Thủ tục

1. Thêm mới Hội viên

Tên thủ tục: sp\_ThemMoiHoiVien

Mục đích: Thêm một hội viên mới vào bảng HocVien và đồng thời tạo một hóa đơn ban đầu trong bảng HoaDon.

Tham số đầu vào:

@Ten: Tên hội viên

@SDT: Số điện thoại

@IDThe: Mã thẻ hội viên

@IDGoiTap: Mã gói tập đã đăng ký

Logic:

Tạo một ID mới cho hội viên.

Tìm thông tin gói tập (GiaTriGoi, ThoiGianGoi) từ @IDGoiTap.

Tính toán NgayKetThuc dựa trên NgayDangKy (ngày hiện tại) và ThoiGianGoi.

Thêm bản ghi vào bảng HocVien.

Thêm bản ghi hóa đơn vào bảng HoaDon với SoTienThu ban đầu là GiaTriGoi.

2. Cập nhật thông tin Hội viên

Tên thủ tục: sp\_CapNhatThongTinHoiVien

Mục đích: Cập nhật các thông tin cá nhân của hội viên.

Tham số đầu vào:

@ID: ID của hội viên cần cập nhật

@Ten: Tên mới

@SDT: Số điện thoại mới

Logic:

Kiểm tra xem @ID có tồn tại trong bảng HocVien không.

Nếu có, thực hiện câu lệnh UPDATE để cập nhật các thông tin.

3. Gia hạn Gói tập

Tên thủ tục: sp\_GiaHanGoiTap

Mục đích: Cập nhật ngày kết thúc của hội viên khi họ mua thêm một gói tập mới hoặc gia hạn gói cũ.

Tham số đầu vào:

@IDHocVien: ID của hội viên cần gia hạn

@IDGoiTapMoi: Mã gói tập mới

@NgayGiaHan: Ngày gia hạn

Logic:

Tìm thông tin ThoiGianGoi của gói tập mới từ @IDGoiTapMoi.

Cập nhật NgayKetThuc trong bảng HocVien bằng cách cộng thêm ThoiGianGoi vào ngày kết thúc hiện tại.

Tạo một hóa đơn mới trong bảng HoaDon cho lần gia hạn này.

4. Xử lý thanh toán hóa đơn

Tên thủ tục: sp\_ThanhToanHoaDon

Mục đích: Ghi nhận một giao dịch thanh toán cho một hóa đơn cụ thể.

Tham số đầu vào:

@IDHoaDon: ID của hóa đơn cần thanh toán

@SoTienThu: Số tiền đã nhận

@NgayTra: Ngày thanh toán

Logic:

Cập nhật bảng HoaDon với SoTienThu và NgayTra.

5. Thống kê doanh thu theo tháng

Tên thủ tục: sp\_ThongKeDoanhThuThang

Mục đích: Tính tổng doanh thu theo tháng của phòng gym.

Tham số đầu vào:

@Nam: Năm cần thống kê

@Thang: Tháng cần thống kê

Logic:

Sử dụng câu lệnh SELECT SUM(SoTienThu) trên bảng HoaDon.

Lọc dữ liệu theo năm và tháng từ cột NgayTra.

Trả về tổng doanh thu.

6. Thống kê trạng thái thiết bị

Tên thủ tục: sp\_ThongKeThietBi

Mục đích: Liệt kê các thiết bị sắp đến hạn bảo dưỡng hoặc đang không hoạt động.

Logic:

Sử dụng SELECT trên bảng DanhSachThietBi.

Sử dụng WHERE để lọc các thiết bị có KhaNangHoatDong = 0 hoặc ThoiGianBaoDuong gần với ngày hiện tại.

Trả về danh sách thiết bị đó.

Hàm

. Hàm tính tuổi của nhân viên hoặc hội viên

Tên hàm: fn\_TinhTuoi

Mục đích: Trả về số tuổi của một người dựa trên ngày sinh.

Tham số đầu vào: @NgaySinh DATE

Logic:

SQL

CREATE FUNCTION fn\_TinhTuoi (@NgaySinh DATE)

RETURNS INT

AS

BEGIN

RETURN DATEDIFF(YEAR, @NgaySinh, GETDATE()) -

CASE

WHEN (MONTH(@NgaySinh) > MONTH(GETDATE())) OR

(MONTH(@NgaySinh) = MONTH(GETDATE()) AND DAY(@NgaySinh) > DAY(GETDATE()))

THEN 1

ELSE 0

END;

END;

2. Hàm tính thời gian còn lại của gói tập

Tên hàm: fn\_TinhThoiGianConLai

Mục đích: Trả về số ngày còn lại của gói tập cho một hội viên.

Tham số đầu vào: @IDHocVien INT

Logic:

SQL

CREATE FUNCTION fn\_TinhThoiGianConLai (@IDHocVien INT)

RETURNS INT

AS

BEGIN

DECLARE @NgayKetThuc smalldatetime;

SELECT @NgayKetThuc = NgayKetThuc FROM HocVien WHERE ID = @IDHocVien;

RETURN DATEDIFF(DAY, GETDATE(), @NgayKetThuc);

END;

3. Hàm lấy tên chức vụ từ ID

Tên hàm: fn\_LayTenChucVu

Mục đích: Trả về tên chức vụ của một nhân viên dựa trên ID chức vụ.

Tham số đầu vào: @IDChucVu INT

Logic:

SQL

CREATE FUNCTION fn\_LayTenChucVu (@IDChucVu INT)

RETURNS NVARCHAR(100)

AS

BEGIN

DECLARE @TenChucVu NVARCHAR(100);

SELECT @TenChucVu = ChucVu FROM ChiTietChucVu WHERE ID = @IDChucVu;

RETURN @TenChucVu;

END;

4. Hàm kiểm tra trạng thái hoạt động của thiết bị

Tên hàm: fn\_KiemTraTrangThaiThietBi

Mục đích: Trả về một chuỗi trạng thái ('Đang hoạt động', 'Hỏng hóc') dựa trên giá trị bit.

Tham số đầu vào: @TrangThai BIT

Logic:

SQL

CREATE FUNCTION fn\_KiemTraTrangThaiThietBi (@TrangThai BIT)

RETURNS NVARCHAR(50)

AS

BEGIN

IF @TrangThai = 1

RETURN N'Đang hoạt động';

ELSE

RETURN N'Hỏng hóc';

END;

RÀNG BUỘC

Dựa trên cấu trúc cơ sở dữ liệu quản lý phòng gym bạn đã cung cấp, đây là các ràng buộc (constraints) cần thiết để đảm bảo tính toàn vẹn và hợp lệ của dữ liệu.

1. Ràng buộc Khóa chính (Primary Key)

Mỗi bảng cần có một khóa chính để xác định duy nhất mỗi bản ghi. Các ràng buộc này đã được định nghĩa trong script của bạn, nhưng cần được nhắc lại để đảm bảo tính nhất quán.

ChiTietChucVu: PK\_ChiTietChucVu trên cột ID.

ChiTietGoiTap: PK\_ChiTietGoiTap trên cột ID.

ChiTietHangThietBi: PK\_ChiTietHangThietBi trên cột ID.

DanhSachNhanVien: PK\_DanhSachNhanVien trên cột ID.

DanhSachThietBi: PK\_DanhSachThietBi trên cột ID.

HoaDon: PK\_HoaDon trên cột IDHoaDon.

HocVien: PK\_HocVien trên cột ID.

2. Ràng buộc Khóa ngoại (Foreign Key)

Khóa ngoại thiết lập mối quan hệ giữa các bảng và đảm bảo rằng dữ liệu trong một bảng tham chiếu đến một bản ghi hợp lệ trong bảng khác.

DanhSachNhanVien:

FK\_DanhSachNhanVien\_ChiTietChucVu: Cột IDChucVu tham chiếu đến ID của bảng ChiTietChucVu.

DanhSachThietBi:

FK\_DanhSachThietBi\_ChiTietHangThietBi: Cột IDHangSanXuat tham chiếu đến ID của bảng ChiTietHangThietBi.

HoaDon:

FK\_HoaDon\_HocVien: Cột IDHocVien tham chiếu đến ID của bảng HocVien.

FK\_HoaDon\_ChiTietGoiTap: Cột IDGoiDangKy tham chiếu đến ID của bảng ChiTietGoiTap.

HocVien:

FK\_HocVien\_ChiTietGoiTap: Cột IDGoiTap tham chiếu đến ID của bảng ChiTietGoiTap.

3. Ràng buộc kiểm tra (Check Constraint)

Ràng buộc này được sử dụng để giới hạn các giá trị có thể có trong một cột, đảm bảo tính hợp lệ của dữ liệu.

DanhSachNhanVien:

Kiểm tra tuổi nhân viên: Đảm bảo ngày sinh phải nhỏ hơn ngày hiện tại để tuổi luôn là số dương. Ví dụ: CHECK (Ngaysinh < GETDATE()).

Kiểm tra lương: Đảm bảo Luong phải lớn hơn 0. Ví dụ: CHECK (Luong > 0).

ChiTietGoiTap:

Kiểm tra giá trị gói: Đảm bảo GiaTriGoi phải lớn hơn 0. Ví dụ: CHECK (GiaTriGoi > 0).

HoaDon:

Kiểm tra số tiền thu: Đảm bảo SoTienThu phải lớn hơn hoặc bằng 0. Ví dụ: CHECK (SoTienThu >= 0).

Kiểm tra khuyến mãi: Đảm bảo KhuyenMai là một giá trị hợp lệ, thường nằm trong khoảng từ 0 đến 1 (0% đến 100%). Ví dụ: CHECK (KhuyenMai BETWEEN 0 AND 1).

DanhSachThietBi:

Kiểm tra ngày bảo dưỡng: Đảm bảo ThoiGianBaoDuong phải lớn hơn hoặc bằng ThoiGianMua để tránh lỗi logic. Ví dụ: CHECK (ThoiGianBaoDuong >= ThoiGianMua).

4. Ràng buộc duy nhất (Unique Constraint)

Ràng buộc này đảm bảo rằng tất cả các giá trị trong một cột (hoặc tập hợp các cột) là duy nhất.

HocVien:

UNIQUE trên cột IDThe: Đảm bảo mỗi hội viên có một mã thẻ duy nhất.

DanhSachNhanVien:

UNIQUE trên cột SDT: Đảm bảo không có hai nhân viên nào có cùng số điện thoại.