

BÀI TẬP HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

Họ và tên: Nguyễn Tuấn Anh

MSSV: K215480106003

Class: K57KMT

Deadline: 3-5-2024

```
SQLQuery1.sql - DESKTOP-FNG6VDN.BTVN (DESKTOP-FNG6VDN\TuanAnh (61)) - Microsoft SQL Server Enterprise Edition - Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
-- Tạo cơ sở dữ liệu BTVN
CREATE DATABASE BTVN;
GO

-- Tạo bảng Môn Học
CREATE TABLE Môn_Học (
    id INT PRIMARY KEY,
    name NVARCHAR(100),
    số_tin_chi INT
);
GO

-- Tạo bảng Sinh viên
CREATE TABLE Sv (
    masv VARCHAR(13) PRIMARY KEY,
    name NVARCHAR(100),
    giới_tính BIT,
    lopsv VARCHAR(10)
);
GO

-- Tạo bảng Giáo viên
CREATE TABLE Gv (
    id INT PRIMARY KEY,
    name NVARCHAR(100),
    bộ_môn NVARCHAR(100)
);
GO

-- Tạo bảng Lớp Học Phần
CREATE TABLE LopHP (
    id INT PRIMARY KEY,
    idMon INT,
    hocKy INT,
    name NVARCHAR(100),
    idGv INT,
    FOREIGN KEY (idMon) REFERENCES Môn_Học(id),
    FOREIGN KEY (idGv) REFERENCES Gv(id)
);
GO

-- Tạo bảng Đăng ký môn học
CREATE TABLE Dkmh (
    id INT PRIMARY KEY,
    idLopHP INT,
    masv VARCHAR(13),
    điểmKt FLOAT,
    điểmThi FLOAT,
    FOREIGN KEY (idLopHP) REFERENCES LopHP(id),
    FOREIGN KEY (masv) REFERENCES Sv(masv)
);
GO

-- Tạo cơ sở dữ liệu
CREATE DATABASE baitapvenha;
GO

-- Tạo bảng Môn Học
CREATE TABLE Môn_Học (
    id INT PRIMARY KEY,
    name NVARCHAR(100),
    số_tin_chi INT
);
GO

-- Tạo bảng Sinh viên
CREATE TABLE Sv (
    masv VARCHAR(13) PRIMARY KEY,
    name NVARCHAR(100),
    giới_tính BIT,
    lopsv VARCHAR(10)
);
GO
```

- Tạo cơ sở dữ liệu BTVN
- Trong cơ sở dữ liệu gồm có các bảng: Môn học(Môn_học), Sinh Viên(sv), Giáo Viên(GV), Lớp học phần(LopHP), Đăng kí môn học(Dkmh)

```

SQLQuery1.sql - DESKTOP-FNG6VDN.BTVN (DESKTOP-FNG6VDN\TuanAnh (61)) - Microsoft SQL Server Enterprise Edition
File Edit View Query Project Tools Window Help
New Query Execute
SQLQuery1.sql - D:\VDN\TuanAnh (61)
-- Tạo bảng Sinh viên
CREATE TABLE Sv (
    masv VARCHAR(13) PRIMARY KEY,
    name NVARCHAR(100),
    giới_tính BIT,
    lopsv VARCHAR(10)
);
GO

-- Tạo bảng Giáo viên
CREATE TABLE Gv (
    id INT PRIMARY KEY,
    name NVARCHAR(100),
    bộ_môn NVARCHAR(100)
);
GO

-- Tạo bảng Lớp Học Phần
CREATE TABLE LopHP (
    id INT PRIMARY KEY,
    idMon INT,
    hocKy INT,
    name NVARCHAR(100),
    idGv INT,
    FOREIGN KEY (idMon) REFERENCES Môn_Học(id),
    FOREIGN KEY (idGv) REFERENCES Gv(id)
);
GO

-- Tạo bảng Đăng ký môn học
CREATE TABLE Dkmh (
    id INT PRIMARY KEY,
    idLopHP INT,
    masv VARCHAR(13),
    điểmKt FLOAT,
    điểmThi FLOAT,
    FOREIGN KEY (idLopHP) REFERENCES LopHP(id),
    FOREIGN KEY (masv) REFERENCES Sv(masv)
);
GO

-- Bài tập 1: Tính điểm trung bình 1 học kỳ của 1 sinh viên
CREATE FUNCTION fn_diem (@hk INT, @masv VARCHAR(13))
RETURNS FLOAT
AS
BEGIN
    DECLARE @diem FLOAT

    SELECT @diem = ((điểmKt * 0.4) + (điểmThi * 0.6))
    FROM Dkmh d
    JOIN LopHP l ON d.idLopHP = l.id
    WHERE l.hocKy = @hk AND d.masv = @masv

    RETURN @diem
END;
GO

-- Bài tập 2: Tính điểm trung bình học kỳ của 1 lớp sinh viên
CREATE FUNCTION fn_diem_lopsv (@hk INT, @lopsz VARCHAR(10))
RETURNS @kq TABLE (masv VARCHAR(13), name NVARCHAR(50), giới_tính BIT, điểm_tb FLOAT)
AS
BEGIN
    INSERT INTO @kq (masv, name, giới_tính, điểm_tb)
    SELECT sv.masv, sv.name, sv.giới_tính, AVG((dkmh.điểmKt * 0.4) + (dkmh.điểmThi * 0.6)) AS điểm_tb
    FROM Sv sv
    JOIN Dkmh dkmh ON sv.masv = dkmh.masv
    JOIN LopHP lop ON dkmh.idLopHP = lop.id
    WHERE lop.hocKy = @hk AND sv.lopsz = @lopsz
    GROUP BY sv.masv, sv.name, sv.giới_tính

    RETURN
END;
GO

```

```

-- Trả về giá trị điểm trung bình
RETURN @diem
END;
GO

```

- Bài tập 2: Tính điểm trung bình học kỳ của 1 lớp sinh viên

```

CREATE FUNCTION fn_diem_lopsv (@hk INT, @lopsz VARCHAR(10))

RETURNS @kq TABLE (masv VARCHAR(13), name NVARCHAR(50), giới_tính BIT, điểm_tb FLOAT)

```

- Bài tập 1: Tính điểm trung bình 1 học kỳ của một sinh viên

```

CREATE FUNCTION fn_diem (@hk INT,
@masv VARCHAR(13))
RETURNS FLOAT
AS
BEGIN
-- Khai báo biến để lưu điểm trung bình
DECLARE @diem FLOAT

```

```

-- Tính điểm trung bình của sinh viên
SELECT @diem = ((điểmKt * 0.4) + (điểmThi
* 0.6))
FROM Dkmh d
JOIN LopHP l ON d.idLopHP = l.id
WHERE l.hocKy = @hk AND d.masv =
@masv

```

AS

BEGIN

-- Tạo bảng kết quả để lưu trữ mã sinh viên, tên, giới tính và điểm trung bình

INSERT INTO @kq (masv, name, giới_tính, điểm_tb)

-- Tính điểm trung bình của sinh viên trong lớp

SELECT sv.masv, sv.name, sv.giới_tính, AVG((dkmh.điểmKt * 0.4) + (dkmh.điểmThi * 0.6)) AS điểm_tb

FROM Sv sv

-- Kết nối bảng Sv với bảng Dkmh để lấy thông tin điểm của sinh viên

JOIN Dkmh dkmh ON sv.masv = dkmh.masv

-- Kết nối bảng Dkmh với bảng LopHP để lấy thông tin về học kỳ

JOIN LopHP lop ON dkmh.idLopHP = lop.id

-- Lọc sinh viên theo học kỳ và lớp sinh viên

WHERE lop.hocky = @hk AND sv.lopsv = @lopsv

-- Nhóm kết quả theo mã sinh viên, tên và giới tính, tính điểm trung bình

GROUP BY sv.masv, sv.name, sv.giới_tính

-- Trả về kết quả

RETURN

END;

GO

```
-- Bài tập 3: Lấy danh mục môn học, lớp học phần và giáo viên dưới dạng JSON
CREATE PROCEDURE sp_danh_muc(@hk INT)
AS
BEGIN
    SELECT (
        SELECT id, name, số_tín_chi
        FROM Môn_Học
        FOR JSON PATH
    ) AS Mon_Hoc,
    (
        SELECT id, idMon, hocky, name, idGv
        FROM LopHP
        WHERE hocky = @hk
        FOR JSON PATH
    ) AS lophp,
    (
        SELECT id, name, bộ_môn
        FROM Gv
        WHERE id IN (SELECT DISTINCT idGv FROM LopHP WHERE hocky = @hk)
        FOR JSON PATH
    ) AS Giáo_viên
    FOR JSON PATH;
END;
GO

-- Bài tập 4: Lấy danh sách đăng ký lớp học phần dưới dạng JSON
CREATE PROCEDURE sp_danh_sach_dk @idLopHP INT
AS
BEGIN
    SELECT *
    FROM Dkmh
    WHERE idLopHP = @idLopHP
    FOR JSON PATH;
END;
GO
```

100 %
Messages
Commands completed successfully.

100 %
Query executed successfully. DESKTOP-FNG6VDN (16.0 RTM) DESKTOP-FNG6VDN\TuanAn... BTVN 00:00:00 0 rows

Ready Ln 194 Col 24 Ch 24 INS

-- Bài tập 3: Lấy danh mục môn học, lớp học phần và giáo viên dưới dạng JSON

```
CREATE PROCEDURE sp_danh_muc(@hk INT)
```

```
AS
```

```
BEGIN
```

```
-- Lấy danh sách môn học và chuyển thành JSON
```

```
SELECT (
```

```
    SELECT id, name, số_tín_chi
```

```
    FROM Môn_Học
```

```
    FOR JSON PATH
```

```
) AS Mon_Hoc,
```

```
-- Lấy danh sách lớp học phần và chuyển thành JSON
```

```
(
```

```
    SELECT id, idMon, hocky, name, idGv
```

```
    FROM LopHP
```

```
    WHERE hocky = @hk
```

```

        FOR JSON PATH
    ) AS lophp,
    -- Lấy danh sách giáo viên và chuyển thành JSON
    (
        SELECT id, name, bộ_môn
        FROM Gv
        WHERE id IN (SELECT DISTINCT idGv FROM LopHP WHERE hocky = @hk)
        FOR JSON PATH
    ) AS Giáo_viên
    -- Kết quả cuối cùng là JSON chứa danh sách môn học, lớp học phần và giáo viên
    FOR JSON PATH;
END;
GO

```

-- Bài tập 4: Lấy danh sách đăng ký lớp học phần dưới dạng JSON

```

CREATE PROCEDURE sp_danh_sach_dk @idLopHP INT
AS
BEGIN
    -- Lấy danh sách đăng ký lớp học phần và chuyển thành JSON
    SELECT *
    FROM Dkmh
    WHERE idLopHP = @idLopHP
    FOR JSON PATH;
END;
GO

```

Truy xuất lấy thông tin

-- Bài tập 1: Tính điểm trung bình 1 học kỳ của 1 sinh viên

```

DECLARE @diem_sv1 FLOAT
EXEC @diem_sv1 = fn_diem 1, 'SV123'
PRINT 'Điểm trung bình của sinh viên Phát trong học kỳ 1 là: ' + CAST(@diem_sv1 AS VARCHAR(10));

```

-- Bài tập 2: Tính điểm trung bình học kỳ của 1 lớp sinh viên

```
SELECT * FROM fn_diem_lopsv(1, 'A1');
```

-- Bài tập 3: Lấy danh mục môn học, lớp học phần và giáo viên dưới dạng JSON

```
EXEC sp_danh_muc 1;
```

-- Bài tập 4: Lấy danh sách đăng ký lớp học phần dưới dạng JSON

```
EXEC sp_danh_sach_dk 101;
```

-- Bài tập 2: Tính điểm trung bình học kỳ của 1 lớp sinh viên

```
SELECT * FROM fn_diem_lopsv(1, 'A1');
```

-- Bài tập 3: Lấy danh mục môn học, lớp học phần và giáo viên dưới dạng JSON

```
EXEC sp_danh_muc 1;
```

-- Bài tập 4: Lấy danh sách đăng ký lớp học phần dưới dạng JSON

```
EXEC sp_danh_sach_dk 101
```