**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 | 18424040 | Lê Hoàng Luật | Cài đặt tình huống | 100% |
| 2 | 18424027 | Nguyễn Xuân Hiếu | Cài đặt tình huống | 100% |
| 3 | 18424010 | Lê Quốc Bình | Cài đặt tình huống | 100% |
| 4 | 18424034 | Huỳnh Anh Khoa | Cài đặt tình huống | 100% |

# XỬ LÝ TÌNH HUỐNG TRANH CHẤP

# Sinh viên thực hiện:

## Tình huống 1: Đăng nhập vào hệ thống

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: *Unrepeatable read***  T1 (User = Nhân viên): Đăng nhập vào hệ thống.  T2 (User = Admin): Reset password của nhân viên | | | |
| **sp\_ReadLogin\_NhanVien** | **Khóa** | **sp\_UpdateUsernamePass\_NhanVien** | **Khóa** |
| ***Input: @username, @password***  ***Output***: ***ID, Ma, Ten, ChiNhanh, ChucVu*** | ***Input***: ***@ID, @Username , @Password***  ***Output***: ***int*** |
| SET TRANSACTION ISOLATION  LEVEL **Repeatable Read** |  |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra Username có tồn tại  IF NOT EXISTS(SELECT \* FROM NhanVien WHERE Username = @username)  BEGIN  PRINT N'NOT EXISTS Username'  ROLLBACK TRAN  RETURN  END | **R(NhanVien)**  //Xin khoá đọc trên bảng NhanVien với dk Username = @username |  |  |
| B2: Kiểm tra Username, Password đúng sai  IF NOT EXISTS(SELECT \* FROM NhanVien WHERE Username = @username and Pass = @pass)  BEGIN  PRINT N'Error Password'  ROLLBACK TRAN  RETURN  END |  |  |  |
| WAITFOR DELAY '00:00:05' |  | BEGIN TRAN |  |
|  |  | B1: Kiểm tra Username, ID có tồn tại  if Exists(select ID from NhanVien where Username = @Username and ID <> @ID)  begin  return -3;  end | **R(NhanVien)**  //Xin khoá đọc trên bảng NhanVien với dk Username = @username and ID <> @ID |
|  |  | B1: Qua được B1, tiến hành update  update NhanVien set Username = @Username, Pass = @Password where ID = @ID | **X(NhanVien)**  //Xin khóa ghi trên bảng NhanVien  //SQL không cấp khóa X do T1 đang giữ khóa S |
|  |  | //Update **NhanVien**: dữ liệu bị lỗi, không thể update **NhanVien** được. Điều này dẫn đến toàn dữ liệu sẽ bị ROLLBACK |  |
| B3: Read thông tin nhân viên  select ID, Ma, Ten, IDChiNhanh, IDChucVu  from NhanVien  where Username = @username and Pass = @pass |  | COMMIT |  |
| COMMIT |  |  |  |

## Tình huống 2: Thống kê hợp đồng trong tháng

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR02: *Phantom***  T1 (User = admin): thống kê lịch xem nhà trong thời gian  T2 (User = nhân viên): xóa 1 lịch xem nhà trong khoảng thời gian admin truy vấn | | | |
| **sp\_Read\_BaoCaoXemNha\_Fix** | **Khóa** | **sp\_DeleteXemNha** | **Khóa** |
| ***Input: @tuNgay, @denNgay, @idChiNhanh***  ***Output***: bảng table sau khi tính toán | ***Input***: @ID  ***Output***: int |
| SET TRANSACTION ISOLATION  LEVEL SERIALIZABLE |  |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra dữ liệu có tồn tại  Declare @count int  SET @count = (Select Count(xn.ID)  from XemNha as xn  inner join KhachHang as kh on xn.IDKH = kh.ID  inner join NhanVien as nv on xn.IDNV = nv.ID  inner join NhaChoThue as nct on xn.IDNhaThue = nct.ID  inner join ChiNhanh as cn on nct.IDChiNhanh = cn.ID  where CONVERT(date, xn.ThoiGianXem) >= @tuNgay  and CONVERT(date, xn.ThoiGianXem) <= @denNgay  and cn.ID = @idChiNhanh) | **R(XemNha)**  //Xin khóa đọc XemNha, KhachHang, NhanVien, NhaChoThue, ChiNhanh điều kiện Thời gian xem ngày trong khoảng ngày |  |  |
| IF @count <= 0  BEGIN  PRINT N'Khong co du lieu'  ROLLBACK TRAN  RETURN  END |  |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1: Kiểm tra dữ liệu có tồn tại  if not Exists(select ID from XemNha where ID = @ID)  begin  ROLLBACK TRAN  return -3;  end | **R(XemNha)**  //Xin khoá đọc trên bảng NhanVien với dk ID = @ID |
|  |  | B2: Xóa lịch xen nhà  delete from XemNha where ID = @ID  set @result = 1 | **X(XemNha)**  //Xin khóa ghi trên bảng XemNha với đk ID = @ID |
|  |  | //Lệnh xóa lịch xem nhà bị lỗi, không thể xóa, dẫn đến ROLLBACK |  |
| B2: Tiến hành đọc dữ liệu  select xn.ID, kh.Ten as KhachHang, nv.Ten as NhanVien, nct.DiaChi, xn.ThoiGianXem, xn.NhanSet, cn.Ten as ChiNhanh, xn.IDKH as IDKhachHang, xn.IDNV as IDNhanVien, xn.IDNhaThue, nct.IDChiNhanh  from XemNha as xn  inner join KhachHang as kh on xn.IDKH = kh.ID  inner join NhanVien as nv on xn.IDNV = nv.ID  inner join NhaChoThue as nct on xn.IDNhaThue = nct.ID  inner join ChiNhanh as cn on nct.IDChiNhanh = cn.ID  where CONVERT(date, xn.ThoiGianXem) >= @tuNgay  and CONVERT(date, xn.ThoiGianXem) <= @denNgay  and cn.ID = @idChiNhanh  order by xn.ID |  | COMMIT |  |
| COMMIT |  |  |  |

## Tình huống 3: Sửa giá thuê nhà

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR03: *Dirty read***  T1 (User = nhân viên): sửa giá thuê nhà  T2 (User = khách hàng): xem giá thuê nhà | | | |
| **sp\_UpdateGiaNhaChoThue** | **Khóa** | **sp\_ReadDanhSachNhaChoThue** | **Khóa** |
| ***Input: @maNha, @giamoi***  ***Output***: true or false | ***Input***:  ***Output***: danh sách nhà cho thuê, giá |
| SET TRANSACTION ISOLATION  LEVEL READ COMMITTED | SET TRANSACTION ISOLATION  LEVEL READ COMMITTED |
| BEGIN TRAN |  |  |  |
| if not Exists(select ID from NhaChoThue where Ma = @Ma and ID = @ID)  begin  return -3; --khong ton tai  end | **R(A)**  //Xin khóa đọc trên bảng NhaChoThue với đk Ma = @Ma and ID = @ID |  |  |
| update NhaChoThue set GiaThue = @Gia where ID = @ID | **X(A)**  //xin khóa ghi với điều kiện ID = @ID |  |  |
| WAITFOR DELAY '00:00:05' |  | BEGIN TRAN |  |
|  |  | select nct.ID, nct.Ma, nct.DiaChi, nct.KhucVuc, nct.SoPhong, nct.GiaThue, l.Ten as LoaiNha, cn.Ten as ChiNhanh,  kh.Ten as ChuNha, nct.IDLoaiNha, nct.IDChiNhanh, nct.IDChuNha  from NhaChoThue as nct  inner join LoaiNha as l on nct.IDLoaiNha = l.ID  inner join ChiNhanh as cn on nct.IDChiNhanh = cn.ID  inner join KhachHang as kh on nct.IDChuNha = kh.ID | R(A)  //xin khóa đọc  //Sql không cấp khóa đọc |
| ROLLBACK TRAN |  |  |  |
| COMMIT |  | COMMIT |  |