóa 1 nhà.			
sp_XemThongTinNha	Khóa	sp_XoaMotNha	Khóa
Input: @id int Output:		Input: @id int Output:	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
B1: Xem danh sách nhà cho thuê SELECT * FROM HOUSE	R(House) //Xin khoá đọc trên bảng House		
WAITFOR DELAY '00:00:05'			
		BEGIN TRAN	
		B1: Xóa một nhà T1 chuẩn bị xem DELETE FROM HOUSE WHERE id = @id	X(House) //Xin khoá ghi trên bảng House
		COMMIT	
B1: Xem danh sách nhà cho thuê SELECT * FROM HOUSE WHERE id = @id Không tồn tại nhà này nữa			
COMMIT			

ERR02: Dirty Read T1 : xem danh sách nhà T2 : thêm một nhà			
sp_ThemMotNha	Khóa	sp_XemDanhSachNha	Khóa
<u>Input:</u> @id int, @streetName varchar(50), @district varchar(50), @city varchar(50), @feemonth float <u>Output:</u>		<u>Input:</u> <u>Output:</u>	
SET TRANSACTION ISOLATION LEVEL READ COMMITED		SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED	
BEGIN TRAN			
B1: Thêm một nhà INSERT INTO HOUSE(ID, STREETNAME, DISTRICT, CITY, FEEMONTH) VALUES(@id, @streetName, @district, @city, @feemonth)	X(House) //Xin khoá ghi trên bảng House		
WAITFOR DELAY '00:00:05'			
		BEGIN TRAN	
		B1: Xem danh sách nhà SELECT * FROM HOUSE	R(House) //Xin khoá đọc trên bảng House
		COMMIT	
B2: Thêm một nhà INSERT INTO HOUSE(ID, STREETNAME, DISTRICT, CITY, FEEMONTH) VALUES(@id, @streetName, @district, @city, @feemonth)	X(House) //Xin khoá ghi trên bảng House		
Error: ROLLBACK vì tạo 2 record cùng primary key			

ERR03: Lost Updated T1 : Chính sửa thông tin nhà T2 : Chính sửa thông tin nhà			
sp_ChinhSuaThongTinNha1 Input: @id int, @streetName varchar(50), @district varchar(50), @city varchar(50), @feemonth float Output:	Khóa	sp_ChinhSuaThongTinNha2 Input: @id int, @streetName varchar(50), @district varchar(50), @city varchar(50), @feemonth float Output:	Khóa
SET TRANSACTION ISOLATION LEVEL READ COMMITED		SET TRANSACTION ISOLATION LEVEL READ COMMITED	
BEGIN TRAN			
B1: Sửa thông tin nhà UPDATE HOUSE SET StreetName = @streetName, District = @district, City = @city, FeeMonth = @fee where id = @id	X(House) //Xin khoá ghi trên bảng House		
WAITFOR DELAY '00:00:05'			
		BEGIN TRAN	
		B1: Sửa thông tin nhà UPDATE HOUSE SET StreetName = @streetName, District = @district, City = @city, FeeMonth = @fee where id = @id	X(House) //Xin khoá ghi trên bảng House
		COMMIT	
COMMIT			
Mất dữ liệu T2			

2. Tình huống 2: Các giao tác ảnh hưởng đến chức năng thuê nhà

ERR01: Dirty read T1 (User = nhân viên 1 (chi nhánh B1)): thực hiện giao tác sửa thông tin chi tiết của người thuê nhà A T2 (User = nhân viên 2 (B1)): thực hiện giao tác lấy thông tin chi tiết của người thuê nhà A			
sp_SuaChiTietNguoiThueNha	Khóa	sp_ChiTietNguoiThueNha	Khóa
Input: maNguoiThue (nvarchar) Output: NguoiThueNha (1 row)		Input: maNguoiThue (nvarchar) Output: NguoiThueNha (1 row)	
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ		SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED	
BEGIN TRAN			
B1: Cập nhật thông tin vào bảng NguoiThueNha với điều kiện maNguoiThue=@maNguoiThue UPDATE NguoiThueNha SET nhaDangThue=5 WHERE maNguoiThue=@maNguoiThue	X(A)		
WAITFOR DELAY '00:00:20'			
		BEGIN TRAN	
		B1: Lấy thông tin người thuê nhà SELET * FROM NguoiThueNha WHERE maNguoiThue = @maNguoiThue	R(A) //Không cần xin khoá
		COMMIT	
IF @error <> 0 SELECT SELECT N'Không thể cập nhật người thuê nhà' ROLLBACK TRAN RETURN End			
ERR02: Dirty read T1 (User = nhân viên 1 (chi nhánh B1)): thực hiện giao tác sửa thông tin chi tiết của nhà A T2 (User = nhân viên 2 (B1)): thực hiện giao tác lấy thông tin chi tiết của nhà A			

sp_SuaChiTietNha	Khóa	sp_ChiTietNha	Khóa
Input: maNha (nvarchar) Output: Nha (1 row)		Input: maNha (nvarchar) Output: Nha (1 row)	
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ		SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED	
BEGIN TRAN			
B1: Cập nhật thông tin vào bảng Nha với điều kiện maNha = @maNha UPDATE Nha SET daDuocThue=True WHERE maNha=@maNha	X(A)		
WAITFOR DELAY '00:00:20'			
		BEGIN TRAN	
		B1: Lấy thông tin người thuê nhà SELET * FROM Nha WHERE maNha = @maNha	R(A) //Không cần xin khoá
		COMMIT	
IF @error <> 0 SELECT SELECT N'Không thể cập nhật nhà' ROLLBACK TRAN RETURN End			

ERR03: Lost Updated

T1 (User = nhân viên 1 (chi nhánh B1)): thực hiện giao tác sửa thông tin chi tiết của người thuê nhà A

T2 (User = nhân viên 2 (B1)): thực hiện giao tác sửa thông tin chi tiết của người thuê nhà As

sp_SuaChiTietNguoiThueNha	Khóa	sp_ChiTietNguoiThueNha	Khóa
Input: maNguoiThue (nvarchar) Output: NguoiThueNha (1 row)		Input: maNguoiThue (nvarchar) Output: NguoiThueNha (1 row)	
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ		SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED	
BEGIN TRAN			
B1: Cập nhật thông tin vào bảng NguoiThueNha với điều kiện maNguoiThue=@maNguoiThue UPDATE NguoiThueNha SET nhaDangThue=5 WHERE maNguoiThue=@maNguoiThue	X(A)		
WAITFOR DELAY '00:00:20'			
		BEGIN TRAN	
		B1: Cập nhật thông tin vào bảng NguoiThueNha với điều kiện maNguoiThue=@maNguoiThues UPDATE NguoiThueNha SET nhaDangThue=4ss WHERE maNguoiThue=@maNguoiThue	X(A)
		COMMIT	
SELECT * FROM NguoiThueNha Where maNguoiThue = @maNguoiThue			

ERR04: Lost Updated

T1 (User = nhân viên 1 (chi nhánh B1)): thực hiện giao tác sửa thông tin chi tiết của nhà A

T2 (User = nhân viên 2 (B1)): thực hiện giao tác sửa thông tin chi tiết của nhà A

sp_SuaChiTietNha	Khóa	sp_ChiTietNha	Khóa
Input: maNha (nvarchar)		Input: maNha (nvarchar)	

Output: Nha (1 row)		Output: Nha (1 row)	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
B1: Cập nhật thông tin vào bảng Nha với điều kiện maNha = @maNha UPDATE Nha SET daDuocThue=True WHERE maNha=@maNha	X(A)		
WAITFOR DELAY '00:00:20'			
		BEGIN TRAN	
		B1: Cập nhật thông tin vào bảng Nha với điều kiện maNha = @maNha UPDATE Nha SET daDuocThue=False WHERE maNha=@maNha	X(A)
		COMMIT	
SELECT * FROM Nha Where maNha = @maNha			

II. Sinh viên thực hiện: Vũ Mạnh Hùng

1. Tình huống 1: Cập nhật – Xem thông tin lương của nhân viên

ERR01: Dirty read T1: User thực hiện cập nhật thông tin nhà T2: User thực hiện xem thông tin nhà			
Employee_InsUpd		Employee_Search	
Input: Danh sách các tham số @pID bigint = -1, @pName nvarchar (255) = '', @pAddress nvarchar (255) = '', @pPhone varchar (15) = '', @pGender bit = 1, @pBirthDay datetime = NULL, @pSalary bigint = 0,	Khóa	Input: Danh sách các tham số @pID bigint = '', @pName varchar (255) = '', @pAddress varchar (255) = '', @pPhone varchar (15) = '', @pGender bit = -1, @pBirthDay varchar (10) = '', @pSalary bigint = -1,	Khóa

<p>@pIDBranchOffice bigint = NULL, @pPassword varchar(255) = ''</p> <p>Output: Kết quả Insert – Update</p>		<p>@pIDBranchOffice bigint = NULL</p> <p>Output: Danh sách thỏa điều kiện tìm kiếm</p>	
		<p>SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED</p>	
<p>BEGIN TRANSACTION</p>			
<p>B1: Khai báo biến lưu thông báo trả về</p> <p>DECLARE @vMessage nvarchar(max)</p> <p>SET @vMessage = ''</p>			
<p>B2: Kiểm tra tham số @pID truyền vào</p> <p>IF EXISTS (@pID = -1)</p> <p>BEGIN</p> <p>B3.1.1: Thêm dữ liệu vào bảng nhân viên</p> <p>INSERT INTO [dbo].[Employee](</p> <p> [Name],</p> <p> [Address],</p> <p> [Phone],</p> <p> [Gender],</p> <p> [Birthday],</p> <p> [Salary],</p> <p> [IDBranchOffice],</p> <p> [IsActive])</p> <p>VALUES (@pName,</p> <p> @pAddress,</p> <p> @pPhone,</p> <p> @pGender,</p> <p> @pBirthday,</p> <p> @pSalary,</p> <p> @pIDBranchOffice,</p> <p> 1)</p> <p>IF @@Error <> 0 GOTO ABORT</p>	<p>X(Employee)</p>		
<p>B3.1.2: Thêm dữ liệu vào bảng người dùng</p> <p>INSERT INTO [dbo].[Users](</p> <p> [UserName],</p> <p> [Password],</p> <p> [IsAdmin],</p> <p> [IsActive])</p>	<p>X(Users)</p>		

<pre>VALUES (IDENT_CURRENT('Employee'), @pPassword, 0, 1) IF @@Error <> 0 GOTO ABORT END</pre>			
<pre>ELSE BEGIN B3.2: Cập nhật dữ liệu nhân viên UPDATE [dbo].[Employee] SET [Name] = @pName, [Address] = @pAddress, [Phone] = @pPhone, [Gender] = @pGender, [Birthday] = @pBirthday, [Salary] = @pSalary, [IDBranchOffice] = @pIDBranchOffice WHERE [ID] = @pID IF @@Error <> 0 GOTO ABORT END</pre>	X(Employee)		
		BEGIN TRANSACTION	
		<pre>B1: Lấy danh sách thỏa SELECT EMP.[ID], EMP.[Name], EMP.[Address], EMP.[Phone], EMP.[Gender], EMP.[Birthday], EMP.[Salary], EMP.[IDBranchOffice], BRO.[Name] AS BranchOfficeName, EMP.[IsActive] FROM Employee EMP LEFT JOIN BranchOffice BRO ON BRO.ID = EMP.IDBranchOffice WHERE EMP.[IsActive] = 1</pre>	R(Employee)

		<pre> AND (EMP.[ID] = @pID OR @pID IS NULL OR @pID = -1) AND (EMP.[Name] LIKE N'%' + @pName + '%' OR @pName IS NULL OR @pName = '') AND (EMP.[Address] LIKE N'%' + @pAddress + '%' OR @pAddress IS NULL OR @pAddress = '') AND (EMP.[Phone] LIKE N'%' + @pPhone + '%' OR @pPhone IS NULL OR @pPhone = '') AND (EMP.[Gender] = @pGender OR @pGender IS NULL OR @pID = -1) AND (CONVERT(varchar(10), EMP.[Birthday], 103) = @pBirthday OR @pBirthday IS NULL OR @pBirthday = CONVERT(varchar(10), getdate(), 103)) AND (EMP.[Salary] = @pSalary OR @pSalary IS NULL OR @pSalary = -1) AND (EMP.[IDBranchOffice] = @pIDBranchOffice OR @pIDBranchOffice IS NULL OR @pIDBranchOffice = -1) </pre>	
		COMMIT TRANSACTION	
<p>COMMIT TRANSACTION</p> <p>B4.1: Thành công trả kết quả</p> <pre> SELECT 1 AS Result, @vMessage ErrorDesc RETURN 1 </pre> <p>ABORT:</p> <p>BEGIN</p> <p>B4.1: Thất bại delay rồi Rollback – Trả kết quả</p> <p>ROLLBACK TRANSACTION</p> <pre> SET @vMessage = 'Thêm nhân viên thất bại!' SELECT 0 AS Result, @vMessage ErrorDesc RETURN 0 </pre> <p>END</p>			
<p>Cài đặt Trigger để phát sinh lỗi</p> <pre> IF((Select Salary From inserted) < 0) BEGIN </pre>			

```
WAITFOR DELAY '0:0:5'
ROLLBACK TRAN
```

END

The image displays two side-by-side screenshots of a web application interface for 'RentHouseBHK - Quản lý thuê nhà - [Nhân viên]' (RentHouseBHK - House Management - [Employee]).

Left Screenshot:

- Form 'Thông tin nhân viên':** Fields include ID, Tên (Name), Điện thoại (Phone), Ngày sinh (Date of Birth), Chi nhánh (Branch), Lương (Salary), and Địa chỉ (Address). The 'Ngày sinh' field is set to 04/10/2019.
- Table:** A table with 7 columns: Tên nhân viên, Địa chỉ, Số điện thoại, Giới tính, Ngày sinh, Lương, and Tên chi nhánh. The first row is 'Vy Tiểu A' (Tân Bình, 032615984..., Female, 01/01/1990, 100, Q1). The second row is 'Test Dirty Re...' (Tân Phú, 012356956..., Female, 03/02/1986, -99999, Q1).

Right Screenshot:

- Form 'Thông tin nhân viên':** Fields are the same as the left screenshot. The 'Ngày sinh' field is set to 03/02/1986.
- Table:** The table structure is the same, but the second row is 'Võ Văn B' (Tân Phú, 012356956..., Female, 03/02/1986, 200, Q1).

Error Dialog Box:

Lỗi

Lỗi hàm EmployeeDAO.Employee_InsUpd: The transaction ended in the trigger. The batch has been aborted.

OK