

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	20120139	Mai Nhật Nam		
2	20120115	Nguyễn Đặng Nam Khánh		
3	20120223	Thái Minh Triết		
4	20120065	Lâm Quang Duy		

CÀI ĐẶT TÌNH HUỐNG TRANH CHẤP

I. Sinh viên thực hiện: Lâm Quang Duy

1. Tình huống 1: Chi nhánh xem các đơn đặt hàng để chọn lọc danh sách các đơn đặt hàng cần phải chuẩn bị. Cùng lúc đó khách hàng muốn huỷ đơn hàng ở trạng thái “chờ nhận” của mình bằng cách Xóa đơn đặt hàng. Chi nhánh xem lại đơn đặt hàng để gửi cho đầu bếp chuẩn bị các món ăn tuy nhiên không xem được nữa.

ERR01: Unrepeatable Read T1 (User = Chi nhánh): thực hiện cập nhật thông tin đơn hàng A để nấu món ăn. T2 (User = khách hàng): thực hiện xóa đơn hàng A của mình (kèm điều kiện tình trạng đơn hàng “Chờ nhận”).			
sp_ChiNhanhKiemTraDonHang	Khóa	sp_KhachHangXoaDonHang	Khóa
Input: Mã chi nhánh, mã đơn đặt hàng. Output: cập nhật đơn đặt hàng.		Input: Mã khách hàng, mã đơn hàng. Output: Xóa đơn đặt hàng trên.	
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ		SET TRANSACTION ISOLATION LEVEL REPEATABLE READ	
BEGIN TRAN			
B1: Chi nhánh kiểm tra các đơn đặt hàng <pre> if(not exists(select * from DonDatHang where idChiNhanh = @MaCN)) begin raiserror('Chi nhánh không có đơn đặt hàng', 16, 1) rollback tran return end </pre>	T1 S(DonDatHang) T1 Xin được khóa S(Share) trên đơn đặt hàng và giữ đến cuối giao tác		
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		B1: Kiểm tra tình trạng đơn hàng <pre> if((select TinhTrang from DonDatHang where idDonDatHang = @idDonDatHang) = N'Chờ nhận') </pre>	

		<pre> begin delete ChiTietDDH where idDonDatHang = @idDonDatHang -- Do ràng buộc khóa chính khóa ngoại delete DonDatHang where idDonDatHang = @idDonDatHang raiserror(N'Xóa thành công', 16,1) end else begin raiserror(N'Đơn hàng của bạn hiện không thể xóa', 16, 1) raiserror(N'Xóa không thành công', 16,1) rollback tran return end rollback tran return end </pre>	<p>T2 X(DonDatHang)</p> <p>T2 không thể xin được khóa X (Exclusive) trên DonDatHang do T1 đang giữ khóa S</p>
		COMMIT TRAN	
<p>B2: Chi nhánh lọc đơn hàng ở trạng thái “chờ nhận”</p> <pre> select * from DonDatHang where TinhTrang = N'Chờ nhận' and idChiNhanh = @MaCN </pre>	<p>T1 S(DonDatHang)</p> <p>T1 xin được khóa S (Share) trên đơn đặt hàng.</p>		
COMMIT TRAN			
<p>Kết quả: Ban đầu chi nhánh xem đơn hàng, sau đó khách hàng gửi yêu cầu xóa đơn hàng. Yêu cầu xóa đơn hàng của khách hàng được bộ lập lịch cho thực hiện sau khi chi nhánh hoàn thành việc xem đơn hàng, nên dữ liệu chi nhánh xem được không thay đổi (có thể đọc lại dữ liệu).</p>			

2. Tình huống 2: Tài xế nhận đơn đặt hàng và bấm xác nhận, chuyển đơn đặt hàng sang trạng thái “Đã nhận đơn hàng”., Chi nhánh kiểm tra đơn đặt hàng thấy hết món nên chuyển sang trạng thái huỷ đơn. Sau đó tài xế cập nhật tình trạng ‘Đã nhận đơn hàng’ ghi đè lên kết quả cập nhật của chi nhánh.

ERR01: Lost Update T1 (User = Khách hàng): thực hiện cập nhật thông tin đơn đặt hàng. T2 (User = Tài xế): thực hiện cập nhật thông tin đơn đặt hàng.			
sp_TaiXeCapNhatDonHang	Khóa	sp_ChiNhanhCapNhatDonHang	Khóa
Input: Mã tài xế, mã đơn đặt hàng. Output: Cập nhật đơn đặt hàng.		Input: Mã chi nhánh, mã đơn đặt hàng. Output: Cập nhật đơn đặt hàng.	
SET TRANSACTION ISOLATION LEVEL SNAPSHOT		SET TRANSACTION ISOLATION LEVEL SNAPSHOT	
BEGIN TRAN			
B1: Kiểm tra tình trạng đơn hàng <pre>if((select TinhTrang from DonDatHang where idDonDatHang = @idDonDatHang) = N'Chờ nhận')</pre>	T1 S(DonDatHang) T1 Xin được khóa S(Share) trên đơn đặt hàng và giữ đến cuối giao tác		
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		B1: Kiểm tra đơn đặt hàng có thuộc chi nhánh <pre>if(not exists(select * from DonDatHang where idDonDatHang = @idDonDatHang and idChiNhanh = @MaCN)) begin raiserror('Chi nhánh không có đơn đặt hàng này', 16, 1) raiserror(N'Cập nhật không thành công', 16,1) rollback tran</pre>	

		<pre> return end </pre>	
		<p>B2: Chi nhánh huỷ đơn hàng</p> <pre> if((select TinhTrang from DonDatHang where idDonDatHang = @idDonDatHang) = N'Chờ nhận') begin update DonDatHang set TinhTrang = N'Đã huỷ' where idDonDatHang = @idDonDatHang update DonDatHang set idChiNhanh = @MaCN where idDonDatHang = @idDonDatHang raiserror(N'Cập nhật thành công', 16,1) end else begin raiserror(N'Dơn hàng của bạn hiện không thể xóa', 16, 1) raiserror(N'Cập nhật không thành công', 16,1) rollback tran return end </pre>	<p>T2 X(DonDatHang)</p> <p>T2 không thể xin được khóa X (Exclusive) trên DonDatHang do T1 đang giữ khóa S</p>
		COMMIT TRAN	
<p>B2: Tài xế cập nhật trạng thái đơn hàng</p> <pre> update DonDatHang set TinhTrang = N'Đã nhận đơn hàng' where idDonDatHang = @idDonDatHang update DonDatHang set idTaiXe = @MaTX where idDonDatHang = @idDonDatHang </pre>	<p>T1 X(DonDatHang)</p> <p>T1 xin được khóa X (Exclusive) trên DonDatHang</p>		
COMMIT TRAN			
<p>Kết quả: Ban đầu tài xế thấy đơn hàng ở trạng thái “chờ nhận” thì xác nhận đơn hàng. Ngay lúc đó cửa hàng thấy hết món nên vào update huỷ đơn hàng. Bộ lập lịch cho phép cửa hàng huỷ đơn hàng, và chặn việc xác nhận đơn hàng của tài xế</p>			

3. Tình huống 3: Chi nhánh thống kê các đơn đặt hàng ở trạng thái “Chờ nhận. Ngay lúc đó khách hàng vào đặt thêm một vài đơn đặt hàng. Và sau đó, chi nhánh đưa danh sách đơn đặt hàng cho đầu bếp thực hiện thì thấy con số thống kê ở trên và số dòng mình in ra không khớp

ERR01: Phantom T1 (User = Chi nhánh): thực hiện thống kê số lượng đơn hàng và xem thông tin đơn đặt hàng. T2 (User = Khách hàng): thực hiện thêm đơn đặt hàng.			
sp_DoiTacThongKe	Khóa	sp_KhachHangThemDongHang	Khóa
<u>Input:</u> Mã chi nhánh. <u>Output:</u> Số lượng đơn hàng, danh sách đơn hàng.		<u>Input:</u> Mã khách hàng, mã đơn đặt hàng, địa chỉ nhận, phí vận chuyển, hình thức thanh toán, mã chi nhánh, số lượng sản phẩm đặt, mã món. <u>Output:</u> Thêm đơn đặt hàng.	
SET TRANSACTION ISOLATION LEVEL SERIALIZABLE			
BEGIN TRAN			
B1: Kiểm tra chi nhánh có tồn tại <pre> if(not exists(select * from ChiNhanh where idChiNhanh = @MaCN)) begin raiserror(N'Mã chi nhánh không tồn tại', 16, 1) raiserror(N'Xóa không thành công', 16,1) rollback tran return end </pre>			
B2: Thống kê số lượng đơn đặt hàng ở trạng thái 'Chờ nhận' <pre> declare @SL_DonHang int set @SL_DonHang = (select count(*) </pre>	T1 S(DonDatHang)		

ChiNhanh CN, DonDatHang DDH where CN.idChiNhanh = DDH.idChiNhanh and DDH.TinhTrang = N'Chờ nhận')	from T1 xin được khóa S (Share) trên đơn đặt hàng và giữ đến cuối giao tác.		
B3: In số lượng đơn đặt hàng print(N'Số lượng đơn hàng ở trạng thái chờ nhận của cửa hàng ' + cast(@MaCN as varchar(4)) + N' là: ' + cast(@SL_DonHang as varchar(4)))			
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		B1: Kiểm tra thông tin của khách hàng if(not exists(select * from KhachHang where idKhachHang = @MaKH)) begin raiserror(N'Mã khách hàng không tồn tại', 16, 1) raiserror(N'Xóa không thành công', 16,1) rollback tran return end	
		B2: Kiểm tra đơn hàng có tồn tại if(exists(select * from DonDatHang where idDonDatHang = @idDonDatHang)) begin raiserror(N'Dơn đặt hàng đã tồn tại', 16,1) rollback tran return end	
		B3: Kiểm tra chi nhánh có nằm trong hệ thống	

		<pre> if(not exists(select * from ChiNhanh where idChiNhanh = @idChiNhanh)) begin raiserror(N'Chi nhánh không tồn tại trong hệ thống', 16,1) rollback tran return end </pre>	
		<p>B4: Thêm đơn đặt hàng</p> <pre> SET IDENTITY_INSERT DonDatHang ON insert into DonDatHang (idDonDatHang, TinhTrang, NgayLap, NgayThanhToan, HinhThucThanhToan, DiaChiNhan, PhiSanPham, PhiVanChuyen, Rating, Comment, idChiNhanh, idTaiXe, idKhachHang) values(@idDonDatHang, @TinhTrang, @NgayLap, @NgayThanhToan, @HinhThucThanhToan, @DiaChiNhan, @PhiSanPham, @PhiVanChuyen, @Rating, @Comment, @idChiNhanh, @idTaiXe, @MaKH) insert into ChiTietDDH values(@MaMon, @idDonDatHang, @SLSanPham, @TuyChon) </pre>	<p>T2 X(DonDatHang)</p> <p>T2 không thể xin khóa X (Exclusive) trên đơn đặt hàng do T1 đang giữ khóa S.</p> <p>T2 không thể INSERT dòng dữ liệu mặc dù đã thỏa điều kiện.</p>
		COMMIT TRAN	
<pre> select DDH.idDonDatHang, DDH.TinhTrang, DDH.DiaChiNhan, DDH.HinhThucThanhToan, DDH.NgayLap, DDH.idTaiXe, DDH.idChiNhanh from ChiNhanh CN, DonDatHang DDH where CN.idChiNhanh = DDH.idChiNhanh and CN.idChiNhanh = @MaCN and DDH.TinhTrang = N'Chờ nhận' </pre>	<p>T1 S(DonDatHang)</p> <p>T1 xin được khóa S (Share) trên đơn đặt hàng.</p>		
COMMIT TRAN			
<p>Kết quả: Ban đầu chi nhánh vào thống kê số lượng đơn hàng ở trạng thái “chờ nhận” của mình. Ngay lúc đó, khách hàng vào đặt thêm đơn đặt hàng. Bộ lập lịch sắp xếp việc khách hàng thêm đơn hàng sau khi hệ thống thực hiện thống kê xong. Nên số lượng đơn hàng thống kê và số lượng đơn hàng in ra của hệ thống là giống nhau. Sau đó mới cho phép khách hàng thêm đơn đặt hàng vào.</p>			

