

TRƯỜNG ĐẠI HỌC XÂY DỰNG

KHOA CÔNG NGHỆ THÔNG TIN



**BÁO CÁO ĐỒ ÁN**

**CƠ SỞ DỮ LIỆU**

***Đề tài: Chương trình quản lý khách sạn***

**Lớp:** *64PM1.*

Học kỳ: *1*

**Nhóm thực hiện:** *Nhóm 09*

1. *Nguyễn Công Bình (Nhóm trưởng)*
2. *Nguyễn Tùng Sơn*
3. *Bùi Thị Hạnh*

**Giảng viên hướng dẫn:** *Thầy Nguyễn Đình Anh.*

**HÀ NỘI,** 01/2023

Mục lục

[**Lời mở đầu** 3](#_Toc126533183)

[**\*\*\*Mục tiêu và ý nghĩa của Đồ án hệ thống quản lý Khách sạn** 4](#_Toc126533184)

[**I.** **Mô tả hệ thống** 4](#_Toc126533185)

[ ***Quản lý khách hàng***: 4](#_Toc126533186)

[ ***Quản lý phòng****:* 4](#_Toc126533187)

[ ***Quản lý đặt phòng****:* 4](#_Toc126533188)

[ ***Quản lý nhân viên:*** 4](#_Toc126533189)

[ ***Quản lý dịch vụ****:* 4](#_Toc126533190)

[**II.** **Phân tích dữ liệu** 5](#_Toc126533191)

[**1.** **Thực thể: Nhân viên** 5](#_Toc126533192)

[**2.** **Thực thể: Tài khoản** 5](#_Toc126533193)

[**3.** **Thực thể: Loại phòng** 5](#_Toc126533194)

[**4.** **Thực thể: Phòng** 5](#_Toc126533195)

[**5.** **Thực thể: Khách hàng** 5](#_Toc126533196)

[**6.** **Thực thể: Đồ ăn** 6](#_Toc126533197)

[**7.** **Thực thể: Đồ uống** 6](#_Toc126533198)

[**III.** **Mô hình hóa các thực thể liên kết về dạng chuẩn 3** 7](#_Toc126533199)

[**1.** **Quan hệ giữa thực thể “Nhân viên” và thực thể “Tài khoản”** 7](#_Toc126533200)

[**2.** **Thuộc tính đa trị trong thực thể “Phòng”** 8](#_Toc126533201)

[**3.** **Quan hệ giữa thực thể “Loại phòng” và thực thể “Phòng”** 9](#_Toc126533202)

[**4.** **Quan hệ giữa thực thể “Nhân viên”, “Khách hàng” và “Phòng”** 10](#_Toc126533203)

[**5.** **Quan hệ giữa thực thể “Phòng” và thực thể “Đồ ăn”** 12](#_Toc126533204)

[**6.** **Quan hệ giữa thực thể “Phòng” và thực thể “Đồ uống”** 13](#_Toc126533205)

[**IV.** **Mô hình dữ liệu quan hệ quản lý khách sạn** 14](#_Toc126533206)

[**V.** **Thiết kế cơ sở dữ liệu trong sql server** 17](#_Toc126533207)

[**1. Tạo cơ sở dữ liệu** 17](#_Toc126533208)

[**2. Tạo hàm kiểm tra điều kiện đầu vào** 17](#_Toc126533209)

[**3. Tạo các bảng Nhân viên,Tài khoản, Loại phòng, Phòng, Trạng thái phòng, Khách hàng, Đồ ăn, Đăng ký đồ ăn, Đồ uống, Đặt phòng** 18](#_Toc126533210)

[**4. Tạo liên kết giữa các bảng** 24](#_Toc126533211)

[**5. Sử dụng PROC thêm nhân viên** 26](#_Toc126533212)

[**6. Sử dụng PROC sửa thông tin nhân viên** 27](#_Toc126533213)

[**7. Sử dụng TRIGGER đăng ký tài khoản** 28](#_Toc126533214)

[**8. Sử dụng PROC để kiểm tra đăng nhập** 30](#_Toc126533215)

[**9. Sử dụng PROC để thêm loại phòng** 31](#_Toc126533216)

[**10. Sử dụng PROC để sửa thông tin loại phòng** 32](#_Toc126533217)

[**11. Sử dụng PROC để thêm khách hàng** 32](#_Toc126533218)

[**12. Sử dụng PROC để sửa thông tin khách hàng** 33](#_Toc126533219)

[**13. Sử dụng TRIGGER để thêm phòng** 34](#_Toc126533220)

[**14. Sử dụng PROC để cập nhật dữ liệu** 36](#_Toc126533221)

[**15. Sử dụng PROC để thêm tự động thêm trạng thái trống cho phòng** 36](#_Toc126533222)

[**(theo ngày nếu nó hiện tại không có ai đặt/hẹn)** 36](#_Toc126533223)

[**16. Sử dụng PROC để đặt phòng** 37](#_Toc126533224)

[**17. Sử dụng PROC để thêm đồ ăn** 41](#_Toc126533225)

[**18. Sử dụng PROC để sửa thông tin đồ ăn** 41](#_Toc126533226)

[**19. Sử dụng PROC để thêm đồ uống** 42](#_Toc126533227)

[**20. Sử dụng PROC để sửa thông tin đồ uống** 43](#_Toc126533228)

[**21. Sử dụng PROC để đặt đồ ăn** 44](#_Toc126533229)

[**22. Sử dụng PROC để đặt đồ uống** 45](#_Toc126533230)

[**23. Sử dụng PROC để thanh toán tiền** 46](#_Toc126533231)

[**24. Sử dụng PROC để in hóa đơn đồ ăn theo phòng** 47](#_Toc126533232)

[**25. Sử dụng PROC để in hóa đơn đồ uống theo phòng** 47](#_Toc126533233)

[**26. Sử dụng PROC để in hóa đơn tiền đặt phòng** 47](#_Toc126533234)

[**27. Sử dụng PROC để in hóa đơn theo phòng gồm tổng chi phí (Phòng, ăn, uống)** 48](#_Toc126533235)

[**28. Sử dụng PROC để in hóa đơn theo khách hàng** 49](#_Toc126533236)

[**29. Sử dụng PROC để tính doanh thu của khách sạng trong tháng/năm** 50](#_Toc126533237)

[**30. Sử dụng PROC để thống kê top3 khách hàng đặt/hẹn phòng nhiều nhất** 51](#_Toc126533238)

[**31. Sử dụng PROC để thống kê những khách hàng quá hạn thanh toán** 51](#_Toc126533239)

[**32. Sử dụng PROC để in ra chi tiết thông tin khách hàng quá hạn thanh toán** 52](#_Toc126533240)

[**33. Sử dụng PROC để thống kê những khách hàng đang hẹn phòng** 52](#_Toc126533241)

[**34. Sử dụng PROC để in ra chi tiết thông tin khách hàng đang hẹn phòng** 52](#_Toc126533242)

[**35. Sử dụng PROC để thống kê những khách hàng đặt phòng trong tháng/năm , số lượng khách hàng đặt** 53](#_Toc126533243)

[**36. Sử dụng PROC để thống kê top3 phòng được đặt nhiều nhất** 53](#_Toc126533244)

[**37. Sử dụng PROC để thống kê top3 món ăn được đặt nhiều nhất** 53](#_Toc126533245)

[**38. Sử dụng PROC để thống kê top3 đồ uống được đặt nhiều nhất** 54](#_Toc126533246)

[**VI.** **Cài đặt chương trình java đồng bộ với cơ sở dữ liệu SQL** 54](#_Toc126533247)

[***1.*** ***Form đăng nhập*** 54](#_Toc126533248)

[***2.*** ***Form đăng ký*** 55](#_Toc126533249)

[***3.*** ***Menu*** 56](#_Toc126533250)

[***4.*** ***Form tài khoản*** 57](#_Toc126533251)

[***5.*** ***Form quản lý đồ ăn/đồ uống/loại phòng*** 58](#_Toc126533252)

[***6.*** ***Form quản lý phòng*** 59](#_Toc126533253)

[***7.*** ***Form quản lý nhân viên*** 60](#_Toc126533254)

[***8.*** ***Form quản lý khách hàng*** 61](#_Toc126533255)

[***9.*** ***Form trạng thái phòng theo ngày*** 62](#_Toc126533256)

[***10.*** ***Form đặt phòng*** 63](#_Toc126533257)

[***11.*** ***Form đặt đồ ăn/đồ uống*** 64](#_Toc126533258)

[***12.*** ***Form báo cáo thống kê*** 65](#_Toc126533259)

[**VII.** **Nhận xét về phần cơ sở dữ liệu** 66](#_Toc126533260)

# **Lời mở đầu**

* Ngày nay, công nghệ thông tin có mặt trong hầu hết các lĩnh vực của đời sống kinh tế xã hội đặc biệt nó là công cụ hỗ trợ đắc lực trong công tác quản lý. Việc đưa tin học vào trong quản lý kinh doanh là một trong những ứng dụng quan trọng nhất trong nhiều ứng dụng của cơ sở dữ liệu. Nhờ vào việc tin học hoá mà công việc quản lý và điều hành doanh nghiệp tỏ ra nhanh chóng và hiệu quả hơn rất nhiều.
* Hiện nay, hầu hết các tổ chức, doanh nghiệp đều đã nhận thức được tầm quan trọng và tác dụng to lớn của việc áp dụng tin học hoá trong công tác quản lý của mình. Vì vậy mà nhu cầu xây dựng những phần mềm quản lý của các doanh nghiệp rất lớn. Hầu hết các doanh nghiệp, tổ chức dù lớn hay nhỏ đều đang có xu hướng muốn áp dụng tin học trong công tác quản lý.
* Cùng với sự phát triển to lớn đó, song song với việc phát triển về chiều rộng thì phát triển về chiều sâu cũng được nhiều nhà phát triển quan tâm. Và một vấn đề đặt ra là Internet đang ngày càng bùng nổ và tiếp cận gẫn gũi với con người nên làm việc trên Internet là một nhu cầu bức thiết. Vì thế dần dần các hệ thống phân bố hay các mô hình phân tán được áp dụng triệt để trong việc phát triển phần mềm và trong các ứng dụng CNTT nói chung để phục vụ tốt hơn nhu cầu của cá nhân cũng như doanh nghiệp trong đời sống và kinh doanh.
* Trong khi đời sống con người nâng cao, nhu cầu du lịch ngày càng phát triển thì nhu cầu tìm những khách sạn có phòng nghỉ chất lượng và dịch vụ tốt để nghỉ ngơi thư giãn ngày càng phổ biến.
* Trước kia khi chưa ứng dụng tin học vào công tác quản lý thì các công việc như cập nhật thông tin khách hàng, xem các thông tin về dịch vụ của khách sạn, thông tin về số phòng, và nhân viên, hoá đơn thanh toán, tất cả đều làm thủ công nên các cán bộ quản lý phải mất rất nhiều thời gian mới tổng hợp được những thông tin cần thiết.
* Mặt khác hiện nay trong các khách sạn, các cán bộ làm công tác quản lý thông tin của khách sạn phải quản lý một lượng lớn thông tin về cán bộ công nhân viên, tình trạng phòng… Việc này mất rất nhiều thời gian và công sức của con người nên việc ứng dụng tin học vào công tác quản lý khách sạn theo yêu cầu của người dùng ngày càng trở nên hết sức cần thiết, quan trọng.

# **\*\*\*Mục tiêu và ý nghĩa của Đồ án hệ thống quản lý Khách sạn**

* Đồ án hệ thống quản lý Khách sạn với mục đích cung cấp cho khách hàng các thông tin chính xác về các phòng tại có trong khách sạn và cách thức đặt phòng / hẹn phòng. Các thông tin được cập nhật thường xuyên và nhanh chóng. Vì vậy, rút ngắn được khoảng cách giữa khách hàng và các khách sạn cung cấp dịch vụ, đưa thông tin về các phòng tại các khách sạn nhanh chóng đến cho khách hàng.

1. **Mô tả hệ thống**

Sau khi khảo sát hiện trạng ở trên, một yêu cầu cấp thiết cần đặt ra là xây dựng và phát triển một phần mềm đáp ứng được nhu cầu tìm kiếm phòng, đặt phòng, các dịch vụ kèm theo phòng… Với việc xây dựng phần mềm thì ta có các yêu cầu về :

* ***Quản lý khách hàng***:

Mỗi khách hàng của khách sạn đều được khách sạn quản lý những thông tin sau: Họ tên, giới tính, Ngày sinh, điện thoại, số CMT,…

* ***Quản lý phòng****:*

Các phòng của khách sạn được quản lý dựa vào mã phòng,tên phòng, loại phòng, mô tả phòng, số người ở tối đa của phòng, giá phòng, trạng thái,…

* ***Quản lý đặt phòng****:*

Tất cả các thông tin đăng ký thuê phòng đều được quản lý dựa trên người đặt phòng, mã khách hàng, ngày đặt phòng, thời gian trả phòng,… Với mỗi phòng được đặt đều có mã đặt phòng do nhân viên thiết lập(mã nhân viên).

* ***Quản lý nhân viên:***

Tất cả những nhân viên làm việc tại khách sạn đều được quản lý các thông tin sau: Mã nhân viên, Tên nhân viên, Ngày sinh, Giới tính, Chức vụ của nhân viên,…

* ***Quản lý dịch vụ****:*

Khách sạn cung cấp một số dịch vụ: Dịch vụ đồ ăn, đồ uống.

1. **Phân tích dữ liệu**
2. **Thực thể: Nhân viên**

\*\*Các thuộc tính:

* Mã nhân viên
* Tên nhân viên
* Ngày sinh
* Giới tính
* CMT (Chứng minh thư nhân dân / thẻ căn cước công dân)
* SDT (Số điện thoại)

1. **Thực thể: Tài khoản**

\*\*Các thuộc tính:

* User (Tài khoản)
* Pass (Mật khẩu)

1. **Thực thể: Loại phòng**

\*\*Các thuộc tính:

* Mã loại phòng
* Mô tả
* Giá

1. **Thực thể: Phòng**

\*\*Các thuộc tính:

* Mã phòng
* Tên phòng
* Mô tả
* Số người tối đa
* Trạng thái phòng ( theo ngày )

1. **Thực thể: Khách hàng**

\*\*Các thuộc tính:

* Mã khách hàng
* Tên khách hàng
* Ngày sinh
* Giới tính
* CMT (Chứng minh thư nhân dân / Thẻ căn cước công dân)
* SDT (Số điện thoại)

1. **Thực thể: Đồ ăn**

\*\*Các thuộc tính:

* Mã đồ ăn
* Tên đồ ăn
* Giá

1. **Thực thể: Đồ uống**

\*\*Các thuộc tính:

* Mã đồ uống
* Tên đồ uống
* Giá

1. **Mô hình hóa các thực thể liên kết về dạng chuẩn 3**
2. **Quan hệ giữa thực thể “Nhân viên” và thực thể “Tài khoản”**

Tài khoản

Nhân viên

Là một

* Từ liên kết giữa 2 thực thể **Nhân viên** và **Tài khoản** thì xác định được:

**\*Bảng Nhân viên**

Table

Description automatically generated

* Có: Khóa chính là **maNV**

**\*Bảng Tài khoản:**

Table

Description automatically generated

* Có: Khóa chính là **maNV** và là khóa ngoại liên kết với **maNV** ở bảng nhân viên
* Mỗi một nhân viên chỉ có một tài khoản duy nhất

1. **Thuộc tính đa trị trong thực thể “Phòng”**

Phòng

* Trạng thái của phòng được tính theo ngày
* **Trạng thái** là đa trị nên ta tách ra thêm 1 bảng:

**\*Bảng Trạng thái phòng:**

Table

Description automatically generated

* Có: Khóa chính là **(maPhong, ngay)**
* **maPhong** là khóa ngoại liên kết với bảng phòng

1. **Quan hệ giữa thực thể “Loại phòng” và thực thể “Phòng”**

1

Phòng

Thuộc

N

Loại phòng

* Từ liên kết giữa 2 thực thể **Loại phòng** và **Phòng** thì xác định được:

**\*Bảng Loại phòng:**

Table

Description automatically generated

* Có: Khóa chính là **maLoaiPhong**

**\*Bảng Phòng:**

Graphical user interface, table

Description automatically generated

* Khóa chính là **maPhong**
* Khóa ngoại là **maLoaiPhong** liên kết với bảng loại phòng

1. **Quan hệ giữa thực thể “Nhân viên”, “Khách hàng” và “Phòng”**

N

N

N

Khách hàng

Phòng

Đặt phòng

Nhân viên

* Từ liên kết giữa thực thể **Nhân viên**, **Khách hàng** và **Phòng** thì xác định được:

**\*Bảng khách hàng:**

**Graphical user interface, table

Description automatically generated**

* Có: Khóa chính là **maKH**

**\*Bảng Đặt phòng:**

Graphical user interface, application, table

Description automatically generated

* Có: Khóa chính là **(maKH, maNV, maPhong, ngayDat)**
* **maKH** là khóa ngoại liên kết với bảng khách hàng
* **maNV** là khóa ngoại liên kết với bảng nhân viên
* **maPhong** là khóa ngoại liên kết với bảng phòng

1. **Quan hệ giữa thực thể “Phòng” và thực thể “Đồ ăn”**

N

N

Phòng

Đồ ăn

Đăng ký

* Từ liên kết giữa thực thể **Phòng** và **Đồ ăn** thì xác định được:

**\*Bảng Đồ ăn:**

Table

Description automatically generated

* Có: Khóa chính là **maDA**

**\*Bảng đăng ký đồ ăn:**

Table

Description automatically generated

* Có: Khóa chính là **(maPhong, maDA)**
* **maPhong** làkhóa ngoại liên kết của bảng Phòng
* **maDA** là khóa ngoại liên kết của bảng Đồ ăn

1. **Quan hệ giữa thực thể “Phòng” và thực thể “Đồ uống”**

N

N

Phòng

Đồ uống

Đăng ký

* Từ liên kết giữa thực thể **Phòng** và **Đồ uống** thì xác định được:

**\*Bảng Đồ uống:**

Table

Description automatically generated

* Có : Khóa chính là **maDU**

**\*Bảng đăng ký đồ uống:**

Table

Description automatically generated

* Có: Khóa chính là **(maPhong, maDU)**
* **maPhong** là khóa ngoại liên kết của bảng Phòng
* **maDU** là khóa ngoại liên kết của bảng Đồ uống

1. **Mô hình dữ liệu quan hệ quản lý khách sạn**

* tbl\_taikhoan(maNV, Taikhoan, maKhau)
* tbl\_nhanvien(maNV, tenNV, ngaySinh, gioiTinh, chucVu, cmt, sdt)
* tbl\_datphong(maKH, maNV, maPhong, ngayHenTra, giaHan)
* tbl\_khachhang(maKH, tenKH, ngaySinh, gioiTinh, cmt, sdt)
* tbl\_phong(maPhong, tenPhong, maLoaiPhong, mota, songuoitoida)
* tbl\_dangkydoan(maPhong,maDA,soLuong)
* tbl\_doAn(maDA, tenDA, gia)
* tbl\_trangthaiphong(maPhong, ngay, tinhTrang)
* tbl\_loaiPhong(maLoaiPhong, moTa, gia)
* tbl\_dangkyDoUong(maPhong, maDU, soLuong)
* tbl\_ doUong(maDU, tenDU, gia)

Diagram, schematic

Description automatically generated

1. **Thiết kế cơ sở dữ liệu trong sql server**

## **1. Tạo cơ sở dữ liệu**

-- Tạo database QLKS

USE master;

GO

IF DB\_ID('QLKS') IS NOT NULL

DROP DATABASE QLKS

GO

CREATE DATABASE QLKS;

GO

USE QLKS;

GO

## **2. Tạo hàm kiểm tra điều kiện đầu vào**

-- Tạo hàm kiểm tra số điện thoại nhập vào

create function funcSDT

(

@SoDT varchar(Max)

)

returns bit

as

begin

Declare @KQ bit;

Set @KQ=0;

Declare @So tinyint;

Set @So=0;

Declare @i tinyint;

Set @i=0;

While (@i<=len(@SoDT))

begin

if (charindex(substring(@SoDT,@i,1),'0123456789')!=0)

Set @So=@So+1;

Set @i=@i+1;

end

if (@SoDT='' or @So=10 or @So=11)

Set @KQ=1;

return @KQ;

END;

GO

## **3. Tạo các bảng Nhân viên,Tài khoản, Loại phòng, Phòng, Trạng thái phòng, Khách hàng, Đồ ăn, Đăng ký đồ ăn, Đồ uống, Đặt phòng**

\*Bảng Nhân viên gồm có:

- Mã nhân viên: maNV

- Tên nhân viên: tenNV

- Ngày sinh: ngaySinh

- Giới tính: gioiTinh

- Chức vụ: chucVu

- Chứng minh thư: cmt

- Số điện thoại: sdt

/\*

Tạo bảng nhân viên (các trường không được để trống)

…

* Trường giới tính (gioitinh) chỉ cho phép

là **Nam** hoặc **Nữ**

* Trường chứng minh thư(cmt) là phải là duy nhất
* Trường số điện thoại (sdt: duy nhất) sử dụng hàm kiểm tra số điện thoại funcSDT(sdt)

\*/

CREATE TABLE tbl\_nhanvien

(

maNV VARCHAR(30) CONSTRAINT kc\_tbl\_nhanvien PRIMARY KEY,

tenNV VARCHAR(50) NOT NULL,

ngaySinh DATE NOT NULL CONSTRAINT ck\_tbl\_nhanvien\_ngaySinh CHECK(DATEDIFF(YEAR,ngaySinh,CAST(GETDATE() AS DATE)) >= 18),

gioiTinh VARCHAR(5) NOT NULL CONSTRAINT ck\_tbl\_nhanvien\_gioitinh CHECK(gioiTinh IN('Nam','Nu')),

chucVu BIT NOT NULL,

cmt VARCHAR(12) UNIQUE NOT NULL,

sdt VARCHAR(11) UNIQUE NOT NULL CONSTRAINT ck\_tbl\_nhanvien\_sdt CHECK(dbo.funcSDT(sdt)=1)

);

GO

\*Bảng Tài khoản gồm có:

- Mã nhân viên: maNV

- Tài khoản: taiKhoan

- Mật khẩu: matKhau

/\*

Tạo bảng tài khoản

(mỗi một nhân viên chỉ có 1 tài khoản duy nhất)

* Tất cả các trường không được để trống

\*/

CREATE TABLE tbl\_taikhoan

(

maNV VARCHAR(30) CONSTRAINT kc\_tbl\_taikhoan PRIMARY KEY,

taiKhoan VARCHAR(50) NOT NULL,

matKhau VARCHAR(20) NOT NULL

);

GO

\*Bảng Loại phòng gồm có:

- Mã loại phòng: maLoaiPhong

- Mô tả: moTa

- Giá: gia

/\*

Tạo bảng loại phòng

(Tất cả các trường không được để trống)

* Giá phòng (gia) phải > 0

\*/

CREATE TABLE tbl\_loaiphong

(

maLoaiPhong VARCHAR(10) CONSTRAINT kc\_tbl\_loaiphong PRIMARY KEY,

moTa VARCHAR(30) NOT NULL,

gia MONEY NOT NULL CONSTRAINT ck\_tbl\_loaiphong\_gia

CHECK(gia > 0), -- gia theo ngay

--donvi VARCHAR(10) NOT NULL DEFAULT 'vnd'

);

GO

\*Bảng Phòng gồm có:

- Mã phòng: maPhong

- Tên phòng: tenPhong

- Mã loại phòng: maLoaiPhong

- Mô tả: mota

- Số người tối đa: songuoitoida

/\*

Tạo bảng phòng

(Tất cả các trường không được để trống)

* Số người tối đa (songuoitoida) phải > 0

\*/

CREATE TABLE tbl\_phong

(

maPhong VARCHAR(30) CONSTRAINT kc\_tbl\_phong PRIMARY KEY,

tenPhong VARCHAR(50) NOT NULL,

maLoaiPhong VARCHAR(10) NOT NULL,

mota VARCHAR(50) NOT NULL,

--trangthai VARCHAR(10) NOT NULL CONSTRAINT ck\_tbl\_phong\_trangthai CHECK(trangthai IN('dung','baotri')),

songuoitoida INT NOT NULL CONSTRAINT ck\_tbl\_phong\_soluongNguoi CHECK(songuoitoida > 0)

);

GO

\*Bảng Trạng thái phòng gồm có:

- Mã phòng: maPhong

- Ngày: ngay

- Tình trạng: tinhTrang

/\*

Tạo bảng trạng thái phòng (xét theo ngày)

(Tất cả các trường không được để trống)

* Tình trạng phòng (tinhTrang) chỉ cho phép

là **trong**, **dat** hoặc **hen**

\*/

CREATE TABLE tbl\_trangthaiphong

(

maPhong VARCHAR(30) NOT NULL,

ngay DATE NOT NULL,

tinhTrang VARCHAR(20) NOT NULL DEFAULT 'trong' CONSTRAINT ck\_tbl\_trangthaiphong CHECK(tinhTrang IN('trong','dat','hen')),

CONSTRAINT kc\_tbl\_trangthaiphong PRIMARY KEY(maPhong,ngay)

);

GO

\*Bảng Khách hàng gồm có:

- Mã khách hàng: maKH

- Tên khách hàng: tenKH

- Ngày sinh: ngaySinh

- Chứng minh thư: cmt

- Số điện thoại: sdt

/\*

Tạo bảng khách hàng

(Tất cả các trường không được trống)

* Trường giới tính(gioiTinh) phải là **Nam** hoặc **Nu**
* Chứng minh thư(cmt) phải là duy nhất
* Số điện thoại(sdt) phải là duy nhất và được kiểm tra bằng hàm funcSDT(sdt)

\*/

CREATE TABLE tbl\_khachhang

(

maKH VARCHAR(30) CONSTRAINT kc\_tbl\_khachhang PRIMARY KEY,

tenKH VARCHAR(50) NOT NULL,

ngaySinh DATE NOT NULL CONSTRAINT ck\_tbl\_khachhang\_ngaySinh CHECK(DATEDIFF(YEAR,ngaySinh,CAST(GETDATE() AS DATE)) >= 18),

gioiTinh VARCHAR(5) NOT NULL CONSTRAINT ck\_tbl\_khachhang\_gioitinh CHECK(gioiTinh IN('Nam','Nu')),

cmt VARCHAR(12) UNIQUE NOT NULL,

sdt VARCHAR(11) UNIQUE NOT NULL CONSTRAINT ck\_tbl\_khachhang\_sdt CHECK(dbo.funcSDT(sdt)=1)

);

GO

\*Bảng Đồ ăn gồm có:

- Mã đồ ăn: maDA

- Tên đồ ăn: tenDA

- Giá: gia

/\*

Tạo bảng đồ ăn (Các trường không được để trống)

* Giá (gia) phải > 0

\*/

CREATE TABLE tbl\_doAn

(

maDA VARCHAR(30) CONSTRAINT kc\_tbl\_doAn PRIMARY KEY,

tenDA VARCHAR(100) NOT NULL,

gia MONEY NOT NULL CONSTRAINT ck\_tbl\_doAn\_gia

CHECK(gia > 0),

--donvi VARCHAR(10) NOT NULL DEFAULT 'vnd'

);

GO

\*Bảng Đồ uống gồm có:

- Mã đồ uống: maDU

- Tên đồ uống: tenDU

- Giá: gia

/\*

Tạo bảng đồ uống (Các trường không được để trống)

* Giá (gia) phải > 0

\*/

CREATE TABLE tbl\_doUong

(

maDU VARCHAR(30) CONSTRAINT kc\_tbl\_doUong PRIMARY KEY,

tenDU VARCHAR(100) NOT NULL,

gia MONEY NOT NULL CONSTRAINT ck\_tbl\_doUong\_gia

CHECK(gia > 0),

--donvi VARCHAR(10) NOT NULL DEFAULT 'vnd'

);

GO

\*Bảng Đăng ký đồ ăn gồm có:

- Ngày: ngay

- Mã phòng: maPhong

- Mã đồ ăn: maDA

- Số lượng: soLuong

/\*

Tạo bảng đăng ký đồ ăn (Các trường không được để trống)

* Ngày đặt (ngay) phải là ngày hiện tại
* Số lượng(soluong) phải > 0

\*/

CREATE TABLE tbl\_dangkyDoAn

(

ngay DATE NOT NULL DEFAULT CAST(GETDATE() AS DATE),

maPhong VARCHAR(30) NOT NULL,

maDA VARCHAR(30) NOT NULL,

soLuong INT NOT NULL DEFAULT 1 CONSTRAINT ck\_tbl\_dangkyDoAn\_soLuong CHECK(soLuong > 0),

CONSTRAINT kc\_tbl\_dangkyDoAn PRIMARY KEY(ngay,maPhong,maDA)

);

GO

\*Bảng Đăng ký đồ uống gồm có:

- Ngày: ngay

- Mã phòng: maPhong

- Mã đồ uống: maDU

- Số lượng: soLuong

/\*

Tạo bảng đăng ký đồ uống(Các trường không được để trống)

* Ngày đặt (ngay) phải là ngày hiện tại
* Số lượng(soluong) phải > 0

\*/

CREATE TABLE tbl\_dangkyDoUong

(

ngay DATE NOT NULL DEFAULT CAST(GETDATE() AS DATE),

maPhong VARCHAR(30) NOT NULL,

maDU VARCHAR(30) NOT NULL,

soLuong INT NOT NULL DEFAULT 1 CONSTRAINT ck\_tbl\_dangkyDoUong\_soLuong CHECK(soLuong > 0),

CONSTRAINT kc\_tbl\_dangkyDoUong PRIMARY KEY(ngay,maPhong,maDU)

);

GO

\*Bảng Đặt phòng gồm có:

- Mã khách hàng: maKH

- Mã nhân viên: maNV

- Mã phòng: maPhong

- Ngày đặt: ngaDat

- Ngày trả: ngayTra

- Trạng thái: trangthai

- Thanh toán: thanhtoan

/\*

Tạo bảng đặt phòng(Các trường không được để trống)

* Trạng thái đặt phòng (trangthai) phải là

**dat** hoặc **hen**

\*/

CREATE TABLE tbl\_datphong

(

maKH VARCHAR(30) NOT NULL,

maNV VARCHAR(30) NOT NULL,

maPhong VARCHAR(30) NOT NULL,

ngayDat DATE NOT NULL,

ngayTra DATE NOT NULL,

trangthai VARCHAR(10) DEFAULT 'dat' NOT NULL CONSTRAINT ck\_tbl\_datphong\_trangthai CHECK(trangthai IN('dat','hen')),

thanhtoan BIT DEFAULT 0 NOT NULL,

CONSTRAINT kc\_tbl\_datPhong PRIMARY KEY(maKH,maNV,maPhong,ngayDat)

);

GO

## **4. Tạo liên kết giữa các bảng**

/\*

Tạo liên kết khóa ngoại **maNV**

của bảng tài khoản với bảng nhân viên

\*/

ALTER TABLE dbo.tbl\_taikhoan

ADD

CONSTRAINT kn\_tbl\_taikhoan\_maNV

FOREIGN KEY(maNV)

REFERENCES dbo.tbl\_nhanvien(maNV);

GO

/\*

Tạo liên kết khóa ngoại **maLoaiPhong**

của bảng phòng với bảng loại phòng

\*/

ALTER TABLE dbo.tbl\_phong

ADD

CONSTRAINT kn\_tbl\_phong\_maLoaiPhong

FOREIGN KEY(maLoaiPhong)

REFERENCES dbo.tbl\_loaiphong(maLoaiPhong)

GO

/\*

Tạo liên kết khóa ngoại **maPhong**

của bảng trạng thái phòng với bảng phòng

\*/

ALTER TABLE dbo.tbl\_trangthaiphong

ADD

CONSTRAINT kn\_tbl\_trangthaiphong\_maPhong

FOREIGN KEY(maPhong)

REFERENCES dbo.tbl\_phong(maPhong)

GO

/\*

Tạo liên kết khóa ngoại **maKH**

của bảng đặt phòng với bảng khách hàng

\*/

ALTER TABLE dbo.tbl\_datphong

ADD

CONSTRAINT kn\_tbl\_datphong\_maKH

FOREIGN KEY(maKH)

REFERENCES dbo.tbl\_khachhang(maKH)

GO

/\*

Tạo liên kết khóa ngoại **maNV**

của bảng đặt phòng với bảng nhân viên

\*/

ALTER TABLE dbo.tbl\_datphong

ADD

CONSTRAINT kn\_tbl\_datphong\_maNV

FOREIGN KEY(maNV)

REFERENCES dbo.tbl\_nhanvien(maNV)

GO

/\*

Tạo liên kết khóa ngoại **maPhong**

của bảng đặt phòng với bảng phòng

\*/

ALTER TABLE dbo.tbl\_datphong

ADD

CONSTRAINT kn\_tbl\_datphong\_maPhong

FOREIGN KEY(maPhong)

REFERENCES dbo.tbl\_phong(maPhong)

GO

/\*

Tạo liên kết khóa ngoại **maPhong**

của bảng đăng ký đồ ăn với bảng phòng

\*/

ALTER TABLE dbo.tbl\_dangkyDoAn

ADD

CONSTRAINT kn\_tbl\_dangkyDoAn\_maPhong

FOREIGN KEY(maPhong)

REFERENCES dbo.tbl\_phong(maPhong)

GO

/\*

Tạo liên kết khóa ngoại **maDA**

của bảng đăng ký đồ ăn với bảng đồ ăn

\*/

ALTER TABLE dbo.tbl\_dangkyDoAn

ADD

CONSTRAINT kn\_tbl\_dangkyDoAn\_maDA

FOREIGN KEY(maDA)

REFERENCES dbo.tbl\_doAn(maDA)

GO

/\*

Tạo liên kết khóa ngoại **maPhong**

của bảng đăng ký đồ uống với bảng phòng

\*/

ALTER TABLE dbo.tbl\_dangkyDoUong

ADD

CONSTRAINT kn\_tbl\_dangkyDoUong\_maPhong

FOREIGN KEY(maPhong)

REFERENCES dbo.tbl\_phong(maPhong)

GO

/\*

Tạo liên kết khóa ngoại **maDU**

của bảng đăng ký đồ uống với bảng đồ uống

\*/

ALTER TABLE dbo.tbl\_dangkyDoUong

ADD

CONSTRAINT kn\_tbl\_dangkyDoUong\_maDU

FOREIGN KEY(maDU)

REFERENCES dbo.tbl\_doUong(maDU)

GO

## **5. Sử dụng PROC thêm nhân viên**

/\*

Kiểm tra thông tin đầu vào của nhân viên

(Các trường không được để trống)

- Chức vụ (cv) dùng để phân quyền

- Số điện thoại với chứng minh thư phải là duy nhất

\*/

-- proc thêm nhân viên

CREATE PROC addNhanVien(@maNV VARCHAR(30), @tenNV VARCHAR(50),

@date DATE, @sex VARCHAR(5), @cv BIT, @cmt VARCHAR(12), @sdt VARCHAR(11))

AS

BEGIN

IF(@maNV='' OR @maNV IS NULL) PRINT('Ma nhan vien khong duoc trong')

ELSE IF(EXISTS(SELECT\*FROM dbo.tbl\_nhanvien WHERE maNV IN(@maNV)))

BEGIN

--PRINT('Ma nhan vien ton tai')

select cast(-3 as int) as ketqua

END

ELSE IF(@tenNV='' OR @tenNV IS NULL) PRINT('Ten nhan vien khong duoc trong')

ELSE IF(@date='' OR @date IS NULL) PRINT('Ngay sinh khong duoc trong')

ELSE IF( DATEDIFF(YEAR,@date,CAST(GETDATE() AS DATE)) < 18)

BEGIN

--PRINT('Nhan vien chua du 18 tuoi')

select cast(-2 as int) as ketqua

END

ELSE IF(@sex NOT IN('Nam','Nu')) PRINT('Gioi tinh phai la Nam/Nu')

ELSE IF(@cv NOT IN(0,1))

BEGIN

PRINT('Chuc vu phai la 0 hoac 1')

END

ELSE IF(@cmt='' OR @cmt IS NULL) PRINT('CMT khong duoc trong')

ELSE IF(EXISTS(SELECT\*FROM dbo.tbl\_nhanvien WHERE cmt IN(@cmt)))

BEGIN

--PRINT('CMT da ton tai')

select cast(-1 as int) as ketqua

END

ELSE IF(@sdt='' OR @sdt IS NULL) PRINT('So dien thoai khong duoc trong')

ELSE IF(EXISTS(SELECT\*FROM dbo.tbl\_nhanvien WHERE sdt IN(@sdt)) OR dbo.funcSDT(@sdt)<>1)

BEGIN

--PRINT('So dien thoai ton tai hoac nhap khong dung dinh dang');

select cast(0 as int) as ketqua

END

ELSE

BEGIN

INSERT INTO dbo.tbl\_nhanvien(maNV,tenNV,ngaySinh,gioiTinh,chucVu,cmt,sdt)VALUES

(@maNV,@tenNV,@date,@sex,@cv,@cmt,@sdt)

--PRINT('Them nhan vien '+@tenNV+' thanh cong!');

select cast(1 as int) as ketqua

END

END;

GO

/\*

Ví dụ thêm nhân viên:

\*/

EXEC dbo.addNhanVien @maNV = 'bibo', @tenNV = 'Nguyen Cong Binh', @date = '3/8/2001', @sex = 'Nam', @cv = 1, @cmt = '001201007344', @sdt = '0971912776'

GO

EXEC dbo.addNhanVien @maNV = 'ts', @tenNV = 'Nguyen Tung Son', @date = '3/8/2001', @sex = 'Nam', @cv = 1, @cmt = '091201207343', @sdt = '0818696268'

GO

EXEC dbo.addNhanVien @maNV = 'bth', @tenNV = 'Bui Thi Hanh', @date = '4/3/2001', @sex = 'Nu', @cv = 1, @cmt = '011201207544', @sdt = '0343066599'

GO

SELECT \* FROM dbo.tbl\_nhanvien

GO

## **6. Sử dụng PROC sửa thông tin nhân viên**

/\*

- Các trường không được để trống

- Số điện thoại với chứng minh thư phải là duy nhất

- Sửa thông tin nhân viên theo mã nhân viên

\*/

-- proc update nhân viên

CREATE PROC updateNhanVien(@maNV VARCHAR(30), @tenNV VARCHAR(50),

@date DATE, @sex VARCHAR(5), @cv BIT, @cmt VARCHAR(12), @sdt VARCHAR(11))

AS

BEGIN

IF(@maNV='' OR @maNV IS NULL) PRINT('Ma nhan vien khong duoc trong')

ELSE IF(NOT EXISTS(SELECT\*FROM dbo.tbl\_nhanvien WHERE maNV IN(@maNV)))

BEGIN

--PRINT('Ma nhan vien khong ton tai')

select cast(-3 as int) as ketqua

END

ELSE IF(@tenNV='' OR @tenNV IS NULL) PRINT('Ten nhan vien khong duoc trong')

ELSE IF(@date='' OR @date IS NULL) PRINT('Ngay sinh khong duoc trong')

ELSE IF(DATEDIFF(YEAR,@date,CAST(GETDATE() AS DATE)) < 18)

BEGIN

--PRINT('Nhan vien chua du 18 tuoi')

select cast(-2 as int) as ketqua

END

ELSE IF(@sex NOT IN('Nam','Nu')) PRINT('Gioi tinh phai la Nam/Nu')

ELSE IF(@cv NOT IN(0,1))

BEGIN

PRINT('Chuc vu phai la 0 hoac 1')

END

ELSE IF(@cmt='' OR @cmt IS NULL) PRINT('CMT khong duoc trong')

ELSE IF((SELECT COUNT(\*) FROM dbo.tbl\_nhanvien WHERE maNV IN(@maNV) AND cmt IN(@cmt)) > 1)

BEGIN

--PRINT('CMT da ton tai')

select cast(-1 as int) as ketqua

END

ELSE IF(@sdt='' OR @sdt IS NULL) PRINT('So dien thoai khong duoc trong')

ELSE IF((SELECT COUNT(\*) FROM dbo.tbl\_nhanvien WHERE maNV IN(@maNV) AND sdt IN(@sdt)) > 1

OR dbo.funcSDT(@sdt)<>1)

BEGIN

--PRINT('So dien thoai ton tai hoac nhap khong dung dinh dang');

select cast(0 as int) as ketqua

END

ELSE

BEGIN

UPDATE dbo.tbl\_nhanvien SET tenNV=@tenNV, ngaySinh=@date, gioiTinh=@sex, chucVu=@cv, cmt=@cmt, sdt=@sdt WHERE maNV IN(@maNV);

--PRINT('Cap nhat du lieu thanh cong!');

select cast(1 as int) as ketqua

END

END;

GO

-- Sửa thông tin nhân viên:

EXEC dbo.updateNhanVien @maNV = 'bth', @tenNV = 'Bui Thi Hanh', @date = '4/3/2001', @sex = 'Nu', @cv = 1, @cmt = '011201207544', @sdt = '0343066599'

GO

## **7. Sử dụng TRIGGER đăng ký tài khoản**

-- trigger đăng ký tài khoản (1 nhân viên / 1 tài khoản)

/\*

* Các trường không được để trống,…
* User và password phải là duy nhất

\*/

CREATE TRIGGER addAccount ON dbo.tbl\_taikhoan INSTEAD OF INSERT

AS

BEGIN

DECLARE @maNV VARCHAR(30);

DECLARE @user VARCHAR(50);

DECLARE @pass VARCHAR(20);

SET @maNV = (SELECT DISTINCT Inserted.maNV FROM Inserted);

SET @user = (SELECT DISTINCT Inserted.taiKhoan FROM Inserted);

SET @pass = (SELECT DISTINCT Inserted.matKhau FROM Inserted);

IF(@maNV='' OR @maNV IS NULL) PRINT('Ma nhan vien khong duoc trong')

ELSE IF(@user='' OR @user IS NULL) PRINT('User khong duoc trong')

ELSE IF(@pass='' OR @pass IS NULL) PRINT('Pass khong duoc trong')

ELSE IF EXISTS (SELECT \* FROM dbo.tbl\_nhanvien WHERE maNV IN (@maNV))

BEGIN

IF EXISTS (SELECT \* FROM dbo.tbl\_taikhoan WHERE maNV IN (@maNV))

BEGIN

--PRINT(N'Mã nhân viên này đã được đăng ký -> Vui lòng nhập khác!');

select cast(-1 as int) as ketqua

END;

ELSE

BEGIN

IF EXISTS (SELECT \* FROM dbo.tbl\_taikhoan WHERE taiKhoan IN (@user))

BEGIN

--PRINT(N'Tài khoản này đã tồn tại!');

--ROLLBACK TRANSACTION;

select cast(0 as int) as ketqua;

END;

ELSE

BEGIN

INSERT INTO dbo.tbl\_taikhoan (maNV,taiKhoan,matKhau)VALUES(@maNV,@user,@pass);

--PRINT('Them tai khoan ' +@user+ ' thanh cong!')

select cast(1 as int) as ketqua;

END

END;

END;

ELSE

BEGIN

-- PRINT(N'Nhân viên có mã ' + @maNV + N' không tồn tại');

select cast(-2 as int) as ketqua;

END

END

GO

-- Ví dụ thêm tài khoản nhân viên

INSERT INTO dbo.tbl\_taikhoan(maNV,taiKhoan,matKhau)VALUES

('bibo','binhboong','1');

go

INSERT INTO dbo.tbl\_taikhoan(maNV,taiKhoan,matKhau)VALUES

('ts','sontung123','123');

GO

INSERT INTO dbo.tbl\_taikhoan(maNV,taiKhoan,matKhau)VALUES

('bth','hanhbui123','123456');

GO

SELECT \* FROM dbo.tbl\_taikhoan

GO

## **8. Sử dụng PROC để kiểm tra đăng nhập**

-- proc kiểm tra đăng nhập

/\*

* Các trường không được để trống
* Giá trị user và password đều trùng khớp

🡪 Đăng nhập thành công

🡪 Ngược lại đăng nhập không thành công

\*/

CREATE PROC checkLogin

@user VARCHAR(50),

@pass VARCHAR(20)

AS

BEGIN

--DECLARE @login INT;

IF (@user=''OR @user IS NULL OR @pass='' OR @pass IS NULL)

begin

select cast(0 as int) as ketqua

--PRINT('user/password khong duoc de trong!');

end

ELSE IF NOT EXISTS (SELECT \* FROM dbo.tbl\_taikhoan WHERE taiKhoan=@user)

BEGIN

--SET @login=0;

--SELECT @login AS [login];

select cast(0 as int) as ketqua

-----------------------------------------------------------------------------

--PRINT('Tai khoan '+@user+' khong ton tai -> Vui long dang ky!');

END

ELSE IF EXISTS (SELECT \* FROM dbo.tbl\_taikhoan WHERE taiKhoan=@user)

BEGIN

IF NOT EXISTS (SELECT \* FROM dbo.tbl\_taikhoan WHERE matKhau=@pass)

BEGIN

--SET @login=0;

--SELECT @login AS [login];

select cast(0 as int) as ketqua

---------------------------------------------------------------------

--PRINT('Mat khau sai -> Vui long nhap lai!');

END

ELSE

BEGIN

DECLARE @name NVARCHAR(50);

SET @name=(SELECT tenNV FROM dbo.tbl\_nhanvien INNER JOIN dbo.tbl\_taikhoan

ON tbl\_taikhoan.maNV = tbl\_nhanvien.maNV AND taiKhoan=@user);

-------------------------------------------------------------------------

--SET @login=1;

--SELECT @login AS [login];

select cast(1 as int) as ketqua

--PRINT('Nhan vien ' + @name + ' dang nhap thanh cong!')

END

END

END

GO

-- Ví dụ kiểm tra đăng nhập

EXECUTE dbo.checkLogin @user='binhboong',@pass='1';

GO

## **9. Sử dụng PROC để thêm loại phòng**

-- proc them loai phong

/\*

- Các trường không được để trống

- Mã loại phòng không được trùng lặp

- Giá phòng phải > 0

\*/

CREATE PROC addLoaiPhong(@maLP VARCHAR(10), @moTa VARCHAR(30), @gia MONEY)

AS

BEGIN

IF(@maLP='' OR @maLP IS NULL ) PRINT('Ma loai phong khong duoc trong')

ELSE IF(EXISTS(SELECT\*FROM dbo.tbl\_loaiphong WHERE maLoaiPhong IN(@maLP)))

begin

--PRINT('Ma loai phong ton tai')

select cast(-1 as int) as ketqua

end

ELSE IF(@moTa='' OR @moTa IS NULL) PRINT('Mo ta khong duoc trong')

ELSE IF(@gia IS NULL OR @gia <= 0)

BEGIN

--PRINT('Gia khong duoc trong va > 0')

SELECT cast(0 as int) as ketqua

END

ELSE

BEGIN

INSERT INTO dbo.tbl\_loaiphong(maLoaiPhong,moTa,gia)VALUES

(@maLP,@moTa,@gia)

--PRINT('Them loai phong ' +@maLP+ ' thanh cong!')

select cast(1 as int) as ketqua

END

END

GO

-- Thêm loại phòng

EXEC dbo.addLoaiPhong @maLP = 'vip', @moTa = 'Phong cao cap', @gia = 700

GO

EXEC dbo.addLoaiPhong @maLP = 'st1', @moTa = 'Phong tieu chuan loai 1', @gia = 500

GO

EXEC dbo.addLoaiPhong @maLP = 'st2', @moTa = 'Phong tieu chuan loai 2', @gia = 400

GO

## **10. Sử dụng PROC để sửa thông tin loại phòng**

-- proc sửa loại phòng

/\*

- Các trường không được để trống

- Giá phòng phải > 0

- Sửa thông tin theo mã loại phòng

\*/

CREATE PROC updateLoaiPhong(@maLP VARCHAR(10), @moTa VARCHAR(30), @gia MONEY)

AS

BEGIN

IF(@maLP='' OR @maLP IS NULL ) PRINT('Ma loai phong khong duoc trong')

ELSE IF(NOT EXISTS(SELECT\*FROM dbo.tbl\_loaiphong WHERE maLoaiPhong IN(@maLP)))

begin

--PRINT('Ma loai phong khong ton tai')

select cast(-1 as int) as ketqua

end

ELSE IF(@moTa='' OR @moTa IS NULL) PRINT('Mo ta khong duoc trong')

ELSE IF(@gia IS NULL OR @gia <= 0)

BEGIN

--PRINT('Gia khong duoc trong va > 0')

SELECT cast(0 as int) as ketqua

END

ELSE

BEGIN

UPDATE dbo.tbl\_loaiphong SET moTa=@moTa, gia=@gia WHERE maLoaiPhong IN(@maLP)

--PRINT('Cap nhat du lieu thanh cong!')

select cast(1 as int) as ketqua

END

END

GO

## **11. Sử dụng PROC để thêm khách hàng**

-- proc them khach hang

/\*

- Các trường không được để trống

- Chứng minh thư và số điện thoại phải là duy nhất

…

\*/

CREATE PROC addKhachHang(@maKH VARCHAR(30), @tenKH VARCHAR(50), @date DATE,

@sex VARCHAR(5), @cmt VARCHAR(12), @sdt VARCHAR(11))

AS

BEGIN

IF(@maKH='' OR @maKH IS NULL) PRINT('Ma khach hang khong duoc trong')

ELSE IF(EXISTS(SELECT\*FROM dbo.tbl\_khachhang WHERE maKH IN(@maKH)))

BEGIN

--PRINT('Ma khach hang ton tai');

SELECT cast(-3 as int) as ketqua

END

ELSE IF(@tenKH='' OR @tenKH IS NULL) PRINT('Ten khach hang khong duoc trong')

ELSE IF(@date='' OR @date IS NULL)

BEGIN

PRINT('Ngay sinh khong duoc trong')

END

ELSE IF( DATEDIFF(YEAR,@date,CAST(GETDATE() AS DATE)) < 18)

BEGIN

--PRINT('Nhan vien chua du 18 tuoi')

select cast(-2 as int) as ketqua

END

ELSE IF(@sex NOT IN('Nam','Nu')) PRINT('Gioi tinh phai la Nam/Nu')

ELSE IF(@cmt='' OR @cmt IS NULL) PRINT('CMT khong duoc trong')

ELSE IF(EXISTS(SELECT\*FROM dbo.tbl\_khachhang WHERE cmt IN(@cmt)))

BEGIN

--PRINT('CMT da ton tai')

SELECT cast(-1 as int) as ketqua

END

ELSE IF(@sdt='' OR @sdt IS NULL) PRINT('So dien thoai khong duoc trong')

ELSE IF(EXISTS(SELECT\*FROM dbo.tbl\_khachhang WHERE sdt IN(@sdt)) OR dbo.funcSDT(@sdt)<>1)

BEGIN

--PRINT('So dien thoai da ton tai hoac nhap khong dung dinh dang');

select cast(0 as int) as ketqua

END

ELSE

BEGIN

INSERT INTO dbo.tbl\_khachhang(maKH,tenKH,ngaySinh,gioiTinh,cmt,sdt)VALUES

(@maKH,@tenKH,@date,@sex,@cmt,@sdt)

select cast(1 as int) as ketqua

--PRINT('Them khach hang ' +@tenKH+ ' thanh cong!')

END

END

GO

-- Thêm khách hàng

EXEC dbo.addKhachHang @maKH = 'kh01', @tenKH = 'Bui Minh Tuan', @date = '3/2/2001', @sex = 'Nam', @cmt = '031412507344', @sdt = '0375912776'

GO

EXEC dbo.addKhachHang @maKH = 'kh02', @tenKH = 'Nguyen Quang Nam', @date = '2/1/2001', @sex = 'Nam', @cmt = '021412507344', @sdt = '0373612776'

GO

EXEC dbo.addKhachHang @maKH = 'kh03', @tenKH = 'Bui Minh Anh', @date = '10/2/2001', @sex = 'Nam', @cmt = '031412507374', @sdt = '0975912776'

GO

EXEC dbo.addKhachHang @maKH = 'kh04', @tenKH = 'Bui Thi Cam Nhung', @date = '11/12/2001', @sex = 'Nu', @cmt = '032412507444', @sdt = '0905912776'

GO

## **12. Sử dụng PROC để sửa thông tin khách hàng**

-- proc sửa thông tin khách hàng

/\*

- Các trường không được để trống

- Chứng minh thư và số điện thoại phải là duy nhất

…

- Sửa thông tin khách hàng theo mã khách hàng

\*/

CREATE PROC updateKhachHang(@maKH VARCHAR(30), @tenKH VARCHAR(50), @date DATE,

@sex VARCHAR(5), @cmt VARCHAR(12), @sdt VARCHAR(11))

AS

BEGIN

IF(@maKH='' OR @maKH IS NULL) PRINT('Ma khach hang khong duoc trong')

ELSE IF(NOT EXISTS(SELECT\*FROM dbo.tbl\_khachhang WHERE maKH IN(@maKH)))

BEGIN

--PRINT('Ma khach khong hang ton tai');

SELECT cast(-3 as int) as ketqua

END

ELSE IF(@tenKH='' OR @tenKH IS NULL) PRINT('Ten khach hang khong duoc trong')

ELSE IF(@date='' OR @date IS NULL)

BEGIN

PRINT('Ngay sinh khong duoc trong')

END

ELSE IF( DATEDIFF(YEAR,@date,CAST(GETDATE() AS DATE)) < 18)

BEGIN

--PRINT('Nhan vien chua du 18 tuoi')

select cast(-2 as int) as ketqua

END

ELSE IF(@sex NOT IN('Nam','Nu')) PRINT('Gioi tinh phai la Nam/Nu')

ELSE IF(@cmt='' OR @cmt IS NULL) PRINT('CMT khong duoc trong')

ELSE IF((SELECT COUNT(\*) FROM dbo.tbl\_khachhang WHERE maKH IN(@maKH) AND cmt IN(@cmt)) > 1)

BEGIN

--PRINT('CMT da ton tai')

SELECT cast(-1 as int) as ketqua

END

ELSE IF(@sdt='' OR @sdt IS NULL) PRINT('So dien thoai khong duoc trong')

ELSE IF((SELECT COUNT(\*) FROM dbo.tbl\_khachhang WHERE maKH IN(@maKH) AND sdt IN(@sdt)) > 1

OR dbo.funcSDT(@sdt)<>1)

BEGIN

--PRINT('So dien thoai da ton tai hoac nhap khong dung dinh dang');

select cast(0 as int) as ketqua

END

ELSE

BEGIN

UPDATE dbo.tbl\_khachhang SET tenKH=@tenKH, ngaySinh=@date, gioiTinh=@sex, cmt=@cmt, sdt=@sdt WHERE maKH IN(@maKH)

select cast(1 as int) as ketqua

--PRINT('Cap nhat du lieu thanh cong!')

END

END

GO

-- Ví dụ khi sửa thông tin khách hàng

EXEC dbo.updateKhachHang @maKH = 'kh01', @tenKH = 'Bui Minh Tuan', @date = '3/2/2001', @sex = 'Nam', @cmt = '031412507344', @sdt = '0375912776'

GO

SELECT \* FROM dbo.tbl\_khachhang;

GO

## **13. Sử dụng TRIGGER để thêm phòng**

-- Trigger thêm phòng

/\*

- Các trường không được để trống

- Mã phòng không được trùng lặp

- Khi thêm phòng -> tự động theo trạng thái phòng trống

(theo ngày)

\*/

CREATE TRIGGER addphong ON dbo.tbl\_phong

INSTEAD OF INSERT AS

BEGIN

DECLARE @maPhong VARCHAR(30);

DECLARE @tenphong VARCHAR(50);

DECLARE @maloaiphong VARCHAR(10);

DECLARE @mota VARCHAR(50);

DECLARE @songuoimax INT;

SET @maPhong = (SELECT DISTINCT Inserted.maPhong FROM Inserted);

SET @tenphong = (SELECT DISTINCT Inserted.tenPhong FROM Inserted);

SET @maloaiphong = (SELECT DISTINCT Inserted.maLoaiPhong FROM Inserted);

SET @mota = (SELECT DISTINCT Inserted.mota FROM Inserted);

SET @songuoimax = (SELECT DISTINCT Inserted.songuoitoida FROM Inserted);

IF(@maPhong='' OR @maPhong IS NULL) PRINT('Ma phong khong duoc trong')

ELSE IF(@tenphong='' OR @tenphong IS NULL) PRINT('Ten phong khong duoc trong')

ELSE IF(@maloaiphong='' OR @maloaiphong IS NULL) PRINT('Ma loai phong khong duoc trong')

ELSE IF(NOT EXISTS(SELECT\*FROM dbo.tbl\_loaiphong WHERE maLoaiPhong IN(@maloaiphong)))

BEGIN

--PRINT('Loai phong '+@maloaiphong+' khong ton tai');

SELECT cast(-2 as int) as ketqua

END

ELSE IF(@mota='' OR @mota IS NULL) PRINT('Mo ta khong duoc trong')

ELSE IF(@songuoimax IS NULL OR @songuoimax <= 0)

BEGIN

--PRINT('Dau vao so luong nguoi max khong dung');

SELECT cast(-1 as int) as ketqua

END

ELSE IF EXISTS (SELECT \* FROM dbo.tbl\_phong WHERE maPhong IN (@maPhong))

BEGIN

--PRINT(N'Phòng có mã ' + @maPhong + N' đã được thêm trước đó' + CHAR(13) + N'Thêm phòng không thành công');

SELECT cast(0 as int) as ketqua

END

ELSE

BEGIN

INSERT INTO dbo.tbl\_phong(maPhong,tenPhong,maLoaiPhong,mota,songuoitoida)VALUES

(@maPhong,@tenphong,@maloaiphong,@mota,@songuoimax);

INSERT INTO dbo.tbl\_trangthaiphong(maPhong,ngay,tinhTrang)VALUES

(@maPhong,GETDATE(),'trong');

--PRINT('Them phong ' +@tenphong+ ' thanh cong!')

--PRINT('Them trang thai trong cho phong thanh cong!')

select cast(1 as int) as ketqua

END

END

GO

-- Thêm phòng

INSERT INTO dbo.tbl\_phong(maPhong,tenPhong,maLoaiPhong,mota,songuoitoida)VALUES

('a1', '10a1','vip','phong tien nghi view dep',5);

GO

INSERT INTO dbo.tbl\_phong(maPhong,tenPhong,maLoaiPhong,mota,songuoitoida)VALUES

('a2', '10a2','vip','phong tien nghi view dep',5);

GO

INSERT INTO dbo.tbl\_phong(maPhong,tenPhong,maLoaiPhong,mota,songuoitoida)VALUES

('a3', '10a3','vip','phong tien nghi view dep',5);

GO

INSERT INTO dbo.tbl\_phong(maPhong,tenPhong,maLoaiPhong,mota,songuoitoida)VALUES

('a4', '10a4','vip','phong tien nghi view dep',5);

GO

INSERT INTO dbo.tbl\_phong(maPhong,tenPhong,maLoaiPhong,mota,songuoitoida)VALUES

('b1', '10b1','st1','phong tien nghi',3);

GO

INSERT INTO dbo.tbl\_phong(maPhong,tenPhong,maLoaiPhong,mota,songuoitoida)VALUES

('b2', '10b2','st1','phong tien nghi',3)

GO

SELECT \* FROM dbo.tbl\_phong

GO

SELECT \* FROM dbo.tbl\_trangthaiphong

GO

## **14. Sử dụng PROC để cập nhật dữ liệu**

-- proc updateData (cho chay khi load app)

/\*

- Cập nhật trạng thái phòng trống ở ngày hiện tại

(Nếu phòng chưa được đặt/hẹn phòng)

- Cập nhật trạng thái phòng với đặt phòng là đặt với các

trường hợp đến lịch hẹn phòng

\*/

CREATE PROC updateData

AS

BEGIN

UPDATE dbo.tbl\_trangthaiphong SET ngay=CAST(GETDATE() AS DATE) WHERE tinhTrang='trong' AND ngay < CAST(GETDATE() AS date);

UPDATE dbo.tbl\_trangthaiphong SET tinhTrang='dat' WHERE tinhTrang='hen' AND ngay <= CAST(GETDATE() AS date);

UPDATE dbo.tbl\_datphong SET trangthai='dat' WHERE ngayDat <= CAST(GETDATE() AS date) AND trangthai='hen';

END

GO

## **15. Sử dụng PROC để thêm tự động thêm trạng thái trống cho phòng**

## **(theo ngày nếu nó hiện tại không có ai đặt/hẹn)**

-- proc them trang thai trong cho 1 phong voi tham so phong truyen vao (neu hien tai phong do khong co ai dat)

CREATE PROC uPhongTrong (@maphong VARCHAR(30))

AS

BEGIN

IF (NOT EXISTS(SELECT maPhong FROM dbo.tbl\_datphong WHERE maPhong IN(@maphong) GROUP BY maPhong HAVING Max(ngayTra) > CAST(GETDATE() AS DATE))

AND NOT EXISTS (SELECT \* FROM dbo.tbl\_trangthaiphong WHERE maPhong IN(@maphong) AND tinhTrang='trong')

AND EXISTS(SELECT\*FROM dbo.tbl\_phong WHERE maPhong IN(@maphong)))

BEGIN

INSERT INTO dbo.tbl\_trangthaiphong(maPhong,ngay,tinhTrang)VALUES

(@maphong, CAST(GETDATE() AS DATE),'trong')

--PRINT('Them trang thai trong cho phong ' +@maphong+ ' thanh cong')

END

--ELSE

-- PRINT('Them trang thai trong khong thanh cong')

END

GO

EXEC dbo.uPhongTrong @maphong = 'a1'

GO

-- proc duyet tat ca cac phong -> them trang thai trong cho phong (neu hien tai phong do khong co ai dat) --> chay khi load app (trong proc co chua proc updateData)

CREATE PROC uPhongTrongs

@cnt INT

AS

BEGIN

EXEC dbo.updateData;

BEGIN

WHILE (@cnt <= (SELECT COUNT(\*) FROM dbo.tbl\_phong))

BEGIN

DECLARE @maPhongTemp VARCHAR(30);

SET @maPhongTemp = (SELECT b.maPhong FROM (SELECT ROW\_NUMBER() OVER (ORDER BY maPhong ASC) AS rownumber, maPhong

FROM (SELECT DISTINCT maPhong FROM dbo.tbl\_phong) AS a

) AS b WHERE b.rownumber=@cnt)

EXEC dbo.uPhongTrong @maphong = @maPhongTemp;

-- PRINT('cnt' + CAST(@cnt AS VARCHAR(100)))

SET @cnt = @cnt + 1;

END;

END

END

GO

EXEC dbo.uPhongTrongs @cnt = 1

GO

## **16. Sử dụng PROC để đặt phòng**

-- proc đặt phòng (them vao bang datphong)

/\*

- Khách hàng có thể đặt/hẹn phòng

- Thời gian đặt phòng phải là hiện tại

- Thời gian đặt/hẹn phòng phải > thời gian trả phòng

gần nhất.

- Với phòng trống thì khách hàng đặt/hẹn phòng bình

thường

- Với phòng đang được hẹn 🡪 khách hàng sẽ đặt phòng nếu thời gian đặt là hiện tại và thời trả phòng phải < thời gian hẹn phòng hiện tại của phòng đó

- Còn với phòng đang được đặt(được sử dụng) thì khách khách hàng sẽ hẹn được phòng nếu thời gian đặt phòng > thời gian trả phòng hiện tại.

\*/

CREATE PROC datphong(@makh VARCHAR(30), @manv VARCHAR(30), @maphong VARCHAR(30), @ngaydat DATE, @ngaytra DATE)

AS

BEGIN

EXEC dbo.updateData;

EXEC dbo.uPhongTrong @maphong;

IF (@makh='' OR @makh IS NULL) PRINT('Ma khach hang khong duoc de trong');

ELSE IF (NOT EXISTS(SELECT \* FROM dbo.tbl\_khachhang WHERE maKH IN(@makh)))

BEGIN

--PRINT('Hay nhap dung ma khach hang')

select cast(-6 as int) as ketqua

END

ELSE IF (@manv='' OR @manv IS NULL) PRINT('Ma nhan vien khong duoc de trong');

ELSE IF (NOT EXISTS(SELECT \* FROM dbo.tbl\_nhanvien WHERE maNV IN(@manv)))

BEGIN

--PRINT('Hay nhap dung ma nhan vien')

select cast(-5 as int) as ketqua

END

ELSE IF (@maphong='' OR @maphong IS NULL) PRINT('Ma phong khong duoc de trong');

ELSE IF (NOT EXISTS(SELECT \* FROM dbo.tbl\_phong WHERE maPhong IN(@maphong)))

BEGIN

--PRINT('Hay nhap dung ma phong')

select cast(-4 as int) as ketqua

END

ELSE IF (@ngaydat='' OR @ngaydat IS NULL) PRINT('Ngay dat khong duoc de trong');

ELSE IF (@ngaytra='' OR @ngaytra IS NULL) PRINT('Ngay tra khong duoc de trong');

ELSE IF(@ngaydat < CAST(GETDATE() AS DATE))

BEGIN

--PRINT('Ngay dat phai la hien tai/tuong lai');

select cast(-3 as int) as ketqua

END

ELSE IF (@ngaytra <= @ngaydat)

BEGIN

--PRINT('Ngay tra phai >= ngaydat + 1');

select cast(-2 as int) as ketqua

-- IF (@ngaydat < CAST(GETDATE() AS date))

--BEGIN

-- PRINT('Thoi gian dat phai la hien tai/tuong lai');

--END

END;

-------------------------------------------------------

ELSE IF (EXISTS (SELECT \* FROM dbo.tbl\_trangthaiphong WHERE tinhTrang='trong' AND maPhong IN(@maphong)))

BEGIN

IF (@ngaydat = CAST(GETDATE() AS date))

BEGIN

INSERT INTO tbl\_datphong(maKH,maNV,maPhong,ngayDat,ngayTra,trangthai)VALUES

(@makh,@manv,@maphong,@ngaydat,@ngaytra,'dat');

UPDATE dbo.tbl\_trangthaiphong SET tinhTrang='dat' WHERE maPhong IN (@maphong) AND ngay=@ngaydat AND tinhTrang='trong'

select cast(1 as int) as ketqua

--PRINT('Dat phong '+@maphong+' thanh cong')

END

-------------------------------------------------------

ELSE IF (@ngaydat > CAST(GETDATE() AS date))

BEGIN

INSERT INTO tbl\_datphong(maKH,maNV,maPhong,ngayDat,ngayTra,trangthai)VALUES

(@makh,@manv,@maphong,@ngaydat,@ngaytra,'hen');

UPDATE dbo.tbl\_trangthaiphong SET ngay=@ngaydat,tinhTrang='hen' WHERE maPhong IN (@maphong) AND tinhTrang='trong'

select cast(2 as int) as ketqua

--PRINT('Hen dat phong '+@maphong+' thanh cong')

END

END

-------------------------------------------------------

ELSE IF (EXISTS (SELECT \* FROM dbo.tbl\_trangthaiphong WHERE tinhTrang IN('dat','hen') AND maPhong IN(@maphong)))

BEGIN

IF (@ngaydat = CAST(GETDATE() AS DATE))

BEGIN

--PRINT('Dat trong khoang time giua')

DECLARE @timedatmax DATE;

DECLARE @timehenmin DATE;

SET @timehenmin=(SELECT TOP 1 ngayDat FROM dbo.tbl\_datphong WHERE maPhong IN(@maphong) AND trangthai='hen' ORDER BY ngayTra ASC);

SET @timedatmax=(SELECT TOP 1 ngayTra FROM dbo.tbl\_datphong WHERE maPhong IN(@maphong) AND trangthai='dat' ORDER BY ngayTra DESC);

if(@timedatmax is null)

begin

if(@ngaytra < @timehenmin)

begin

INSERT INTO tbl\_datphong(maKH,maNV,maPhong,ngayDat,ngayTra,trangthai)VALUES

(@makh,@manv,@maphong,@ngaydat,@ngaytra,'dat');

INSERT INTO dbo.tbl\_trangthaiphong(maPhong,ngay,tinhTrang)VALUES(@maphong,@ngaydat,'dat');

select cast(1 as int) as ketqua

--PRINT('Dat phong '+@maphong+' thanh cong')

end

end

else IF ((@ngaydat > @timedatmax) AND (@ngaytra < @timehenmin))

BEGIN

INSERT INTO tbl\_datphong(maKH,maNV,maPhong,ngayDat,ngayTra,trangthai)VALUES

(@makh,@manv,@maphong,@ngaydat,@ngaytra,'dat');

INSERT INTO dbo.tbl\_trangthaiphong(maPhong,ngay,tinhTrang)VALUES(@maphong,@ngaydat,'dat');

select cast(1 as int) as ketqua

--PRINT('Dat phong '+@maphong+' thanh cong')

END

ELSE

begin

select cast(-1 as int) as ketqua

--PRINT('Phong '+@maphong+' hien tai chua dat duoc')

end

END

-------------------------------------------------------

ELSE IF (@ngaydat > CAST(GETDATE() AS DATE))

BEGIN

DECLARE @timehenmax DATE; -- dat trong tuong lai

SET @timehenmax=(SELECT TOP 1 ngayTra FROM dbo.tbl\_datphong WHERE maPhong IN(@maphong) ORDER BY ngayTra DESC);

IF ((select datediff(day,@timehenmax, @ngaydat) as timeDiff) < 1) PRINT('Dat phong '+@maphong+' khong thanh cong');

--------------------------------------------------------

ELSE

BEGIN

INSERT INTO tbl\_datphong(maKH,maNV,maPhong,ngayDat,ngayTra,trangthai)VALUES

(@makh,@manv,@maphong,@ngaydat,@ngaytra,'hen');

INSERT INTO dbo.tbl\_trangthaiphong(maPhong,ngay,tinhTrang)VALUES(@maphong,@ngaydat,'hen');

select cast(2 as int) as ketqua

--PRINT('Hen dat phong '+@maphong+' thanh cong')

END

END

END

-------------------------------------------------------

ELSE

BEGIN

select cast(0 as int) as ketqua

--PRINT('Dat phong '+@maphong+' khong thanh cong')

END

END

GO

-- Ví dụ về đặt/hẹn phòng

EXEC dbo.datphong @makh = 'kh01',@manv = 'bibo',@maphong = 'a1',@ngaydat = '2023-1-2',@ngaytra = '2023-10-1'

GO

EXEC dbo.datphong @makh = 'kh01',@manv = 'ts',@maphong = 'a1',@ngaydat = '2023-10-2',@ngaytra = '2023-10-20'

GO

EXEC dbo.datphong @makh = 'kh01',@manv = 'bth',@maphong = 'a3',@ngaydat = '2023-1-2',@ngaytra = '2023-1-30'

GO

EXEC dbo.datphong @makh = 'kh02',@manv = 'bibo',@maphong = 'a4',@ngaydat = '2023-2-2',@ngaytra = '2023-2-10'

GO

EXEC dbo.datphong @makh = 'kh02',@manv = 'bibo',@maphong = 'a4',@ngaydat = '2023-2-11',@ngaytra = '2023-2-20'

GO

EXEC dbo.datphong @makh = 'kh02',@manv = 'bibo',@maphong = 'b1',@ngaydat = '2023-1-2',@ngaytra = '2023-1-20'

GO

EXEC dbo.datphong @makh = 'kh02',@manv = 'bibo',@maphong = 'b1',@ngaydat = '2023-1-21',@ngaytra = '2023-2-10'

GO

EXEC dbo.datphong @makh = 'kh02',@manv = 'bibo',@maphong = 'b1',@ngaydat = '2023-2-11',@ngaytra = '2023-2-20'

GO

SELECT \* FROM dbo.tbl\_datphong

GO

SELECT \* FROM dbo.tbl\_trangthaiphong

GO

## **17. Sử dụng PROC để thêm đồ ăn**

-- proc thêm đồ ăn

/\*

* Các trường không được để trống
* Giá (gia) phải > 0

\*/

CREATE PROC addDoAn(@maDoAn VARCHAR(30), @tenDA VARCHAR(100), @gia INT)

AS

BEGIN

IF(@maDoAn='' OR @maDoAn IS NULL) PRINT('Ma do an khong duoc de trong')

ELSE IF(EXISTS (SELECT \* FROM dbo.tbl\_doAn WHERE maDA IN(@maDoAn)))

BEGIN

--PRINT('Ma do an ton tai -> Chen do an that bai')

select cast(-1 as int) as ketqua

END

ELSE IF(@tenDA='' OR @tenDA IS NULL) PRINT('Ten do an khong duoc de trong')

ELSE IF(@gia IS NULL OR @gia <= 0)

BEGIN

--PRINT('Gia mon an phai duong')

select cast(0 as int) as ketqua

END

ELSE

BEGIN

INSERT INTO dbo.tbl\_doAn(maDA,tenDA,gia)VALUES

(@maDoAn,@tenDA,@gia);

--PRINT('Them mon an '+@tenDA+' thanh cong!');

select cast(1 as int) as ketqua

END;

END;

GO

-- Thêm đồ ăn

EXEC dbo.addDoAn @maDoAn = 'gahap', @tenDA = 'ga hap', @gia = 150

GO

EXEC dbo.addDoAn @maDoAn = 'chimquay', @tenDA = 'chim cut quay', @gia = 100

GO

EXEC dbo.addDoAn @maDoAn = 'tomhap', @tenDA = 'tom cang xanh hap', @gia = 200

GO

EXEC dbo.addDoAn @maDoAn = 'cachien', @tenDA = 'ca chep chien gion', @gia = 80

GO

EXEC dbo.addDoAn @maDoAn = 'boxao', @tenDA = 'thit bo xao toi', @gia = 150

GO

## **18. Sử dụng PROC để sửa thông tin đồ ăn**

-- proc sửa thông tin đồ ăn

/\*

* Các trường không được để trống
* Giá (gia) phải > 0
* Sửa thông tin theo mã đồ ăn

\*/

CREATE PROC updateDoAn(@maDoAn VARCHAR(30), @tenDA VARCHAR(100), @gia INT)

AS

BEGIN

IF(@maDoAn='' OR @maDoAn IS NULL) PRINT('Ma do an khong duoc de trong')

ELSE IF(NOT EXISTS (SELECT \* FROM dbo.tbl\_doAn WHERE maDA IN(@maDoAn)))

BEGIN

--PRINT('Ma do an khong ton tai')

select cast(-1 as int) as ketqua

END

ELSE IF(@tenDA='' OR @tenDA IS NULL) PRINT('Ten do an khong duoc de trong')

ELSE IF(@gia IS NULL OR @gia <= 0)

BEGIN

--PRINT('Gia mon an phai duong')

select cast(0 as int) as ketqua

END

ELSE

BEGIN

UPDATE dbo.tbl\_doAn SET tenDA=@tenDA, gia=@gia WHERE maDA IN (@maDoAn)

--PRINT('Cap nhat du lieu thanh cong!');

select cast(1 as int) as ketqua

END;

END;

GO

SELECT \* FROM dbo.tbl\_doAn

GO

## **19. Sử dụng PROC để thêm đồ uống**

-- proc thêm đồ uống

/\*

* Các trường không được để trống
* Giá (gia) phải > 0

\*/

CREATE PROC addDoUong(@maDoUong VARCHAR(30), @tenDoUong VARCHAR(100), @gia INT)

AS

BEGIN

IF(@maDoUong='' OR @maDoUong IS NULL) PRINT('Ma do uong khong duoc de trong')

ELSE IF(EXISTS (SELECT \* FROM dbo.tbl\_doUong WHERE maDU IN(@maDoUong)))

BEGIN

--PRINT('Ma do uong ton tai -> Chen do uong that bai')

select cast(-1 as int) as ketqua

END

ELSE IF(@tenDoUong='' OR @tenDoUong IS NULL) PRINT('Ten do uong khong de trong')

ELSE IF(@gia IS NULL OR @gia <= 0)

BEGIN

--PRINT('Gia do uong phai duong')

SELECT cast(0 as int) as ketqua

END

ELSE

BEGIN

INSERT INTO dbo.tbl\_doUong(maDU,tenDU,gia)VALUES

(@maDoUong,@tenDoUong,@gia);

--PRINT('Them do uong '+@tenDoUong+' thanh cong!')

select cast(1 as int) as ketqua

END;

END;

-- Thêm đồ uống

GO

EXEC dbo.addDoUong @maDoUong = 'cep', @tenDoUong = 'nuoc chanh ep', @gia = 10

GO

EXEC dbo.addDoUong @maDoUong = 'camv', @tenDoUong = 'nuoc cam vat', @gia = 20

GO

EXEC dbo.addDoUong @maDoUong = 'stbo', @tenDoUong = 'sinh to bo', @gia = 50

GO

EXEC dbo.addDoUong @maDoUong = 'nl', @tenDoUong = 'nuoc loc', @gia = 5

GO

EXEC dbo.addDoUong @maDoUong = 'cafed', @tenDoUong = 'cafe den', @gia = 30

GO

EXEC dbo.addDoUong @maDoUong = 'cafes', @tenDoUong = 'cafe sua', @gia = 10

GO

## **20. Sử dụng PROC để sửa thông tin đồ uống**

-- proc sửa thông tin đồ uống

/\*

* Các trường không được để trống
* Giá (gia) phải > 0
* Sửa thông tin đồ uống theo mã đồ uống

\*/

CREATE PROC updateDoUong(@maDoUong VARCHAR(30), @tenDoUong VARCHAR(100), @gia INT)

AS

BEGIN

IF(@maDoUong='' OR @maDoUong IS NULL) PRINT('Ma do uong khong duoc de trong')

ELSE IF(NOT EXISTS (SELECT \* FROM dbo.tbl\_doUong WHERE maDU IN(@maDoUong)))

BEGIN

--PRINT('Ma do uong khong ton tai')

select cast(-1 as int) as ketqua

END

ELSE IF(@tenDoUong='' OR @tenDoUong IS NULL) PRINT('Ten do uong khong de trong')

ELSE IF(@gia IS NULL OR @gia <= 0)

BEGIN

--PRINT('Gia do uong phai duong')

SELECT cast(0 as int) as ketqua

END

ELSE

BEGIN

UPDATE dbo.tbl\_doUong SET tenDU=@tenDoUong, gia=@gia WHERE maDU IN(@maDoUong);

--PRINT('Cap nhat du lieu thanh cong!')

select cast(1 as int) as ketqua

END;

END;

GO

SELECT \* FROM dbo.tbl\_doUong

GO

## **21. Sử dụng PROC để đặt đồ ăn**

-- proc đặt đồ ăn

/\*

* Các trường không được để trống
* Ngày đặt(ngay) phải là ngày hiện tại

(Vì đã để mặc định nên không cần thêm)

* Số lượng (soluong) phải > 0
* Vì đặt đồ ăn theo ngày nên nếu cùng 1 ngày cùng 1 phòng và cùng một món ăn thì chỉ cập nhật lại trường số lượng (soluong)

\*/

CREATE PROC datDoAn(@maPhong VARCHAR(30), @maDoAn VARCHAR(30), @soluong INT)

AS

BEGIN

DECLARE @ngay DATE = CAST(GETDATE() AS DATE)

IF(@maPhong='' OR @maPhong IS NULL) PRINT('Ma phong khong duoc trong')

ELSE IF(@maDoAn='' OR @maDoAn IS NULL) PRINT('Ma do an khong duoc trong')

ELSE IF(NOT EXISTS(SELECT\*FROM dbo.tbl\_doAn WHERE maDA IN(@maDoAn)))

BEGIN

--PRINT('Do an nhap vao khong ton tai -> dat do an that bai')

SELECT cast(-2 as int) as ketqua

END

ELSE IF(@soluong IS NULL OR @soluong <= 0)

BEGIN

--PRINT('So luong phai duong')

SELECT cast(-1 as int) as ketqua

END

ELSE IF(NOT EXISTS(SELECT\*FROM dbo.tbl\_datphong WHERE maPhong IN(@maPhong) AND ngayTra > CAST(GETDATE() AS date) AND trangthai='dat'))

BEGIN

--PRINT('Ma phong khong hop le(phong trong/khong ton tai,...)')

select cast(0 as int) as ketqua

END

ELSE IF(EXISTS(SELECT\*FROM dbo.tbl\_dangkyDoAn WHERE ngay=@ngay AND maPhong=@maPhong AND maDA=@maDoAn))

BEGIN

UPDATE dbo.tbl\_dangkyDoAn SET soLuong=soLuong+@soluong WHERE ngay=@ngay AND maPhong=@maPhong AND maDA=@maDoAn

--PRINT('Dat do an thanh cong!')

select cast(1 as int) as ketqua

END

ELSE

BEGIN

INSERT INTO dbo.tbl\_dangkyDoAn(ngay,maPhong,maDA,soLuong)VALUES

(@ngay,@maPhong,@maDoAn,@soluong);

--PRINT('Dat do an thanh cong!')

select cast(1 as int) as ketqua

END

END

GO

-- Ví dụ đặt đồ ăn

EXEC dbo.datDoAn @maPhong = 'a1', @maDoAn = 'tomhap', @soluong = 3

GO

EXEC dbo.datDoAn @maPhong = 'a1', @maDoAn = 'boxao', @soluong = 3

GO

EXEC dbo.datDoAn @maPhong = 'a1', @maDoAn = 'cachien', @soluong = 3

GO

EXEC dbo.datDoAn @maPhong = 'a1', @maDoAn = 'gahap', @soluong = 3

GO

EXEC dbo.datDoAn @maPhong = 'a3', @maDoAn = 'tomhap', @soluong = 3

GO

EXEC dbo.datDoAn @maPhong = 'a3', @maDoAn = 'boxao', @soluong = 3

GO

EXEC dbo.datDoAn @maPhong = 'a3', @maDoAn = 'cachien', @soluong = 3

GO

EXEC dbo.datDoAn @maPhong = 'a3', @maDoAn = 'gahap', @soluong = 3

GO

EXEC dbo.datDoAn @maPhong = 'b1', @maDoAn = 'tomhap', @soluong = 3

GO

EXEC dbo.datDoAn @maPhong = 'b1', @maDoAn = 'boxao', @soluong = 3

GO

EXEC dbo.datDoAn @maPhong = 'b1', @maDoAn = 'cachien', @soluong = 3

GO

EXEC dbo.datDoAn @maPhong = 'b1', @maDoAn = 'gahap', @soluong = 3

GO

SELECT \* FROM dbo.tbl\_dangkyDoAn

GO

## **22. Sử dụng PROC để đặt đồ uống**

-- proc đặt đồ uống

/\*

* Các trường không được để trống
* Ngày đặt(ngay) phải là ngày hiện tại

(Vì đã để mặc định nên không cần thêm)

* Số lượng (soluong) phải > 0
* Vì đặt đồ uống theo ngày nên nếu cùng 1 ngày cùng 1 phòng và cùng một đồ uống thì chỉ cập nhật lại trường số lượng (soluong)

\*/

CREATE PROC datDoUong(@maPhong VARCHAR(30), @maDoUong VARCHAR(30), @soluong INT)

AS

BEGIN

DECLARE @ngay DATE = CAST(GETDATE() AS DATE)

IF(@maPhong='' OR @maPhong IS NULL) PRINT('Ma phong khong duoc trong')

ELSE IF(@maDoUong='' OR @maDoUong IS NULL) PRINT('Ma do uong khong duoc trong')

ELSE IF(NOT EXISTS(SELECT\*FROM dbo.tbl\_doUong WHERE maDU IN(@maDoUong)))

BEGIN

--PRINT('Do uong nhap vao khong ton tai -> dat do uong that bai')

SELECT cast(-2 as int) as ketqua

END

ELSE IF(@soluong IS NULL OR @soluong <= 0)

BEGIN

--PRINT('So luong phai duong')

select cast(-1 as int) as ketqua

END

ELSE IF(NOT EXISTS(SELECT\*FROM dbo.tbl\_datphong WHERE maPhong IN(@maPhong) AND ngayTra > CAST(GETDATE() AS date) AND trangthai='dat'))

BEGIN

--PRINT('Ma phong khong hop le(phong trong/khong ton tai,...)')

select cast(0 as int) as ketqua

END

ELSE IF(EXISTS(SELECT\*FROM dbo.tbl\_dangkyDoUong WHERE ngay=@ngay AND maPhong=@maPhong AND maDU=@maDoUong))

BEGIN

UPDATE dbo.tbl\_dangkyDoUong SET soLuong=soLuong+@soluong WHERE ngay=@ngay AND maPhong=@maPhong AND maDU=@maDoUong

--PRINT('Dat do uong thanh cong')

select cast(1 as int) as ketqua

END

ELSE

BEGIN

INSERT INTO dbo.tbl\_dangkyDoUong(ngay,maPhong,maDU,soLuong)VALUES

(@ngay,@maPhong,@maDoUong,@soluong);

--PRINT('Dat do uong thanh cong')

select cast(1 as int) as ketqua

END

END

GO

-- Ví dụ về đặt đồ uống

EXEC dbo.datDoUong @maPhong = 'a1', @maDoUong = 'cep', @soluong = 3

GO

EXEC dbo.datDoUong @maPhong = 'a1', @maDoUong = 'nl', @soluong = 3

GO

EXEC dbo.datDoUong @maPhong = 'a1', @maDoUong = 'stbo', @soluong = 3

GO

EXEC dbo.datDoUong @maPhong = 'a1', @maDoUong = 'camv', @soluong = 3

GO

EXEC dbo.datDoUong @maPhong = 'a3', @maDoUong = 'cep', @soluong = 3

GO

EXEC dbo.datDoUong @maPhong = 'a3', @maDoUong = 'nl', @soluong = 3

GO

EXEC dbo.datDoUong @maPhong = 'a3', @maDoUong = 'stbo', @soluong = 3

GO

EXEC dbo.datDoUong @maPhong = 'a3', @maDoUong = 'camv', @soluong = 3

GO

EXEC dbo.datDoUong @maPhong = 'b1', @maDoUong = 'cep', @soluong = 3

GO

EXEC dbo.datDoUong @maPhong = 'b1', @maDoUong = 'nl', @soluong = 3

GO

EXEC dbo.datDoUong @maPhong = 'b1', @maDoUong = 'stbo', @soluong = 3

GO

EXEC dbo.datDoUong @maPhong = 'b1', @maDoUong = 'camv', @soluong = 3

GO

SELECT \* FROM dbo.tbl\_dangkyDoUong

GO

## **23. Sử dụng PROC để thanh toán tiền**

-- proc thanh toán tiền (dat thanhtoan=1)

--> thanh toán cho 1 người / 1 phòng

CREATE PROC thanhToan(@maKH VARCHAR(30), @maPhong VARCHAR(30), @ngayDat DATE)

AS

BEGIN

IF(NOT EXISTS(SELECT\*FROM dbo.tbl\_datphong WHERE maKH IN(@maKH) AND maPhong IN(@maPhong) AND ngayDat=@ngayDat AND CAST(GETDATE() AS DATE) >= ngayTra))

BEGIN

SELECT cast(0 as int) as ketqua

END

ELSE

BEGIN

UPDATE dbo.tbl\_datphong SET thanhtoan=1 --, ngayTra=CAST(GETDATE() AS date)

WHERE maKH=@maKH AND maPhong=@maPhong AND ngayDat=@ngayDat;

SELECT cast(1 as int) as ketqua

END

END

GO

-- Ví dụ về thanh toán

EXEC dbo.thanhToan @maKH = 'kh01',@maPhong = 'a1', @ngayDat = '2023-01-02'

GO

SELECT \* FROM dbo.tbl\_datphong

GO

## **24. Sử dụng PROC để in hóa đơn đồ ăn theo phòng**

-- proc in ra hóa đơn đồ ăn theo phòng

CREATE PROC hoaDonDoAn(@maPhong VARCHAR(30), @ngayDat DATE, @ngayTra DATE)

AS

BEGIN

SELECT a.maPhong, @ngayDat AS [Ngay Dat Phong], @ngayTra AS [Ngay Tra Phong] , SUM(b.gia\*a.soLuong) AS [Gia Do An] FROM dbo.tbl\_dangkyDoAn a JOIN dbo.tbl\_doAn b ON b.maDA = a.maDA

AND a.maPhong IN(@maPhong) AND a.ngay BETWEEN @ngayDat AND @ngayTra GROUP BY a.maPhong

END

GO

-- Ví dụ

EXEC dbo.hoaDonDoAn @maPhong = 'a1', @ngayDat = '2023-01-02', @ngayTra = '2023-10-01'

GO

## **25. Sử dụng PROC để in hóa đơn đồ uống theo phòng**

-- proc in ra hóa đơn đồ uống theo phòng

CREATE PROC hoaDonDoUong(@maPhong VARCHAR(30), @ngayDat DATE, @ngayTra DATE)

AS

BEGIN

SELECT a.maPhong, @ngayDat AS [Ngay Dat Phong], @ngayTra AS [Ngay Tra Phong] , SUM(b.gia\*a.soLuong) AS [Gia Do Uong] FROM dbo.tbl\_dangkyDoUong a JOIN dbo.tbl\_doUong b ON b.maDU = a.maDU

AND a.maPhong IN(@maPhong) AND a.ngay BETWEEN @ngayDat AND @ngayTra GROUP BY a.maPhong

END

GO

-- Ví dụ

EXEC dbo.hoaDonDoUong @maPhong = 'a1', @ngayDat = '2023-01-02', @ngayTra = '2023-10-01'

GO

## **26. Sử dụng PROC để in hóa đơn tiền đặt phòng**

/\*

proc in ra hóa đơn tiền đặt phòng

(Chưa tính tiền đồ ăn, đồ uống(nếu có))

\*/

CREATE PROC hoaDonPhong(@maPhong VARCHAR(30), @ngayDat DATE, @ngayTra DATE, @thanhtoan bit)

AS

BEGIN

SELECT a.maPhong, @ngayDat AS [Ngay Dat Phong], @ngayTra AS [Ngay Tra Phong], c.gia\*datediff(day,@ngayDat, @ngayTra) AS [Gia Phong] FROM dbo.tbl\_datphong a JOIN dbo.tbl\_phong b ON b.maPhong = a.maPhong JOIN dbo.tbl\_loaiphong c ON c.maLoaiPhong = b.maLoaiPhong

AND a.maPhong IN(@maPhong) AND a.ngayDat=@ngayDat AND a.ngayTra=@ngayTra AND a.trangthai='dat' AND a.thanhtoan=@thanhtoan

END

GO

-- SELECT DATEDIFF(DAY,'2023-01-02','2023-10-01')

-- Ví dụ

EXEC dbo.hoaDonPhong @maPhong = 'a1', @ngayDat = '2023-01-02', @ngayTra = '2023-10-01' , @thanhtoan=0

GO

## **27. Sử dụng PROC để in hóa đơn theo phòng gồm tổng chi phí (Phòng, ăn, uống)**

-- Thanh toan = 0 la chưa thanh toán;

-- Thanh toan = 1 -> thanh toan roi -> gia=0

/\*

proc in ra hóa đơn theo phòng bao gồm tổng chi phí

- Tiền đặt phòng

- Tiền đồ ăn, đồ uống (nếu có)

\*/

CREATE PROC hoaDonTheoPhong(@maPhong VARCHAR(30), @ngayDat DATE, @ngayTra DATE)

AS

BEGIN

IF(NOT EXISTS(SELECT\*FROM dbo.tbl\_datphong WHERE maPhong IN(@maPhong) AND ngayDat=@ngayDat AND @ngayTra=@ngayTra AND trangthai='dat' AND thanhtoan=0))

BEGIN

SELECT 0 AS [Tong Hoa Don]

END

ELSE

BEGIN

DECLARE @tienPhong MONEY, @tienDoAn MONEY, @tienDoUong MONEY;

SET @tienPhong=(SELECT c.gia\*datediff(day,a.ngayDat, a.ngayTra) AS [Gia Phong] FROM dbo.tbl\_datphong a JOIN dbo.tbl\_phong b ON b.maPhong = a.maPhong JOIN dbo.tbl\_loaiphong c ON c.maLoaiPhong = b.maLoaiPhong

AND a.maPhong IN(@maPhong) AND (a.ngayDat=@ngayDat AND a.ngayTra=@ngayTra) AND a.trangthai='dat' AND a.thanhtoan=0);

----------------------------------------------------------------------------------------------------------------------------------

IF(EXISTS(SELECT\*FROM dbo.tbl\_dangkyDoAn WHERE maPhong IN(@maPhong) AND (ngay BETWEEN @ngayDat AND @ngayTra)))

BEGIN

SET @tienDoAn=(SELECT SUM(b.gia\*a.soLuong) AS [Gia Do An] FROM dbo.tbl\_dangkyDoAn a JOIN dbo.tbl\_doAn b ON b.maDA = a.maDA

AND a.maPhong IN(@maPhong) AND (a.ngay BETWEEN @ngayDat AND @ngayTra) GROUP BY a.maPhong);

END

----------------------------------------------------------------------------------------------------------------------------------

ELSE

BEGIN

SET @tienDoAn=0;

END

----------------------------------------------------------------------------------------------------------------------------------

IF(EXISTS(SELECT\*FROM dbo.tbl\_dangkyDoUong WHERE maPhong IN(@maPhong) AND (ngay BETWEEN @ngayDat AND @ngayTra)))

BEGIN

SET @tienDoUong=(SELECT SUM(b.gia\*a.soLuong) AS [Gia Do Uong] FROM dbo.tbl\_dangkyDoUong a JOIN dbo.tbl\_doUong b ON b.maDU = a.maDU

AND a.maPhong=@maPhong AND a.ngay BETWEEN @ngayDat AND @ngayTra GROUP BY a.maPhong);

END

----------------------------------------------------------------------------------------------------------------------------------

ELSE

BEGIN

SET @tienDoUong=0;

END

-- Hien thi

--SELECT c.maKH, c.tenKH, b.tenPhong, a.ngayDat AS [Ngay Dat Phong], a.ngayTra AS [Ngay Tra Phong], (@tienPhong+@tienDoAn+@tienDoUong) AS [Tong Hoa Don] FROM dbo.tbl\_datphong a

-- JOIN dbo.tbl\_phong b ON b.maPhong = a.maPhong

-- JOIN dbo.tbl\_khachhang c ON c.maKH = a.maKH AND a.ngayDat=@ngayDat AND a.ngayTra=@ngayTra

RETURN (@tienPhong+@tienDoAn+@tienDoUong);

END

END

GO

-- Ví dụ

EXEC dbo.hoaDonTheoPhong @maPhong = 'a1', @ngayDat = '2023-01-02', @ngayTra = '2023-10-01'

GO

EXEC dbo.hoaDonTheoPhong @maPhong = 'a3', @ngayDat = '2023-01-02', @ngayTra = '2023-01-30'

GO

EXEC dbo.hoaDonTheoPhong @maPhong = 'b1', @ngayDat = '2023-01-02', @ngayTra = '2023-01-20'

GO

## **28. Sử dụng PROC để in hóa đơn theo khách hàng**

/\*

proc in ra hóa đơn theo khách hàng(mã khách hàng)

nếu khách hàng chưa thanh toán

🡪 Bao gồm tiền đặt phòng + đồ ăn + đồ uống

\*/

CREATE PROC hoaDonTheoKhachHang(@maKH VARCHAR(30))

AS

BEGIN

IF(EXISTS(SELECT\*FROM dbo.tbl\_datphong WHERE maKH IN(@maKH)))

BEGIN

DECLARE @cnt INT;

SET @cnt=1;

DECLARE @hoaDon MONEY; SET @hoaDon=0;

WHILE(@cnt <= (SELECT COUNT(\*) FROM dbo.tbl\_datphong WHERE maKH IN(@maKH) AND trangthai='dat' AND thanhtoan=0))

BEGIN

DECLARE @maPhong VARCHAR(30);

DECLARE @ngayDat DATE, @ngayTra DATE; DECLARE @temp MONEY;

SET @maPhong=(SELECT b.maPhong FROM(SELECT ROW\_NUMBER() OVER (ORDER BY a.maPhong ASC) AS num, maPhong

FROM (SELECT maPhong FROM dbo.tbl\_datphong WHERE maKH IN(@maKH) AND trangthai='dat' AND thanhtoan=0) AS a)

AS b WHERE b.num=@cnt)

-------------------------------------------------------------------------------------------------------

SET @ngayDat=(SELECT b.ngayDat FROM(SELECT ROW\_NUMBER() OVER (ORDER BY a.maPhong ASC) AS num, a.ngayDat

FROM (SELECT \* FROM dbo.tbl\_datphong WHERE maKH IN(@maKH) AND trangthai='dat' AND thanhtoan=0) AS a)

AS b WHERE b.num=@cnt)

-------------------------------------------------------------------------------------------------------

SET @ngayTra=(SELECT b.ngayTra FROM(SELECT ROW\_NUMBER() OVER (ORDER BY a.maPhong ASC) AS num, a.ngayTra

FROM (SELECT \* FROM dbo.tbl\_datphong WHERE maKH IN(@maKH) AND trangthai='dat' AND thanhtoan=0) AS a)

AS b WHERE b.num=@cnt)

-------------------------------------------------------------------------------------------------------

EXEC @temp=dbo.hoaDonTheoPhong @maPhong = @maPhong, @ngayDat = @ngayDat, @ngayTra = @ngayTra

SET @hoaDon = @hoaDon + @temp;

SET @cnt = @cnt + 1;

END

SELECT maKH, tenKH, @hoaDon AS [Tong Hoa Don] FROM dbo.tbl\_khachhang WHERE maKH IN(@maKH)

END

END

GO

-- Ví dụ

EXEC hoaDonTheoKhachHang @maKH='kh01'

GO

EXEC hoaDonTheoKhachHang @maKH='kh02'

GO

## **29. Sử dụng PROC để tính doanh thu của khách sạng trong tháng/năm**

-- Tính doanh thu của khách sạn trong tháng=?, năm=?

CREATE PROC doanhThu(@ngayBD DATE, @ngayKT DATE)

AS

BEGIN

DECLARE @tienPhong MONEY, @tienDoAn MONEY, @tienDoUong MONEY;

SET @tienPhong=(SELECT SUM(c.gia\*datediff(day,a.ngayDat, a.ngayTra)) AS [Gia Phong] FROM dbo.tbl\_datphong a JOIN dbo.tbl\_phong b ON b.maPhong = a.maPhong JOIN dbo.tbl\_loaiphong c ON c.maLoaiPhong = b.maLoaiPhong

AND a.ngayDat>=@ngayBD AND a.ngayTra<=@ngayKT AND a.trangthai='dat');

SET @tienDoAn=(SELECT SUM(b.gia\*a.soLuong) AS [Gia Do An] FROM dbo.tbl\_dangkyDoAn a JOIN dbo.tbl\_doAn b ON b.maDA = a.maDA

AND a.ngay BETWEEN @ngayBD AND @ngayKT);

SET @tienDoUong=(SELECT SUM(b.gia\*a.soLuong) AS [Gia Do Uong] FROM dbo.tbl\_dangkyDoUong a JOIN dbo.tbl\_doUong b ON b.maDU = a.maDU

AND a.ngay BETWEEN @ngayBD AND @ngayKT);

IF(@tienPhong IS NULL) SET @tienPhong=0;

IF(@tienDoAn IS NULL) SET @tienDoAn=0;

IF(@tienDoUong IS NULL) SET @tienDoUong=0;

-- Hien thi

SELECT @ngayBD AS tu, @ngayKT AS den, (@tienPhong+@tienDoAn+@tienDoUong) AS doanhThu

END

GO

-- Ví dụ

EXEC dbo.doanhThu @ngayBD = '2023-01-02', @ngayKT = '2023-10-01'

GO

EXEC dbo.doanhThu @ngayBD = '2023-01-02', @ngayKT = '2023-01-02'

GO

---------------------- PHẦN BÁO CÁO THỐNG KÊ --------------------

## **30. Sử dụng PROC để thống kê top3 khách hàng đặt/hẹn phòng nhiều nhất**

-- proc thống kê top 3 khách hàng đặt/hẹn phòng nhiều nhất

CREATE PROC khachHangDatPhongMax

AS

BEGIN

SELECT TOP 3 a.maKH, b.tenKH, b.sdt, COUNT(a.maKH) AS dem

FROM tbl\_datphong a JOIN tbl\_khachhang b ON b.maKH = a.maKH GROUP BY a.maKH, b.tenKH, b.sdt ORDER BY COUNT(a.maKH) DESC

END

GO

-- Ví dụ

EXEC khachHangDatPhongMax

GO

## **31. Sử dụng PROC để thống kê những khách hàng quá hạn thanh toán**

-- proc thống kê những khách hàng quá hạn thanh toán

CREATE PROC khachHangQH

AS

BEGIN

SELECT b.maKH, b.tenKH, b.sdt FROM dbo.tbl\_datphong a

JOIN tbl\_khachhang b ON b.maKH = a.maKH AND a.trangthai='dat' AND a.thanhtoan=0 AND a.ngayTra < CAST(GETDATE() AS DATE)

GROUP BY b.maKH, b.tenKH,b.sdt

END

GO

-- Ví dụ

EXEC dbo.khachHangQH

GO

## **32. Sử dụng PROC để in ra chi tiết thông tin khách hàng quá hạn thanh toán**

/\*

proc in ra phòng, ngày đặt, ngày trả, trạng thái,

thanh toán=0 của khách hàng quá hạn thanh toán

\*/

CREATE PROC chiTietKhachHangQH (@maKH VARCHAR(30))

AS

BEGIN

SELECT b.maPhong AS[Ma Phong],b.tenPhong AS[Ten Phong],a.ngayDat AS[Ngay Dat], a.ngayTra AS[Ngay Tra], a.trangthai AS[Trang Thai], a.thanhtoan AS[Thanh Toan]

FROM tbl\_datphong a JOIN tbl\_phong b ON b.maPhong = a.maPhong AND a.maKH IN(@maKH) AND a.trangthai='dat' AND a.thanhtoan=0 AND a.ngayTra < CAST(GETDATE() AS DATE)

END

GO

-- Ví dụ

EXEC chiTietKhachHangQH @maKH='kh01'

GO

## **33. Sử dụng PROC để thống kê những khách hàng đang hẹn phòng**

-- proc thống kê những khách hàng đang hẹn phòng

CREATE PROC khachHangHenPhong

AS

BEGIN

SELECT b.maKH, b.tenKH, b.sdt FROM dbo.tbl\_datphong a

JOIN tbl\_khachhang b ON b.maKH = a.maKH AND a.trangthai='hen'

GROUP BY b.maKH, b.tenKH,b.sdt

END

GO

-- Ví dụ

EXEC dbo.khachHangHenPhong;

GO

## **34. Sử dụng PROC để in ra chi tiết thông tin khách hàng đang hẹn phòng**

-- Hiển thị ra phong, ngay dat, ngay tra

CREATE PROC chiTietKhachHangHenPhong (@maKH VARCHAR(30))

AS

BEGIN

SELECT b.maPhong AS[Ma Phong],b.tenPhong AS[Ten Phong],a.ngayDat AS[Ngay Dat], a.ngayTra AS[Ngay Tra], a.trangthai AS[Trang Thai], a.thanhtoan AS[Thanh Toan]

FROM tbl\_datphong a JOIN tbl\_phong b ON b.maPhong = a.maPhong AND a.maKH IN(@maKH) AND a.trangthai='hen'

END

GO

-- Ví dụ

EXEC chiTietKhachHangHenPhong @maKH='kh01'

GO

## **35. Sử dụng PROC để thống kê những khách hàng đặt phòng trong tháng/năm , số lượng khách hàng đặt**

-- proc thống kê những khách hàng đặt phòng trong tháng=?, năm=? --> Số lượng khách hàng

CREATE PROC tkKhachHangDatPhongTheoMY(@thang VARCHAR(10), @nam VARCHAR(100))

AS

BEGIN

IF(@thang='' OR @thang IS NULL) PRINT('Thang khong duoc trong')

ELSE IF(@thang NOT LIKE'%[0-9]%') PRINT('Dau vao thang khong dung')

ELSE IF(CAST(@thang AS INT) NOT BETWEEN 1 AND 12) PRINT('Dau vao thang khong dung')

ELSE IF(@nam='' OR @nam IS NULL) PRINT('Nam khong duoc trong')

ELSE IF(@nam NOT LIKE '%[0-9]%') PRINT('Dau vao nam khong dung')

ELSE

BEGIN

-- Hien thi ra khach hang

SELECT b.maKH AS[Ma Khach Hang], b.tenKH AS[Ten Khach Hang], b.sdt AS[So Dien Thoai] FROM tbl\_datphong a JOIN tbl\_khachhang b ON b.maKH = a.maKH AND MONTH(a.ngayDat)=@thang AND YEAR(a.ngayDat)=@nam

AND a.trangthai='dat' GROUP BY b.maKH, b.tenKH, b.sdt

-- Hien thi ra tong so luong

SELECT COUNT(DISTINCT a.maKH) AS[Tong So Khach Hang] FROM tbl\_datphong a JOIN tbl\_khachhang b ON b.maKH = a.maKH AND MONTH(a.ngayDat)=@thang AND YEAR(a.ngayDat)=@nam

AND a.trangthai='dat'

END

END

GO

-- Ví dụ

EXEC tkKhachHangDatPhongTheoMY @thang='1', @nam='2023'

GO

## **36. Sử dụng PROC để thống kê top3 phòng được đặt nhiều nhất**

-- proc thống kê top 3 phòng được đặt nhiều nhất

CREATE PROC phongDatMax

AS

BEGIN

SELECT TOP 3 a.maPhong, b.tenPhong, COUNT(a.maPhong) AS dem

FROM tbl\_datphong a JOIN tbl\_phong b ON b.maPhong = a.maPhong GROUP BY a.maPhong,b.tenPhong ORDER BY COUNT(a.maPhong) DESC

END

GO

-- Ví dụ

EXEC phongDatMax

GO

## **37. Sử dụng PROC để thống kê top3 món ăn được đặt nhiều nhất**

-- proc thống kê top 3 món ăn được đặt nhiều nhất

CREATE PROC doAnDatMax

AS

BEGIN

SELECT TOP 3 a.maDA, b.tenDA, SUM(a.soLuong) AS soLuong

FROM tbl\_dangkyDoAn a JOIN tbl\_doAn b ON b.maDA = a.maDA GROUP BY a.maDA,b.tenDA ORDER BY SUM(a.soLuong) DESC

END

GO

-- Ví dụ

EXEC doAnDatMax

GO

## **38. Sử dụng PROC để thống kê top3 đồ uống được đặt nhiều nhất**

-- proc thống kê top 3 đồ uống được đặt nhiều nhất

CREATE PROC doUongDatMax

AS

BEGIN

SELECT TOP 3 a.maDU, b.tenDU, SUM(a.soLuong) AS soLuong

FROM tbl\_dangkyDoUong a JOIN tbl\_doUong b ON b.maDU = a.maDU GROUP BY a.maDU, b.tenDU ORDER BY SUM(a.soLuong) DESC

END

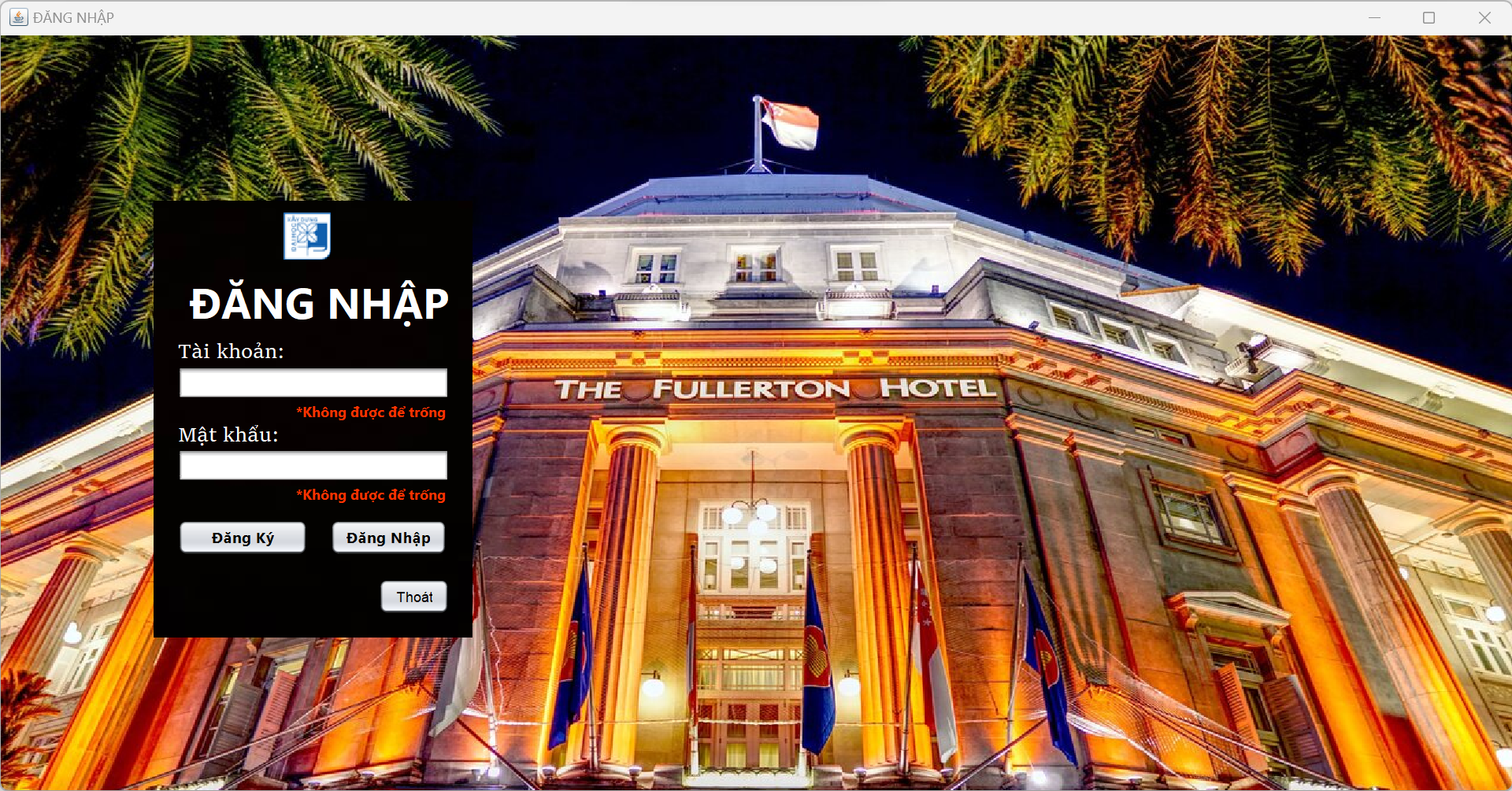
GO

-- Ví dụ

EXEC doUongDatMax

# **Cài đặt chương trình java đồng bộ với cơ sở dữ liệu SQL**

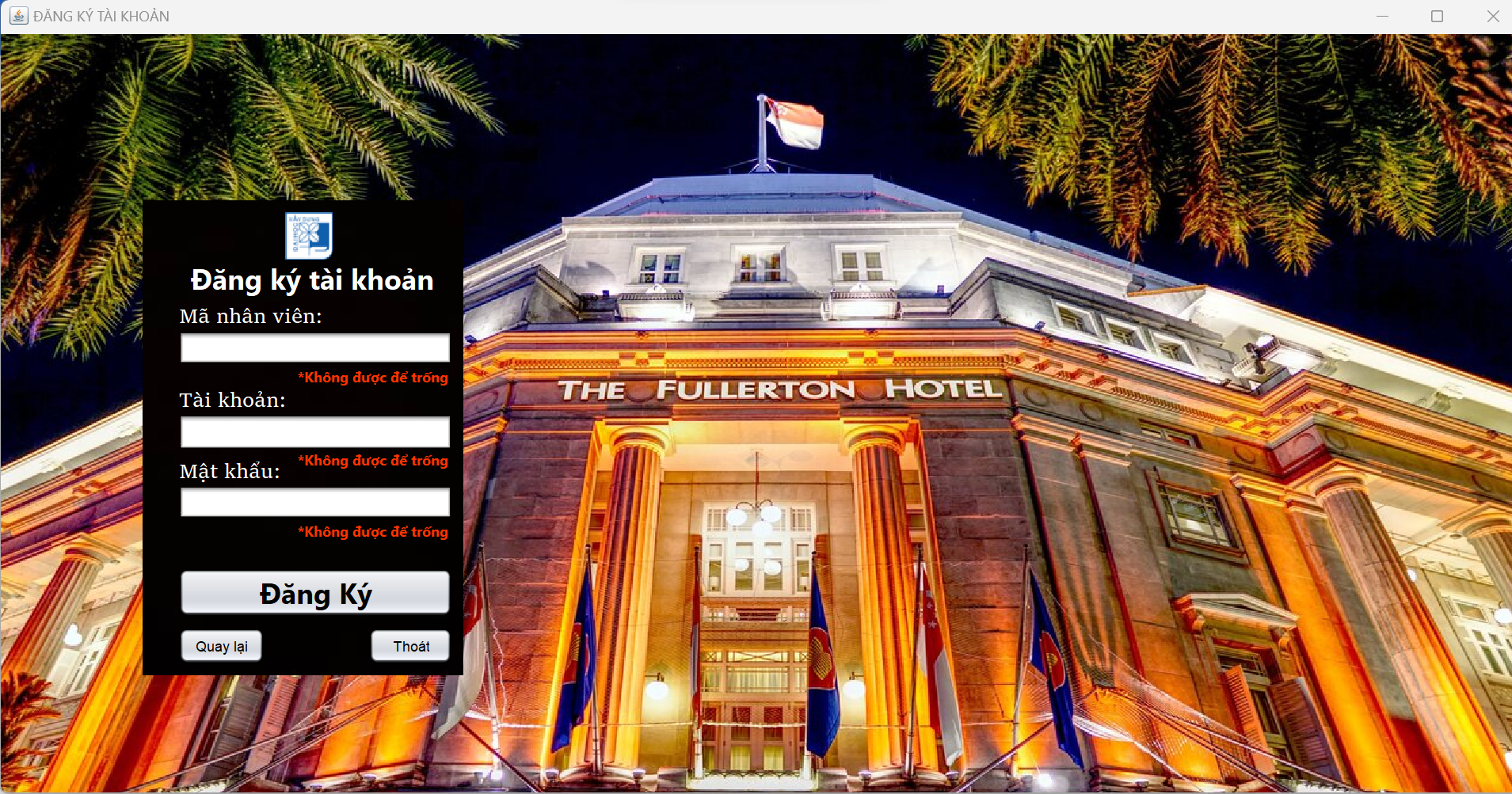
(Có ứng dụng stored procedure với trigger tương ứng với các chức năng cụ thể)

1. ***Form đăng nhập***

**\*\*Chú thích:**

* Mỗi một nhân viên chỉ được phép đăng ký 1 tài khoản duy nhất để đăng nhập.
* Khi bạn nhập đúng **tài khoản** và **mật khẩu** 🡪 đăng nhập thành công.
* Nếu bạn chưa có tài khoản 🡪 nhần nút **Đăng ký** 🡪 vào form đăng ký tài khoản.
* Nhấn **Thoát** để thoát chương trình.

1. ***Form đăng ký***



**\*\*Chú thích:**

* Mỗi một nhân viên chỉ được có 1 tài khoản duy nhất.
* Nhập đúng **mã nhân viên**

🡪 Nhập 1 tài khoản duy nhất và mật khẩu 🡪 đăng ký tài khoản thành công.

* Nhấn **Thoát** để thoát chương trình.

1. Graphical user interface, application

   Description automatically generated***Menu***

**\*\*Chú thích:**

* Nhấn **Tài khoản** 🡪 Form thông tin tài khoản đăng nhập.
* Nhấn **Đồ ăn / Đồ uống / Loại phòng** 🡪 Form quản lý việc thêm, sửa thông tin đồ ăn/ đồ uống/ loại phòng.
* Nhấn **Phòng** 🡪 Form quản lý phòng
* Nhấn **Nhân Viên** 🡪 Form quản lý thông tin nhân viên.
* Nhấn **Khách Hàng** 🡪 Form quản lý thông tin khách hàng.
* Nhấn **Refresh** 🡪 cập nhật lại cơ sở dữ liệu mới nhất

(mặc định sẽ tự động refresh sau khi mở chương trình).

* Nhấn **Trạng thái phòng theo ngày** 🡪 Form hiện thị thông tin trạng thái (trống, đặt, hẹn) của phòng theo ngày cụ thể.
* Nhấn **Đặt phòng** 🡪 Form quản lý việc đặt/hẹn phòng.
* Nhấn **Đặt đồ ăn / đồ uống** 🡪 Form quản lý đặt đồ ăn / đồ uống
* Nhấn **Báo cáo – Thống kê** 🡪 Form hiện thị doanh thu trong 1 khoảng thời gian nhập vào; hiển thị một số form thống kê nổi bật.
* Nhấn **Đăng xuất** 🡪 Form đăng nhập.
* Nhấn **Thoát** 🡪 Thoát chương trình.

1. Graphical user interface, application

   Description automatically generated***Form tài khoản***

**\*\*Chú thích:**

* Nhấn **Sửa** 🡪 enable các trường và nút **Lưu**

(Cho phép sửa tất cả các trường trừ mã nhân viên)

* Nhấn **Lưu 🡪** Nhập đầy đủ các trường (CMT/CCCD và số điện thoại phải là duy nhất)
* Nhấn **Danh sách nhân viên 🡪** Form quản lý nhân viên.

1. ***Form quản lý đồ ăn/đồ uống/loại phòng***

Graphical user interface, application

Description automatically generated

**\*\*Chú thích:**

* Có 3 **Lựa chọn** (Đồ ăn, Đồ uống, Loại phòng)
* Ứng với từng lựa chọn 🡪 Bạn sẽ có các chức năng **Tìm kiếm**, **Thêm**, **Sửa**, **Lưu** và **bảng hiện thị thông tin** tương ứng.
* Nhấn **Tìm kiếm** 🡪 Hiển thị thông tin gần đúng xuống bảng thông tin.
* Nhấn **Lưu (**với chức năng thêm dữ liệu**) 🡪** Lưu dữ liệu thành công nếu nhập đầy đủ các trường và **mã** nhập vào là duy nhất.
* **Click** vào bảng dữ liệu 🡪 Ánh xạ dữ liệu tương ứng lên các trường text bên trên.
* Nhấn **Sửa** 🡪 Sửa dữ liệu theo **mã** thành công nếu nhập đầy đủ các trường.

1. ***Form quản lý phòng***

Graphical user interface

Description automatically generated

**\*\*Chú thích:**

* Nhấn **Tìm kiếm 🡪** ánh xạ dữ liệu theo text tìm kiếm xuống bảng bên dưới.
* Nhấn **Thêm 🡪 enable** các trường.
* Nhấn **Lưu 🡪** Mã phòng phải là duy nhất, nhập đầy đủ các trường 🡪 Lưu thành công.
* Khi thêm phòng thành công 🡪 tự động thêm trạng thái trống cho phòng vào bảng trạng thái phòng theo ngày.
* Khi **click** vào bảng dữ liệu 🡪 ánh xạ dữ liệu lên các trường text tương ứng bên trên.
* Nhấn **Sửa 🡪** Sửa dữ liệu theo mã phòng 🡪 Nhập đầy đủ các trường
* Sửa dữ liệu thành công.

1. ***Form quản lý nhân viên***

Graphical user interface

Description automatically generated

**\*\*Chú thích:**

* Nếu bạn đăng nhập với tài khoản có quyền quản trị viên
* Mới có quyền thêm, sửa thông tin nhân viên.
* Các chức năng **Tìm kiếm**, **Thêm**, **Lưu**, **Sửa** được thiết kế tương tự như Form quản lý phòng.

1. Graphical user interface, application

   Description automatically generated***Form quản lý khách hàng***

**\*\*Chú thích:**

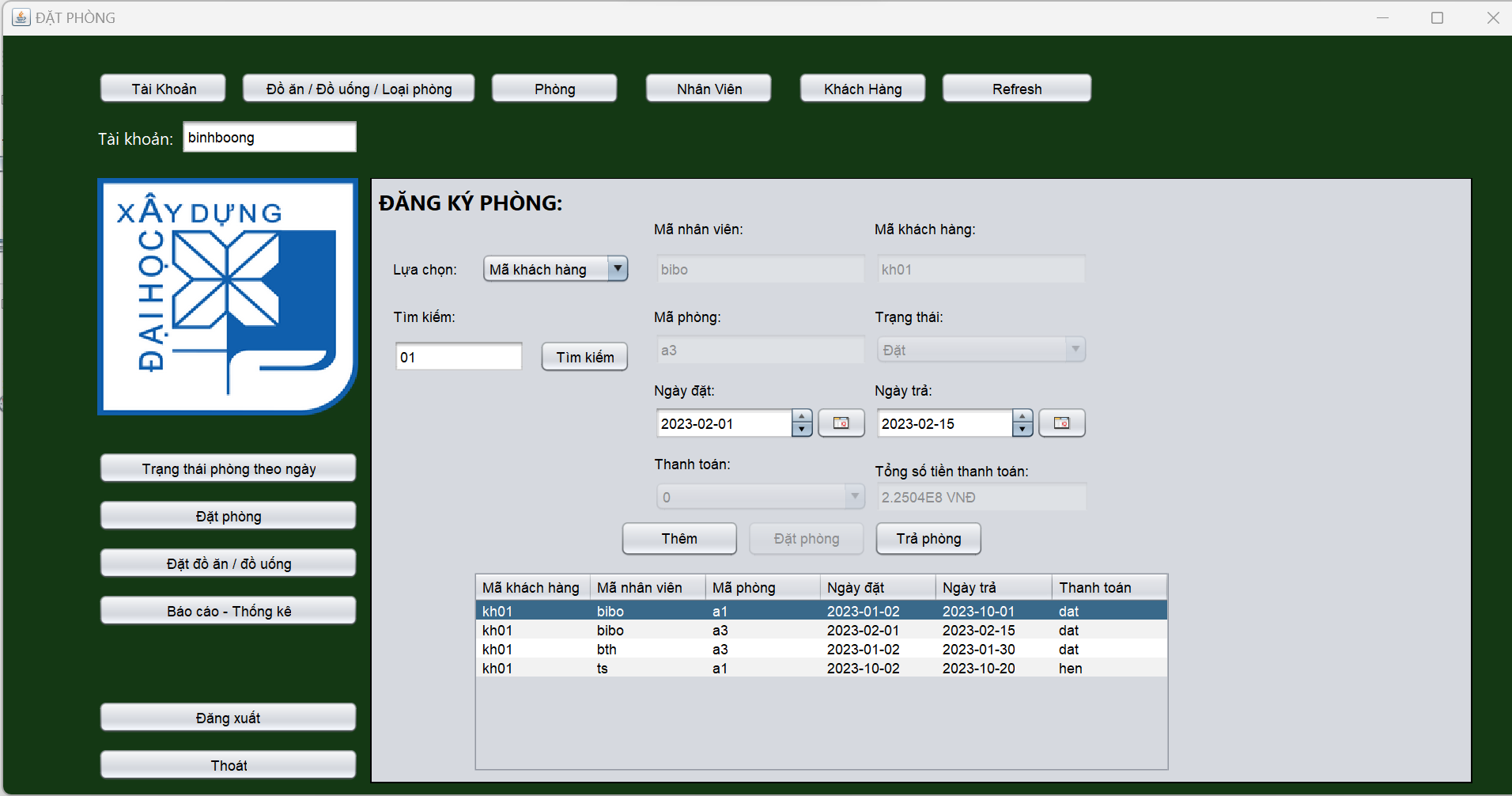
* Các chức năng **Tìm kiếm**, **Thêm**, **Lưu**, **Sửa** được thiết kế tương tự như Form quản lý phòng.

1. Graphical user interface, table

   Description automatically generated***Form trạng thái phòng theo ngày***

**\*\*Chú thích:**

* **Trạng thái (**Trống, Đặt, Hẹn) tương ứng với thời gian hiện tại mà phòng đang trống **hoặc** thời gian đặt/hẹn phòng cụ thể với từng mã phòng.
* Ánh xạ dữ liệu tương ứng vào bảng hiển thị bên dưới.

1. ***Form đặt phòng***

**\*\*Chú thích:**

* Có 2 lựa chọn tìm kiếm theo mã phòng và mã khách hàng
* Ánh xạ dữ liệu tương ứng với text tìm kiếm vào bảng hiển thị dữ liệu.
* Nhấn **Thêm 🡪** enable tất cả các trường (trừ **mã nhân viên**), **mã nhân viên** được gán là mã nhân viên đang đăng nhập vào chương trình.

**(Note)**

- Khách hàng có thể đặt/hẹn phòng

- Thời gian đặt phòng phải là hiện tại

- Thời gian đặt/hẹn phòng phải > thời gian trả phòng gần nhất.

- Với phòng trống thì khách hàng đặt/hẹn phòng bình thường

- Với phòng đang được hẹn 🡪 khách hàng sẽ đặt phòng nếu thời gian đặt là hiện tại và thời trả phòng phải < thời gian hẹn phòng hiện tại của phòng đó

- Còn với phòng đang được đặt(được sử dụng) thì khách khách hàng sẽ hẹn

được phòng nếu thời gian đặt phòng > thời gian trả phòng hiện tại.

* Nhấn **Đặt phòng 🡪** Nhập đầy đủ các trường và thỏa mã điều kiện trên
* Nếu nhập thời gian đặt phòng là hiện tại 🡪 Đặt phòng thành công.
* Nếu nhập thời gian đặt phòng là tương lai 🡪 Hẹn đặt phòng thành công.
* Nhấn **Trả phòng 🡪** Nếu đến ngày trả phòng thì trả phòng thành công
* **Click** vào bảng dữ liệu 🡪 Ánh xạ dữ liệu lên các trường text tương ứng bên trên và hiển thị tổng số tiền mà khách hàng đó phải thanh toán (tiền đặt phòng, tiền đặt đồ ăn/đồ uống)

1. Graphical user interface, table

   Description automatically generated***Form đặt đồ ăn/đồ uống***

**\*\*Chú thích:**

* Có 2 lựa chọn **Dịch vụ (**Đồ ăn, Đồ uống**)**
* Các chức năng **Đặt mới**, **Tìm kiếm**, **Đặt** và thông tin hiện thị ở 2 bảng sẽ tương ứng với lựa chọn dịch vụ của bạn.
* Nhấn **Tìm kiếm 🡪** Ánh xạ dữ liệu ứng với mã đồ ăn/đồ uống tương ứng vào 2 bảng dữ liệu.
* Nhấn **Đặt mới 🡪** Xóa trắng các trường nhập đầu vào 🡪 enable nút **Đặt.**
* Nhấn nút **Đặt** 🡪 Thời gian đặt mặc định là thời gian hiện tại, nhập đầy đủ các trường 🡪 Đặt đồ ăn/đồ uống thành công.

…

1. ***Form báo cáo thống kê***

Graphical user interface, application

Description automatically generated

**\*\*Chú thích:**

* Nhấn **Doanh thu 🡪** Tính toán doanh thu theo từ ngày ? đến ngày ? nhập vào

🡪 hiển thị ở text doanh thu bên dưới

* Ở phần báo cáo – thống kê: Nhấn **Hiển thị 🡪** Hiển thị lên các form tương ứng với **Lựa chọn.**

# **Nhận xét về phần cơ sở dữ liệu**

1. ***Ưu điểm***

* Đáp ứng được nghiệp vụ cơ bản của 1 khách sạn: đặt/hẹn phòng,

đặt đồ ăn/đồ uống.

* Các truy vấn đúng trọng tâm, ưu tiên truy vấn theo thời gian (theo ngày)
* Có ứng dụng **stored procedure** và **trigger** trong truy vấn dữ liệu
* Đảm bảo được dữ liệu đầu vào.

1. ***Hạn chế và ý tưởng cải thiện***

* Thời gian đặt/hẹn phòng, đặt đồ ăn/đồ uống mới chỉ được tính theo ngày

🡪 Cần cải thiện để xử lý được đầu vào thời gian tính theo từng giờ, phút, giây

để đáp ứng tối đa được nhu cầu của khách hàng 🡪 Tăng thu nhập cho khách sạn

🡪 Ví dụ 1 khách hàng chỉ đặt phòng trong 3 tiếng thì sau 3 tiếng đó + với thời gian dọn dẹp phòng thì khách hàng mới có thể đặt ngay phòng đó.

* Về trường hợp bảo trì phòng:

***\*\*Đưa ra ý tưởng về cách xử lý:***

**+ Nếu phòng (bảo trì/ đột ngột bảo trì)🡪 không đặt được phòng**

.Nếu phòng hiện tại trống

🡪 bảo trì bình thường

.Nếu phòng đang có người đặt

🡪 cập nhật ngày trả là ngày hiện tại thực hiện bảo trì

.Nếu phòng có người hẹn đặt phòng

🡪 đổi phòng / cập nhật ngày trả là ngày đặt (nếu không đổi được phòng)

**++Tính được thời gian hoàn thành bảo trì phòng (nếu được)**

**🡪** Cập nhật trạng thái trống cho phòng sớm nhất

🡪 Tận dụng được tối đa năng suất phục vụ của khách sạn

🡪 Tăng nguồn thu cho khách sạn

**…**