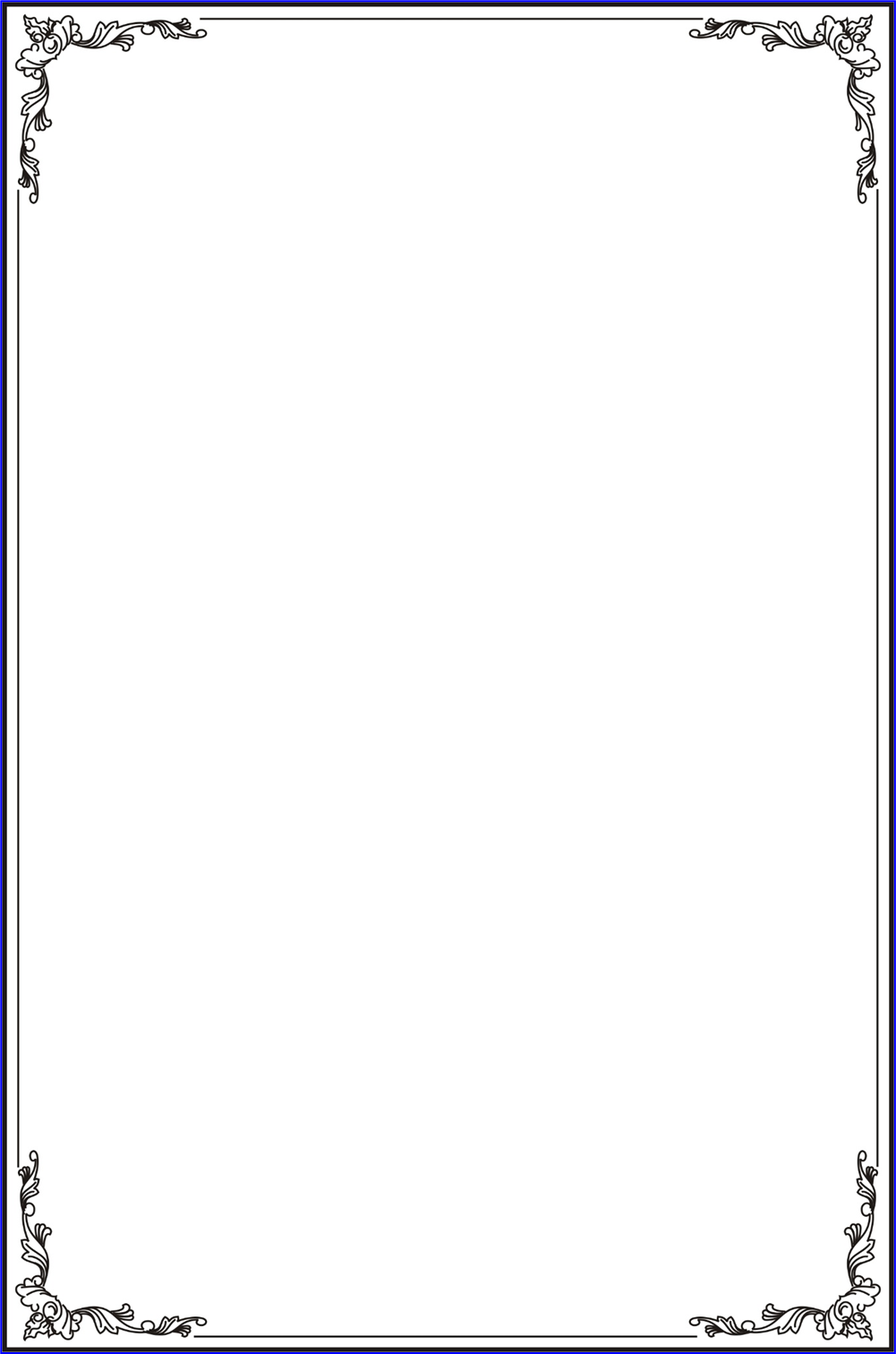
****

**Nhóm sinh viên thực hiện:**

Hồ Thành Danh 20110207

Nguyễn Đưc Hiển 20110643

Phạm Quang Huy 20110653

Nguyễn Đức Thịnh 20110729

**Giảng viên: TS. Nguyễn Thành Sơn**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**'Y&&Y'**

**BÁO CÁO CUỐI KỲ**

**QUẢN LÝ PHÒNG TRỌ**

**MỤC LỤC**

[I.ĐẶC TẢ ĐỀ TÀI 1](#_Toc120017809)

[II.KẾT CẤU CƠ SỞ DỮ LIỆU: 2](#_Toc120017810)

[1.Lược đồ ERD 2](#_Toc120017811)

[2.Class Diagram 3](#_Toc120017812)

[3. Table 3](#_Toc120017813)

[4. Trigger 8](#_Toc120017814)

[5. Function 19](#_Toc120017815)

[6. Procedure 21](#_Toc120017816)

[7. View: 26](#_Toc120017817)

[8. Role: 27](#_Toc120017818)

[III.ỨNG DỤNG SQL SERVER SECURITY: 28](#_Toc120017819)

# I.ĐẶC TẢ ĐỀ TÀI

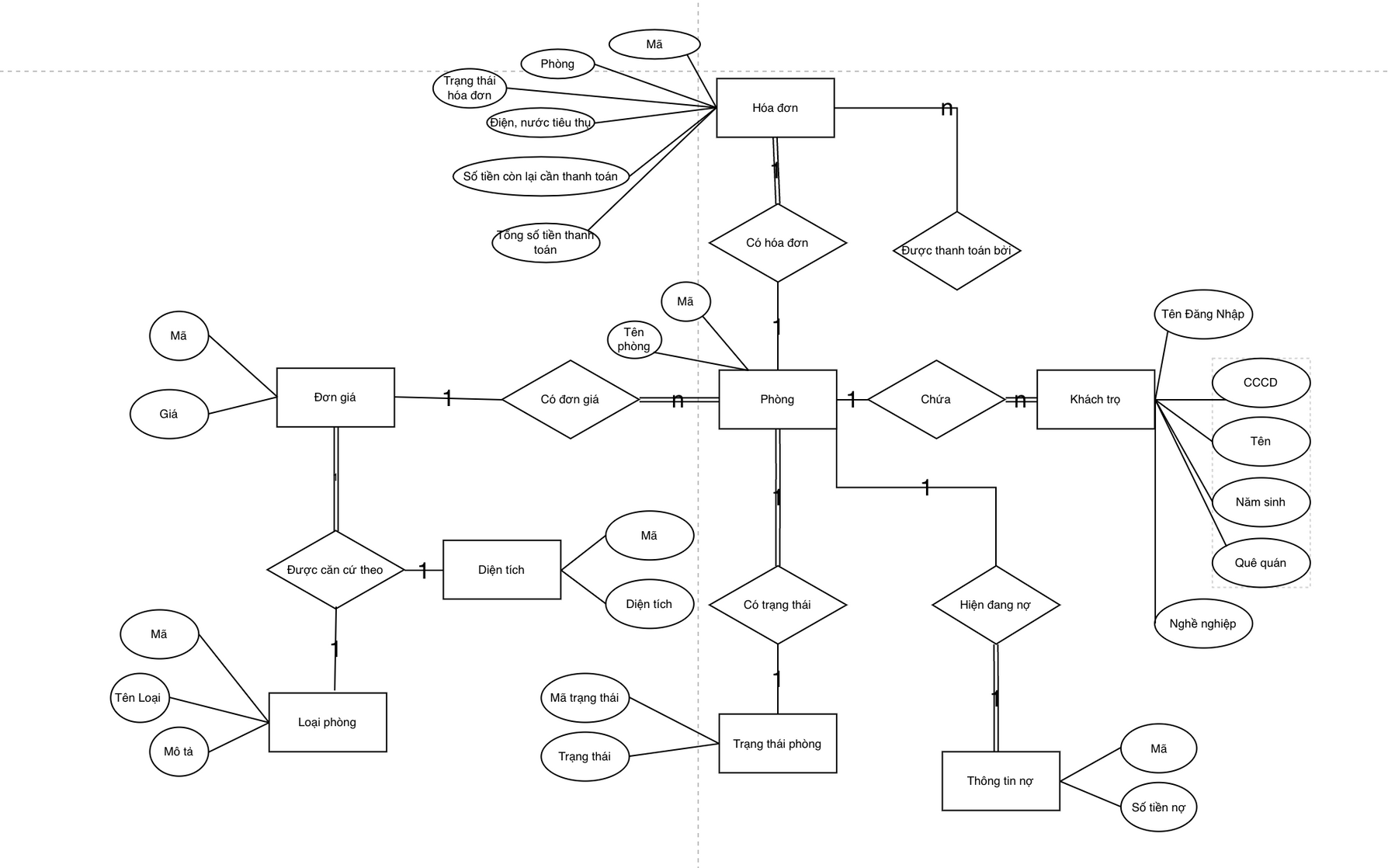
Một chương trình quản lý 1 phòng trọ gồm nhiều phòng. Mỗi phòng tại một thời điểm sẽ có một hoặc nhiều khách thuê trọ. Phòng có các thông tin như mã số, diện tích, giá phòng ứng với diện tích, số lượng người ở tại thời điểm hiện tại. Phòng phải có loại phòng ví dụ như có gác lửng, có sẵn nội thất, v.v…. Ngoài ra, mỗi phòng sẽ còn có tình trạng phòng ứng với thời điểm hiện tại.Khách thuê nhà cần các thông tin để đăng ký tạm trú như tên, số cccd, quê quán. Hóa đơn tiền phòng được tính ra dựa trên chỉ số điện và chỉ số nước cũng như giá tiền phòng cứng và được xuất ra bới admin quản lý. Mỗi tháng admin sẽ tạo ra một hóa đơn tương ứng với thời điểm hiện tại. Mỗi hóa đơn được phân biệt bởi mã phòng, tháng, và năm của hóa đơn đó, các thông tin như tổng số tiền cần thanh toán, kì hạn thanh toán và gồm số tiền còn cần phải thanh toán, trạng thái cho biết hóa đơn đó còn thanh toán được hay không, đã bao giờ được thanh toán chưa,.... Thanh toán chứa thông tin thanh toán thực sự của khách trọ được phân biệt bởi ngày tháng, năm, số tiền và nếu không thanh toán hết số tiền của hóa đơn thì số tiền đó được cộng vào nợ.

Có hai đối tượng đăng nhập là admin và các khách trọ được phân quyền dựa vào thông tin đăng nhập.Admin sẽ có tài khoản riêng trong code ứng dụng còn khách trọ sẽ có tên đăng nhập và mật khẩu được tạo bởi admin khi đăng ký phòng.

Chương trình cho phép admin kiểm trả các phòng vi phạm nội quy nhà trọ. Nội quy đó là nếu phòng đó có nợ hiện vượt hơn hai triệu hoặc có số hóa đơn thanh toán thiếu trong sáu tháng gần nhất lớn hơn 3 thì xem như vi phạm và admin có lựa chọn mới tất cả khách trọ thuộc các phòng đó rời phòng

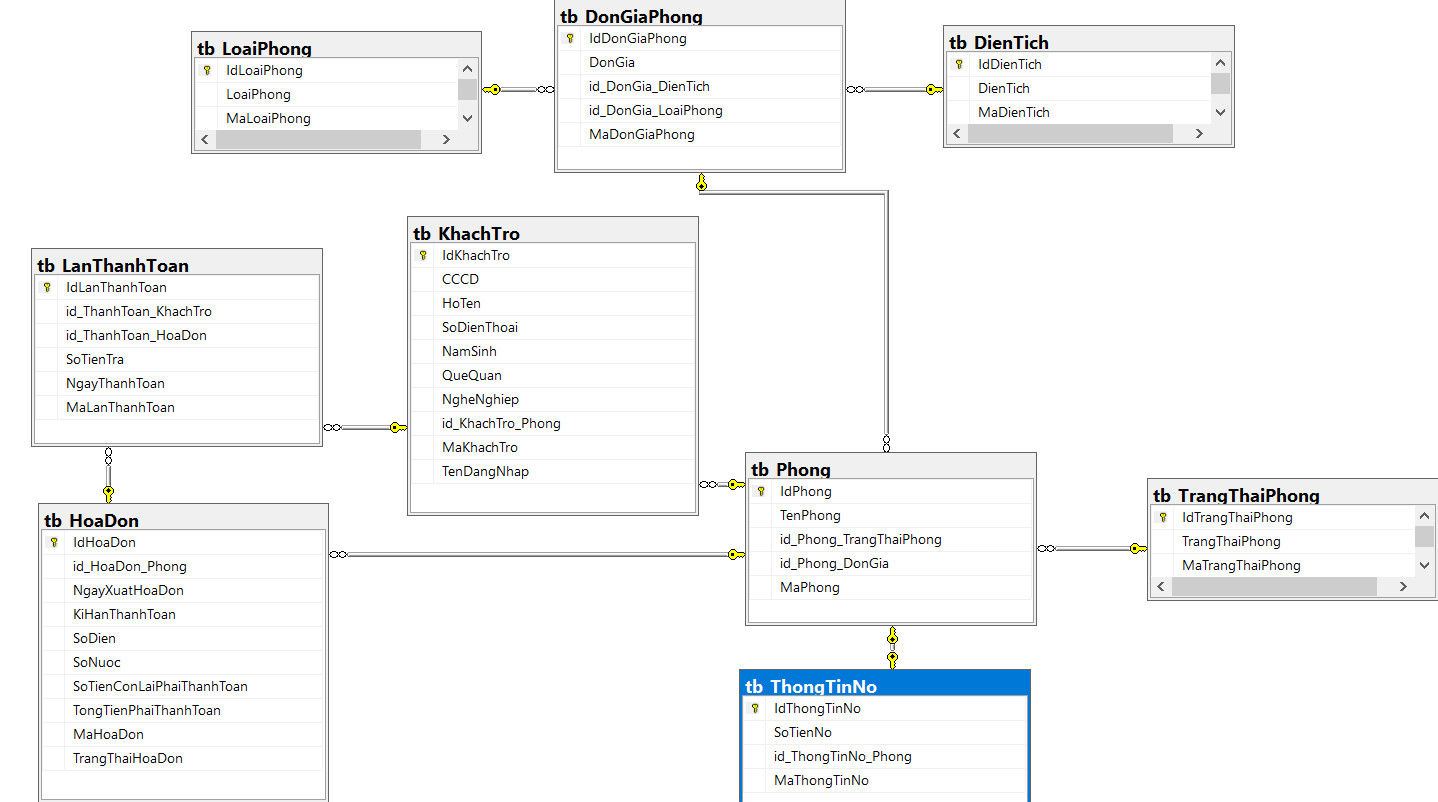
# II.KẾT CẤU CƠ SỞ DỮ LIỆU:

## 1.Lược đồ ERD

****

## 

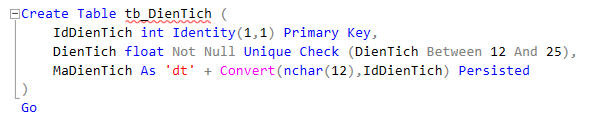
## 2.Class Diagram

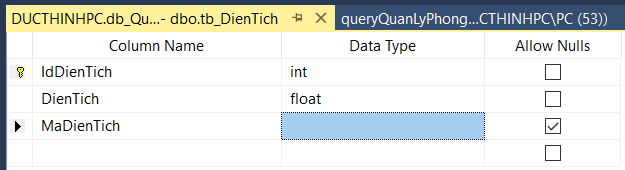
****

## 3. Table

***Tb\_DienTich***

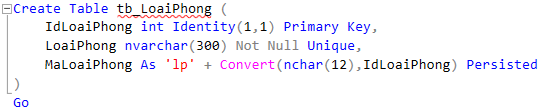
Lưu thông tin về các diện tích phòng trọ

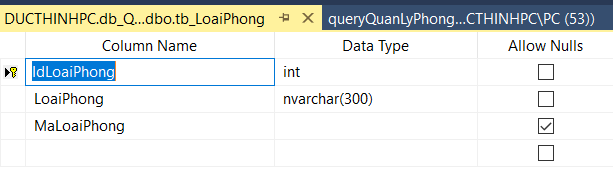




***Tb\_LoaiPhong***

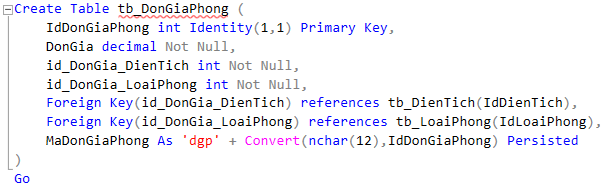
Lưu thông tin về các loại phòng tồn tại

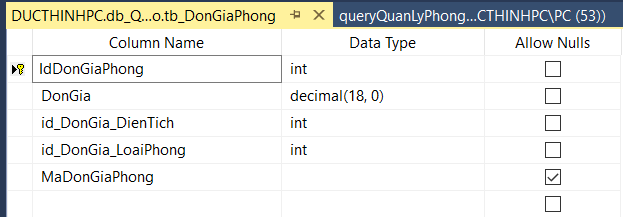




***Tb\_DonGiaPhong***

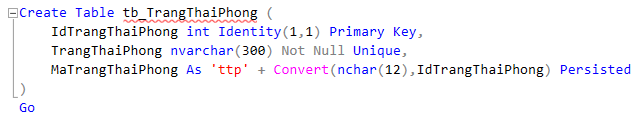
Bảng này sẽ can cứ theo cặp loại phòng và diện tích để căn cứ ra một đơn giá

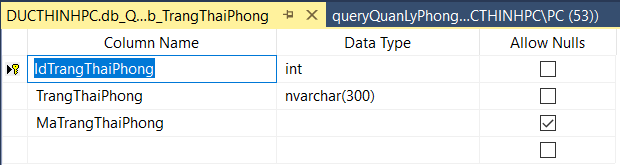




***Tb\_TrangThaiPhong***

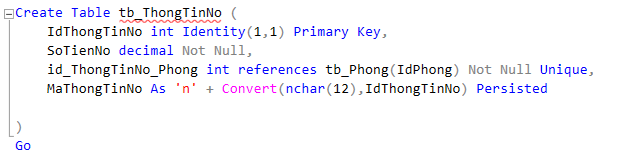
Bảng lưu thông tin về các loại trạng thái phòng. Có ba loại trang thái phòng là phòng trống, phòng có người ở và phòng đầy

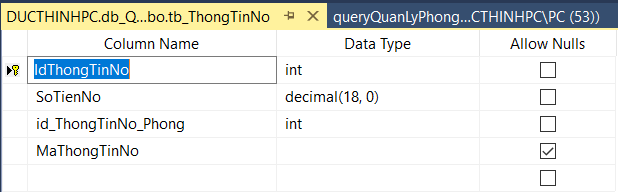




***Tb\_ThongTinNo***

Bảng lưu số tiền nợ của một phòng tại một thời điểm. Mỗi khi xuất một hóa đơn mới, số tiền phải trả của hóa đơn đó tự công vào nợ. Nếu khách trọ thanh toán thì số nợ sẽ giảm xuống. Thanh toán đủ thì không còn nợ còn không đủ thì nợ sẽ tồn và cộng dồn tiếp cho tháng kế tiếp.



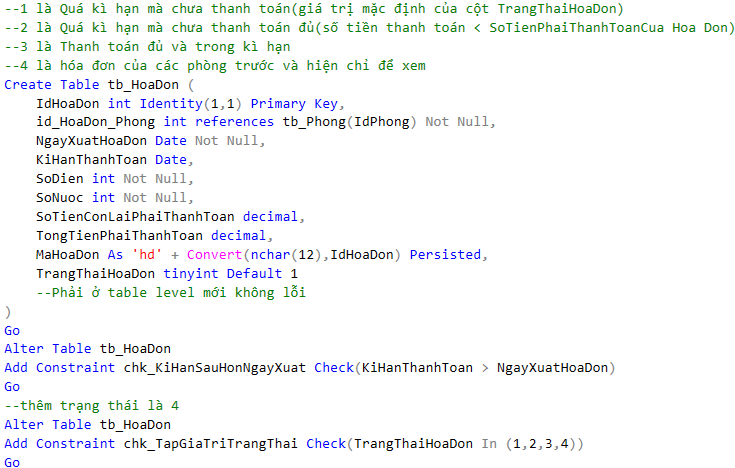


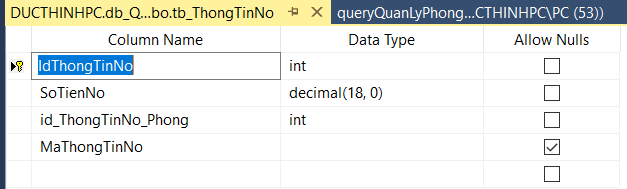
***Tb\_HoaDon***

Bảng lưu thông tin hóa đơn của một phòng,ngày xuất hóa đơn, hạn thanh toán, điện nước và tính ra số tiền phải trả trong tháng. Cột SoTienConLaiPhaiThanhToan là để theo dõi hóa đơn này còn phải trả bao nhiêu mới xong,giá trị cột này liên tục thay đổi khi khách thanh toán còn cột TongTienPhaiThanhToan thì không biến động. TrangThaiHoaDon cho biết hóa đơn đã được thanh toán đầy đủ, chưa đủ hoặc chưa hề được thanh toán trong kì hạn.

Khi các khách vi phạm nội quy và bị mời ra khỏi phòng thì toàn bộ hóa đơn ứng với phòng lúc này sẽ chuyển sang trạng thái là 4, tức là các hóa đơn của khách trước không còn ở của phòng này

Bảng có một số ràng buộc check để đảm bảo tính toàn vẹn dữ liệu

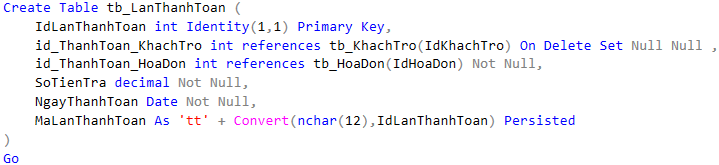


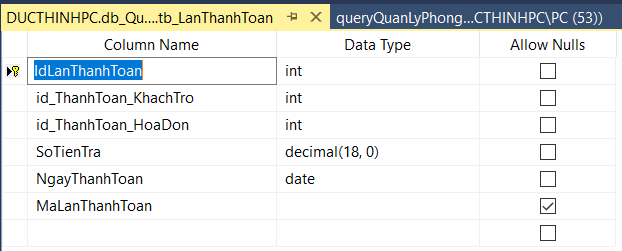


***Tb\_LanThanhToan***

Bảng lưu thông tin thanh toán của một khách trọ cho một hóa của phòng mà khách trọ đó đang ở, vì vậy cần có trigger để đảm bảo khách phòng nào chỉ được thanh toán cho hóa đơn phòng đó

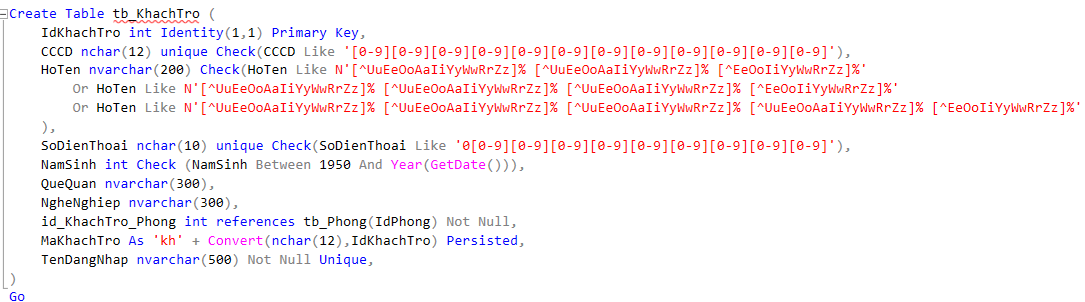
Nếu khách trọ được tham chiếu không còn(do trả phòng hoặc bị đuổi) thì khóa ngoại id\_ThanhToan\_KhachTro được chuyển thành null, nhằm giữ lại thông tin lần thanh toán hiện tại và để mở rộng cho các chức năng khác trong tương lai như quản lý doanh thu,v.v...

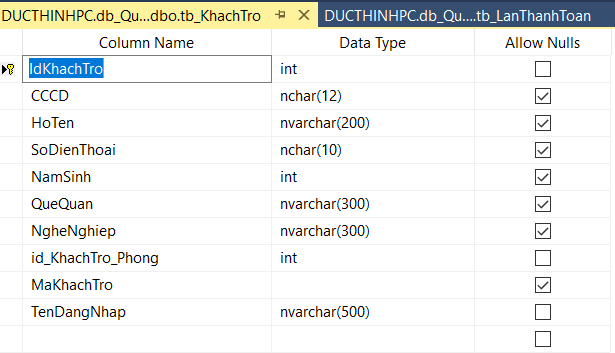




***Tb\_KhachTro***

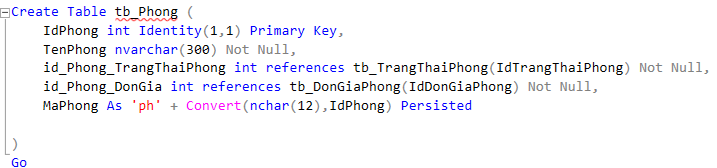
Bảng lưu thông tin về các khách trọ. Số CCCD dân và họ tên cần đảm bảo đúng cấu trúc

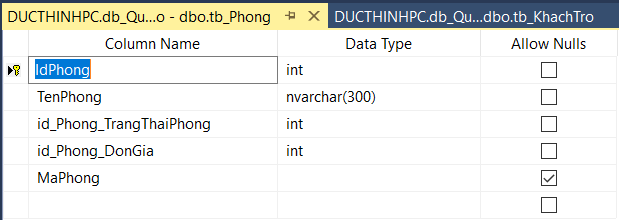




***Tb\_Phong***

Bảng lưu thông tin về phòng, có khóa ngoại tới bảng TrangThaiPhong và DonGia





Một số các ràng buộc phức tạp không thể sử dụng ràng buộc check để thể hiện nên sẽ sử dụng trigger để thay thế. Trigger còn được sử dụng để tính toán giá trị của một số cột

## 4. Trigger

***- Trigger GioiHanSoKhach:*** trigger này nhằm mục đích kiểm tra và thông báo trạng thái phòng trọ có đầy do đạt tối đa số lượng khách trong một phòng hay chưa sau khi thực hiện thêm hoặc cập nhập thông tin khách trọ vào trong phòng đó. Mỗi phòng chỉ cho phép tối đa 4 khách trọ. Nếu thêm hoặc sửa phòng khách trọ không làm số lượng khách trong phòng đó vượt quá 4 thì trigger sẽ cho phép thực hiện giao tác đó, nếu không, trigger sẽ thông báo thực hiện giao tác không thành công và rollback.

Code:

Create Trigger trg\_GioiHanSoKhach On tb\_KhachTro

After Insert, Update

As

Begin

Declare @IdPhong int

Select @IdPhong = ins.id\_KhachTro\_Phong

From inserted ins

If dbo.func\_SoKhachThuocPhong(@IdPhong) < 4

Begin

--Cột identity và computed không thể thêm vào được vì nó được sinh ra tự động

If Not Exists(Select \* From deleted) --Hanh Động chắc chắn là insert

Begin

Print N'Thêm khách không làm đầy phòng rb1'

End

Else

Print N'Sửa phòng khách ở không làm đầy phòng mới/cũ đó rb1'

End

Else

If dbo.func\_SoKhachThuocPhong(@IdPhong) > 4

Begin

Print N'Không thêm khách được vì phòng đã đầy: 4 người rb1'

Rollback

End

End

Go

***- Trigger ToanVenTrangThaiPhong:*** Trigger này nhằm mục đích thiết lập trạng thái phòng sau khi thực hiện Insert, Update hoặc Delete trên bảng Phong. Phòng có 3 trạng thái: trống (tương ứng số 1), chưa đầy (tương ứng số 2), đầy (tương ứng số 3). Sau khi thực hiện:

* Insert: Nếu sau khi thêm khách trọ vào làm số lượng khách trong phòng đạt 4, Trigger sẽ thực hiện thiết lập trạng thái 3 (đầy) cho phòng đó và thông báo. Nếu thực hiện Insert không làm số lượng khách trong phòng đạt 4 thì thiết lâp trạng thái 2 cho phòng. Việc thực hiện Insert khách trọ vào làm số lượng khách trọ trong phòng đạt mức trên 4 không thể xảy ra do đã có Trigger GioiHanSoKhach kiểm tra.
* Delete: Nếu sau khi xóa khách trọ ra khỏi phòng mà làm cho số lượng khách trong phòng đó bằng 0 thì sẽ thông báo và cập nhật trạng thái phòng bằng 1. Nếu sau khi xóa khách ra khỏi phòng mà không làm cho số lượng khách trong phòng đó bằng 0 thì sẽ cập nhật trạng thái phòng bằng 2.
* Update: Nếu sau khi cập nhật khách vào trong phòng mới làm số lượng khách trong phòng đó chưa đạt 4, trigger sẽ thiết lập trạng thái phòng bằng 2 và thông báo; Nếu đạt 4, trigger sẽ thiết lập trạng thái phòng bằng 3 và thông báo. Nếu sau khi cập nhật khách vào trong phòng mới đồng thời làm cho số lượng khách trong phòng cũ bằng 0, trigger sẽ cập nhật trạng thái phòng cũ bằng 1 và thông báo; Nếu chưa bằng 0, trigger sẽ cập nhật trạng thái phòng cũ bằng 2 và thông báo.

Code:

Create Trigger trg\_ToanVenTrangThaiPhong On tb\_KhachTro

After Insert, Update, Delete

As

Begin

Declare @IdPhong int, @HanhDong as char(1)

Set @HanhDong = (Case When Exists(Select \* From inserted) And Exists(Select Top 1 \* From deleted)

Then 'U' -- Set @HanhDong thực hiện là update

When Exists(Select\* From inserted)

Then 'I'

When Exists(Select \* From deleted)

Then 'D'

End)

If @HanhDong = 'I'

Begin

Select @IdPhong = ins.id\_KhachTro\_Phong

From inserted ins

If dbo.func\_SoKhachThuocPhong(@IdPhong) = 4

Begin

Update tb\_Phong

Set id\_Phong\_TrangThaiPhong = 3 --3 là id của trạng thái đã đầy

Where IdPhong = @IdPhong

Print N'Phòng đã đầy sau khi khách mới vào'

End

Else If dbo.func\_SoKhachThuocPhong(@IdPhong) < 4

Begin

Update tb\_Phong

Set id\_Phong\_TrangThaiPhong = 2

Where IdPhong = @IdPhong

End

End

Else

Begin

If @HanhDong = 'D'

Begin

Select @IdPhong = del.id\_KhachTro\_Phong

From deleted del

If dbo.func\_SoKhachThuocPhong(@IdPhong) = 0

Begin

Print N'Phòng đã trống sau khi khách đi'

Update tb\_Phong

Set id\_Phong\_TrangThaiPhong = '1' --1 là id của trạng thái trống

Where IdPhong = @IdPhong

End

Else If dbo.func\_SoKhachThuocPhong(@IdPhong) > 4

Begin

Update tb\_Phong

Set id\_Phong\_TrangThaiPhong = 2

Where IdPhong = @IdPhong

End

End

Else -- Hanh Dong chắc chắn là update

Begin

Declare @IdPhongMoi int, @IdPhongCu int

Select @IdPhongMoi = ins.id\_KhachTro\_Phong, @IdPhongCu = del.id\_KhachTro\_Phong

From inserted ins, deleted del

If @IdPhongCu = @IdPhongMoi

Return

If dbo.func\_SoKhachThuocPhong(@IdPhongMoi) = 4

Begin

Update tb\_Phong

Set id\_Phong\_TrangThaiPhong = 3 --3 là id của trạng thái đã đầy

Where IdPhong = @IdPhong

Print N'Phòng đã đầy sau khi khách mới vào'

End

Else

Begin

Update tb\_Phong

Set id\_Phong\_TrangThaiPhong = 2

Where IdPhong = @IdPhong

Print N'Phòng vẫn có người ở sau khi khách vô'

End

If dbo.func\_SoKhachThuocPhong(@IdPhongCu) = 0

Begin

Update tb\_Phong

Set id\_Phong\_TrangThaiPhong = '1' --1 là id của trạng thái trống

Where IdPhong = @IdPhong

Print N'Phòng đã trống sau khi khách đi'

End

Else

Begin

Update tb\_Phong

Set id\_Phong\_TrangThaiPhong = 2

Where IdPhong = @IdPhong

Print N'Phòng vẫn có người ở sau khi khách đi'

End

End

End

End

Go

***- Trigger ToanVenPhongDcThanhToan:*** Trigger này nhằm mục đích kiểm tra hóa đơn mà khách trọ thanh toán có đúng hóa đơn của phòng mình đang ở hay không, nếu không, trigger sẽ thông báo và Rollback, không cho thực hiện giao tác đó.

Code:

Create Trigger trg\_ToanVenPhongDcThanhToan On tb\_LanThanhToan

After Insert

As

Begin

Declare @IdKhachTro int, @IdHoaDon int,

@IdPhongKhachO int, @IdPhongDcThanhToanHoaDon int

Select @IdKhachTro = ins.id\_ThanhToan\_KhachTro, @IdHoaDon = ins.id\_ThanhToan\_HoaDon

From inserted ins

Select @IdPhongKhachO = kt.id\_KhachTro\_Phong

From tb\_KhachTro kt Where kt.IdKhachTro = @IdKhachTro

Select @IdPhongDcThanhToanHoaDon = hd.id\_HoaDon\_Phong

From tb\_HoaDon hd Where hd.IdHoaDon = @IdHoaDon

If @IdPhongKhachO != @IdPhongDcThanhToanHoaDon

Begin

Print N'Khách thanh toán cho sai phòng'

--Không dùng Delete record đã thêm

Rollback

End

End

Go

***- Trigger NgayThanhToanHopLe:*** Trigger này nhằm mục đích kiểm tra ngày thanh toán của khách trọ có hợp lệ hay không. Nếu ngày khách trọ thanh toán vượt ngày kì hạn thanh toán hoặc sớm hơn ngày xuất hóa đơn thì trigger sẽ thông báo và rollback, không cho thực hiện thanh toán.

Code:

Create Trigger trg\_NgayThanhToanHopLe On tb\_LanThanhToan

After Insert

As

Begin

Declare @NgayThanhToan Date, @IdHoaDonThanhToan int,@KiHanThanhToanHoaDonHienTai Date,@NgayXuatHoaDon Date

Select @NgayThanhToan = ins.NgayThanhToan, @IdHoaDonThanhToan = ins.id\_ThanhToan\_HoaDon

From inserted ins

Select @KiHanThanhToanHoaDonHienTai = hd.KiHanThanhToan,@NgayXuatHoaDon = hd.NgayXuatHoaDon

From tb\_HoaDon hd Where hd.IdHoaDon = @IdHoaDonThanhToan

If @NgayThanhToan > @KiHanThanhToanHoaDonHienTai Or @NgayThanhToan < @NgayXuatHoaDon

Begin

Print N'Đã quá kì hạn nên không thể thanh toán cho hóa đơn này hoặc ngày thanh toán trước ngày xuất hóa đơn !!?'

Rollback

End

End

Go

***- Trigger TinhSoTienPhaiTraChoHoaDon:***

Trigger này thực hiện tính số tiền mà một phòng trọ phải trả cho hóa đơn sau khi cập nhật thông tin vào trong bảng HoaDon. Trigger sẽ cập nhật tiền thanh toán hóa đơn của tháng hiện tại vào trong bảng HoaDon. Tiền thanh toán hóa đơn có công thức: Giá phòng + Số điện sử dụng\*2500 + Số nước sử dụng\*5000.

Đồng thời, trigger cập nhật số tiền nợ vào trong bảng ThongTinNo. Nếu phòng trọ (các) tháng trước đó không có nợ nhưng tháng hiện tại nợ, trigger sẽ thêm thông tin nợ (số tiền nợ, phòng nợ) của phòng đó vào bảng ThongTinNo. Nếu phòng trọ có (các) tháng nợ trước đó, trigger sẽ cập nhật nợ lại theo công thức: nợ mới = nợ cũ + số tiền nợ tháng này và gọi thủ tục GanNoMoiHoacXoaNo.

Code:

Create Trigger trg\_TinhSoTienPhaiTraChoHoaDon On tb\_HoaDon

After Insert

As

Begin

Declare @IdPhongPhaiThanhToan int,@IdDonGiaCuaPhong int,@IdHoaDon int,

@DonGia decimal, @SoDien int,@SoNuoc int, @TienThanhToanHoaDon decimal

Select @IdHoaDon = ins\_Hd.IdHoaDon, @IdPhongPhaiThanhToan = ins\_Hd.id\_HoaDon\_Phong,

@SoDien = ins\_Hd.SoDien, @SoNuoc = ins\_Hd.SoNuoc

From inserted ins\_Hd

Select @IdDonGiaCuaPhong = p.id\_Phong\_DonGia

From tb\_Phong p Where p.IdPhong = @IdPhongPhaiThanhToan

Select @DonGia = dg.DonGia

From tb\_DonGiaPhong dg Where dg.IdDonGiaPhong = @IdDonGiaCuaPhong

Set @TienThanhToanHoaDon = @DonGia + @SoDien \* 2500 + @SoNuoc \* 5000

--Cập nhật bảng hóa đơn tại bản ghi vừa thêm,tong tiền phải thanh toán, tiền còn lại phải thanh toán

Update tb\_HoaDon

Set TongTienPhaiThanhToan = @TienThanhToanHoaDon,SoTienConLaiPhaiThanhToan = @TienThanhToanHoaDon

Where IdHoaDon = @IdHoaDon

--Cập nhật lại cứ mỗi khi xuất hóa đơn sẽ tự cộng vào nợ, mỗi lần thanh toán thì nợ sẽ bớt dần

--hoặc nếu hiện tại không có nợ(do tháng trước thanh toán hết nên xóa) thì tạo mới

If Not Exists(Select Top 1 \* From tb\_ThongTinNo ttn Where ttn.id\_ThongTinNo\_Phong = @IdPhongPhaiThanhToan)

Begin

Insert Into tb\_ThongTinNo(SoTienNo,id\_ThongTinNo\_Phong) Values(@TienThanhToanHoaDon,@IdPhongPhaiThanhToan)

End

Else

--Hiện đang còn nợ từ các tháng trước

Begin

Declare @SoTienDangNo decimal,@TienNoMoi decimal

Select @SoTienDangNo = ttn.SoTienNo From tb\_ThongTinNo ttn Where ttn.id\_ThongTinNo\_Phong = @IdPhongPhaiThanhToan

Set @TienNoMoi = @SoTienDangNo + @TienThanhToanHoaDon

Exec dbo.proc\_GanNoMoiHoacXoaNo @TienNoMoi,@IdPhongPhaiThanhToan

End

End

Go

***- Trigger CapNhatTrangThaiHoaDon:***

Trigger này nhằm mục đích cập nhật lại trạng thái hóa đơn và số tiền còn lại phải thanh toán của phòng đó trong bảng HoaDon. Hóa đơn gồm có 3 trạng thái: quá kì hạn mà chưa thanh toán (tương ứng số 1), quá kì hạn mà chưa thanh toán đủ (tương ứng số 2), thanh toán đủ và trong kì hạn (tương ứng số 3). Trạng thái hóa đơn được cập nhật:

* Nếu số tiền thanh toán nhỏ hơn số tiền cần phải thanh toán, trigger sẽ cập nhật lại trạng thái hóa đơn bằng 2, cập nhật số tiền còn lại phải thanh toán = số tiền phải thanh toán - só tiền đã thanh toán
* Nếu số tiền thanh toán lớn hơn hoặc bằng số tiền cần phải thanh toán, trigger sẽ cập nhật lại trạng thái hóa đơn bằng 3 và set số tiền còn lại phải thanh toán bằng 0.

Code:

Create Trigger trg\_CapNhatTrangThaiHoaDon On tb\_LanThanhToan

After Insert

As

Begin

Declare @SoTienTra decimal,@SoTienPhaiThanhToan decimal,

@IdHoaDon int

Select @SoTienTra = ins\_TT.SoTienTra, @IdHoaDon = ins\_TT.id\_ThanhToan\_HoaDon

From inserted ins\_TT

Select @SoTienPhaiThanhToan = hd.SoTienConLaiPhaiThanhToan

From tb\_HoaDon hd Where hd.IdHoaDon = @IdHoaDon

--Update trạng thái hóa đơn là 2, không cần kiểm tra ngày thanh toán vì đã có trigger khác đảm bảo ngày thanh toán phải lớn hơn ngày xuất hóa đơn

--Cập nhật lại cả số tiền cần phải trả

--Nên cập tại trigger này để đảm bảo khi kiểm tra thì kiểm tra đúng số

If @SoTienTra < @SoTienPhaiThanhToan

Begin

Update tb\_HoaDon

Set TrangThaiHoaDon = 2,SoTienConLaiPhaiThanhToan = @SoTienPhaiThanhToan - @SoTienTra

Where IdHoaDon = @IdHoaDon

End

--Update trạng thái hóa đơn là 3 và chắc chắn số tiền còn lại phải thanh toán là 0 vì trả nhiều hơn rồi

Else If @SoTienTra >= @SoTienPhaiThanhToan

Begin

Update tb\_HoaDon

Set TrangThaiHoaDon = 3,SoTienConLaiPhaiThanhToan = 0

Where IdHoaDon = @IdHoaDon

End

End

Go

***- Trigger ToanVenTienThanhToan:***

Trigger này đảm bảo xử lý toàn vẹn tiền thanh toán trong nhiều trường hợp khi trả hóa đơn thanh toán và trả nợ:

+ Nếu số tiền thanh toán lớn hơn số tiền đang nợ (đã bao gồm số tiền cần phải trả cho hóa đơn tháng hiện tại) thì trigger sẽ thông báo và rollback.

+ Nếu số tiền thanh toán bằng số tiền đang nợ, trigger sẽ thông báo thanh toán đủ và không còn nợ và gọi thủ tục GanNoMoiHoacXoaNo

+ Nếu số tiền thanh toán nhỏ hơn số tiền đang nợ, tại đây, ta có thêm 2 trường hợp.

++ Trường hợp 1: số tiền thanh toán lớn hơn số tiền phải trả cho hóa đơn tháng hiện tại nhưng nhỏ hơn tổng số tiền cần phải trả (bao gồm số tiền đang nợ + số tiền phải trả cho hóa đơn tháng hiện tại), trigger sẽ trừ số tiền đang nợ bằng số tiền thanh toán theo công thức: @TienNoMoi = @SoTienDangNo - @TienThanhToan, sau đó, trigger thông báo trả hết hóa đơn tháng hiện tại nhưng vẫn còn nợ và thực thi thủ tục GanNoMoiHoacXoaNo;

++ Trường hợp 2: số tiền thanh toán nhỏ hơn số tiền cần phải trả cho hóa đơn tháng hiện tại, trigger sẽ trừ số tiền đang nợ bằng số tiền thanh toán theo công thức: @TienNoMoi = @SoTienDangNo - @TienThanhToan, sau đó, trigger thông báo đã thanh toán nhưng chưa trả hết hóa đơn tháng hiện tại và thực thi thủ tục GanNoMoiHoacXoaNo

Code:

Create Trigger trg\_ToanVenTienThanhToan On tb\_LanThanhToan

After Insert

As

Begin

Declare @TienThanhToan decimal(18,0),@SoTienConLaiHoaDon decimal(18,0), @SoTienDangNo decimal(18,0),

@IdHoaDon int, @IdPhongDcThanhToan int, @IdLanThanhToanNay int

Select @TienThanhToan = ins.SoTienTra, @IdHoaDon = ins.id\_ThanhToan\_HoaDon,@IdLanThanhToanNay = ins.IdLanThanhToan

From inserted ins

Select @IdPhongDcThanhToan = hd.id\_HoaDon\_Phong,@SoTienConLaiHoaDon = hd.SoTienConLaiPhaiThanhToan

From tb\_HoaDon hd Where hd.IdHoaDon = @IdHoaDon

--Bây giờ số tiền đang nợ đã bao gồm tiền còn cần phải trả của hóa đơn tháng đó

Select @SoTienDangNo = n.SoTienNo

From tb\_ThongTinNo n Where n.id\_ThongTinNo\_Phong = @IdPhongDcThanhToan

If @TienThanhToan > @SoTienDangNo

Begin

Print N'Trả dư tiền rồi !!?'

Rollback

End

Else

Begin

--Xóa thông tin nợ hiện tại của phòng đó

If @TienThanhToan = @SoTienDangNo

Begin

Print N'Đã thanh toán hết hóa đơn tháng này và thanh toán hết nợ'

Exec proc\_GanNoMoiHoacXoaNo 0,@IdPhongDcThanhToan

End

Else

Begin

Declare @TienNoMoi decimal

If @TienThanhToan > @SoTienConLaiHoaDon And @TienThanhToan < @SoTienDangNo

Begin

Set @TienNoMoi = @SoTienDangNo - @TienThanhToan

Print N'Đã thanh toán hết hóa đơn tháng này nhưng vẫn còn nợ'

Exec proc\_GanNoMoiHoacXoaNo @TienNoMoi, @IdPhongDcThanhToan

End

--Cũng vẫn cập nhật nợ vì nợ là bao gồm cả số tiền còn lại cần thanh toán của hóa đơn nên cứ mỗi lần thanh toán đều cần phải cập nhật nợ

Else If @TienThanhToan < @SoTienConLaiHoaDon

Begin

Set @TienNoMoi = @SoTienDangNo - @TienThanhToan

Print N'Đã thanh toán nhưng chưa hết hóa đơn tháng này'

Exec proc\_GanNoMoiHoacXoaNo @TienNoMoi, @IdPhongDcThanhToan

End

End

--Cập nhật thông tin nợ của phòng đó

End

End

Go

***- Trigger KhongTonTaiHaiHoaDonTrongCungThang:***

Mỗi khi có hóa đơn mới được xuất,trigger đảm bảo hóa đơn vừa xuất của một phòng và hóa đơn xuất gần đây nhất của phòng đó cách nhau ít nhất 1 tháng sử dụng hàm DateDiff có sẵn của SQL Server

Code:

Create Trigger trg\_KhongTonTaiHaiHoaDonTrongCungThang On tb\_HoaDon

After Insert

As

Begin

Declare @NgayXuatHoaDonHienTai date,@IdPhongDuocThanhToan int,@IdHoaDonHienTai int,

@NgayXuatHoaDonGanNhat date

Select @NgayXuatHoaDonHienTai = ins.NgayXuatHoaDon,@IdPhongDuocThanhToan = ins.id\_HoaDon\_Phong,@IdHoaDonHienTai = ins.IdHoaDon

From inserted ins

--Nếu phòng đó có tồn tại ít nhất nhất một hóa đơn

If Exists (Select Top(1) \* From tb\_HoaDon Where id\_HoaDon\_Phong = @IdPhongDuocThanhToan And IdHoaDon != @IdHoaDonHienTai)

Begin

Select Top(1) @NgayXuatHoaDonGanNhat = NgayXuatHoaDon

From tb\_HoaDon hd

Where id\_HoaDon\_Phong = @IdPhongDuocThanhToan And IdHoaDon != @IdHoaDonHienTai

Order By NgayXuatHoaDon Desc

If DateDiff(Month,@NgayXuatHoaDonGanNhat,@NgayXuatHoaDonHienTai) < 1

Begin

Raiserror (N'Hóa đơn không hợp lệ vì, ngày xuất hóa đơn phải cách ngày xuất hóa đơn mới nhất hơn 1 tháng',16,1)

Rollback

End

End

End

## 5. Function

***- Function SoKhachThuocPhong***: Function này thực hiện việc đếm số lượng khách trong một phòng

Code:

Create Function func\_SoKhachThuocPhong(@IdPhong int)

Returns int

Begin

Declare @SoLuongKhach int

Select @SoLuongKhach = Count(kt.IdKhachTro)

From tb\_KhachTro kt Where kt.id\_KhachTro\_Phong = @IdPhong

Return @SoLuongKhach

End

Go

***- Function NoPhong***: Function này sẽ trả về tổng số tiền nợ của một phòng

Code:

Create Function func\_NoPhong(@IdPhong int)

Returns decimal

Begin

Declare @TongTienNo decimal

Select @TongTienNo = ttn.SoTienNo

From tb\_ThongTinNo ttn Where ttn.id\_ThongTinNo\_Phong = @IdPhong

Return @TongTienNo

End

Go

***- Function SoLanThanhToanThieuSauThang***: Function này thực hiện đếm số lần đóng tiền thiếu của một phòng trọ trong sáu tháng.

Code:

Create Function func\_SoLanThanhToanThieuSauThang(@IdPhong int)

Returns tinyint

Begin

Declare @SoLanDongThieu tinyint

Select @SoLanDongThieu = Count(hd.IdHoaDon)

From tb\_HoaDon hd

Where DATEDIFF(month,hd.NgayXuatHoaDon,GETDATE()) <= 6 And hd.TrangThaiHoaDon = 2 And hd.id\_HoaDon\_Phong = @IdPhong

Return @SoLanDongThieu

End

Go

***- Function CacHoaDonThanhToanDuocCuaPhong*** : Function này thực hiện trả về bảng các hóa đơn của phòng có mã truyền vào mà hiện còn thanh toán được . Tức thời điểm hiện tại nằm trong khoảng thời gian xuất hóa đơn đến khi hết kì hạn thanh toán và hóa đơn vẫn còn một số tiền phải trả(ta biết được điều này dựa vào trạng thái hóa đơn)

Code:

Create Function func\_CacHoaDonThanhToanDuocCuaPhong(@IdPhong int)

Returns Table

As Return (Select \* From tb\_HoaDon hd

Where hd.TrangThaiHoaDon In (1,2) And (GETDATE() Between hd.NgayXuatHoaDon And hd.KiHanThanhToan) And hd.id\_HoaDon\_Phong = @IdPhong)

Go

***- Function DangNhap***: Function này thực hiện trả về idKhachTro mà có cột tên đăng nhập khớp với TenDangNhap được truyền xuống từ form đăng nhập

Code:

Create Function func\_DangNhap(@TenDangNhap nvarchar(Max))

Returns int

Begin

Declare @IdKhachTroDangNhap int

Select @IdKhachTroDangNhap = kt.IdKhachTro

From tb\_KhachTro kt

Where kt.TenDangNhap = @TenDangNhap

Return @IdKhachTroDangNhap

End

Go

## 6. Procedure

***- Proc GanNoMoiHoacXoaNo:*** Proc này thực hiện việc gán hoặc xóa nợ của một phòng trọ trong bảng ThongTinNo:

* Nếu phòng trọ trả hết nợ (Nợ = 0), proc sẽ xóa thông tin nợ của phòng đó khỏi bảng ThongTinNo.
* Nếu phòng trọ còn nợ, proc sẽ cập nhật nợ mới nhất của phòng đó vào trong bảng ThongTinNo

Code:

Create Procedure proc\_GanNoMoiHoacXoaNo

@NoMoi decimal(18, 0), @IdPhong int

As

Begin

If @NoMoi = 0

Delete From tb\_ThongTinNo Where @IdPhong = tb\_ThongTinNo.id\_ThongTinNo\_Phong;

Else

Update tb\_ThongTinNo

Set SoTienNo = @NoMoi

Where @IdPhong = tb\_ThongTinNo.id\_ThongTinNo\_Phong;

End

Go

***- Proc XoaCacKhachThuocPhong***: Proc này thực hiện việc xóa thông tin toàn bộ khách trong một phòng ra khỏi bảng KhachTro

Code:

Create Procedure proc\_XoaCacKhachThuocPhong(@IdPhong int)

As

Begin

Delete From tb\_KhachTro

Where id\_KhachTro\_Phong = @IdPhong

End

Go

***- Proc DuoiKhachTroViPhamNoiQuy***: Proc này thực hiện việc xóa tất cả thông tin khách trọ trong một phòng ra khỏi bảng KhachTro khi phòng đó vi phạm một trong các nội quy sau: số tiền nợ vượt quá 200000 đồng hoặc số lần thanh toán thiếu từ 4 tháng liên tục trở lên.

Code:

Create Proc proc\_DuoiKhachTroViPhamNoiQuy

As

Begin

Declare @IdPhongDangXet int

--Khởi tạo Cursor sẽ là danh sách các id của các phòng

Declare IdPhong\_Cursor Cursor For

Select IdPhong From tb\_Phong

--Mở cursor ra để lặp qua từng mã phòng trong danh sách

Open IdPhong\_Cursor

--Lấy hàng dữ liệu đầu trong cursor ra và gán cho biến @IdPhongDangXet

Fetch Next From IdPhong\_Cursor Into @IdPhongDangXet

--Lặp qua từng id phòng, điều kiện dừng là việc fetch thất bại

While @@FETCH\_STATUS = 0

Begin

--Nếu vi phạm nội quy -> Xóa Tất cả khách thuộc phòng đó và các khoản nợ hiện tại của phòng

If dbo.func\_NoPhong(@IdPhongDangXet) > 2000000 Or dbo.func\_SoLanThanhToanThieuSauThang(@IdPhongDangXet) > 3

Begin

Exec proc\_XoaCacKhachThuocPhong @IdPhongDangXet

Exec proc\_GanNoMoiHoacXoaNo 0, @IdPhongDangXet

--Thêm để cập nhật các hóa đơn của phòng hiện tại,xem lại trạng thái

Update tb\_HoaDon Set TrangThaiHoaDon = 4 Where TrangThaiHoaDon !=4 And id\_HoaDon\_Phong = @IdPhongDangXet

End

Fetch Next From IdPhong\_Cursor Into @IdPhongDangXet

End

Close IdPhong\_Cursor

DeAllocate IdPhong\_Cursor

End

Go

***- Proc XoaMotKhachTro:***

Chỉ xóa một khách trọ ra khỏi phòng đó, sẽ sử dụng khi khách muốn rời khỏi phòng

Code:

Create Proc proc\_XoaMotKhachTro(@IdKhach int)

As

Begin

Delete From tb\_KhachTro

Where IdKhachTro=@IdKhach

End

Go

***Proc XuatHoaDonPhong:***

Tạo ra một hóa đơn mới cho một phòng, thường được thực hiện trong tháng mới bởi admin

Code:

Create Proc proc\_XuatHoaDonPhong(

@IdPhongXuatHoaDon int,

@NgayXuatHoaDon Date,

@KiHanThanhToan Date,

@SoDien int,

@SoNuoc int

)

As

Begin

Insert Into tb\_HoaDon (id\_HoaDon\_Phong,NgayXuatHoaDon,KiHanThanhToan,SoDien,SoNuoc)

Values (@IdPhongXuatHoaDon, @NgayXuatHoaDon,@KiHanThanhToan,@SoDien,@SoNuoc)

End

Go

***Proc ThanhToanHoaDon:***

Thêm một lần thanh toán vào bảng LanThanhToan, lần thanh toán này được thực hiện bởi một khách trọ cho một hóa đơn. Đã đảm bảo khách thanh toán đúng hóa đơn của phòng mình nhờ trigger ToanVenPhongDuocThanhToan

Code:

Create Procedure proc\_ThanhToanHoaDon

@IdKhachThanhToan int,

@IdHoaDonThanhToan int,

@SoTienThanhToan decimal,

@NgayThanhToan Date

As

Begin

Insert Into tb\_LanThanhToan(id\_ThanhToan\_KhachTro,id\_ThanhToan\_HoaDon,SoTienTra,NgayThanhToan)

Values (@IdKhachThanhToan,@IdHoaDonThanhToan,@SoTienThanhToan,@NgayThanhToan)

End

Go

***Proc ThemKhachTro:***

Thêm khách trọ mới vào bảng KhachTro, thủ tục này sẽ được gọi khi admin đăng ký tài khoản cho một khách trọ mới

Thực hiện tạo Login với TenDangNhap và MatKhau lấy từ form đăng ký

Tạo User có tên là user\_TenDangNhap cho login vừa tạo trong cơ sở dữ liệu hiện tại

Tiếp theo là gán user đó vào role\_KhachTro đã được tạo sẵn khi tạo cơ sở dữ liệu. Như vậy khi khách trọ login vào thì họ chỉ được phép thực hiện các quyền hạn trong role\_KhachTro

Code:

Create Procedure proc\_ThemKhachTro

@CCCD nchar(12),

@Hoten nvarchar(200),

@Sdt nchar(10),

@NamSinh int,

@QueQuan nvarchar(300),

@Nghe nvarchar(300),

@idPhong int,

@TenDangNhap varchar(500),

@MatKhau nvarchar(Max)

As

Begin Tran

Insert Into tb\_KhachTro(CCCD, HoTen, SoDienThoai, NamSinh, QueQuan, NgheNghiep, id\_KhachTro\_Phong, TenDangNhap)

Values (@CCCD,@Hoten,@Sdt,@NamSinh,@QueQuan,@Nghe,@idPhong, @TenDangNhap)

Declare @Sql nvarchar(Max)

Set @Sql = 'Create Login ' + @TenDangNhap +

' With Password = ''' + @MatKhau + '''; ' +

'Create User user\_' + @TenDangNhap + ' For Login ' + @TenDangNhap

Exec (@Sql)

Declare @TenUser varchar(505)

Set @TenUser = Concat('user\_', @TenDangNhap)

Exec sp\_addrolemember @rolename= 'role\_KhachTro',@membername= @TenUser

If @@ERROR = 0

Begin

Commit

End

Else

Begin

Print 'Xảy ra lỗi trong proc\_ThemKhachTro'

Rollback

End

Go

***Proc CapNhatThongTinKhachTro:***

Cập nhật thông tin khách trọ trong bảng KhachTro

Code:

Create Proc proc\_CapNhatThongTinKhachTro (

@idKhachTro int,

@cccd nchar(12),

@hoTen nvarchar(200),

@sdt nchar(10),

@namSinh int,

@queQuan nvarchar(300),

@ngheNghiep nvarchar(300)

)

As

Begin

Update tb\_KhachTro

Set

CCCD = @cccd,

HoTen = @hoTen,

SoDienThoai = @sdt,

NamSinh = @namSinh,

QueQuan = @queQuan,

NgheNghiep = @ngheNghiep

Where IdKhachTro = @idKhachTro

End

Go

***Proc CapNhatDonGia:***

Cập nhật thông tin đơn giá trong bảng DonGia

Code:

Create Proc proc\_CapNhatDonGia (@idDonGiaPhong int, @donGia DECIMAL)

As

Begin

Update tb\_DonGiaPhong

Set DonGia = @donGia

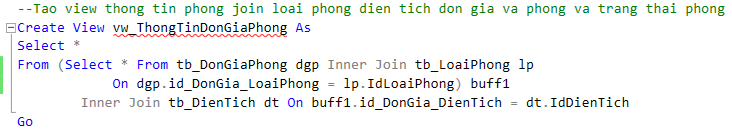
Where IdDonGiaPhong = @idDonGiaPhong

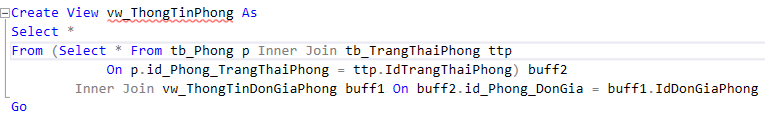
End

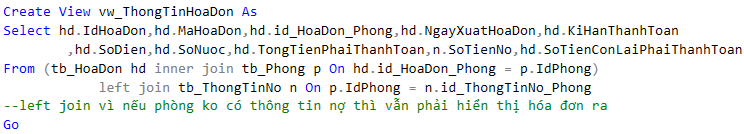
Go

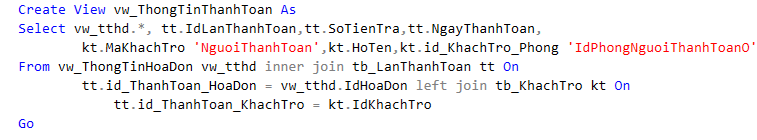
## 7. View:

*Các khách trọ sẽ truy xuất dữ liệu trong cơ sở dữ liệu thông các bảng ảo là view sau:*

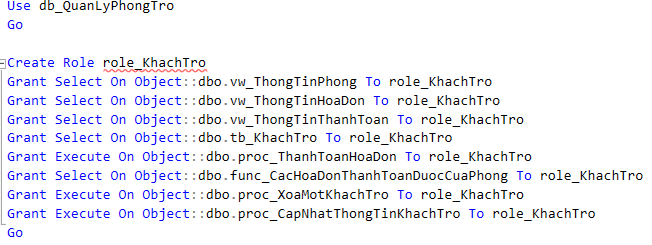








## 8. Role:



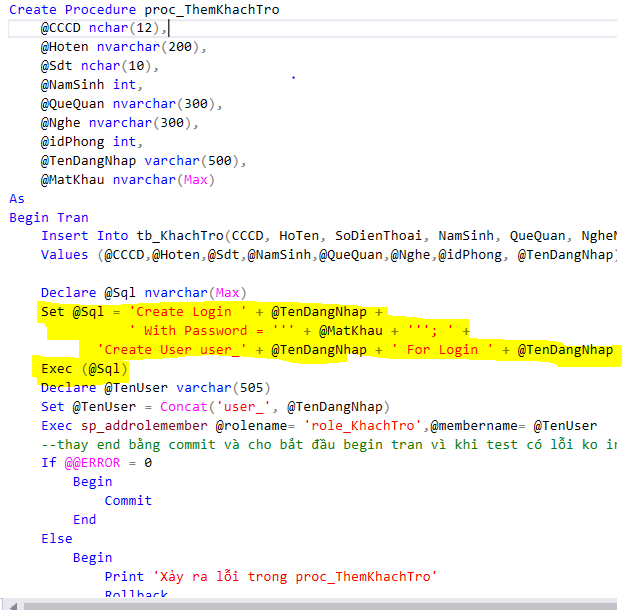
Tạo role khách trọ, có quyền select các view, bảng tb\_KhachTro và các hóa đơn hiện thanh toán được từ func\_CacHoaDonThanhToanDuocCuaPhon

Khách trọ còn có thể thực hiện proc\_ThanhToanHoaDon, proc\_XoaMotKhachTro nếu khách đó muốn rời phòng hoặc cập nhật thông tin khách trọ. Mục đích khi tạo role là để các khách trọ chỉ được phép truy vấn cơ sở dữ liệu ở một mức quyền hạn cụ thể và có giới hạn.

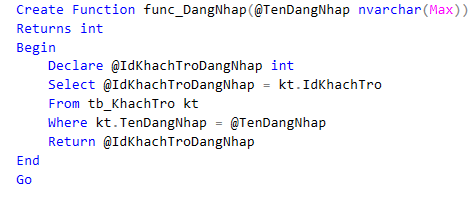
# III.ỨNG DỤNG SQL SERVER SECURITY:

Ứng dụng có sử dụng chức năng security của SQL Server để thực hiện đăng nhập và truy vấn dữ liệu cho khách trọ.

Khi admin đăng ký khách trọ trên ứng dụng với một tên đăng nhập và mật khẩu, proc\_ThemKhachTro sẽ tạo login ứng vơi thông tin đó và tạo user trong database ứng với login đó. Sau đó thêm user vào role\_KhachTro



Khi khách trọ thực hiện đăng nhập họ nhập TenDangNhap và MauKhau, khi nhấn đăng nhập sẽ thực hiện func\_DangNhap sẽ so khớp với bảng tb\_KhachTro và trả về Id của khách có thông tin cột TenDangNhap khớp với tham số @TenDangNhap truyền vào(nếu có). Nếu không thì sẽ thông báo đăng nhập thất bại



Khi func\_DangNhap thành công, ứng dụng sẽ thay đổi connection string để kết nối với cơ sở dữ liệu theo SQL Server Authentication với TenDangNhap và MauKhau vừa nhập vào form, nếu có kết nối và truy vấn thành công, chứng tỏ có tồn tại login và user ứng với hai thông tin vừa nhập. Lúc này, quá trình đăng nhập mới thực sự thành công