**TRƯỜNG ĐẠI HỌC KINH TẾ - ĐẠI HỌC ĐÀ NẴNG**

**KHOA THỐNG KÊ – TIN HỌC**



**BÁO CÁO HỌC PHẦN CƠ SỞ DỮ LIỆU**

**Hệ thống quản lý công việc của giáo viên trong BMYC**

**Lớp** :48K21.2

**Nhóm** : 48K212.03

**Giáo viên hướng dẫn** : Cao Thị Nhâm

**Thành viên** : Nguyễn Thị Thảo Phương

Trần Hiệp Lực

Vy Mai Nhật Thúy

Hồ Thị Thu Hà

Trần Tiến Hùng

**Đà Nẵng, 05/2024**

**MỤC LỤC**

Table of Contents

[CHƯƠNG 1. MÔ TẢ KHÁI QUÁT HỆ THỐNG 4](#_Toc180436106)

[CHƯƠNG 2. THIẾT KẾ BẢNG 5](#_Toc180436107)

[1. Thiết kế chi tiết các bảng: 5](#_Toc180436108)

[1.1. Giáo viên: 5](#_Toc180436109)

[1.2. Lớp học: 5](#_Toc180436110)

[1.3. Khóa học: 6](#_Toc180436111)

[1.4. Lịch dạy: 6](#_Toc180436112)

[1.5. Phiếu xin nghỉ: 6](#_Toc180436113)

[1.6. Lương: 7](#_Toc180436114)

[1.7. Tài khoản: 7](#_Toc180436115)

[1.8. Bằng cấp: 7](#_Toc180436116)

[1.9. Bảng cấp chi tiết: 8](#_Toc180436117)

[2. Sơ đồ quan hệ: 8](#_Toc180436118)

[CHƯƠNG 3. XÂY DỰNG MODULE 9](#_Toc180436119)

[1. Các module tạo dữ liệu dump cho các bảng trong cơ sở dữ liệu: 9](#_Toc180436120)

[1.1. Giáo viên: 9](#_Toc180436121)

[1.2. Lớp học: 9](#_Toc180436122)

[1.3. Khóa học: 10](#_Toc180436123)

[1.4. Lịch dạy: 10](#_Toc180436124)

[1.5. Phiếu xin nghỉ: 11](#_Toc180436125)

[1.6. Lương: 12](#_Toc180436126)

[1.7. Tài khoản: 12](#_Toc180436127)

[1.8. Bằng cấp: 13](#_Toc180436128)

[2. Các module phục vụ các thao tác xử lý dữ liệu: 14](#_Toc180436129)

[2.1. Trigger: 14](#_Toc180436130)

[2.2. Thủ tục 19](#_Toc180436131)

**DANH MỤC HÌNH ẢNH**

[Hình 1. Sơ đồ quan hệ 12](#_Toc180415488)

# MÔ TẢ KHÁI QUÁT HỆ THỐNG

Dự án này được thực hiện nhằm tạo ra một phần mềm hỗ trợ công việc của nhân viên công ty BMYC. Cụ thể, mục đích chính của hệ thống bao gồm:

* Quản lý lịch trình: Hỗ trợ quản lý và giáo viên trong việc lập kế hoạch quản lý và giảng cũng như đảm bảo hiệu quả trong việc sắp xếp thời gian và nguồn lực.
* Quản lý thông tin học viên/giáo viên: Lưu trữ và quản lý thông tin cá nhân, tiến trình học tập, thông tin lịch dạy và các thông tin khác của học viên/giáo viên, từ đó quản lý có thể nắm bắt rõ trạng thái của giáo viên và giáo viên hiểu rõ trạng thái của học sinh
* Quản lý tài chính: Cung cấp thông tin và báo cáo về thu nhập của giáo viên theo thời gian, hỗ trợ việc đưa ra quyết định chi tiêu và tạo động lực trong công việc
* Tính minh bạch: Tạo điều kiện cho việc trao đổi thông tin một cách dễ dàng và minh bạch giữa quản lý và giáo viên. Từ việc đăng ký lịch dạy đến quản lý thu nhập và các hoạt động khác, thông tin sẽ được lưu trữ, gửi nhận một cách nhanh chóng và chính xác từ nội dung đến thời gian.
* Tối ưu hóa hiệu suất công việc: Hỗ trợ giáo viên và quản lý trong việc tối ưu hóa thời gian và quy trình công việc

Tóm lại, mục đích của hệ thống là tạo ra một một môi trường làm việc tốt hơn, giúp người dùng quản lý công việc hàng ngày một cách dễ dàng, làm việc chuyên nghiệp, và nâng cao chất lượng dịch vụ giảng dạy.

# THIẾT KẾ BẢNG

## Thiết kế chi tiết các bảng:

## Giáo viên:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Mô tả | Kiểu dữ liệu | Ràng buộc |
| **1** | Magiaovien | Mã giáo viên | Char (10) | Primary Key Not Null |
| **2** | Tengiaovien | Tên giáo viên | Varchar(50) | Not Null |
| **3** | Ngaysinh | Ngày sinh | Date | Not Null |
| **4** | Diachi | Địa chỉ | Varchar(50) | Not Null |
| **5** | Email | Email | Varchar(50) | Unique Not Null |
| **6** | Sdt | Số điện thoại | Char(10) | Unique Not Null |

## Lớp học:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Mô tả | Kiểu dữ liệu | Ràng buộc |
| **1** | Malophoc | Mã lớp học | Char (10) | Primary Key Not Null |
| **2** | Magiaovien | Mã giáo viên | Char(10) | Foreign Key Not Null |
| **3** | Tenlophoc | Tên lớp học | Varchar (30) | Not Null |
| **4** | Khunggioday | Khung giờ dạy | Time | Not Null |

## Khóa học:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Mô tả | Kiểu dữ liệu | Ràng buộc |
| **1** | Makhoahoc | Mã khóa học | Char(10) | Primary Key Not Null |
| **2** | Magiaovien | Mã giáo viên | Char(10) | Foreign Key Not Null |
| **3** | Tenkhoahoc | Tên khóa học | Varchar(50) | Not Null |
| **4** | Thoigianbatdau | Thời gian bắt đầu | Date | Not Null |
| **5** | Thoigianketthuc | Thời gian kết thúc | Date | Not Null |

## Lịch dạy:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Mô tả | Kiểu dữ liệu | Ràng buộc |
| **1** | Malichday | Mã đăng ký lịch dạy | Char(10) | Primary Key Not Null |
| **2** | Magiaovien | Mã giáo viên | Char(10) | Foreign Key Not Null |
| **3** | Malophoc | Mã lớp học | Char(10) | Foreign Key Not Null |
| **4** | Thoigiandangki | Thời gian đăng ký | Date | Not Null |
| **5** | Khunggiohoc | Khung giờ học | Datetime | Not Null |

## Phiếu xin nghỉ:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Mô tả | Kiểu dữ liệu | Ràng buộc |
| **1** | Maphieuxinnghi | Mã phiếu xin nghỉ | Char (10) | Primary Key Not Null |
| **2** | Magiaovien | Mã giáo viên | Char(10) | Foreign Key Not Null |
| **3** | Malophoc | Mã lớp học | Char(10) | Foreign Key Not Null |
| **4** | Lydoxinnghi | Lý do xin nghỉ | Varchar(100) | Not Null |
| **5** | Thoigianxinnghi | Thời gian xin nghỉ | Date | Not Null |

## Lương:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Mô tả | Kiểu dữ liệu | Ràng buộc |
| **1** | Maluong | Mã lương | Char(10) | Primary Key Not Null |
| **2** | Magiaovien | Mã giáo viên | Char(10) | Foreign Key Not Null |
| **3** | Hesoluong | Hệ số lương | Demical(10,2) | Not Null |
| **4** | Motaluong | Mô tả lương | Varchar(50) | Not Null |
| **5** | Luongcoban | Lương cơ bản | Varchar(20) | Not Null |

## Tài khoản:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Mô tả | Kiểu dữ liệu | Ràng buộc |
| **1** | Mataikhoan | Mã tài khoản | Char(10) | Primary Key Not Null |
| **2** | Magiaovien | Mã giáo viên | Char(10) | Foreign Key Not Null |
| **3** | Tentaikhoan | Tên tài khoản | Varchar(50) | Not Null |
| **4** | Matkhau | Mật khẩu | Varchar(30) | Not Null |

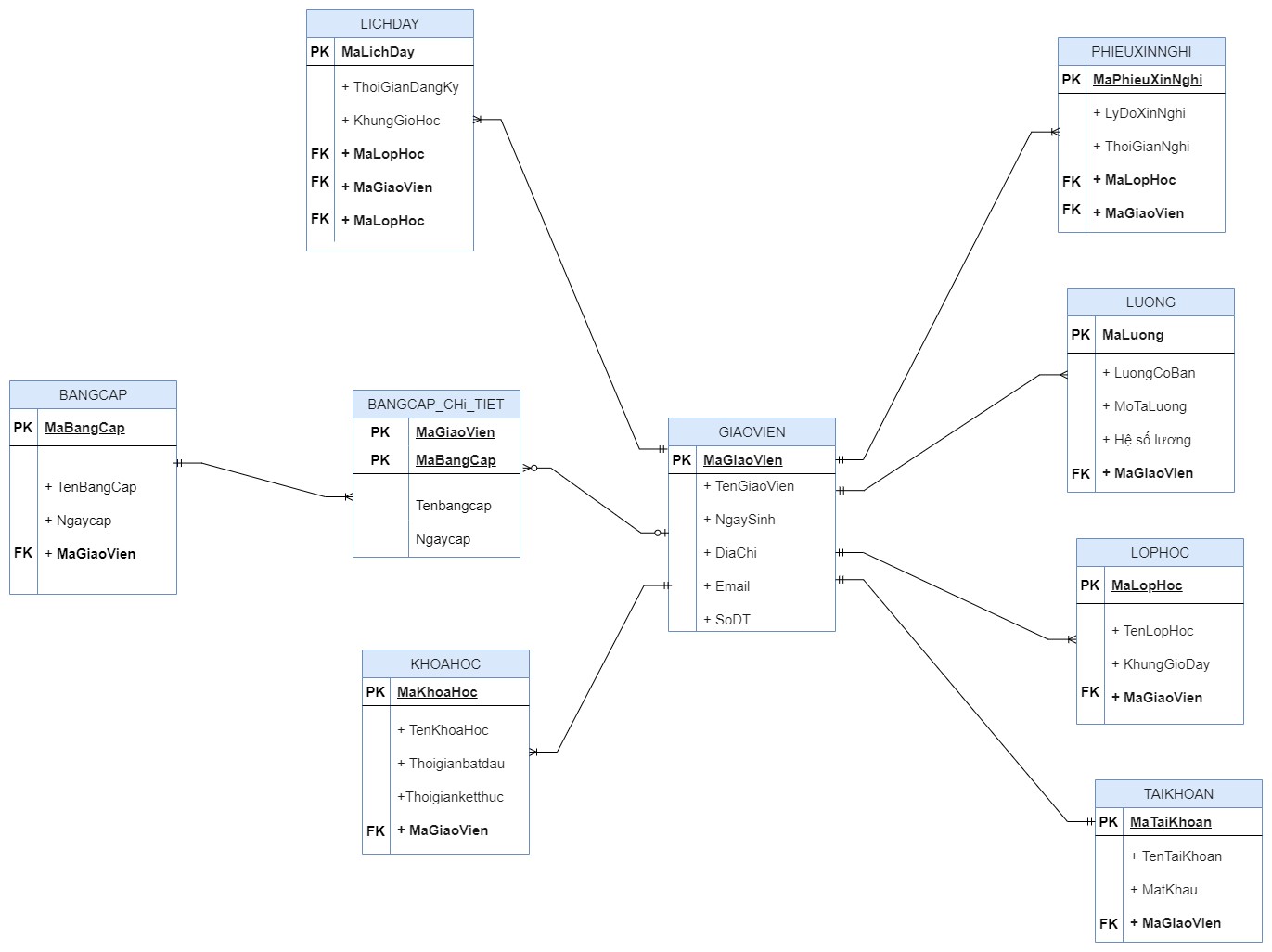
## Bằng cấp:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Mô tả | Kiểu dữ liệu | Ràng buộc |
| **1** | Mabangcap | Mã bằng cấp | Char(10) | Primary Key Not Null |
| **2** | Magiaovien | Mã giáo viên | Char(10) | Foreign Key Not Null |
| **3** | Tenbangcap | Tên bằng cấp | Varchar(50) | Not Null |
| **4** | Ngaycap | Ngày cấp | Date | Not Null |

## Bảng cấp chi tiết:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Mô tả | Kiểu dữ liệu | Ràng buộc |
| **1** | Mabangcap | Mã bằng cấp | Char(10) | Primary Key Not Null |
| **2** | Magiaovien | Mã giáo viên | Char(10) | Primary Key Not Null |
| **3** | Tenbangcap | Tên bằng cấp | Varchar(50) | Not Null |
| **4** | Ngaycap | Ngày cấp | Date | Not Null |

## Sơ đồ quan hệ:



Hình 1. Sơ đồ quan hệ

# XÂY DỰNG MODULE

## Các module tạo dữ liệu dump cho các bảng trong cơ sở dữ liệu:

## Giáo viên:

create procedure giaovien\_dump

as

begin

declare @i int = 1;

while @i <= 1000

begin

insert into Giaovien (Magiaovien, Tengiaovien, Ngaysinh, Diachi, Email, Sdt)

values (

'GV' + right('00000000' + CAST(@i AS VARCHAR(10)), 8),

'Giáo viên ' + cast(@i as varchar(10)),

dateadd(year, -round(rand() \* 30 + 22, 0), getdate()),

'Địa chỉ ' + cast(@i as varchar(10)),

'Email' + cast(@i as varchar(10)) + '@gmail.com',

right('0000000000' + cast(cast(rand() \* 10000000000 as bigint) as varchar(10)), 10)

);

set @i = @i + 1;

end

end;

exec giaovien\_dump;

select \* from giaovien;

## Lớp học:

create procedure lophoc\_dump

as

begin

declare @i int = 1;

while @i <= 1000

begin

insert into Lophoc (Malophoc, Tenlophoc, Khunggioday, Magiaovien)

values (

'LH' + right('00000000' + CAST(@i AS VARCHAR(10)), 8),

N'Lớp ' + cast(@i as varchar(10)),

cast ((cast (8 + floor(rand() \* 5) as varchar) + ':00:00') as time ),

'GV' + RIGHT('00000000' + CAST(FLOOR(RAND() \* 1000) + 1 AS VARCHAR(10)), 8)

);

set @i = @i + 1;

end

end;

exec lophoc\_dump;

select \* from lophoc;

## Khóa học:

create procedure khoahoc\_dump

as

begin

declare @i int = 1;

while @i <= 1000

begin

insert into Khoahoc (Makhoahoc, Tenkhoahoc, Thoigianbatdau, Thoigiankethuc, Magiaovien)

values (

'KH' + right('00000000' + CAST(@i AS VARCHAR(10)), 8),

'Khóa học ' + cast(@i as varchar(10)),

dateadd(day, @i, getdate()),

dateadd(day, @i + 30, getdate()),

'GV' + RIGHT('00000000' + CAST(FLOOR(RAND() \* 1000) + 1 AS VARCHAR(10)), 8)

);

set @i = @i + 1;

end

end;

exec khoahoc\_dump;

select \* from khoahoc;

## Lịch dạy:

create procedure lichday\_dump

as

begin

declare @i int = 1;

while @i <= 1000

begin

insert into Lichday (Madangkilichday, Thoigiandangki, khunggiohoc, Magiaovien, Makhoahoc, Malophoc)

values (

'LD' + right('00000000' + CAST(@i AS VARCHAR(10)), 8),

getdate(),

dateadd(hour, (rand() \* 4) + 8, cast(getdate() as datetime)),

'GV' + RIGHT('00000000' + CAST(FLOOR(RAND() \* 1000) + 1 AS VARCHAR(10)), 8),

'KH' + RIGHT('00000000' + CAST(FLOOR(RAND() \* 1000) + 1 AS VARCHAR(10)), 8),

'LH' + RIGHT('00000000' + CAST(FLOOR(RAND() \* 1000) + 1 AS VARCHAR(10)), 8)

);

set @i = @i + 1;

end

end;

exec lichday\_dump;

select \* from lichday;

## Phiếu xin nghỉ:

create procedure phieuxinnghi\_dump

as

begin

declare @i int = 1;

while @i <= 1000

begin

insert into Phieuxinnghi (Maphieuxinnghi, Magiaovien, Lydoxinnghi, Thoigiannghi, Malophoc)

values (

'PN' + right('00000000' + CAST(@i AS VARCHAR(10)), 8),

'GV' + RIGHT('00000000' + CAST(FLOOR(RAND() \* 1000) + 1 AS VARCHAR(10)), 8),

'lý do nghỉ ' + cast(@i as varchar(10)),

dateadd(day, -(@i % 30), getdate()),

'LH' + RIGHT('00000000' + CAST(FLOOR(RAND() \* 1000) + 1 AS VARCHAR(10)), 8)

);

set @i = @i + 1;

end

end;

exec phieuxinnghi\_dump;

select \* from phieuxinnghi;

## Lương:

create procedure luong\_dump

as

begin

declare @i int = 1;

while @i <= 1000

begin

insert into Luong (Maluong, Magiaovien, Luongcoban, Motaluong, Hesoluong)

values (

'BL' + right('00000000' + CAST(@i AS VARCHAR(10)), 8),

'GV' + RIGHT('00000000' + CAST(FLOOR(RAND() \* 1000) + 1 AS VARCHAR(10)), 8),

cast(3000000 + (rand() \* 2000000) as varchar(20)),

'mô tả lương ' + cast(@i as varchar(10)),

round(rand() \* 3 + 1, 2)

);

set @i = @i + 1;

end

end;

exec luong\_dump;

select \* from luong;

## Tài khoản:

create procedure taikhoan\_dump

as

begin

declare @i int = 1;

while @i <= 1000

begin

insert into Taikhoan (Mataikhoan, Tentaikhoan, Matkhau, Magiaovien)

values (

'TK' + right('00000000' + CAST(@i AS VARCHAR(10)), 8),

'Tài khoản ' + cast(@i as varchar(10)),

'Matkhau\_' + cast(@i as varchar(10)),

'GV' + RIGHT('00000000' + CAST(FLOOR(RAND() \* 1000) + 1 AS VARCHAR(10)), 8)

);

set @i = @i + 1;

end

end;

exec taikhoan\_dump;

select \* from taikhoan;

## Bằng cấp:

create procedure bangcap\_dump

as

begin

declare @i int = 1;

while @i <= 1000

begin

insert into Bangcap (Mabangcap, Tenbangcap, Magiaovien, Ngaycap)

values (

'BC' + right('00000000' + CAST(@i AS VARCHAR(10)), 8),

'Bằng cấp ' + cast(@i as varchar(10)),

'GV' + RIGHT('00000000' + CAST(FLOOR(RAND() \* 1000) + 1 AS VARCHAR(10)), 8) ,

dateadd(year, -(@i % 5), getdate())

);

set @i = @i + 1;

end

end;

exec bangcap\_dump;

select \* from bangcap;

## Các module phục vụ các thao tác xử lý dữ liệu:

## Trigger:

**a. Khi thêm mới dữ liệu vào khóa học hãy đảm bảo thời gian bắt đầu nhỏ hơn thời gian kết thúc. Nếu thời gian kết thúc > thời gian bắt đầu thì in ra thông báo Không hợp lệ và hủy thao tác đã thực hiện:**

|  |  |
| --- | --- |
| Mục đích | Khi thêm dữ liệu vào khóa học, việc kiểm tra thời gian bắt đầu và kết thúc giúp tránh các lỗi không đáng có, đảm bảo thông tin được ghi nhận chính xác và có thể sử dụng trong các phép toán sau này. |
| Input | Không |
| OutPut | Không |
| Bảng kí sinh | Khoahoc |
| Loại | For |
| Sự kiện | Insert |
| Process | 1. Lấy ra thoigianbatdau,thoigiankethuc trong bảng inserted  ⇒ @thoigianbatdau, @thoigianketthuc  2. Nếu @thoigianbatdau > @thoigianketthuc  print ' Thời gian bắt không hợp lệ ' + rollback |
| Code | create trigger tg\_khoahoc  on Khoahoc  for insert  as  begin  declare @thoigianbatdau date  declare @thoigianketthuc date  select @thoigianbatdau=thoigianbatdau ,  @thoigianketthuc= thoigiankethuc  from inserted  if @thoigianbatdau > @thoigianketthuc  begin  print (N' Thời gian bắt không hợp lệ ')  rollback  end  end    insert into Khoahoc values ('0000002001','Thu ha','2024-10-11','2024-11-11','0000000002')  select \* from Khoahoc |

**b. Khi cập nhật hoặc sửa dữ liệu tên  giáo viên trong bảng Giáo viên, hãy đảm bảo tên giáo viên không ít hơn 5 kí tự:**

|  |  |
| --- | --- |
| Mục đích | Mục đích của việc đảm bảo tên giáo viên không ít hơn 5 kí tự để đảm bảo tính hợp lệ , duy trì chất lượng dữ liệu để nâng cao chất lượng quản lý  thông tin trong hệ thống |
| Bảng kí sinh | Giaovien |
| Loại | For |
| Sự kiện | Update |
| Process | 1. Lấy magiaovien , tengiaovien trong bảng inserted ⇒  @magiaovien,@tengiaovien  2. Nếu @Tengiaovien  <5  print ‘tên giáo viên không hợp lệ’ + rollback  3. Ngược lại update Giaovien ;dieukien magiaovien =@magiaovien |
| Code | create trigger tg\_giaovien  on Giaovien  for update  as  begin  declare @magiaovien varchar (10)  declare @tengiaovien varchar(30)  select @tengiaovien = tengiaovien  from inserted  if len(@tTengiaovien ) <5  begin  print (N' Tên giáo viên khong hop le ')  rollback  end  else  begin  update Giaovien  set Tengiaovien =@tengiaovien  where magiaovien=@magiaovien  end  end  update Giaovien  set tengiaovien ='Thu Hà '  where magiaovien ='GV00000001'  select \* from Giaovien |

**c. Khi thêm mới dữ liệu vào bảng Giaovien, hãy kiểm tra tên  giáo viên và số điện thoại.  Nếu Đã tồn tại trong bảng thì đưa ra thông báo ‘đã tồn tại giáo viên’ và hủy toàn bộ thao tác:**

|  |  |
| --- | --- |
| Mục đích | Mục đích của trigger này là để ngăn chặn dữ liệu trùng lặp trong bảng Giaovien giúp quản lý dữ liệu chất lượng và nâng cao hiệu quả làm việc |
| Bảng kí sinh | Giaovien |
| Loại | For |
| Sự kiện | Insert |
| Process | 1.Lấy tengiaovien , sdt  trong bảng inserted  ⇒ @tengiaovien ,@sdt  2. Nếu tên giáo viên và sdt >=1 đkien @tengiaovien =tengiaovien, @sdt = sdt thì print Đã tồn tại giáo viên + rollback |
| Code | create trigger tg\_themgiaovien  on Giaovien  for insert  as  begin  declare @tengiaovien varchar(20)  declare @sdt varchar(10)  select @tengiaovien =tengiaovien,  @sdt = sdt  from inserted  if ( select count(\*)  from Giaovien  where tengiaovien =@tengiaovien and  sdt = @sdt ) >= 1  begin  print ('Đã tồn tại giáo viên')  rollback  end  end  insert into Giaovien values ('GV00002002','Giáo viên 3','2004-01-01','Quang Tri ','thuhaaaheo@gmail.com','3920306656')  select \* from Giaovien |

**d. Khi thêm mới dữ liệu vào bảng taikhoan, hãy kiểm tra  mã giáo viên Nếu mã giáo viên  tồn tại trong bảng giaovien thì đưa ra thông báo ‘giao viên đã  tồn tại và hủy toàn bộ thao tác:**

|  |  |
| --- | --- |
| Mục đích | Mục đích module này để ngăn chặn việc tạo tài khoản với mã giáo viên đã tồn tại, giúp duy trì tính nhất quán và toàn vẹn của dữ liệu trong hệ thống. Thông báo “giáo viên tồn tại” cung cấp phản hồi ngay lập tức cho người dùng, giúp họ điều chỉnh thông tin kịp thời và hiệu quả. |
| Bảng kí sinh | Taikhoan |
| Loại | For |
| Sự kiện | Insert |
| Process | 1. Lấy magiaovien trong bảng inserted => @magiaovien  2. Nếu magiaovien tồn tại trong bảng => print ‘Mã giáo viên tồn tại’ + rollback |
| Code | create trigger Trupdata0  on taikhoan  for insert  as  begin  declare @magiaovien varchar(10)  select @magiaovien = Magiaovien from inserted  if exists (select 1 from Giaovien  where Magiaovien = @magiaovien)  begin  Print N'Mã giáo viên tồn tại';  rollback;  end  end  select \* from giaovien  insert into taikhoan values('TK00990004','taikhon2','matkhau1','GV00000003') |

**e. Khi cập nhật hoặc sửa dữ liệu số điện thoại trong bảng Giáo viên, hãy đảm bảo số điện thoại đủ 10 số. Nếu số điện thoại không phải 10 số đưa ra thông báo Số điện thoại không hợp lệ và hủy toàn bộ thao tác:**

|  |  |
| --- | --- |
| Mục đích | Mục đích của module này là đảm bảo rằng số điện thoại được cập nhật trong bảng Giáo viên phải có độ dài chính xác là 10 số. Điều này giúp duy trì tính chính xác và hợp lệ của thông tin liên lạc. |
| Bảng kí sinh | Giaovien |
| Loại | For |
| Sự kiện | Update |
| Process | 1. Lấy magiaovien ,sđt trong bảng inserted => @magiaovien,@sdt  2. nếu độ dài số  = 10, update giao vien ; Sdt=@sdt , điều kiện Magiaovien =@magiaovien  3. Ngược lại => in ra ‘SDT không hợp lệ ’ + rollback |
| Code | create trigger Tupdatetel12  on giaovien  for update  as  begin  declare @sdt char(10),@magiaovien char(10)  select @sdt = Sdt, @magiaovien = Magiaovien from inserted  if len(@sdt) = 10  begin  update giaovien  set Sdt = @sdt  where Magiaovien = @magiaovien  end  else  begin  print N'Số điện thoại không hợp lệ'  rollback  end  end  select \* from giaovien  update giaovien  set Sdt = '1234567890'  where Magiaovien = 'GV00000002' |

## Thủ tục

**a. Nếu biết mã giáo viên , hãy tính tổng số lần nghỉ của giáo viên trong năm:**

|  |  |
| --- | --- |
| Mục đích | Mục đích của module này là tính toán tổng số lần nghỉ của một giáo viên trong năm dựa trên mã giáo viên.Và thông tin này có thể hỗ trợ trong việc đưa ra quyết định về khen thưởng, xử lý kỷ luật hoặc lập kế hoạch giảng dạy |
| Input | Magiaovien |
| Output | Đưa ra số lần xin nghỉ của Giáo viên |
| Process | 1.Đếm số lượng phiếu xin nghỉ của giáo viên, Đk: Magiaovien = @magaovien và year(Thoigiannghi) =@year  2. Nếu số lượng phiếu >0 , Đưa ra số lần nghỉ   của giáo viên  3. Ngược lại số lần nghỉ =0; |
| Code | create proc Testtt (@magiaovien char(10),@year int, @solanxinnghi int output)  as  begin  declare @count int  select @count = count(\*) from phieuxinnghi  where Magiaovien = @magiaovien and  year(Thoigiannghi) = @year  if @count>0  begin  set @solanxinnghi = @count  return  end  else  begin  set @solanxinnghi = 0  return  end  end  Declare @b int;  exec Testtt 'GV00000001','2024' , @solanxinnghi = @b OUTPUT;  select @b |

**b. Thêm một bản ghi vào bảng khóa học nếu biết Mã khóa học , tên khóa học ,thời gian bắt đầu , thời gian kết thuc, mã giáo viên . Kiểm tra nếu thời gian bắt đầu lớn hơn hoặc bằng thời gian kết thúc thì báo Lỗi ngày bắt đầu phải nhỏ hơn ngày kết thúc. Ngược lại thì thêm khóa học mới vào bảng khóa học:**

|  |  |
| --- | --- |
| Mục đích | Khi thêm dữ liệu vào khóa học, việc kiểm tra thời gian bắt đầu và kết thúc giúp tránh các lỗi không đáng có, đảm bảo thông tin được ghi nhận chính xác và có thể sử dụng trong các phép toán sau này. |
| Input | makhoahoc, magiaovien, tenkhoahoc, thoigianbatdau, thoigianketthuc |
| Output | Thêm bản ghi thành công |
| Process | 1. nếu @thoigianbatdau>=@thoigianketthuc, in ra 'ngày bắt đầu khóa học phải nhỏ hơn ngày kết thúc.' 2. ngược lại, thêm khóa học mới vào bảng và thông báo 'thêm khóa học thành công.' |
| Code | create or alter proc themkhoahocmoi  @makhoahoc char(10),  @magiaovien char(10),  @tenkhoahoc varchar(50),  @thoigianbatdau date,  @thoigianketthuc date  as  begin  if @thoigianbatdau >= @thoigianketthuc  begin  print N'Lỗi! Ngày bắt đầu khóa học phải nhỏ hơn ngày kết thúc.';  end  else  begin  insert into khoahoc (Makhoahoc, Magiaovien, Tenkhoahoc, Thoigianbatdau, Thoigiankethuc)  values (@makhoahoc, @magiaovien, @tenkhoahoc, @thoigianbatdau, @thoigianketthuc);  print N'Thêm khóa học thành công.';  end  end  exec themkhoahocmoi 'KH80000001', 'GV00000001', 'tiếng anh B3', '2024-11-01', '2024-12-31';  select \* from khoahoc where makhoahoc = 'KH80000001';  exec themkhoahocmoi 'KH50000001', 'GV00000002', 'tieng anh bo tuc', '2024-10-30', '2024-10-25'; |

**c. Khi lớp học kết thúc , hãy xóa lớp học nếu biết được mã lớp học trong bảng Lớp học:**

|  |  |
| --- | --- |
| Mục đích | Khi một lớp học kết thúc , hãy xóa lớp học đó để tránh việc nhầm lẫn giữa các lớp học và quản lý các lớp học khác tốt hơn . |
| Input | Malophoc |
| Output | Lớp học đã được xóa thành công |
| Process | 1. khai báo @dem, đếm số lớp học điều kiện  Malophoc = @Malophoc 2. 2a.Neu @dem > 0, xóa  Lophoc điều kiện  Malophoc = @malophoc  và đưa ra thông báo ‘ Đã xóa lớp học thành công ‘   2b. nguoc lai, @dem =0, thông báo 'Lớp học không tồn tại.' |
| Code | create or alter proc XoaLopHoc  @Malophoc char(10)  as  begin  declare @dem int;  select @dem = count(\*) from LopHoc where Malophoc = @Malophoc;  if @dem > 0  begin  delete from LopHoc where Malophoc = @Malophoc;  print (N'Đã xóa lớp học thành công.')  end  else  begin  print (N'Lớp học không tồn tại.')  end  end  exec XoaLopHoc '0000000002'  select \* from Lophoc where Malophoc = '0000000002' |

**d. Thủ xóa tài khoản của giáo viên khi giáo viên nghỉ dạy . Biết được mã giáo viên:**

|  |  |
| --- | --- |
| Mục đích | Mục đích của module này là quản lý tài khoản giáo viên một cách hiệu quả, đảm bảo rằng các tài khoản không còn hoạt động được xóa khi giáo viên nghỉ dạy. |
| Input | Magiaovien |
| Output | Tài khoản giáo viên đã xóa |
| Process | **1.** khai báo biến @dem, đếm số lượng tài khoản , điều kiện  Magiaovien =@magiaovien  2a. Nếu @dem>0, Xóa tài khoản  điều kiện Magiaovien = @magiaovien  Hiển thị thông báo ‘đã xóa tài khoản thành công.’  2b. Ngược lại, @dem<0 thông báo 'tài khoản không tồn tại.' |
| Code | create or alter proc xoaTaiKhoan  @magiaovien char(10)  as  begin  declare @dem int;  select @dem = count(\*) from TaiKhoan where Magiaovien = @magiaovien;  if @dem > 0  begin  delete from TaiKhoan where Magiaovien = @magiaovien;  print N'đã xóa tài khoản giáo viên thành công.';  end  else  begin  print N'tài khoản giáo viên không tồn tại.';  end  end  exec xoaTaiKhoan 'GV00001234'  exec xoaTaiKhoan '0000000079'  select \* from TaiKhoan where Magiaovien = '0000000079' |

**e. Hiển thị danh sách  giáo viên gần nghỉ hưu. (Người gần nghỉ hưu sẽ có tuổi từ 50 trở lên):**

|  |  |
| --- | --- |
| Mục đích | Mục đích của module hiển thị danh sách giáo viên gần nghỉ hưu là để quản lý và theo dõi tình trạng nhân sự trong trung tâm . Bằng cách xác định những giáo viên từ 50 tuổi trở lên |
| Input | Ngày sinh |
| Output | Danh sách giáo viên |
| Process | 1. Kiểm tra  GV có số tuổi >50 2. Hiển thị danh sách giáo viên |
| Code | create proc Giaovien50tuoi  as  begin  select \*  from Giaovien  where datediff (year , NgaySinh, getdate ()) > 50;  end  exec Giaovien50tuoi |

**f. Tìm tên , số điện thoại , địa chỉ , email của giáo viên có hệ số lương cao nhất:**

|  |  |
| --- | --- |
| Mục đích | Mục đích của module này để xác định và hiển thị thông tin giáo viên có hệ số lương cao nhất nhằm đánh giá hiệu suất và công nhận |
| Input | Không |
| Output | Tên giáo viên,Số điện thoại,Địa chỉ ,Email của giáo viên, mã giáo viên,hệ số lương |
| Process | 1.  Kiểm tra tồn tại của bản ghi trong bảng Giaovien   nếu không có print “Không có”  2.Kiểm tra giáo viên có hệ số lương cao nhất và đưa ra thông báo Giáo viên có hệ số lương cao nhất |
| Code | create procedure TimGiaoVienHSLLonNhat  as  begin  if not exists (select 1 from Giaovien)  begin  print N'Không có';  return;  end  select TenGiaovien, Sdt, diaChi, Email ,hesoluong  from Giaovien  join Luong on giaovien.Magiaovien = luong.Magiaovien  where HesoLuong = (select max(HesoLuong) from Luong);  print N'giáo viên có hệ số lương cao nhất:';  end;  exec TimGiaoVienHSLLonNhat |

**g. Tìm tên lớp học , mã lớp học khi biết mã giáo viên:**

|  |  |
| --- | --- |
| Mục đích | Mục đích của module này để truy xuất thông tin về lớp học dựa trên mã giáo viên nhằm quản lý thông tin lớp học hiệu quả ,tránh việc giáo viên bị trùng lớp học |
| Input | Magiaovien |
| Output | TenLophoc, MaLophoc. |
| Process | 1a. Nếu không tồn tại lịch dạy , điều kiện Magiaovien=@Magiaovien ,print  “Giáo viên không có lớp học nào”  1b. Ngược lại tìm tên lớp học , mã lớp học điều kiện Magiaovien = @Magiaovien |
| Code | create procedure TimLopHocTheoMaGiaoVien  @Magiaovien varchar(10)  as  begin  if not exists (  select 1  from Lichday  where Magiaovien = @Magiaovien  )  begin  print N'Giáo viên không có lớp học nào';  return;  end  else  begin  select Lophoc.Tenlophoc, Lophoc.Malophoc  from Lophoc  where Magiaovien = @Magiaovien;  print N'Đã có lớp học của giáo viên';  end  end;  --–kiểm tra:  exec TimLopHocTheoMaGiaoVien 'GV00002000'; |