**d**

BÁO CÁO ĐỒ ÁN THỰC HÀNH

MÔN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

GVHD: Tiết Gia Hồng

# **THÔNG TIN NHÓM**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **MSSV** | **Họ tên** | **Công việc** | **% Hoàn thành** |
| **1** | **18120358** | **Nguyễn Văn Hảo** | **Viết script và báo cáo** | **100%** |
| **2** | **18120422** | **Trần Thái Đăng Khoa** | **Viết script và báo cáo** | **100%** |
| **3** | **18120418** | **Phạm Minh Khoa** | **Viết script và báo cáo** |  |
| **4** | **18120383** | **Huỳnh Ngọc Hòa** | **Viết script và báo cáo** | **100%** |
| **5** |  |  |  |  |

# CÀI ĐẶT TÌNH HUỐNG TRANH CHẤP

# Sinh viên thực hiện: Phạm Minh Khoa

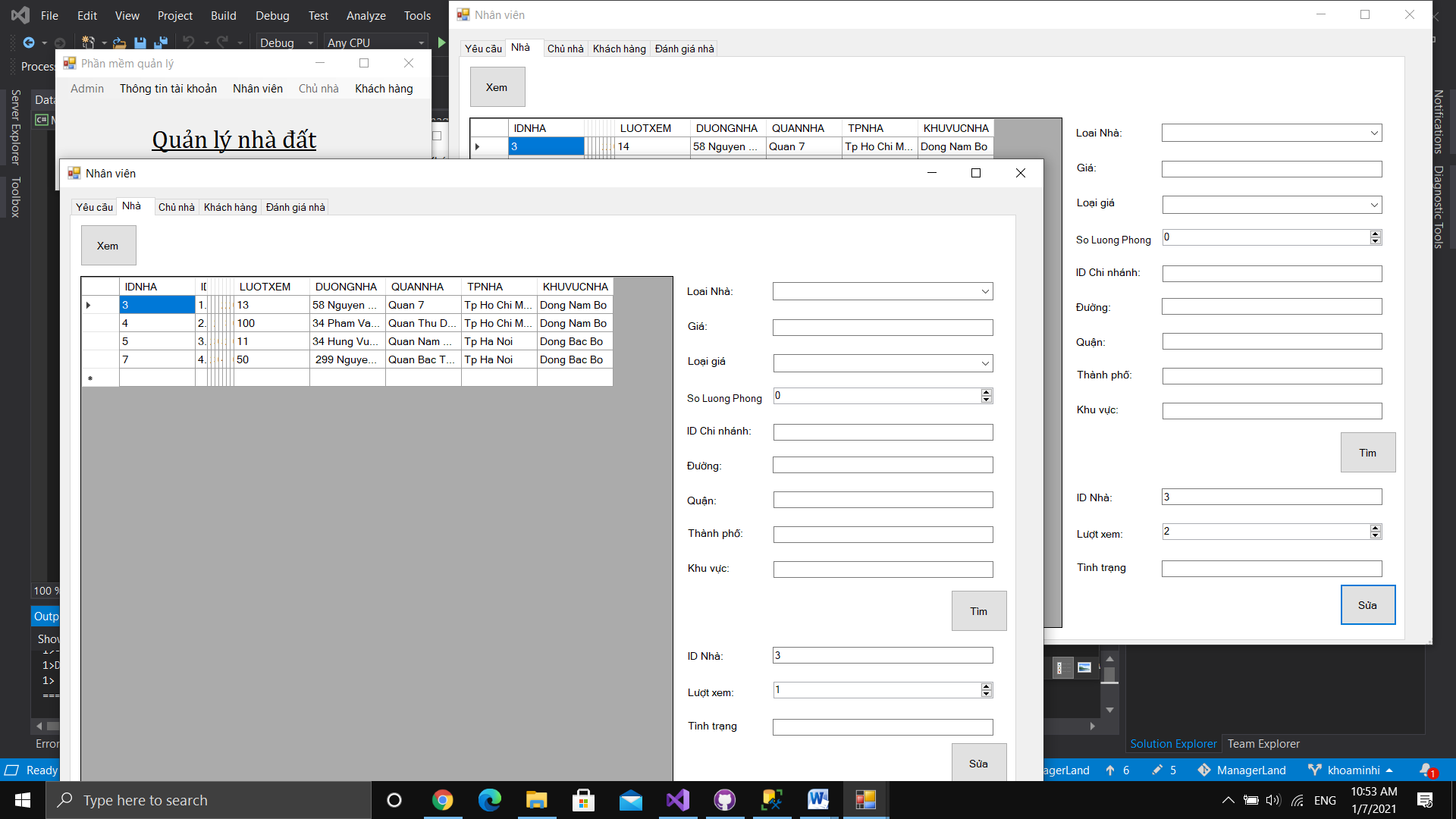
## Tình huống 1: Hai nhân viên cùng cập nhật lượt xem của 1 nhà.

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: *Lost Update***  T1 (User = Nhân viên): thực hiện cập nhật lượt xem của nhà A  T2 (User = Nhân viên): thực hiện cập nhật lượt xem của nhà A | | | |
| **sp\_updateView\_T1** | **Khóa** | **sp\_updateView\_T2** | **Khóa** |
| ***Input:*** *lượt xem nhà*  ***Output***: lượt xem mới | ***Input***: lươt xem nhà  ***Output***: lượt xem nhà mới |
| SET TRANSACTION ISOLATION  LEVEL READ COMMITTED (mặc định) | SET TRANSACTION ISOLATION  LEVEL READ COMMITTED (mặc định) |
| BEGIN TRAN |  |  |  |
| B1: Lấy lượt xem hiên tại của nhà A.  declare @currentViews int  set @currentViews = (select LUOTXEM from NHA where IDNHA = @houseID) | **R(A)**  đọc lượt xem của nhà A trên bảng NHA  Hệ quản trị phát khóa S. Đọc xong trả lại S. | BEGIN TRAN |  |
|  |  | B1: Đọc lượt xem hiện tại của nhà A.  declare @currentViews int  set @currentViews = (select LUOTXEM from NHA where IDNHA = @houseID) | R(A)  xin khóa S, thỏa điều kiện.  Đọc xong trả lại khóa S. |
| waitfor delay '00:00:05' |  |  |  |
|  |  | B2: Cập nhật lượt xem mới vào nhà A  update NHA  set LUOTXEM = @currentViews + @views  where IDCNHA = @houseID | W(A)  xin khóa X, thỏa bảng tương thích.  Giữ cho đến khi commited. |
| B2: Cập nhật lượt xem mới vào nhà A  update NHA  set LUOTXEM = @currentViews + @views  where IDCNHA = @houseID | W(A)  xin khóa X, không thỏa bảng tương thích vì T2 đang giữ khóa X trên A. Chờ T2 commit hoặc rollback |  |  |
|  |  | COMMIT |  |
| Tiếp tục B2 | W(A)  xin khóa X, thỏa vì T2 commit xong đã trả khóa.  Giữ cho đến khi commited. |  |  |
| COMMIT |  |  |  |

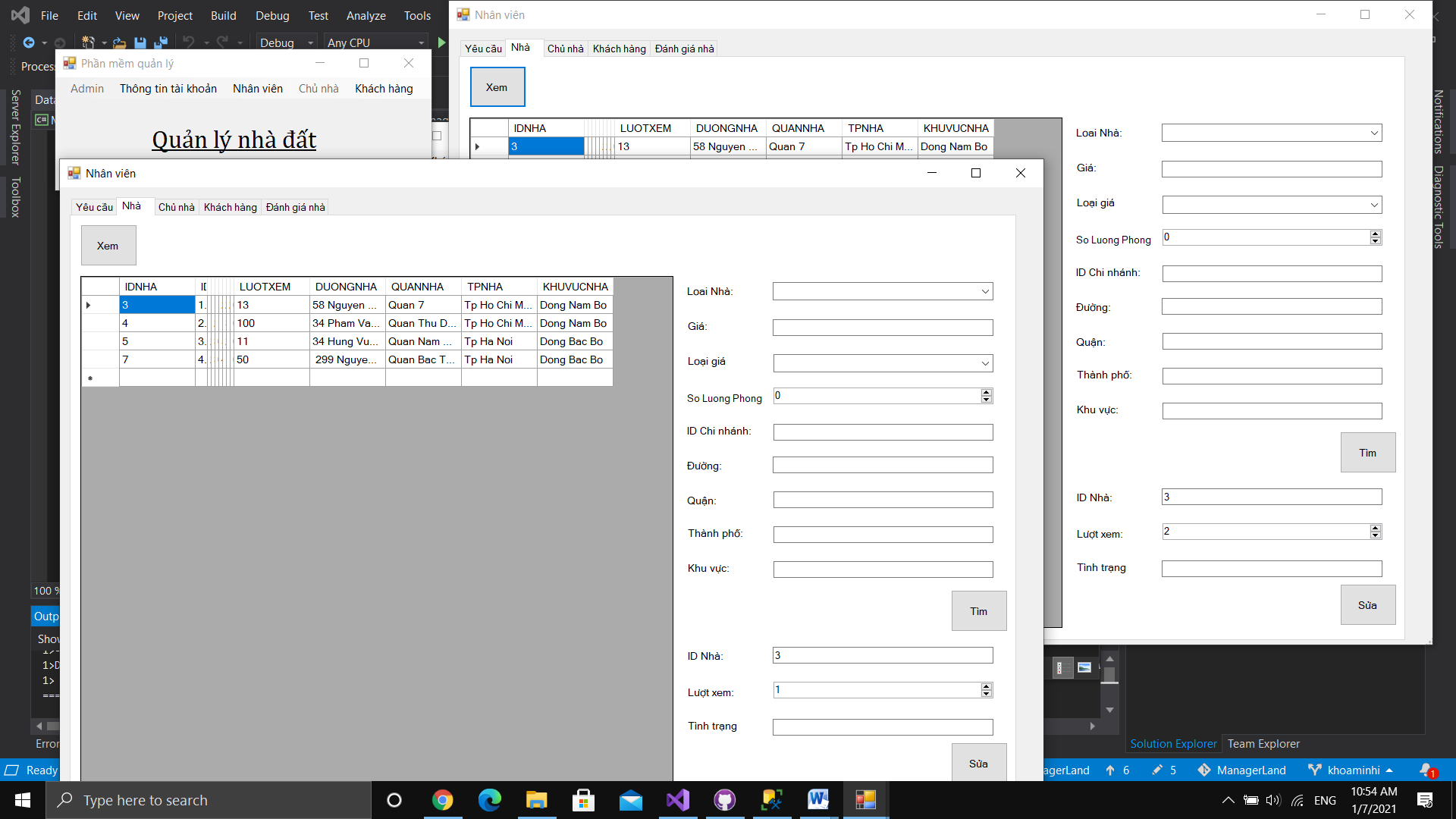
1. Trước khi update. Lượt view của nhà id = 3 của 2 file .exe đều là 12. T1 tăng view lên 1, T2 tăng view lên 2.

## C:\Users\Khoa MinHi\Pictures\Screenshots\Screenshot (79).png

1. Sau khi update và bị lost update. File .exe 1 nhà 3 có view = 13, nhưng file .exe 2 có view = 14.



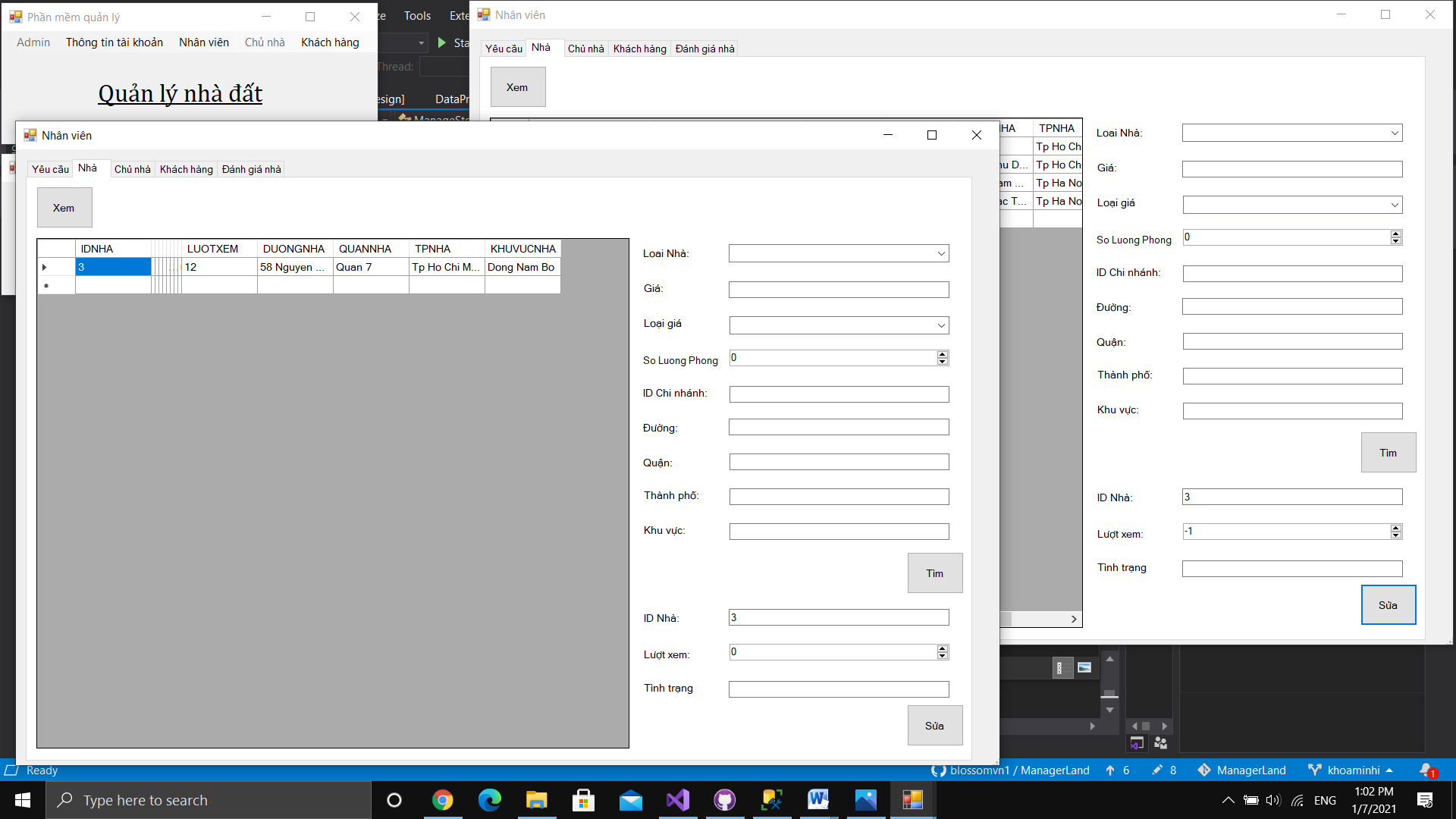
1. Sau khi lọc lại thì cả hai view đều là 13.



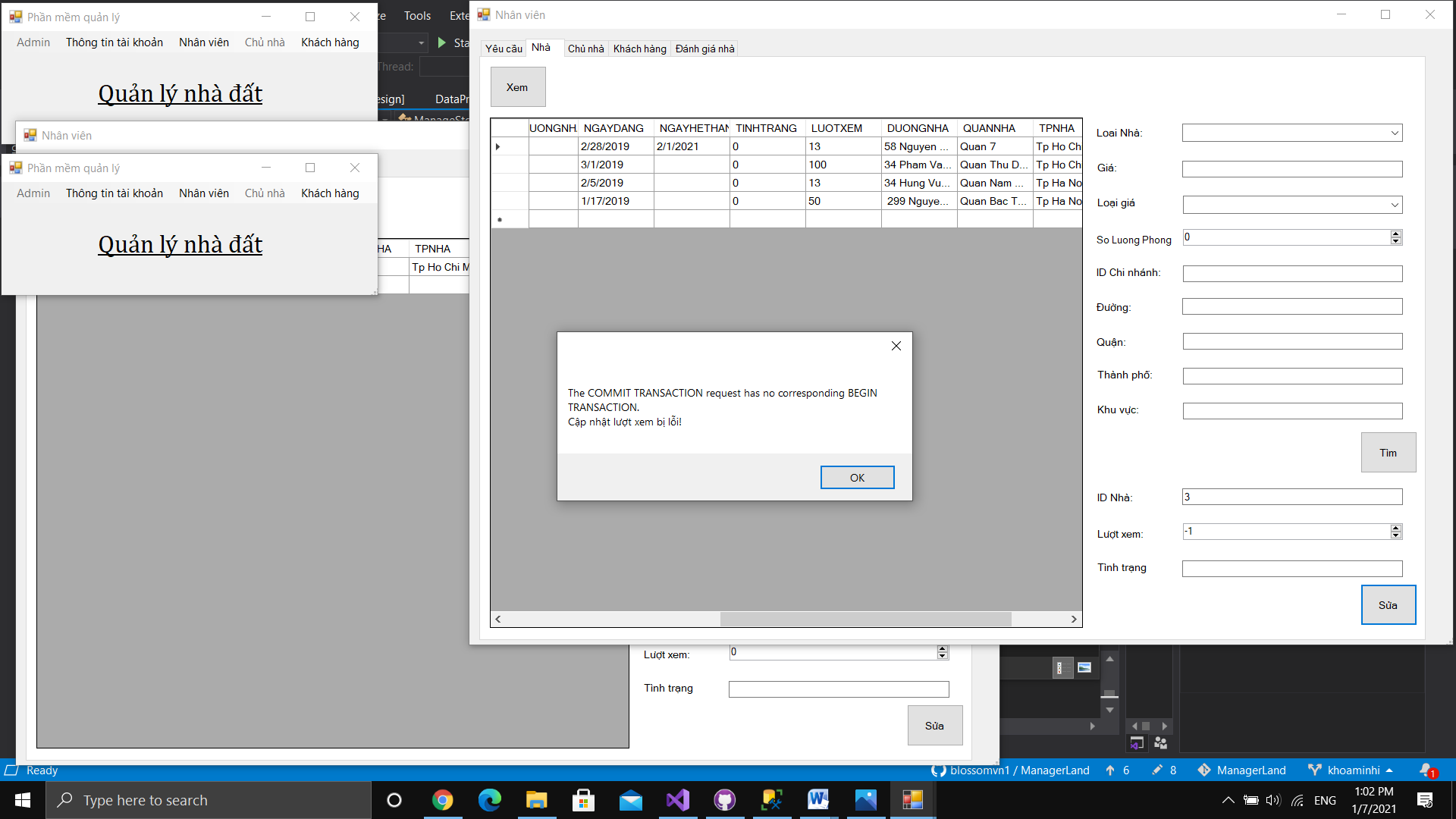
## Tình huống 2: Nhân viên cập nhật lượt xem và chủ nhà vào xem lượt xem.

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR02: Dirty Read**  T1 (User = Nhân viên): thực hiện cập nhật lượt xem của nhà A  T2 (User = Chủ nhà): thực hiện xem lượt xem của nhà A | | | |
| sp\_updateView\_2\_T1 | **Khóa** | **sp\_getView** | **Khóa** |
| ***Input:*** *lượt xem nhà*  ***Output***: rollback | ***Input***: IDNHA  ***Output***: lượt xem rác |
| SET TRANSACTION ISOLATION  LEVEL READ COMMITTED (mặc định) | SET TRANSACTION ISOLATION  LEVEL READ UNCOMMITTED |
| BEGIN TRAN |  |  |  |
| B1: Lấy lượt xem hiên tại của nhà A.  declare @currentViews int  set @currentViews = (select LUOTXEM from NHA where IDNHA = @houseID) | **R(A)**  đọc lượt xem của nhà A trên bảng NHA  Hệ quản trị phát khóa S. Đọc xong trả lại S. |  |  |
| B2: Cập nhật lượt xem tại nhà A  update NHA  set LUOTXEM = @currentViews + @views  where IDCNHA = @houseID | W(A)  xin khóa X, thỏa hqt cấp cho khóa X.  Giữ tới cuối giao tác |  |  |
| waitfor delay '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B2: Đọc lượt xem  select \* from NHA where IDNHA = @houseID | **R(A)**  **Không xin khóa vì bên dưới không có write(A) và không có thao tác gì nữa. Không cin vì để tránh xin gây chậm hệ thống hoặc xẩy ra những tinh huống khác. (Cho nên set uncommitted)** |
|  |  | COMMIT |  |
| B3: Kiểm tra điều kiện lượt xem có thỏa không  if(@currentViews > @currentViews + @views)  begin  print N'Cập nhật lượt xem bị lỗi!'  rollback tran  end |  |  |  |
| ROLLBACK |  |  |  |

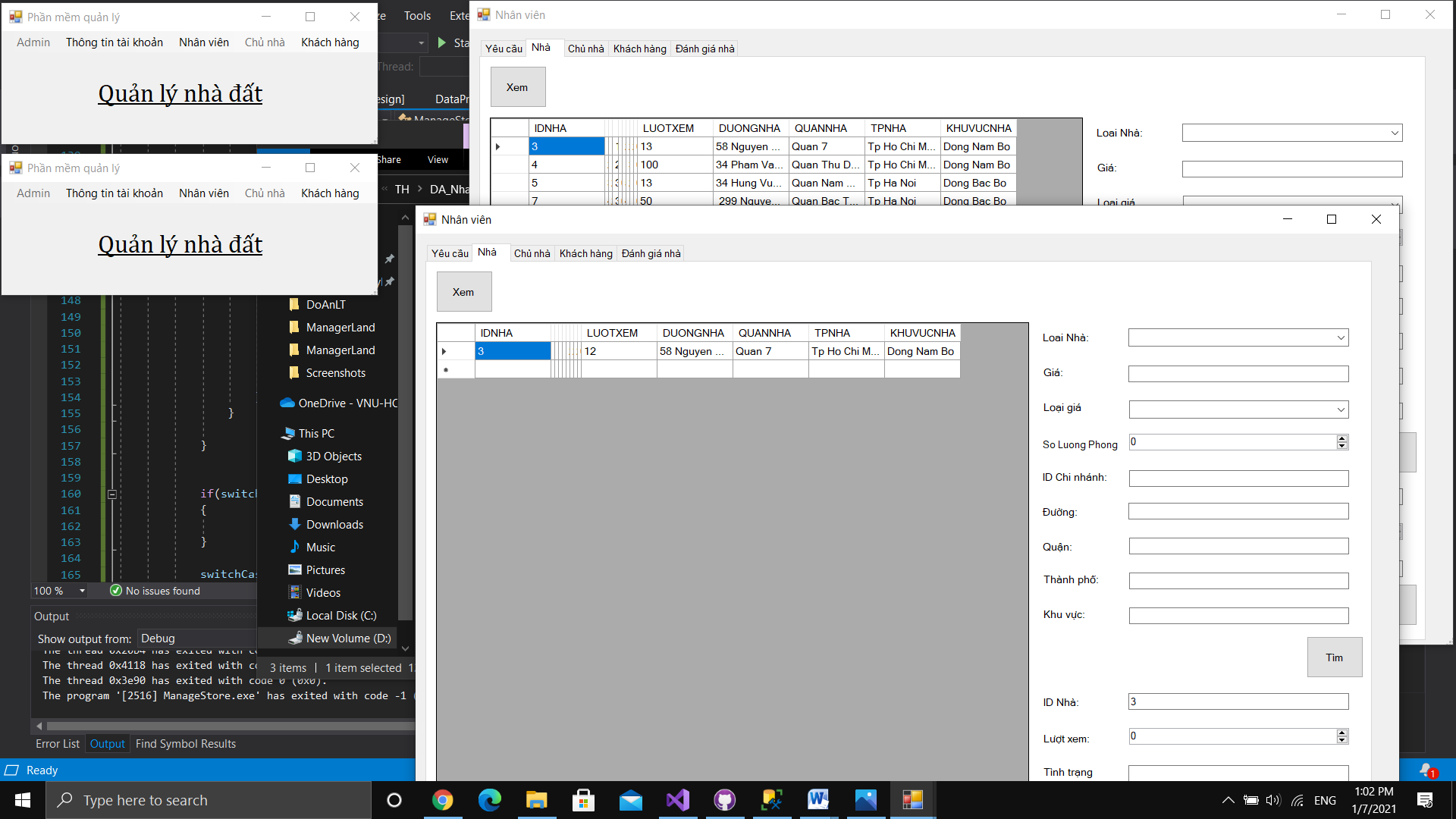
1. Lúc đầu trong nhà id = 3 có view =13. Sau khi T1 cập nhật thì view = 12 (chưa commit), T2 vô dọc được view = 12



1. T1 bị rollback



1. So sánh kết quả tranh chấp



# Sinh viên thực hiện:Huỳnh Ngọc Hòa

## Tình huống 1: Chủ nhà thêm 1 nhà nhưng chưa commit thì nhân viên vào xem thông tin nhà

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: *Dirty Read***  T1 (User = Chủ nhà): thực hiện thêm 1 nhà  T2 (User = Nhân viên): thực hiện xem thông tin nhà | | | |
| **sp\_ThemNha** | **Khóa** | **sp\_XemThongTinNha** | **Khóa** |
| ***Input:*** *Thông tin nhà*  ***Output***: Rollback | ***Input***: không  ***Output***: thông tin tất cả nhà |
| SET TRANSACTION ISOLATION  LEVEL READ COMMITTED (mặc định) | SET TRANSACTION ISOLATION  LEVEL READ COMMITTED |
| BEGIN TRAN |  |  |  |
| B1: Thêm 1 nhà vào bảng NHA.  Insert into NHA values (Thông tin nhà)  waitfor delay '00:00:10' | **W(NHA)**  Chèn thông tin nhà mới vào bảng NHA  Hệ quản trị cấp khóa X và giữ đến hết giao tác | BEGIN TRAN |  |
|  |  | Đọc thông tin nhà.  select \*  from NHA | R(NHA)  **Xin khóa đọc trên bảng NHA nhưng không được vì T1 đang giữ khóa ghi trên nhà(phải chờ)** |
|  |  |  |  |
| B2: kiểm tra điều kiện  if(DATEDIFF(day,@NgayHetHan,@NgayDang)>0)  begin  print N'Ngày hết hạn không được nhỏ hơn ngày đăng'  rollback tran  end |  |  |  |
| ROLLBACK | Trả khóa đang giữ trên bảng NHA |  |  |
|  |  | Tiếp tục đọc thông tin nhà vì T1 đã trả khóa  select \*  from NHA | R(NHA)  Xin khóa đọc trên bảng nhà |
|  |  | COMMIT |  |

## Tình huống 2: Nhân viên đang đọc thông tin nhà thì chủ nhà cập nhập lại nhà khiến lần đọc sau đó của nhân viên không còn chính xác nữa

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: *Unrepeatable Read***  T1 (User = Nhân viên): Đọc thông tin nhà  T2 (User = Chủ nhà): Cập nhập lại địa chỉ nhà | | | |
| **sp\_XemThongTinNha** | **Khóa** | **sp\_CapNhapNha** | **Khóa** |
| ***Input:*** *không*  ***Output***: không | ***Input***: Thông tin mới của nhà cần cập nhập  ***Output***: không |
| SET TRANSACTION ISOLATION  LEVEL REPEATABLE READ | SET TRANSACTION ISOLATION  LEVEL REPEATABLE READ |
| BEGIN TRAN |  |  |  |
| B1: Đọc thông tin nhà.  select \*  from NHA  waitfor delay '00:00:10' | **R(NHA)**  Xin khóa S trên bảng NHA và giữ đến hết giao tác | BEGIN TRAN |  |
|  |  | Cập nhập thông tin nhà.  update NHA  set NHA.DUONGNHA =@DuongNha,  NHA.IDCNHA = @idChuNha,  NHA.IDCNHANH = @idChiNhanh,  NHA.IDLOAINHA = @idLoaiNha,  NHA.IDNV = @idNhanVien,  NHA.KHUVUCNHA = @KhuVucNha,  NHA.LUOTXEM = @LuotXem,  NHA.NGAYDANG=@NgayDang,  NHA.NGAYHETHANG = @NgayHetHan,  NHA.QUANNHA = @QuanNha,  NHA.SOLUONGNHA = @SoLuongNha,  NHA.TINHTRANG = @TinhTrang,  NHA.TPNHA = @TPNha  where NHA.IDNHA = @idNha | W(NHA)  **Xin khóa X trên bảng NHA nhưng không dược do T1 đang giữ khóa S trên NHA (phải chờ)** |
|  |  |  |  |
| B2: Đọc thông tin nhà.  select \*  from NHA | **R(NHA)**  Xin khóa S trên bảng NHA |  |  |
| COMMIT | **Trả khóa S đang giữ trên bảng NHA** |  |  |
|  |  | Tiếp tục cập nhập do khóa đã được trả  update NHA  set NHA.DUONGNHA =@DuongNha,  NHA.IDCNHA = @idChuNha,  NHA.IDCNHANH = @idChiNhanh,  NHA.IDLOAINHA = @idLoaiNha,  NHA.IDNV = @idNhanVien,  NHA.KHUVUCNHA = @KhuVucNha,  NHA.LUOTXEM = @LuotXem,  NHA.NGAYDANG=@NgayDang,  NHA.NGAYHETHANG = @NgayHetHan,  NHA.QUANNHA = @QuanNha,  NHA.SOLUONGNHA = @SoLuongNha,  NHA.TINHTRANG = @TinhTrang,  NHA.TPNHA = @TPNha  where NHA.IDNHA = @idNha | W(NHA)  Xin khóa X trên bảng NHA và giữ đến hết giao tác |
|  |  | COMMIT |  |

# Sinh viên thực hiện:Nguyễn Văn Hảo

## Tình huống 1: 1 admin thực hiện xem thông tin nhân viên thì có một admin khác thêm nhân viên

|  |  |  |  |
| --- | --- | --- | --- |
| 1. **ERR01: *Phantom*** 2. T1 (User = chủ nhà): thực hiện xem 1 thông tin nhân viên 3. T2 (User = admin): thực hiện them một nhân viên | | | |
| **Sp\_XemThongTinNhanVien** | **Khóa** | **sp\_ThemNhanVien** | **Khóa** |
| ***Input:*** Không có  ***Output***: Bảng NHANVIEN | ***Input***: @idNV,@idChiNhanh,@tenNV,@dienthoaiNV,  @gioitinhNV, @ngaysinhNV,@luong, @diachiNV  ***Output***: Không có |
| SET TRANSACTION ISOLATION  LEVEL REPEATABLE READ |  | SET TRANSACTION ISOLATION  LEVEL REPEATABLE READ |  |
| ***BEGIN TRAN*** |  |  |  |
| Select \* from NHANVIEN | **R(NHANVIEN)**  Xin khóa đọc trên bảng NHANVIEN |  |  |
| WAITFOR DELAY '00:00:05.000' |  |  |  |
|  |  | ***BEGIN TRAN*** |  |
|  |  | B1: Kiểm tra thông tin nhân viên  IF EXISTS (SELECT \* FROM NHANVIEN WHERE NHANVIEN.IDNV=@idNV)  BEGIN    END | **R(NHANVIEN)**  //Xin khoá đọc trên bảng NHANVIEN với đk có IDNV=@idNV |
|  |  | Insert into NHANVIEN  Values(  IDNV=@idNV  IDCHINHANH=@idChiNhanh  TENNV=@tenNV  DIENTHOAINV=@dienthoaiNV  GIOITINHNV=@gioitinhNV  NGAYSINHNV=@ngaysinhNV  LUONG=@luong  DIACHINV=@diachiNV) | X(NHANVIEN)  Xin khóa ghi trên bảng NHANVIEN |
|  |  | ***COMMIT*** |  |
| Select \* from NHANVIEN | **R(NHANVIEN)**  Xin khóa đọc trên bảng NHANVIEN |  |  |
| ***COMMIT*** |  |  |  |

## Tình huống 2: Admin thực hiện xem thông tin nhân viên và admin khác thực hiện cập nhật nhân viên

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR02: *Unrepeatable Read***  T1 (User = chủ nhà): thực hiện xem 1 thông tin nhân viên  T2 (User = admin): thực hiện cập nhật một nhân viên | | | |
| **Sp\_XemThongTinNhanVien** | **Khóa** | **sp\_ThemNhanVien** | **Khóa** |
| ***Input:*** Không có  ***Output***: Bảng NHANVIEN | ***Input***: @idNV,@idChiNhanh,@tenNV,@dienthoaiNV,  @gioitinhNV, @ngaysinhNV,@luong, @diachiNV  ***Output***: Không có |
| SET TRANSACTION ISOLATION  LEVEL REPEATABLE READ |  | SET TRANSACTION ISOLATION  LEVEL REPEATABLE READ |  |
| ***BEGIN TRAN*** |  |  |  |
| B1: Xem thông tin nhân viên  Select \* from NHANVIEN | **R(NHANVIEN)**  Xin khóa đọc trên bảng NHANVIEN |  |  |
| WAITFOR DELAY '00:00:05.000' |  |  |  |
|  |  | ***BEGIN TRAN*** |  |
|  |  | B1: Kiểm tra thông tin nhân viên  IF EXISTS (SELECT \* FROM NHANVIEN WHERE NHANVIEN.IDNV=@idNV)  BEGIN    END | **R(NHANVIEN)**  //Xin khoá đọc trên bảng NHANVIEN với đk có IDNV=@idNV |
|  |  | B2: Thêm thông tin vào bảng nhân viên  UPDATE NHANVIEN  SET  IDNV=@idNV  IDCHINHANH=@idChiNhanh  TENNV=@tenNV  DIENTHOAINV=@dienthoaiNV  GIOITINHNV=@gioitinhNV  NGAYSINHNV=@ngaysinhNV  LUONG=@luong  DIACHINV=@diachiNV  Where IDNV=@idNV | **X(NHANVIEN)**  Xin khóa ghi trên bảng NHANVIEN |
|  |  | ***COMMIT*** |  |
| B2: Xem thông tin nhân viên  Select \* from NHANVIEN | **R(NHANVIEN)**  Xin khóa đọc trên bảng NHANVIEN |  |  |
| ***COMMIT*** |  |  |  |

# Sinh viên thực hiện: Trần Thái Đăng Khoa

## Tình huống 1: 1 Nhân viên đang xem nhà cho khách hàng thì chủ nhà thêm một nhà mới

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: *Phantom***  T1 (User = nhân viên): thực hiện tìm nhà nhà cho khách hàng  T2 (User = chủ nhà): thực hiện thêm một nhà mới | | | |
| **Sp\_XemNha** | **Khóa** | **sp\_ThemNha** | **Khóa** |
| ***Input:*** Không có  ***Output***: Bảng NHA | ***Input***: @idChuNha,@idChiNhanh,@idLoaiNha,  @idNhanVien,@KhuVucNha,@NgayDang, @NgayHetHan,@QuanNha,@SoLuongNha,  @TinhTrang,@DuongNha,@TPNha  ***Output***: Không có |
| SET TRANSACTION ISOLATION  LEVEL REPEATABLE READ |  | SET TRANSACTION ISOLATION  LEVEL REPEATABLE READ |  |
| ***BEGIN TRAN*** |  |  |  |
| Select \* from NHA | **R(NHA)**  Xin khóa đọc trên bảng NHA |  |  |
| WAITFOR DELAY '00:00:05' |  | ***BEGIN TRAN*** |  |
|  |  | Insert into NHA  Values( @idChuNha,  @idChiNhanh,  @idLoaiNha,  @idNhanVien,  @KhuVucNha,  @NgayDang,  @NgayHetHan,  @QuanNha,  @SoLuongNha,  @TinhTrang,  @DuongNha,  @TPNha) | **X(NHA)**  Xin khóa ghi trên bảng NHA |
|  |  | ***COMMIT*** |  |
| Select \* from NHA | **R(NHA)**  Xin khóa đọc trên bảng NHA |  |  |
| ***COMMIT*** |  |  |  |

## Tình huống 2: 2 admin đang thao tác trên cùng bảng nhân viên và 1 admin thêm 1 nhân viên, 1 admin cập nhật nhân viên

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR02: *Conversion deadlock***  T1 (User = admin): xem danh sách nhân viên và thực hiện thêm 1 thông tin nhân viên  T2 (User = admin): xem danh sách nhân viên và thực hiện cập nhật thông tin 1 nhân viên | | | |
| **Sp\_ThemNhanVien** | **Khóa** | **sp\_UpdateNhanVien** | **Khóa** |
| ***Input:*** @idNV,@idChiNhanh,@tenNV,@dienthoaiNV,  @gioitinhNV, @ngaysinhNV,@luong, @diachiNV  ***Output***: Bảng NHANVIEN | ***Input***: @idNV,@idChiNhanh,@tenNV,@dienthoaiNV,  @gioitinhNV, @ngaysinhNV,@luong, @diachiNV  ***Output***: Bảng NHANVIEN |
| SET TRAN ISOLATION LEVEL SERIALIZABLE |  | SET TRAN ISOLATION LEVEL SERIALIZABLE |  |
| ***BEGIN TRAN*** |  | ***BEGIN TRAN*** |  |
| B1: Xem thông tin nhân viên  Select \* from NHANVIEN | **R(NHANVIEN)**  Xin khóa S trên bảng NHANVIEN và giữ cho đến khi đọc xong | B1: Xem thông tin nhân viên  Select \* from NHANVIEN | **R(NHANVIEN)**  Xin khóa S trên bảng NHANVIEN và giữ cho đến khi đọc xong |
| WAITFOR DELAY '00:00:05' |  | WAITFOR DELAY '00:00:05' |  |
| B2: Thêm 1 nhân viên  INSERT INTO dbo.NHANVIEN  (  @idnv,  @idcnhanh,  @tennv,  @dienthoainv,  @gioitinhnv,  @ngaysinhnv,  @luongnv,  @diachinv  ) | **X(NHANVIEN)**  Yêu cầu X-lock trên bảng NHANVIEN | B2: Cập nhật thông tin 1 nhân viên  UPDATE NHANVIEN  SET  IDNV=@idNV  IDCHINHANH=@idChiNhanh  TENNV=@tenNV  DIENTHOAINV=@dienthoaiNV  GIOITINHNV=@gioitinhNV  NGAYSINHNV=@ngaysinhNV  LUONG=@luong  DIACHINV=@diachiNV  Where IDNV=@idNV | **X(NHANVIEN)**  Yêu cầu X-lock trên bảng NHANVIEN |
| ***COMMIT*** |  | ***COMMIT*** |  |

## Tình huống 3: 2 chủ nhà cùng thao tác trên bảng nhà và cả 2 đồng thời thêm vào 1 nhà

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR02: *Conversion deadlock***  T1 (User = chủ nhà): xem thông tin nhà và thực hiện thêm 1 nhà  T2 (User = chủ nhà): xem thông tin nhà và thực hiện thêm 1 nhà | | | |
| **Sp\_ThemNha** | **Khóa** | **sp\_ThemNha** | **Khóa** |
| ***Input:*** @idChuNha,@idChiNhanh,@idLoaiNha,  @idNhanVien,@KhuVucNha,@NgayDang, @NgayHetHan,@QuanNha,@SoLuongNha,  @TinhTrang,@DuongNha,@TPNha  ***Output***: Bảng NHA | ***Input***: @idChuNha,@idChiNhanh,@idLoaiNha,  @idNhanVien,@KhuVucNha,@NgayDang, @NgayHetHan,@QuanNha,@SoLuongNha,  @TinhTrang,@DuongNha,@TPNha  ***Output***: Bảng NHA |
| SET TRAN ISOLATION LEVEL SERIALIZABLE |  | SET TRAN ISOLATION LEVEL SERIALIZABLE |  |
| ***BEGIN TRAN*** |  | ***BEGIN TRAN*** |  |
| B1: Xem thông tin nhà  Select \* from NHA | **R(NHA)**  Xin khóa S trên bảng NHA và giữ cho đến khi đọc xong | B1: Xem thông tin nhà  Select \* from NHA | **R(NHA)**  Xin khóa S trên bảng NHA và giữ cho đến khi đọc xong |
| WAITFOR DELAY '00:00:05' |  | WAITFOR DELAY '00:00:05' |  |
| B2: Thêm 1 nhà  Insert into NHA  Values( @idChuNha,  @idChiNhanh,  @idLoaiNha,  @idNhanVien,  @KhuVucNha,  @NgayDang,  @NgayHetHan,  @QuanNha,  @SoLuongNha,  @TinhTrang,  @DuongNha,  @TPNha) | **X(NHA)**  Yêu cầu X-lock trên bảng NHA | B2: Thêm 1 nhà  Insert into NHA  Values( @idChuNha,  @idChiNhanh,  @idLoaiNha,  @idNhanVien,  @KhuVucNha,  @NgayDang,  @NgayHetHan,  @QuanNha,  @SoLuongNha,  @TinhTrang,  @DuongNha,  @TPNha) | **X(NHA)**  Yêu cầu X-lock trên bảng NHA |
| ***COMMIT*** |  | ***COMMIT*** |  |

1. Sinh viên thực hiện: Nguyễn Minh Hiếu
2. Tình huống 1: Nhân viên cập nhật 1 nhà thành đã thuê, sau đó trở về lại trang thông tin của minh. Khách hàng xem thông tin nhân viên, sau đó xem danh sách nhà quản lý của Nhân viên đó

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: *Cycle Deadlock***  T1 (User = Nhân viên): cập nhật 1 nhà thành đã thuê, sau đó trở về lại trang thông tin của minh  T2 (User = Admin): cập nhật thông tin nhân viên, sau đó xem danh sách nhà quản lý của Nhân viên đó | | | |
| **sp\_update** | **Khóa** | **sp\_update2** | **Khóa** |
| ***Input:*** ID Nhân Viên, ID Nhà  ***Output***: thông tin nhân viên | ***Input***: ID Nhân Viên, @ Tên  ***Output***: thông tin Nhân Viên và danh sách nhà |
| SET TRANSACTION ISOLATION  LEVEL READ COMMITTED (mặc định) | SET TRANSACTION ISOLATION  LEVEL READ COMMITTED (mặc định) |
| BEGIN TRAN |  |  |  |
| B1: Cập nhật 1 nhà.  update NHA  set TINHTRANG = 1  where IDNHA = @houseID | **W(A)** |  |  |
| waitfor delay '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1: Thông tin nhân viên  update NHANVIEN  set TENNV = @Ten  where IDNV = @NhanVienID | W(B) |
|  |  | B2: Cập nhật lượt xem mới vào nhà A  select \*  from NHA  where IDNV = @NhanVienID | R(A) |
| B2: Xem lại thông tin của mình  select \*  from NHANVIEN  where IDNV = @NhanVienID | R(B) |  |  |
|  |  | COMMIT |  |
| COMMIT |  |  |  |

1. Tình huống 2: Chủ nhà đổi tên chủ nhà, đổi tình trạng 1 căn nhà, Nhân Viên chọn nhà để quản lý xem thông tin chủ

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | **ERR01: *Cycle Deadlock***  T1 (User = Chủ Nhà): Chủ nhà đổi tên chủ nhà, đổi tình trạng 1 căn nhà  T2 (User = Nhân Viên): chọn nhà để quản lý xem thông tin chủ | | | | | **sp\_update** | **Khóa** | **sp\_view** | **Khóa** | | ***Input:*** ID Chủ Nhà, ID Nhà, Tên  ***Output***: cập nhật thành công | ***Input***: ID Nhân Viên, ID Nhà  ***Output***: thông tin Nhân Viên và danh sách nhà | | SET TRANSACTION ISOLATION  LEVEL READ COMMITTED (mặc định) | SET TRANSACTION ISOLATION  LEVEL READ COMMITTED (mặc định) | | BEGIN TRAN |  |  |  | | B1: Cập nhật tên chủ nhà  update CHUNHA  set TENCNHA = @Ten  where IDCNHA = @ChuNhaID | **W(A)** |  |  | | waitfor delay '00:00:05' |  |  |  | |  |  | BEGIN TRAN |  | |  |  | B1: Chọn nhà quản lý  update NHA  set IDNV = @NhanVienID  where IDNHA = @houseID    declare @IDChuNha int  set @IDChuNha = (select IDCNHA from NhA where IDNHA = @houseID) | W(B) | |  |  | B2: Xem thông tin chủ  select \*  from CHUNHA  where IDCNHA = @IDChuNha | R(A) | | B2: Cập nhật 1 nhà.  update NHA  set TINHTRANG = 1  where IDNHA = @houseID | W(B) |  |  | |  |  | COMMIT |  | | COMMIT |  |  |  | |