**Bài kiểm tra : Nhóm 1 – PHẠM PHƯỚC TIẾN**

**Dữ liệu**

USE [22T1020763]

GO

/\*\*\*\*\*\* Object: User [normal] Script Date: 12/9/2024 7:45:56 PM \*\*\*\*\*\*/

CREATE USER [normal] FOR LOGIN [normal\_user] WITH DEFAULT\_SCHEMA=[dbo]

GO

/\*\*\*\*\*\* Object: UserDefinedFunction [dbo].[func\_CountPassed] Script Date: 12/9/2024 7:45:56 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

create function [dbo].[func\_CountPassed](@examineeId int)

returns int

as

begin

declare @sl int

select @sl = COUNT(\*)

from Registration

where ExamResult >= 5 and ExamineeId = @examineeId

return @sl

end

GO

/\*\*\*\*\*\* Object: UserDefinedFunction [dbo].[func\_totalbydate] Script Date: 12/9/2024 7:45:56 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

create function [dbo].[func\_totalbydate](@from date,@to date)

returns @thongke table(ngay date,sl int)

as

begin

if @from > @to

begin

return

end

declare @i date = @from

while @i <= @to

begin

declare @tt int

select @tt = Count(\*)

from Registration

where RegisterTime = @i

insert into @thongke(ngay,sl)

values(@i,@tt)

set @i = DATEADD(day,1,@i)

end

return

end

GO

/\*\*\*\*\*\* Object: Table [dbo].[Certificate] Script Date: 12/9/2024 7:45:56 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Certificate](

[CertificateId] [int] NOT NULL,

[CertificateName] [nvarchar](100) NOT NULL,

[NumberOfRegister] [int] NOT NULL,

[NumberOfPass] [int] NOT NULL,

CONSTRAINT [PK\_Certificate] PRIMARY KEY CLUSTERED

(

[CertificateId] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Examinee] Script Date: 12/9/2024 7:45:56 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Examinee](

[ExamineeId] [int] NOT NULL,

[FirstName] [nvarchar](50) NOT NULL,

[LastName] [nvarchar](50) NOT NULL,

[BirthDate] [date] NOT NULL,

[Email] [nvarchar](50) NOT NULL,

[Address] [nvarchar](255) NOT NULL,

CONSTRAINT [PK\_Examinee] PRIMARY KEY CLUSTERED

(

[ExamineeId] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Registration] Script Date: 12/9/2024 7:45:56 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Registration](

[ExamineeId] [int] NOT NULL,

[CertificateId] [int] NOT NULL,

[RegisterTime] [date] NOT NULL,

[ExamResult] [int] NOT NULL,

CONSTRAINT [PK\_Registration] PRIMARY KEY CLUSTERED

(

[ExamineeId] ASC,

[CertificateId] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

INSERT [dbo].[Certificate] ([CertificateId], [CertificateName], [NumberOfRegister], [NumberOfPass]) VALUES (1, N'Tin Học', 5, 4)

INSERT [dbo].[Certificate] ([CertificateId], [CertificateName], [NumberOfRegister], [NumberOfPass]) VALUES (2, N'Ngoại Ngữ', 5, 2)

INSERT [dbo].[Examinee] ([ExamineeId], [FirstName], [LastName], [BirthDate], [Email], [Address]) VALUES (1, N'Phạm Phước', N'Tiến', CAST(N'1993-03-05' AS Date), N'phamtien@gmail.com', N'Huế')

INSERT [dbo].[Examinee] ([ExamineeId], [FirstName], [LastName], [BirthDate], [Email], [Address]) VALUES (2, N'Trần Việt', N'Lân', CAST(N'1994-04-06' AS Date), N'tranlan@gmail.com', N'Huế')

INSERT [dbo].[Examinee] ([ExamineeId], [FirstName], [LastName], [BirthDate], [Email], [Address]) VALUES (3, N'Lê Thanh', N'Thuyết', CAST(N'1995-01-05' AS Date), N'lethuyet@gmail.com', N'Hà Nội')

INSERT [dbo].[Examinee] ([ExamineeId], [FirstName], [LastName], [BirthDate], [Email], [Address]) VALUES (4, N'Nguyễn Hữu', N'Thiện', CAST(N'1992-02-11' AS Date), N'nguyenthien@gmail.com', N'Quảng Bình')

INSERT [dbo].[Examinee] ([ExamineeId], [FirstName], [LastName], [BirthDate], [Email], [Address]) VALUES (5, N'Hồng', N'Phương', CAST(N'1993-03-12' AS Date), N'hongphuong@gmail.com', N'Đà Nẵng')

INSERT [dbo].[Registration] ([ExamineeId], [CertificateId], [RegisterTime], [ExamResult]) VALUES (1, 1, CAST(N'2004-01-05' AS Date), 9)

INSERT [dbo].[Registration] ([ExamineeId], [CertificateId], [RegisterTime], [ExamResult]) VALUES (2, 1, CAST(N'2004-02-03' AS Date), 6)

INSERT [dbo].[Registration] ([ExamineeId], [CertificateId], [RegisterTime], [ExamResult]) VALUES (3, 2, CAST(N'2005-01-02' AS Date), 7)

INSERT [dbo].[Registration] ([ExamineeId], [CertificateId], [RegisterTime], [ExamResult]) VALUES (4, 2, CAST(N'2006-01-02' AS Date), 3)

INSERT [dbo].[Registration] ([ExamineeId], [CertificateId], [RegisterTime], [ExamResult]) VALUES (5, 1, CAST(N'2005-02-03' AS Date), 7)

INSERT [dbo].[Registration] ([ExamineeId], [CertificateId], [RegisterTime], [ExamResult]) VALUES (6, 2, CAST(N'2005-02-03' AS Date), 7)

INSERT [dbo].[Registration] ([ExamineeId], [CertificateId], [RegisterTime], [ExamResult]) VALUES (7, 1, CAST(N'2005-02-03' AS Date), 2)

INSERT [dbo].[Registration] ([ExamineeId], [CertificateId], [RegisterTime], [ExamResult]) VALUES (8, 1, CAST(N'2024-12-09' AS Date), 0)

INSERT [dbo].[Registration] ([ExamineeId], [CertificateId], [RegisterTime], [ExamResult]) VALUES (9, 2, CAST(N'2024-12-09' AS Date), 0)

INSERT [dbo].[Registration] ([ExamineeId], [CertificateId], [RegisterTime], [ExamResult]) VALUES (10, 2, CAST(N'2024-12-09' AS Date), 0)

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[proc\_CountRegisteringByDate] Script Date: 12/9/2024 7:45:56 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE proc [dbo].[proc\_CountRegisteringByDate](@from date,

@to date)

as

begin

if @from > @to

begin

return

end

declare @tb table(ngay date,

sl int)

declare @i date = @from

while @i <= @to

begin

declare @tt int

select @tt = Count(\*)

from Registration

where RegisterTime = @i

insert into @tb(ngay,sl)

values(@i,@tt)

set @i = DATEADD(day,1,@i)

end

select \*

from @tb

end

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[proc\_Registration\_Add] Script Date: 12/9/2024 7:45:56 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE proc [dbo].[proc\_Registration\_Add](@ExamineeId int,

@CertificateId int,

@Result nvarchar(255) output)

as

begin

if exists(select 1 from Registration where ExamineeId = @ExamineeId)

begin

set @Result = N'ExamineeId đã tồn tại!!!!'

return

end

if not exists(select 1 from Certificate where CertificateId = @CertificateId)

begin

set @Result = N'CertificateId không tồn tại!!!'

return

end

insert into Registration(ExamineeId,CertificateId,RegisterTime,ExamResult)

values(@ExamineeId,@CertificateId,GETDATE(),0)

set @Result = ''

end

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[proc\_SaveExamResult] Script Date: 12/9/2024 7:45:56 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE proc [dbo].[proc\_SaveExamResult](@ExamineeId int,

@CertificateId int,

@ExamResult int,

@Result nvarchar(255) output)

as

begin

if not exists(select 1 from Registration where ExamineeId = @ExamineeId)

begin

set @Result = N'ExamineeId không tồn tại!!!!'

return

end

if not exists(select 1 from Certificate where CertificateId = @CertificateId)

begin

set @Result = N'CertificateId không tồn tại!!!'

return

end

if not(@ExamResult between 0 and 10)

begin

set @Result = N'Điểm thi không thuộc trong khoảng 0 - 10'

return

end

set @Result = ''

update Registration

set ExamResult = @ExamResult

where ExamineeId = @ExamineeId

end

GO

**Câu 1:**

**Câu1a:**

create trigger trg\_Registration\_Insert on Registration

for insert

as

begin

update Certificate

set NumberOfRegister = NumberOfRegister + 1

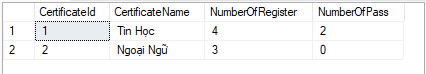
from Certificate ce inner join inserted i

on ce.CertificateId = i.CertificateId

end

insert into Registration(ExamineeId,CertificateId,RegisterTime,ExamResult)

values(7,1,N'2005-02-03',2)

****

**Câu1b:**

create trigger trg\_Registration\_UPDATE on Registration

for update

as

begin

update Certificate

set NumberOfPass = NumberOfPass + 1

from Certificate ce inner join inserted i

on ce.CertificateId = i.CertificateId

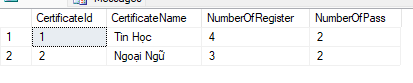
where i.ExamResult >= 5

end

update Registration

set ExamResult = 7

where ExamineeId = 6

****

**Câu 2:**

**Câu 2a:**

alter proc proc\_Registration\_Add(@ExamineeId int,

@CertificateId int,

@Result nvarchar(255) output)

as

begin

if exists(select 1 from Registration where ExamineeId = @ExamineeId)

begin

set @Result = N'ExamineeId đã tồn tại!!!!'

return

end

if not exists(select 1 from Certificate where CertificateId = @CertificateId)

begin

set @Result = N'CertificateId không tồn tại!!!'

return

end

insert into Registration(ExamineeId,CertificateId,RegisterTime,ExamResult)

values(@ExamineeId,@CertificateId,GETDATE(),0)

set @Result = 'aâ'

end

declare @resulta nvarchar(255)

exec proc\_Registration\_Add @examineeId = 9,@certificateId= 3,@result=@resulta output

print @resulta

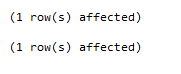
**Trường hợp1:**

****

**Trường hợp2:**

****

**Trường hợp3:**

****

**Câu 2b:**

alter proc proc\_SaveExamResult(@ExamineeId int,

@CertificateId int,

@ExamResult int,

@Result nvarchar(255) output)

as

begin

if not exists(select 1 from Registration where ExamineeId = @ExamineeId)

begin

set @Result = N'ExamineeId không tồn tại!!!!'

return

end

if not exists(select 1 from Certificate where CertificateId = @CertificateId)

begin

set @Result = N'CertificateId không tồn tại!!!'

return

end

if not(@ExamResult between 0 and 10)

begin

set @Result = N'Điểm thi không thuộc trong khoảng 0 - 10'

return

end

set @Result = ''

update Registration

set ExamResult = @ExamResult

where ExamineeId = @ExamineeId

end

declare @result nvarchar(255)

exec proc\_SaveExamResult @examineeid=1,@certificateid=1 ,@examresult=9 ,@result=@Result output

print @result

**Trường hợp1:**

****

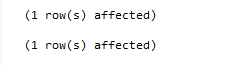
**Trường hợp2:**

****

**Trường hợp3:**

****

**Trường hợp4:**

****

**Câu2C:**

**Câu2D:**

alter proc proc\_CountRegisteringByDate(@from date,

@to date)

as

begin

if @from > @to

begin

return

end

declare @tb table(ngay date,

sl int)

declare @i date = @from

while @i <= @to

begin

declare @tt int

select @tt = Count(\*)

from Registration

where RegisterTime = @i

insert into @tb(ngay,sl)

values(@i,@tt)

set @i = DATEADD(day,1,@i)

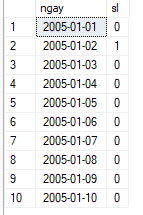
end

select \*

from @tb

end

exec proc\_CountRegisteringByDate @from = N'2005-01-01',@to = N'2005-01-10'

****

**Câu3:**

**Câu3a:**

create function func\_CountPassed(@examineeId int)

returns int

as

begin

declare @sl int

select @sl = COUNT(\*)

from Registration

where ExamResult >= 5 and ExamineeId = @examineeId

return @sl

end

select dbo.func\_countpassed(10)

****

**Câu3b:**

create function func\_totalbydate(@from date,@to date)

returns @thongke table(ngay date,sl int)

as

begin

if @from > @to

begin

return

end

declare @i date = @from

while @i <= @to

begin

declare @tt int

select @tt = Count(\*)

from Registration

where RegisterTime = @i

insert into @thongke(ngay,sl)

values(@i,@tt)

set @i = DATEADD(day,1,@i)

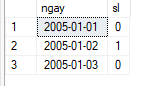
end

return

end

select \*

from dbo.func\_totalbydate(N'2005-01-01',N'2005-01-03')

****

**Câu4:**

- Tạo tài khoản có tên là **normal\_user** với mật khẩu là **123456**

create login normal\_user with password = '123456'

create user normal for login normal\_user

****

- Cấp phát cho tài khoản trên các quyền sau:

o Được phép thực hiện lệnh SELECT trên bảng **Certificate**

o Được phép sử dụng các thủ tục và hàm đã tạo ở trên

grant select

on Certificate

to normal

****

grant execute

on dbo.proc\_Registration\_Add

to normal

with grant option

grant execute

on dbo.proc\_SaveExamResult

to normal

with grant option

grant execute

on dbo.proc\_Examinee\_Select

to normal

with grant option

grant execute

on dbo.proc\_CountRegisteringByDate

to normal

with grant option

grant execute

on dbo.func\_CountPassed

to normal

with grant option

grant execute

on dbo.func\_TotalByDate

to normal

with grant option