

BÁO CÁO ĐỒ ÁN THỰC HÀNH
MÔN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

GVHD:

Lý thuyết: Thầy Tuấn Nguyễn Hoài Đức

Thực Hành: Cô Tiết Gia Hồng

Cô Hồ Thị Hoàng Vy

THÔNG TIN NHÓM N02

STT	MSSV	Họ tên	Công việc	% Hoàn thành
1	20127099	Võ Ngọc Khánh Vy	Cài đặt tình huống Dirty read,	100%
2	20127154	Nguyễn Thị Thanh Hằng	Cài đặt tình huống lost update	100%
3	20127577	Phan Nguyễn Phước Nguyên	Cài đặt tình huống Phantom,	100%
4	20127317	Phạm Minh Tài	Cài đặt tình huống unrepeatable read	100%

MỤC LỤC

THÔNG TIN NHÓM N02	2
SỬA TÌNH HUỐNG TRANH CHẤP.....	4
I. Sinh viên thực hiện: 20127154	4
1. Tình huống 1: LOST UPDATE	4
2. Tình huống 2: LOST UPDATE	8
II. Sinh viên thực hiện: 20127099	16
1. Tình huống 2: Dirty read	16
2. Tình huống 3: Dirty read	23
III. Sinh viên thực hiện: 20127317	28
1. Tình huống 1: Unrepeatable read	28
2. Tình huống 2: Unrepeatable read	38
IV. Sinh viên thực hiện: 20127577	45
1. Tình huống 1: Phantom	45
2. Tình huống 2: Phantom.....	50


SỬA TÌNH HUỐNG TRANH CHẤP



I. Sinh viên thực hiện: 20127154

1. Tình huống 1: LOST UPDATE




Tài xế A cập nhật tình trạng đơn hàng (tiến hành nhận đơn để giao), cùng lúc đó tài xế B cũng cập nhật đúng đơn hàng tài xế A vừa cập nhật.

Sửa lỗi: thiết lập uplock trên bảng DON_DH, đổi mức cô lập của 2 giao tác thành read committed

ERR01: Lost update T1 (User = tài xế): cập nhật tình trạng đơn hàng T2 (User = tài xế): cập nhật tình trạng đơn hàng			
sp_CapNhat_DH_TX		sp_CapNhat_DH_TX	
Input: @TRANGTHAIDH, @MATX, @MADH Output:		Input: @TRANGTHAIDH, @MATX, @MADH Output:	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
B1: kiểm tra đơn hàng có tồn tại hay không  IF NOT EXISTS(SELECT MADH FROM DON_DH WHERE MADH=@MADH)	T1 xin khóa S trên bảng DON_DH tại dòng MADH=@MADH SQL: cấp khóa S(DON_DH) và trả khóa S		

<pre> BEGIN ROLLBACK TRAN RAISERROR ('KHONG TON TAI DON HANG',16,1) RETURN END </pre>			
<p>B2: kiểm tra tài xế có tồn tại hay không</p>  <pre> IF NOT EXISTS(SELECT MATX FROM TAIXE WHERE MATX=@MATX) BEGIN ROLLBACK TRAN RAISERROR ('KHONG TON TAI TAI XE',16,1) RETURN END </pre>	<p>T1 xin khóa S trên bảng TAIXE tại dòng MATX=@MATX</p> <p>SQL: cấp khóa S(TAIXE) và trả khóa S</p>		
<p>B3.1: kiểm tra xem có được phép cập nhật trạng thái đơn hàng sang 'DELEVERING' không</p>  <pre> IF (@TRANGTHAIDH='DELIVERING') BEGIN </pre>	<p>T1 xin khóa S trên bảng DON_DH tại dòng MADH=@MADH AND TRANGTHAIDH ='PROCESSING'</p>		

IF NOT EXISTS (SELECT MADH FROM DON_DH with(updlock) WHERE MADH=@MADH AND TRANGTHAIDH ='PROCESSING') BEGIN ROLLBACK TRAN RAISERROR ('KHONG NHAN DON DUOC',16,1) RETURN END	SQL: cấp khóa S và giữ đến khi commit		
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		B1: kiểm tra đơn hàng có tồn tại hay không IF NOT EXISTS(SELECT MADH FROM DON_DH WHERE MADH=@MADH) BEGIN ROLLBACK TRAN RAISERROR ('KHONG TON TAI DON HANG',16,1) RETURN END	T2 chờ T1 commit và nhả khóa khỏi bảng mới được kiểm tra
B3.2: cập nhật trạng thái đơn hàng UPDATE DON_DH SET MATX=@MATX, TRANGTHAIDH=@TRANGTHAIDH WHERE MADH=@MADH IF @@ERROR <>0 BEGIN ROLLBACK TRAN RAISERROR ('KHONG NHAN DON DUOC',16,1) RETURN END COMMIT TRAN	T1 xin khóa X(DON_DH) trên dòng có MATX=@MATX SQL cấp khóa X giữ tới khi commit		

RETURN END	T1 commit và kết thúc giao tác		
		B1: kiểm tra đơn hàng có tồn tại hay không IF NOT EXISTS(SELECT MADH FROM DON_DH WHERE MADH=@MADH) BEGIN ROLLBACK TRAN RAISERROR ('KHONG TON TAI DON HANG',16,1) RETURN END	T2 xin khóa S trên bảng DON_DH SQL: cấp khóa S và trả khóa S
		B2: kiểm tra tài xế có tồn tại hay không IF NOT EXISTS(SELECT MATX FROM TAIXE WHERE MATX=@MATX) BEGIN ROLLBACK TRAN RAISERROR ('KHONG TON TAI TAI XE',16,1) RETURN END	T2 xin khóa S trên bảng TAIXE SQL: cấp khóa S(TAIXE) và trả khóa S
		B3: kiểm tra xem có được phép cập nhật trạng thái đơn hàng sang 'DELEVERING' không IF (@TRANGTHAIDH='DELIVERING') BEGIN IF NOT EXISTS (SELECT MADH FROM DON_DH with(updlock) WHERE MADH=@MADH AND TRANGTHAIDH='PROCESSING') BEGIN	T2 xin khóa S trên bảng DON_DH tại dòng MADH=@MADH AND

		ROLLBACK TRAN RAISERROR ('KHONG NHAN DON DUOC',16,1) RETURN END UPDATE DON_DH SET MATX=@MATX, TRANGTHAIDH=@TRANGTHAIDH WHERE MADH=@MADH IF @@ERROR <>0 BEGIN ROLLBACK TRAN RAISERROR ('KHONG NHAN DON DUOC',16,1) RETURN END COMMIT TRAN RETURN END	TRANGTHAIDH ='PROCESSING' SQL: cấp khóa S và giữ đến khi commit T2 xin khóa X(DON_DH) trên dòng có MATX=@MATX SQL cấp khóa X giữ tới khi commit T2 commit và kết thúc giao tác
--	--	--	---


2. Tình huống 2: LOST UPDATE


Khách hàng A tiến hành đặt cùng lúc 2 đơn hàng (mở 2 web), nhưng tài khoản ví chỉ bị trừ tiền 1 lần

Sửa: Thiết lập uplock khi đọc trên bảng KHACHHANG và đổi mức cô lập của 2 giao tác thành read committed

ERR02: Lost Update T1 (User = khách hàng): thực hiện đặt đơn hàng T2 (User = khách hàng): thực hiện đặt đơn hàng			
sp_Tao_DH	Khóa	sp_Tao_DH	Khóa
Input: @HINHTHUCTT, @MADIACHI, @MAKH, @MACN, @PHIVANCHUYEN, @LIST_MONAN		Input: @HINHTHUCTT, @MADIACHI, @MAKH, @MACN, @PHIVANCHUYEN, @LIST_MONAN	



Output:		Output:	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
<p>B1: Kiểm tra tình trạng cửa hàng có đang hoạt động hay không và hình thức thanh toán</p> <p>IF (@HINHTHUCTT='CHUYEN KHOAN' OR @HINHTHUCTT='VI' OR @HINHTHUCTT='NGAN HANG')</p> <p>BEGIN</p> <p>IF NOT EXISTS(SELECT MACN FROM CUAHANG WHERE MACN=@MACN AND TINHTRANGCH='AVAILABLE')</p> <p>BEGIN</p> <p>ROLLBACK TRAN</p> <p>RAISERROR ('CUA HANG KHONG HOAT DONG',16,1)</p> <p>RETURN</p> <p>END</p>	<p>T1 xin khóa S trên bảng CUAHANG</p> <p>SQL: cấp khóa S(CUAHANG) và trả khóa S</p>		


 <p>B2: Tạo biến lưu lại số dư ví và tạo tự động các thông tin cần cho đơn hàng</p> <pre> DECLARE @SODUVI INT=(SELECT SODUVI FROM KHACHHANG with(updlock) WHERE MAKH=@MAKH) DECLARE @MADH INT EXEC sp_IDMAX 'DON_DH',@MADH OUT DECLARE @NGAYTAO DATETIME=GETDATE() DECLARE @MAKHUVUC VARCHAR(10) = NULL Thêm đơn hàng mới EXEC sp_ThemDonDH @MADH,'AVAILABLE',@NGAYTAO,@HINHTHUC TT,0,@PHIVANCHUYEN,@MADIACHI, @MAKH, @MACN,NULL,@MAKHUVUC IF @@ERROR<>0 BEGIN ROLLBACK TRAN RAISERROR ('KHONG TAO DUOC DON HANG',16,1) RETURN </pre>	<p>T1 xin khóa S(KHACHHANG)</p> <p>SQL: cấp khóa S và giữ đến khi commit</p> <p>T1 xin khóa X trên bảng PHAMVIBANG với dòng có TENPHAMVI='DON_D H'</p> <p>SQL cấp khóa X(TENPHAMVI) và giữ đến khi T1 commit</p> <p>T1 xin khóa X trên bảng DON_DH</p> <p>SQL cấp khóa X(DON_DH) và giữ đến khi T1 commit</p> <p>T1 xin khóa X trên THUCDONDATHANG</p> <p>SQL cấp khóa X(THUCDONDATHAN G) và giữ đến khi T1 commit</p>		
--	--	--	--

<pre>END DECLARE @STT INT =(SELECT MAX(STT) FROM @list_MONAN) DECLARE @I INT = 1 WHILE (@I<= @STT) BEGIN DECLARE @MAMONAN VARCHAR(10) DECLARE @SOLUONG INT SELECT @MAMONAN=MAMONAN,@SOLUONG=SOLUON G FROM @list_MONAN WHERE STT=@I EXEC sp_ThemThucDonDatHang @MAMONAN,@MADH,@SOLUONG IF @@ERROR <>0 BEGIN ROLLBACK TRAN RAISERROR ('KHONG TAO DUOC DON HANG',16,1) RETURN END SET @I=@I+1 END</pre> 			
---	--	--	--

<p>B3: Tính tổng tiền của đơn hàng</p> <pre> IF (@HINHTHUCTT='VI') BEGIN DECLARE @TONGTIEN INT=(SELECT PHIVANCHUYEN+PHISP FROM DON_DH WHERE MADH=@MADH) </pre>	<p>T1 xin khóa S trên bảng DON_DH SQL: cấp khóa S</p>		
<p>WAITFOR DELAY '00:00:10'</p>			
		<p>BEGIN TRAN</p>	
		<p>B1: Kiểm tra tình trạng cửa hàng có đang hoạt động hay không</p> <pre> IF (@HINHTHUCTT='CHUYEN KHOAN' OR @HINHTHUCTT='VI' OR @HINHTHUCTT='NGAN HANG') BEGIN IF NOT EXISTS(SELECT MACN FROM CUAHANG WHERE MACN=@MACN AND TINHTRANGCH='AVAILABLE') BEGIN ROLLBACK TRAN RAISERROR ('CUA HANG KHONG HOAT DONG',16,1) RETURN END </pre>	<p>T2 xin khóa S trên bảng CUAHANG SQL: cấp khóa S và trả khóa S</p>

		<p>B2: tạo biến lưu số dư ví DECLARE @SODUVI INT=(SELECT SODUVI FROM KHACHHANG with(updlock) WHERE MAKH=@MAKH)</p>	<p>T2 xin khóa S trên bảng KHACHANG SQL: không cấp, T2 đợi T1 commit mới được tiếp tục</p>
<p>B4: cập nhật lại số dư ví UPDATE KHACHHANG SET SODUVI=@SODUVI- @TONGTIEN WHERE MAKH=@MAKH IF @@ERROR <>0 BEGIN ROLLBACK TRAN RETURN END END ELSE BEGIN ROLLBACK TRAN RAISERROR ('HINH THUC THANH TOAN KHONG HOP LE',16,1) RETURN END COMMIT</p>	<p><i>T1 xin khóa X trên bảng KHACHHANG tại dòng có MAKH = @MAKH SQL cấp X(KHACHHANG) và giữ đến khi commit T1 commit và nhả tất cả khóa</i></p>		

		<p>B2: tạo biến lưu số dư ví</p> <pre>DECLARE @SODUVI INT=(SELECT SODUVI FROM KHACHHANG with(updlock) WHERE MAKH=@MAKH)</pre>	<p>T2 xin khóa S(KHACHHANG)</p> <p>SQL: cấp khóa S và giữ đến khi commit</p>
		<p>B3: tạo tự động các thông tin cần cho đơn hàng</p> <pre>DECLARE @MADH INT EXEC sp_IDMAX 'DON_DH',@MADH OUT DECLARE @MAKHUVUC VARCHAR(10)=NULL DECLARE @NGAYTAO DATETIME=GETDATE() EXEC sp_ThemDonDH @MADH,'AVAILABLE',@NGAYTAO,@HINHTH UCTT,0,@PHIVANCHUYEN,@MADIACHI, @MAKH, @MACN,NULL,@MAKHUVUC IF @@ERROR<>0 BEGIN ROLLBACK TRAN RAISERROR ('KHONG TAO DUOC DON HANG',16,1) RETURN END DECLARE @STT INT =(SELECT MAX(STT) FROM @list_MONAN)</pre>	<p>T2 xin khóa X trên bảng PHAMVIBANG với dòng có TENPHAMVI='DON_DH'</p> <p>SQL cấp khóa X(TENPHAMVI) và giữ đến khi T2 commit</p> <p>T2 xin khóa X trên bảng DON_DH</p> <p>SQL cấp khóa X(DON_DH) và giữ đến khi T2 commit</p> <p>T2 xin khóa X trên THUCDONDATHANG</p> <p>SQL cấp khóa X(THUCDONDATHANG) và giữ</p>

		<pre> DECLARE @I INT = 1 WHILE (@I <= @STT) BEGIN DECLARE @MAMONAN VARCHAR(10) DECLARE @SOLUONG INT SELECT @MAMONAN=MAMONAN,@SOLUONG=SOLUONG FROM @list_MONAN WHERE STT=@I EXEC sp_ThemThucDonDatHang @MAMONAN,@MADH,@SOLUONG IF @@ERROR <> 0 BEGIN ROLLBACK TRAN RAISERROR ('KHONG TAO DUOC DON HANG',16,1) RETURN END SET @I=@I+1 END </pre>	đến khi T2 commit
		<p>B3: Tính tổng tiền của đơn hàng</p> <pre> IF (@HINHTHUCTT='VI') BEGIN DECLARE @TONGTIEN INT=(SELECT PHIVANCHUYEN+PHISP FROM DON_DH WHERE MADH=@MADH) </pre>	T2 xin khóa S trên bảng DON_DH SQL: cấp khóa S

		<p>B4: cập nhật lại số dư ví</p> <pre> UPDATE KHACHHANG SET SODUVI=@SODUVI- @TONGTIEN WHERE MAKH=@MAKH IF @@ERROR <>0 BEGIN ROLLBACK TRAN RETURN END END ELSE BEGIN ROLLBACK TRAN RAISERROR ('HINH THUC THANH TOAN KHONG HOP LE',16,1) RETURN END COMMIT </pre>	<p><i>T2 đang giữ khóa X nên tiếp tục thực hiện T2 commit và nhả tất cả khóa</i></p>
--	--	--	--


II. Sinh viên thực hiện: 20127099


1. Tình huống 2: Dirty read



Khách hàng đặt đơn hàng nhưng số dư ví không đủ nên hệ thống cho rollback thao tác của khách hàng, cùng lúc đó chi nhánh vào xem đơn cần xác nhận thì thấy đơn hàng của khách hàng này


Sửa: sửa đổi mức cô lập cho giao tác T2 thành read committed


ERR05: Dirty read			
T1 (User = khách hàng): đặt đơn			
T2 (User = chi nhánh): xem đơn			
sp_Tao_DH	Khóa	sp_DonHang_CN_XuLy	Khóa
<u>Input:</u> @HINHTHUCTT, @MADIACHI, @MAKH, @MACN, @PHIVANCHUYEN, @LIST_MONAN		<u>Input:</u> @MACN, @TRANGTHAIDH, @TGBD, @TGKT	
<u>Output:</u>		<u>Output:</u>	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
B1: Kiểm tra tình trạng cửa hàng có đang hoạt động hay không	T1 xin khóa S trên bảng (CUAHANG) SQL cấp khóa S(CUAHANG) và trả khóa S		

 <pre> IF (@HINHTHUCTT='CHUYEN KHOAN' OR @HINHTHUCTT='VI' OR @HINHTHUCTT='NGAN HANG') BEGIN SELECT *FROM @list_MONAN IF NOT EXISTS(SELECT MACN FROM CUAHANG WHERE MACN=@MACN AND TINHTRANGCH ='AVAILABLE') BEGIN ROLLBACK TRAN RAISERROR ('CUA HANG KHONG HOAT DONG',16,1) RETURN END </pre>			
<p>B2: Tạo biến lưu lại số dư ví và tạo tự động các thông tin cần cho đơn hàng</p>	<p>T1 xin khóa S(KHACHHANG)</p>		

 <pre> DECLARE @SODUVI INT=(SELECT SODUVI FROM KHACHHANG WHERE MAKH=@MAKH) DECLARE @MADH INT EXEC sp_IDMAX 'DON_DH',@MADH OUT IF @@ERROR<>0 BEGIN ROLLBACK TRAN RAISERROR ('KHONG TAO DUOC DON HANG',16,1) RETURN END DECLARE @STT INT =(SELECT MAX(STT) FROM @list_MONAN) DECLARE @I INT = 1 WHILE (@I<= @STT) BEGIN DECLARE @MAMONAN VARCHAR(10) DECLARE @SOLUONG INT </pre>	<p>SQL: cấp khóa S(KHACHHANG) và trả khóa S</p> <p>T1 xin khóa X trên bảng PHAMVIBANG với dòng có có TENPHAMVI='DON_DH'</p> <p>SQL cấp khóa X(TENPHAMVI) và giữ đến khi T1 commit</p> <p>T1 xin khóa X trên bảng DON_DH</p> <p>SQL cấp khóa X(DON_DH) và giữ đến khi T1 commit</p> <p>T1 xin khóa X trên THUCDONDATHANG</p> <p>SQL cấp khóa X(THUCDONDATHANG) và giữ đến khi T1 commit</p>		
---	--	--	--

<pre> SELECT @MAMONAN=MAMONAN,@SOLUONG=SOLU ONG FROM @list_MONAN WHERE STT=@I EXEC sp_ThemThucDonDatHang @MAMONAN,@MADH,@SOLUONG IF @@ERROR <>0 BEGIN ROLLBACK TRAN RAISERROR ('KHONG TAO DUOC DON HANG',16,1) RETURN END SET @I=@I+1 END </pre>			
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		<pre> SELECT * FROM DON_DH DH JOIN KHACHHANG KH ON DH.MAKH=KH.MAKH JOIN CHINHANH CN ON DH.MACN=CN.MACN JOIN CUAHANG CH ON CH.MACN=CN.MACN WHERE DH.MACN=@MACN AND DH.TRANGTHAIDH=@TRANGTHAIDH </pre>	T2 chờ T1 commit mới có thể truy cập xem thông tin

		SELECT COUNT (DON.MADH) FROM (SELECT DH.MADH FROM DON_DH DH JOIN KHACHHANG KH ON DH.MAKH=KH.MAKH JOIN CHINHANH CN ON DH.MACN=CN.MACN JOIN CUAHANG CH ON CH.MACN=CN.MACN WHERE DH.MACN=@MACN AND DH.TRANGTHAIDH=@TRANGTHAIDH) DON	
B3: Tính tổng tiền của đơn hàng và cập nhật lại số dư ví IF (@HINHTHUCTT='VI') BEGIN DECLARE @TONGTIEN INT=(SELECT PHIVANCHUYEN+PHISP FROM DON_DH WHERE MADH=@MADH) UPDATE KHACHHANG SET SODUVI=@SODUVI- @TONGTIEN WHERE MAKH=@MAKH IF @@ERROR <>0 BEGIN ROLLBACK TRAN RAISERROR ('SO DU KHONG DU DE THANH TOAN',16,1) RETURN END	 T1 xin khóa S(DON_DH) SQL: cấp khóa S(DON_DH) và trả khóa S T1 xin khóa X trên bảng (KHACHHANG) tại dòng MAKH = @MAKH SQL: cấp khóa X(KHACHHANG) và giữ đến khi commit T1 commit và nhả tất cả khóa		

END END ELSE BEGIN ROLLBACK TRAN RAISERROR ('HINH THUC THANH TOAN KHONG HOP LE',16,1) RETURN END COMMIT			
		<pre> SELECT * FROM DON_DH DH JOIN KHACHHANG KH ON DH.MAKH=KH.MAKH JOIN CHINHANH CN ON DH.MACN=CN.MACN JOIN CUAHANG CH ON CH.MACN=CN.MACN WHERE DH.MACN=@MACN AND DH.TRANGTHAIDH=@TRANGTHAIDH SELECT COUNT (DON.MADH) FROM (SELECT DH.MADH FROM DON_DH DH JOIN KHACHHANG KH ON DH.MAKH=KH.MAKH JOIN CHINHANH CN ON DH.MACN=CN.MACN JOIN CUAHANG CH ON CH.MACN=CN.MACN WHERE DH.MACN=@MACN AND DH.TRANGTHAIDH=@TRANGTHAIDH) DON </pre>	T2 xin khóa S SQL: cấp khóa S và trả khóa



		COMMIT TRAN	
--	--	-------------	--


2. Tình huống 3: Dirty read



Chi nhánh thêm món ăn vào cửa hàng nhưng trong lúc điền thông tin món ăn lại điền nhầm mô tả vào tên món ăn khiến hệ thống báo lỗi và cho rollback thao tác, cùng lúc đó khách hàng xem danh sách món ăn của chi nhánh thì xem được món ăn vừa thêm



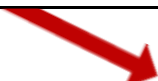
Sửa: Đổi mức cô lập cho giao tác T2 thành read committed

ERR06: Dirty read			
T1 (User = Chi nhánh): Thêm món ăn			
T2 (User = Khách hàng): Xem món ăn			
sp_ThemThucDon	Khóa	sp_MonAn_ChiNhanh	Khóa
<u>Input:</u> @TENMONAN, @MOTA, @GIA, @MACN, @MALAT		<u>Input:</u> @MaCN	
<u>Output:</u>		<u>Output:</u>	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
B1: Khai báo biến chứa lỗi			

DECLARE @ERROR INT = 0			
<p>B2: Kiểm tra chi nhánh có tồn tại hay không</p>  <pre>IF NOT EXISTS(SELECT MACN FROM CHINHANH WHERE MACN=@MACN) BEGIN ROLLBACK TRAN RAISERROR ('KHONG TON TAI CHI NHANH',16,1) RETURN END</pre>	<p>T1 xin khóa S trên bảng CHINHANH</p> <p>SQL cấp khóa S(CHINHANH) và trả khóa khi đọc xong</p>		
<p>B2: Kiểm tra loại ẩm thực có tồn tại hay không</p>  <pre>IF NOT EXISTS(SELECT MALAT FROM LOAIAMTHUC WHERE MALAT=@MALAT)</pre>	<p>T1 xin khóa S trên bảng LOAIAMTHUC</p> <p>SQL cấp khóa S(LOAIAMTHUC)</p>		

BEGIN ROLLBACK TRAN RAISERROR ('KHONG TON TAI LOAI AM THUC NAY',16,1) RETURN END			
B3: kiểm tra món ăn đã tồn tại hay không  IF EXISTS (SELECT MAMONAN FROM THUCDON WHERE MACN=@MACN AND TENMONAN=@TENMONAN) BEGIN ROLLBACK TRAN RAISERROR ('DA TON TAI MON AN NAY',16,1) RETURN END	T1 xin khóa S trên bảng THUCDON SQL cấp khóa S và trả khóa khi đọc xong		

<p>B4: cài đặt ID cho món ăn vào thêm vào thực đơn</p>  <pre> DECLARE @MAMONAN CHAR(10) EXEC sp_IDMAX 'THUCDON',@MAMONAN OUT INSERT INTO THUCDON VALUES (@MAMONAN,@TENMONAN, @MOTA,@GIA,'AVAILABLE',@HINHANHTD, @DIACHIHINHANHTD,@MACN,@MALAT) IF @@ERROR<>0 SET @ERROR=@ERROR+1 </pre>	<p>T1 xin khóa X trên PHAMVIBANG</p> <p>SQL cấp khóa X(PHAMVIBANG) và giữ đến khi T1 commit</p> <p>T1 xin khóa X trên bảng THUCDON</p> <p>SQL cấp khóa X(THUCDON) và giữ đến khi commit</p>		
<p>WAITFOR DELAY '00:00:10'</p>			
		<p>BEGIN TRAN</p>	

		IF EXISTS(SELECT MACN FROM CHINHANH WHERE MACN=@MACN) SELECT * FROM THUCDON WHERE MACN=@MACN ELSE RAISERROR ('KHONG TON TAI CHI NHANH',16,1) COMMIT TRAN	T2 xin khóa S trên bảng THUCDON SQL: không cấp T2 đợi T1 commit rồi mới được truy cập xem thông tin
B4: kiểm tra lỗi, nếu có thì hệ thống rollback IF (@ERROR<>0) BEGIN PRINT N'KHÔNG THỂ THÊM MÓN ĂN' ROLLBACK TRAN RETURN 0 END COMMIT TRAN	 T1 trả khóa X(THUCDON)		
		IF EXISTS(SELECT MACN FROM CHINHANH WHERE MACN=@MACN)	T2 xin khóa S trên bảng CHINHANH


		SELECT * FROM THUCDON WHERE MACN=@MACN ELSE RAISERROR ('KHONG TON TAI CHI NHANH',16,1) COMMIT TRAN	SQL: cấp khóa S và trả khóa S T2 xin khóa S trên bảng THUCDON SQL: cấp khóa S và trả khóa S
--	--	--	---

III. Sinh viên thực hiện: 20127317


1. Tình huống 1: Unrepeatable read

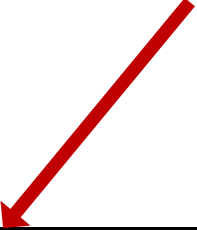

Chi nhánh nhận được đơn hàng nhưng quyết định hủy đơn, cùng lúc đó Khách Hàng cũng quyết định hủy đơn làm cho nhưng vì không thể đọc lại khiến cho đơn hàng bị hủy 2 lần làm cho tài khoản khách hàng được hoàn 2 lần tiền
Sửa lỗi đặt XLOCK trên đơn hàng lúc kiểm tra tồn tại đơn hàng để giao tác còn lại phải chờ tới khi giao tác cập nhật xong mới được đọc

ERR07: Unrepeatable read T1 (User = Chi nhánh) T2 (User = Khách hàng)			
sp_CapNhat_DH_CN	Khóa	sp_CapNhat_DH_KH	Khóa
<u>Input:</u> @TRANGTHAIDH CHAR(20) (CANCELED),		<u>Input:</u> @TRANGTHAIDH CHAR(20) (CANCELED), @MAKH VARCHAR(10),	




@MACN VARCHAR(10), @MADH VARCHAR(10)		@MADH VARCHAR(10)	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
B1: KIỂM TỒN TẠI ĐƠN HÀNG IF NOT EXISTS(SELECT MADH FROM DON_DH WHERE MADH=@MADH) BEGIN ROLLBACK TRAN RAISERROR ('KHONG TON TAI DON HANG',16,1) RETURN END	 T1 Xin khóa S(DON_DH) trên dòng MADH=@MADH và trả sau khi đọc SQL: cấp khóa S		
KIỂM TRA TỒN TẠI CHI NHÁNH	T1 Xin khóa S(CHINHANH) trên dòng MACN=@MACN và trả sau khi đọc SQL: cấp khóa S		

<p>IF NOT EXISTS(SELECT MACN FROM CHINHANH WHERE MACN=@MACN)</p> <p>BEGIN</p> <p> ROLLBACK TRAN</p> <p> RAISERROR ('KHONG TON TAI CHI NHANH',16,1)</p> <p> RETURN</p> <p>END</p>			
<p>KIỂM TRA GIÁ TRỊ TRẠNG THÁI</p> <p>IF (@TRANGTHAIDH='PROCESSING' OR @TRANGTHAIDH='CANCELED')</p>			
<p>BEGIN</p>			
<p>KIỂM TRA TRẠNG THÁI ĐƠN HÀNG CẦN ĐƯỢC CẬP NHẬT</p> <p>IF NOT EXISTS(SELECT MADH FROM DON_DH WITH(XLOCK,ROWLOCK) WHERE MADH=@MADH AND MACN=@MACN AND TRANGTHAIDH='AVAILABLE')</p>	<p>T1 Xin khóa X(DON_DH) trên dòng MADH=@MADH và TRANGTHAIDH='AVAILABLE' giữ tới khi commit</p>		

BEGIN ROLLBACK TRAN RAISERROR ('KHONG NHAN DON DUOC',16,1) RETURN END	SQL: cấp khóa X		
WAITFOR DELAY '0:00:10'			
		BEGIN TRAN	
		KIỂM TRA SỰ TỒN TẠI CỦA ĐƠN HÀNG IF NOT EXISTS(SELECT MADH FROM DON_DH WITH (XLOCK,ROWLOCK) WHERE MADH=@MADH) BEGIN ROLLBACK TRAN	T2 Xin khóa X(DON_DH) trên dòng MADH=@MA DH SQL: không cấp khóa X

		RAISERROR ('KHONG TON TAI DON HANG',16,1) RETURN END	
			
CẬP NHẬT TRẠNG THÁI MỚI CHO ĐƠN HÀNG UPDATE DON_DH SET TRANGTHAIDH=@TRANGTHAIDH WHERE MADH=@MADH IF @@ERROR <>0 BEGIN ROLLBACK TRAN RAISERROR ('KHONG NHAN DON DUOC',16,1) RETURN	 T1 đã có khóa X(DON_DH) trên dòng MADH=@MADH		

END			
<p>NẾU LÀ HỦY ĐƠN THÌ HOÀN TIỀN LẠI CHO KHÁCH HÀNG</p> <p>IF @TRANGTHAIDH='CANCELED'</p> <p>BEGIN</p> <p>LẤY MÃ KHÁCH HÀNG</p> <p>DECLARE @MAKH VARCHAR(10) = (SELECT MAKH FROM DON_DH WHERE MADH=@MADH)</p> <p>LẤY SỐ TIỀN CẦN PHẢI HOÀN</p> <p>DECLARE @TONGTIEN INT = (SELECT PHIVANCHUYEN+ PHISP FROM DON_DH WHERE MADH=@MADH)</p>	<p>T1 xin khóa S(KHACHHANG) trên dòng có MAKH=@MAKH và trả sau khi đọc xong</p> <p>SQL cấp khóa s</p>		
<p>HOÀN TIỀN CHO KHÁCH HÀNG</p>	<p>T1 xin khóa X(KHACHHANG) trên dòng có MAKH=@MAKH</p> <p>SQL cấp khóa X</p>		

<pre> UPDATE KHACHHANG SET SODUVI=SODUVI+@TONGTIEN WHERE MAKH=@MAKH IF @@ERROR <>0 BEGIN ROLLBACK TRAN RAISERROR ('LOI HE THONG THANH TOAN',16,1) RETURN END END </pre> 			
<p>COMMIT TRAN</p> <pre> RETURN </pre>	<p>T1 COMMIT và kết thúc</p> 		
		<p>KIỂM TRA SỰ TỒN TẠI CỦA ĐƠN HÀNG</p> <pre> IF NOT EXISTS(SELECT MADH FROM DON_DH WHERE MADH=@MADH) BEGIN ROLLBACK TRAN </pre> 	<p>T2 Xin khóa S(DON_DH) trên dòng MADH=@MADH và trả sau khi đọc</p> <p>SQL: không cấp khóa S</p>

		RAISERROR ('KHONG TON TAI DON HANG',16,1) RETURN END	
		KIỂM TRA SỰ TỒN TẠI CỦA KHÁCH HÀNG IF NOT EXISTS(SELECT MAKH FROM KHACHHANG WHERE MAKH=@MAKH) BEGIN ROLLBACK TRAN RAISERROR ('KHONG TON TAI KHACH HANG',16,1) RETURN END	T2 Xin khóa S(KHACHHAN G) trên dòng MAKH=@MA KH và trả sau khi đọc SQL: cấp khóa S
		KIỂM TRA TRẠNG THÁI KHÁCH HÀNG MUỐN CẬP NHẬT IF (@TRANGTHAIDH='CANCELED') BEGIN KIỂM TRA TRẠNG THÁI CỦA ĐƠN HÀNG CẦN ĐƯỢC CẬP NHẬT CÓ PHÙ HỢP KHÔNG	T2 Xin khóa S(DON_DH) trên dòng MADH=@MA DH AND MACN=@MAC N AND TRANGTHAID H='AVAILABL

		<pre> IF NOT EXISTS (SELECT MADH FROM DON_DH MADH=@MADH AND MAKH=@MAKH AND TRANGTHAIDH='AVAILABLE') BEGIN ROLLBACK TRAN RAISERROR ('KHONG HUY DON DUOC',16,1) RETURN END </pre>	<p>E' và trả sau khi đọc</p> <p>SQL: cấp khóa S</p>
		<p>NẾU TRẠNG THÁI ĐƯỢC PHÉP HỦY THÌ CẬP NHẬT HỦY ĐƠN CHO ĐƠN HÀNG</p> <pre> UPDATE DON_DH SET TRANGTHAIDH=@TRANGTHAIDH WHERE MADH=@MADH IF @@ERROR <>0 BEGIN ROLLBACK TRAN </pre>	<p>T2 Xin khóa X(DON_DH) trên dòng MADH=@MA DH và giữ tới khi commit</p> <p>SQL: cấp khóa X</p>


		RAISERROR ('KHONG HUY DON DUOC',16,1) RETURN END	
		LẤY SỐ TIỀN KHÁCH HÀNG ĐÃ TRẢ CHO ĐƠN HÀNG DECLARE @TONGTIEN INT = (SELECT PHIVANCHUYEN+ PHISP FROM DON_DH WHERE MADH=@MADH)	T2 đọc trên DON_DH
		HOÀN TIỀN ĐƠN HÀNG CHO KHÁCH HÀNG UPDATE KHACHHANG SET SODUVI=SODUVI+@TONGTIEN WHERE MAKH=@MAKH IF @@ERROR <>0 BEGIN ROLLBACK TRAN RAISERROR ('LOI HE THONG THANH TOAN',16,1)	T2 xin khóa X(KHACHHANG) trên dòng có MAKH=@MAKH SQL cấp khóa X T2 COMMIT và kết thúc

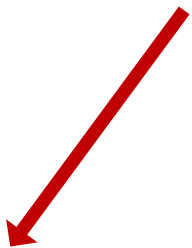
		RETURN END COMMIT TRAN RETURN	
--	--	---	--

2. Tình huống 2: Unrepeatable read


Chi nhánh cập nhật lại tình trạng cửa hàng đóng cửa thành công nhưng vẫn có đơn hàng được đặt tới chi nhánh


ERR05: Unrepeatable read T1 (User = khách hàng): đặt đơn T2 (User = chi nhánh): cập nhật tình trạng cửa hàng			
sp_Tao_DH	Khóa	sp_DonHang_CN_XuLy	Khóa
<u>Input:</u> @HINHTHUCTT, @MADIACHI, @MAKH, @MACN, @PHIVANCHUYEN, @LIST_MONAN <u>Output:</u>		<u>Input:</u> @MACN, @TRANGTHAIDH, @TGBD, @TGKT <u>Output:</u>	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			


<p>B1: Kiểm tra tình trạng cửa hàng có đang hoạt động hay không</p> <pre> IF (@HINHTHUCTT='CHUYEN KHOAN' OR @HINHTHUCTT='VI' OR @HINHTHUCTT='NGAN HANG') IF NOT EXISTS(SELECT MACN FROM CUAHANG WITH(HOLDLOCK,ROWLOCK) WHERE MACN=@MACN AND TINHTRANGCH ='AVAILABLE') BEGIN ROLLBACK TRAN RAISERROR ('CUA HANG KHONG HOAT DONG',16,1) RETURN END </pre>	<p>T1 xin khóa S trên bảng CUAHANG trên dòng có MACN=@MACN và giữ tới khi commit</p> <p>SQL cấp khóa S(CUAHANG)</p>		
<p>WAITFOR DELAY '0:00:10'</p>			
		<p>BEGIN TRAN</p>	

		<p>CẬP NHẬT TRẠNG THÁI</p> <pre> UPDATE CUAHANG SET TENCH=@TENCH,@TG_HD_MO=@TG_HD_MO ,@TG_HD_DONG= @TG_HD_DONG,TINHTRANGCH= @TINHTRANGCH WHERE MACN=@MACN IF @@ERROR<>0 BEGIN PRINT N'LỖI HỆ THỐNG' ROLLBACK TRAN RETURN END </pre>	<p>T2 xin khóa X trên bảng CUAHANG</p> <p>SQL Không cấp khóa X</p>
<p>B2: Tạo biến lưu lại số dư ví và tạo tự động các thông tin cần cho đơn hàng</p> <pre> DECLARE @SODUVI INT=(SELECT SODUVI FROM KHACHHANG WHERE MAKH=@MAKH) </pre>	<p>T1 Xin khóa S(KHACHHANG) trên dòng có MAKH=@MAKH và trả sau khi đọc</p> <p>SQL cấp khóa S</p>		

<p>LẤY MÃ ĐƠN MỚI</p> <pre> DECLARE @MADH INT EXEC sp_IDMAX 'DON_DH',@MADH OUT DECLARE @MAKHUVUC VARCHAR(10)=NULL DECLARE @NGAYTAO DATETIME=GETDATE() TẠO ĐƠN HÀNG EXEC sp_ThemDonDH @MADH,'AVAILABLE',@NGAYTAO,@HINHTH UCTT,0,@PHIVANCHUYEN,@MADIACHI, @MAKH, @MACN,NULL,@MAKHUVUC IF @@ERROR<>0 BEGIN ROLLBACK TRAN RAISERROR ('KHONG TAO DUOC DON HANG',16,1) RETURN END LẤY SỐ MÓN CẦN PHẢI THÊM DECLARE @STT INT =(SELECT MAX(STT) FROM @list_MONAN) </pre>	<p>T1 xin khóa X trên PHAMVIBANG trên dòng có TENPHAMVI='DON_DH ,</p> <p>SQL cấp khóa X(PHAMVIBANG)</p> <p>T1 xin khóa X trên bảng DON_DH</p> <p>SQL cấp khóa X(DON_DH)</p> <p>T1 xin khóa X trên THUCDONDATHANG</p> <p>SQL cấp khóa X(THUCDONDATHANG)</p>		
--	---	--	--

<pre> DECLARE @I INT = 1 WHILE (@I<= @STT) BEGIN DECLARE @MAMONAN VARCHAR(10) DECLARE @SOLUONG INT LẤY THÔNG TIN MÓN ĐỂ THÊM SELECT @MAMONAN=MAMONAN,@SOLUONG=SOLU ONG FROM @list_MONAN WHERE STT=@I THÊM MÓN VÀO ĐƠN HÀNG EXEC sp_ThemThucDonDatHang @MAMONAN,@MADH,@SOLUONG IF @@ERROR <>0 BEGIN ROLLBACK TRAN RAISERROR ('KHONG TAO DUOC DON HANG',16,1) RETURN END SET @I=@I+1 END </pre> 			
--	--	--	--

<p>B3: Tính tổng tiền của đơn hàng và cập nhật lại số dư ví</p> <pre> IF (@HINHTHUCTT='VI') BEGIN DECLARE @TONGTIEN INT=(SELECT PHIVANCHUYEN+PHISP FROM DON_DH WHERE MADH=@MADH) UPDATE KHACHHANG SET SODUVI=@SODUVI- @TONGTIEN WHERE MAKH=@MAKH IF @@ERROR <>0 BEGIN ROLLBACK TRAN RAISERROR ('SO DU KHONG DU DE THANH TOAN',16,1) RETURN END END ELSE BEGIN ROLLBACK TRAN </pre>	 <p>T1 đọc trên bảng DON_DH</p> <p>T1 xin khóa X trên bảng KHACHHANG</p> <p>SQL: cấp khóa X(KHACHHANG) và trả khóa khi commit</p>		
--	--	--	--

RAISERROR ('HÌNH THỨC THANH TOAN KHONG HOP LE',16,1) RETURN END COMMIT			
		CẬP NHẬT TRẠNG THÁI UPDATE CUAHANG SET TENCH=@TENCH,@TG_HD_MO=@TG_HD_MO ,@TG_HD_DONG= @TG_HD_DONG,TINHTRANGCH= @TINHTRANGCH WHERE MACN=@MACN IF @@ERROR<>0 BEGIN PRINT N'LỖI HỆ THỐNG' ROLLBACK TRAN RETURN END	T2 xin khóa X trên bảng CUAHANG trên dòng có MACN=@MAC N SQL cấp khóa X

		COMMIT TRAN	

IV. Sinh viên thực hiện: 20127577



1. Tình huống 1: Phantom

Khách hàng lấy danh sách món ăn theo thứ tự tăng dần của giá nhưng lúc đó chi nhánh thêm một món vào làm cho số món hiện ra khác với số lượng món trả về

ERR10: Phantom T1 (User = Khách Hàng): T2 (User = Chi Nhánh):			
sp_MonAn_TheoGia	Khóa	sp_ThemThucDon	Khóa
<u>Input:</u> @SL,@THUTU <u>Output:</u> @OUTPUT		<u>Input:</u> @TENMONAN, @MOTA, @GIA, @HINHANHTD , @DIACHIHINHANHTD , @MACN , @MALAT	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	

BEGIN TRAN			
<p>BƯỚC 1 LẤY CÁC MÓN CÓ GIÁ THẤP NHẤT</p> <p>SELECT TOP (@SL) * FROM THUCDON WITH (HOLDLOCK,TABLOCK)ORDER BY GIA ASC</p>	<p>T1 xin khóa S(THUCDON) trên toàn bảng và ngăn không insert và giữ tới khi commit</p> <p>SQL: cấp khóa S</p>		
WAITFOR DELAY '0:00:10'			
		BEGIN TRAN	
		<p>BƯỚC 1 KIỂM TRA CHI NHÁNH CÓ TỒN TẠI KHÔNG</p> <p>IF NOT EXISTS(SELECT MACN FROM CHINHANH WHERE MACN=@MACN)</p> <p>BEGIN</p> <p>ROLLBACK TRAN</p> <p>RAISERROR ('KHONG TON TAI CHI NHANH',16,1)</p> <p>RETURN</p> <p>END</p>	<p>T2 xin khóa S(CHINHANH) tại dòng có MACN=@MAC N và trả sau khi đọc</p> <p>SQL: cấp khóa S(CHINHANH)</p>

		<p>KIỂM TRA ĐÃ TỒN TẠI LOẠI ẨM THỰC NÀY HAY CHƯA</p> <pre> IF NOT EXISTS(SELECT MALAT FROM LOAIAMTHUC WHERE MALAT=@MALAT) BEGIN ROLLBACK TRAN RAISERROR ('KHONG TON TAI LOAI AM THUC NAY',16,1) RETURN END </pre>	<p>T2 xin khóa S(LOAIAMTHUC) tại dòng có MALAT=@MALAT và trả sau khi đọc</p> <p>SQL: cấp khóa S</p>
		<p>KIỂM TRA XEM CHI NHÁNH ĐÃ CÓ MÓN ĂN CÓ TÊN NÀY CHƯA</p> <pre> IF EXISTS (SELECT MAMONAN FROM THUCDON WHERE MACN=@MACN AND TENMONAN=@TENMONAN) ROLLBACK TRAN RAISERROR ('DA TON TAI MON AN NAY',16,1) RETURN END </pre>	<p>T2 xin khóa S(THUCDON) tại dòng có MACN=@MACN AND TENMONAN=@TENMONAN và trả sau khi đọc</p> <p>SQL: cấp khóa S</p>

		<p>LẤY MÃ MÓN ĂN MỚI</p> <pre> DECLARE @MAMONAN VARCHAR(10) EXEC sp_IDMAX 'THUCDON',@MAMONAN OUT </pre>	<p>T2 xin khóa X(PHAMVIBA NG) tại dòng có TENPHAMVI='THUCDON' và giữ cho tới khi commit</p> <p>SQL cấp khóa X</p> 
		<p>THÊM MÓN ĂN MỚI VÀO BẢNG</p> <pre> INSERT INTO THUCDON VALUES(@MAMONAN,@TENMONAN,@MOTA,@GIA,'AVAILABLE',@HINHANHTD,@DIACHIHINHANHTD, @MACN, @MALAT) IF @@ERROR<>0 BEGIN PRINT 'KHONG THEM MON DUOC' ROLLBACK TRAN RETURN END </pre>	<p>T2 xin khóa X(THUCDON) và giữ tới khi commit</p> <p>SQL không cấp khóa X</p>



SELECT COUNT(TD.MAMONAN) SOLUONG FROM (SELECT TOP (@SL) * FROM THUCDON ORDER BY GIA ASC) TD	T1 đã có khóa trên S(THUCDON)		
COMMIT TRAN	T1 commit		
		THÊM MÓN ĂN MỚI VÀO BẢNG INSERT INTO THUCDON VALUES(@MAMONAN,@TENMONAN,@MOTA, @GIA,'AVAILABLE',@HINHANHTD,@DIACHIHI NHANHTD, @MACN, @MALAT) IF @@ERROR<>0 BEGIN PRINT 'KHONG THEM MON DUOC' ROLLBACK TRAN RETURN END	T2 xin khóa X(THUCDON) và giữ tới khi commit SQL cấp khóa X

		COMMIT TRAN	T2 Commit
--	--	-------------	-----------



2. Tình huống 2: Phantom


T2 sau khi kiểm tra tập dữ liệu ngân hàng của mình chưa có trong cơ sở dữ liệu chuẩn bị tiến hành thêm vào đó thì T1 thêm vào cơ sở dữ liệu đó khiến xuất hiện 2 tài khoản giống nhau

ERR11: PHANTOM T1 (User = Khách Hàng 1): Đăng ký Ngân hàng T2 (User = Khách hàng 2): Đăng ký Ngân hàng			
sp_CapNhatNganHang	Khóa	sp_CapNhatNganHang	Khóa
<u>Input:</u> @MAND , @LOAIND , @TENNGANHANG, @TENCHINHANH, @STK		<u>Input:</u> @MAND , @LOAIND , @TENNGANHANG, @TENCHINHANH, @STK	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			

<pre> IF @LOAIND<0 OR @LOAIND>5 BEGIN RAISERROR (N'KHÔNG TỒN TẠI NGƯỜI DỪNG',16,1); ROLLBACK RETURN END </pre>			
<pre> IF @LOAIND=5 IF NOT EXISTS(SELECT MAKH FROM KHACHHANG WHERE MAKH=@MAND) BEGIN RAISERROR (N'KHÔNG TỒN TẠI NGƯỜI DỪNG',16,1); ROLLBACK RETURN END </pre>		<p>T1 xin khóa S(KHACHHANG) trên dòng có MAKH=@MAND và trả sau khi đọc</p> <p>SQL: Cấp khóa S</p>	
<pre> DECLARE @MATKNH INT= (SELECT CONVERT(INT, MASTKNGANHANG) FROM </pre>		<p>T1 xin khóa X(NGANHANG) trên toàn bảng và ngăn</p>	

NGANHANG WITH(TABLOCKX) NH WHERE STK=@STK AND TENNGANHANG=@TENNGANHANG AND TENCHINHANH=@TENCHINHANH)	không cho insert và giữ tới khi commit SQL: Cấp khóa X		
WAITFOR DELAY '0:00:10'			
		BEGIN TRAN	
		IF @LOAIND<0 OR @LOAIND>5 BEGIN RAISERROR (N'KHÔNG TỒN TẠI NGƯỜI DÙNG',16,1); ROLLBACK RETURN END	

		<pre> IF @LOAIND=4 IF NOT EXISTS(SELECT MATX FROM TAIXE WHERE MATX=@MAND) BEGIN RAISERROR (N'KHÔNG TỒN TẠI NGƯỜI DÙNG',16,1); ROLLBACK RETURN END </pre> 	<p>T2 xin khóa S(TAIXE) trên dòng có MATX=@MAND và trả sau khi đọc</p> <p>SQL: Cấp khóa S</p>
		<pre> DECLARE @MATKNH INT= (SELECT CONVERT(INT, MASTKNGANHANG) FROM NGANHANG NH WITH(TABLOCKX) WHERE STK=@STK AND TENNGANHANG=@TENNGANHANG AND TENCHINHANH=@TENCHINHANH) </pre>	<p>T2 xin khóa X(NGANHANG) và ngăn không cho insert và giữ tới khi commit</p> <p>SQL: Cấp KHÔNG khóa</p>

<pre> IF @MATKNH>0 DECLARE @DONOTHING INT =0 ELSE BEGIN EXEC sp_IDMAX 'NGANHANG',@MATKNH OUT EXEC sp_ThemNganHang @MATKNH,@TENNGANHANG,@TENCHINHANH,@STK IF @@ERROR <>0 BEGIN RAISERROR (N'KHONG THÊM ĐƯỢC TÀI KHOẢN NGÂN HÀNG',16,1) ROLLBACK RETURN END END </pre>	 <p>T1 xin khóa X(PHAMVIBANG) trên dòng có TENPHAMVI='NGAN HANG' và giữ tới khi commit</p> <p>SQL: Cấp khóa X</p> <p>T1 đã có khóa X(NGANHANG)</p>		
---	---	--	--

<pre> IF @LOAIND=5 BEGIN UPDATE KHACHHANG SET MATKNGANHANG=@MATKNH WHERE MAKH=@MAND IF @@ERROR <>0 BEGIN RAISERROR (N'KHONG THEM ĐƯỢC TÀI KHOẢN NGÂN HÀNG',16,1) ROLLBACK TRAN RETURN END COMMIT TRAN RETURN END </pre>	<p>T1 xin khóa X(KHACHHANG) tại dòng có MAKH=@MAND và dữ tới khi commit</p> <p>SQL: Cấp khóa X</p> <p>T1 COMMIT</p>		
		<pre> DECLARE @MATKNH INT= (SELECT CONVERT(INT, MASTKNGANHANG) FROM NGANHANG NH WITH(TABLOCKX) WHERE STK=@STK AND </pre>	<p>T2 xin khóa X(NGANHANG) và ngăn không cho insert và giữ</p>

		TENNGANHANG=@TENNGANHANG AND TENCHINHANH=@TENCHINHANH)	tới khi commit SQL: Cấp khóa X
		WAITFOR DELAY '0:00:10'	
		IF @MATKNH>0 DECLARE @DONOTHING INT =0 ELSE BEGIN EXEC sp_IDMAX 'NGANHANG',@MATKNH OUT EXEC sp_ThemNganHang @MATKNH,@TENNGANHANG,@TENCHINHANH,@STK IF @@ERROR <>0 BEGIN RAISERROR (N'KHONG THÊM ĐƯỢC TÀI KHOẢN NGÂN HÀNG',16,1) END END	T2 xin khóa X(NGANHANG) và giữ tới khi commit SQL: Cấp khóa X T2 xin khóa X(NGANHANG) và giữ tới khi commit SQL: Cấp khóa X

		ROLLBACK RETURN END END	
		IF @LOAIND=4 BEGIN UPDATE TAIXE SET MATKNGANHANG=@MATKNH WHERE MATX=@MAND IF @@ERROR <>0 BEGIN RAISERROR (N'KHONG THÊM ĐƯỢC TÀI KHOẢN NGÂN HÀNG',16,1) ROLLBACK TRAN RETURN END COMMIT TRAN RETURN END	T2 xin khóa X(TAIXE) tại dòng có MATX=@MAN D và dữ tới khi commit SQL: Cấp khóa X T1 COMMIT

