

Chương 5

NGÔN NGỮ SQL

GIẢNG VIÊN: PHẠM THỊ XUÂN HIỀN

MỤC TIÊU CHƯƠNG

- Biết cách định nghĩa CSDL
- Biết thao tác (**tìm kiếm, thêm, xoá, sửa**) trên CSDL

GIỚI THIỆU SQL

- SQL: Structured Query Language
- SQL là ngôn ngữ chuẩn của nhiều HQT CSDL, gồm các câu lệnh định nghĩa dữ liệu, truy vấn và cập nhật dữ liệu
- SQL sơ khai được gọi là SEQUEL (Structured English Query Language) do IBM phát triển trong hệ thống System R, 1974-1976
- Gồm các phiên bản:
 - Chuẩn SQL-86 (SQL1) do ANSI (American National Standards Institute) và ISO (International Standards Organization)
 - Chuẩn SQL-92 (SQL2)
 - Chuẩn SQL-99 (SQL3)

PHÂN LOẠI

SQL gồm 3 nhóm câu lệnh:

- Nhóm định nghĩa dữ liệu (DDL - Data Definition Language): **CREATE, ALTER, DROP**
- Ngôn ngữ thao tác dữ liệu (DML - Data Manipulation Language): **SELECT, INSERT, UPDATE, DELETE**
- Nhóm điều khiển dữ liệu (DCL – Data Control Language commands): **DENY, GRANT, REVOKE => HỆ QTCSDL**

DDL – ĐỊNH NGHĨA DỮ LIỆU

Nhóm lệnh định nghĩa dữ liệu (Data Definition Language) gồm các lệnh:

- Tạo CSDL
- Tạo bảng
- Kiểu dữ liệu
- Khai báo ràng buộc dữ liệu

PHÂN LOẠI

SQL gồm 3 nhóm câu lệnh:

- Nhóm định nghĩa dữ liệu (DDL - Data Definition Language): **CREATE, ALTER, DROP**
- Ngôn ngữ thao tác dữ liệu (DML - Data Manipulation Language): **SELECT, INSERT, UPDATE, DELETE**
- Nhóm điều khiển dữ liệu (DCL – Data Control Language commands): **DENY, GRANT, REVOKE => HỆ QTCSDL**

DML – THAO TÁC DỮ LIỆU

- Truy vấn cơ giản
- Hàm kết hợp, gom nhóm
- Truy vấn lồng
- Truy vấn lồng nâng cao (phép chia) (đọc thêm)

TRUY VẤN CƠ BẢN

SQL vs ĐSQL

π	-----	SELECT <danh sách các cột>
\times	-----	FROM <danh sách các bảng>
σ	-----	WHERE <điều kiện>

```
SELECT GV.MAGV  
FROM GIAOVIEN GV, BOMON BM  
WHERE PHAI='Nam' AND GV.MABM = BM.MABM
```

⋮

$$\pi_{\text{MAGV}} (\sigma_{\text{PHAI}='Nam'} (\text{BOMON} \times \text{GIAOVIEN}))$$

TRUY VẤN CƠ BẢN

Biểu thức điều kiện trong mệnh đề WHERE

- Là biểu thức **Boolean**, sử dụng để chọn dòng (filtering rows)
- Biểu thức Boolean sử dụng
 - Phép toán so sánh dữ liệu số : **>, >=, <=, =, <>, between ... and**
 - Phép toán so sánh chuỗi : **LIKE** , nối chuỗi **+**, ký hiệu đại diện **%** ,
IS NULL, IS NOT NULL
 - Các phép toán so sánh tập hợp **IN, NOT IN**
 - **ANY, ALL , EXISTS, NOT EXISTS** dùng với subquery
 - Dùng toán tử **AND, OR** nối các biểu thức con
 - Toán tử phủ định **NOT : NOT LIKE, NOT between... and**
 - Sử dụng **các hàm** hỗ trợ bởi T-SQL để tính toán

TRUY VẤN CƠ BẢN

Ví dụ Biểu thức điều kiện trong mệnh đề **WHERE**

Biểu thức điều kiện trong WHERE	Công dụng
Ngaysinh like '1998-01-21'	Lọc row có Ngaysinh là 21-01-1998
Ngaysinh like '1999____-01'	Lọc row có Ngaysinh là ngày 1, năm 1999
year(ngaysinh) = 1999 AND day(ngaysinh) = 1	Lọc row có Ngaysinh là ngày 1, năm 1999 (sử dụng hàm Year() và day())
Ngaysinh IS NULL	Lọc row có Ngaysinh là NULL
Year(ngaysinh) <> 1999	Lọc row có Ngaysinh thỏa năm không là 1999 (sử dụng hàm Year())
Year(ngaysinh) != 1999	Lọc row có Ngaysinh thỏa năm không là 1999 (sử dụng hàm Year())
TenSV IN ('hoa', 'khanh')	Lọc row có TenSV là 'hoa' hoặc 'khanh'

PHÉP TOÁN TẬP HỢP TRONG SQL

Phép toán: Hội (**UNION**), Giao (**INTERSECT**), Trừ (**EXCEPT**).

NẾU chúng ta muốn **giữ lại các bộ trùng** thì sử dụng từ khoá **ALL** đi kèm.

SELECT <danh sách các cột> **FROM** <danh sách các bảng> **WHERE** <điều kiện>

UNION [ALL]

SELECT <danh sách các cột> **FROM** <danh sách các bảng> **WHERE** <điều kiện>

SELECT <danh sách các cột> **FROM** <danh sách các bảng> **WHERE** <điều kiện>

INTERSECT [ALL]

SELECT <danh sách các cột> **FROM** <danh sách các bảng> **WHERE** <điều kiện>

SELECT <danh sách các cột> **FROM** <danh sách các bảng> **WHERE** <điều kiện>

EXCEPT [ALL]

SELECT <danh sách các cột> **FROM** <danh sách các bảng> **WHERE** <điều kiện>

MỆNH ĐỀ SELECT

Lấy hết tất cả các cột

```
SELECT *  
FROM GIAOVIEN  
WHERE PHAI = 'NAM'
```

$\sigma_{\text{PHAI}='NAM'}(\text{GIAOVIEN})$

Results Messages

	MAGV	HOTEN	LUONG	PHAI	NGAYSINH	DIACHI	GVQLCM	MABM
1	001	Nguyễn Hoài An	2000	Nam	1973-02-15 00:00:00.000	25/3 Lạc Long Quân, Q.10, TP HCM	NULL	MMT
2	004	Trương Nam Sơn	2300	Nam	1959-06-20 00:00:00.000	215 Lí Thùơng Kiệt, TP Biên Hòa	NULL	VS
3	005	Lý Hoàng Hà	2500	Nam	1954-10-23 00:00:00.000	22/5 Nguyễn Xí, Q Bình Thạnh, TP HCM	NULL	VLĐT
4	007	Nguyễn Anh Trung	2100	Nam	1976-06-05 00:00:00.000	234 3/2, TP Biên Hòa	NULL	HPT
5	008	Trần Trung Hiếu	1800	Nam	1977-08-06 00:00:00.000	22/11 Lý Th?ong Ki?t, TP M? Tho	007	HPT
6	009	Trần Hoàng Nam	2000	Nam	1975-11-12 00:00:00.000	234 Trần Năo, An Phú, TP HCM	001	MMT
7	010	Phạm Nam Thanh	1500	Nam	1980-12-12 00:00:00.000	221 Hùng Vương, Q.5 TP HCM	007	HPT

MỆNH ĐỀ SELECT

Lấy một số cột

VD1: LẤY MAGV, HOTEN CỦA GIÁO VIÊN LÀ PHÁI NAM

$\pi_{\text{MAGV}, \text{HOTEN}} (\sigma_{\text{PHAI} = \text{'NAM'}} (\text{GIAOVIEN}))$

```
SELECT MAGV, HOTEN  
FROM GIAOVIEN  
WHERE PHAI = 'NAM'
```

Results		Messages
	MAGV	HOTEN
1	001	Nguyễn Hoài An
2	004	Trương Nam Sơn
3	005	Lý Hoàng Hà
4	007	Nguyễn Anh Trung
5	008	Trần Trung Hiếu
6	009	Trần Hoàng Nam
7	010	Phạm Nam Thanh

MỆNH ĐỀ SELECT

Tên bí danh

$\rho_{\text{THONG TIN}}(\pi_{\text{MAGV-HOTEN}}(\sigma_{\text{PHAI}=\text{'NAM'}}(\text{GIAOVIEN})))$

```
SELECT MAGV + ' - ' + HOTEN AS 'THÔNG TIN'  
FROM GIAOVIEN  
WHERE PHAI = 'NAM'
```

```
SELECT 'THÔNG TIN' = MAGV + ' - ' + HOTEN  
FROM GIAOVIEN  
WHERE PHAI = 'NAM'
```

Results		Messages
	THÔNG TIN	
1	001 - Nguyễn Hoài An	
2	004 - Trương Nam Sơn	
3	005 - Lý Hoàng Hà	
4	007 - Nguyễn Anh Trung	
5	008 - Trần Trung Hiếu	
6	009 - Trần Hoàng Nam	
7	010 - Phạm Nam Thanh	

MỆNH ĐỀ SELECT

Mở rộng

VD2: TÍNH LƯƠNG GIÁO VIÊN SAU KHI TĂNG 10%

```
SELECT MAGV, 'LƯƠNG 10%' = LUONG *1.1  
FROM GIAOVIEN
```

$\rho_{\text{MAGV,LUONG10\%}}(\pi_{\text{MAGV,LUONG*1.1}}(\text{GIAOVIEN}))$

Results		Messages
	MAGV	LƯƠNG 10%
1	001	2200
2	002	2750

MỆNH ĐỀ SELECT

Mở rộng

TÌM 5% GIÁO VIÊN ĐẦU TIÊN

```
SELECT * FROM GIAOVIEN --11 NGƯỜI
```

```
SELECT TOP 10 PERCENT * FROM GIAOVIEN
```


MỆNH ĐỀ SELECT

VD3: TÌM GIÁO VIÊN CÓ NGƯỜI THÂN

Loại bỏ các dòng trùng nhau

```
SELECT MAGV  
FROM NGUOITHAN
```

Results		Messages
	MAGV	▼
1	001	
2	001	
3	003	
4	003	
5	007	
6	007	
7	008	
8	009	
9	010	

```
SELECT DISTINCT(MAGV)  
FROM NGUOITHAN
```

Results		Messages
	MAGV	▼
1	001	
2	003	
3	007	
4	008	
5	009	
6	010	

MỆNH ĐỀ FROM

KHÔNG SỬ DỤNG MỆNH ĐỀ WHERE

```
SELECT HOTEN, TENBM  
FROM BOMON, GIAOVIEN
```

WHERE true (luôn luôn đúng),
Giống phép tích (x) lấy hết tất cả
các bộ

	HOTEN	TENBM
1	Nguyễn Hoài An	Công ngh? tri th?c
2	Trần Trà Hương	Công ngh? tri th?c
3	Nguyễn Ngọc Ánh	Công ngh? tri th?c
4	Trương Nam Sơn	Công ngh? tri th?c
5	Lý Hoàng Hà	Công ngh? tri th?c
6	Trần Bạch Tuyết	Công ngh? tri th?c
7	Nguyễn Anh Trung	Công ngh? tri th?c
8	Trần Trung Hiếu	Công ngh? tri th?c
9	Trần Hoàng Nam	Công ngh? tri th?c
10	Phạm Nam Thanh	Công ngh? tri th?c
11	Nguyễns Thành Chí Tâm	Công ngh? tri th?c
12	Nguyễn Hoài An	Hóa H?u Co

Ln 38, Col 14 Spaces: 4 UTF-8 LF SQL 110 rows

MỆNH ĐỀ WHERE

VD4: CHO BIẾT MAGV, HOTEN CỦA GIÁO VIÊN THUỘC BỘ MÔN HỆ THỐNG THÔNG TIN

```
SELECT GV.MAGV, GV.HOTEN
FROM BOMON BM inner join GIAOVIEN GV
      on BM.MABM = GV.MABM
WHERE BM.TENBM = N'Hệ thống thông tin'
```

```
SELECT GV.MAGV, GV.HOTEN
FROM BOMON BM, GIAOVIEN GV
WHERE BM.MABM = GV.MABM
      AND BM.TENBM = N'Hệ thống thông tin'
```

Biểu thức luận lý

TRUE AND TRUE: CÙNG ĐÚNG THOẢ ĐIỀU KIỆN

$\pi_{MAGV, HOTEN} (\sigma_{TENBM='Hệ thống thông tin'} (BOMON \bowtie GIAOVIEN))$

MỆNH ĐỀ WHERE

Độ ưu tiên

```
SELECT GV.MAGV, GV.HOTEN, BM.TENBM
FROM BOMON BM, GIAOVIEN GV
WHERE (BM.TENBM = N'Hệ thống thông tin' OR
       BM.TENBM = N'Mạng máy tính')
AND BM.MABM = GV.MABM
```

Độ ưu tiên mặc định của phép toán: **từ trái qua phải**

Nên sử dụng dấu ngoặc thể hiện tường minh sự ưu tiên của các phép toán

Results		Messages	
	MAGV	HOTEN	TENBM
1	001	Nguyễn Hoài An	Mạng máy tính
2	002	Trần Trà Hương	Hệ thống thông tin
3	003	Nguyễn Ngọc Ánh	Hệ thống thông tin
4	009	Trần Hoàng Nam	Mạng máy tính

MỆNH ĐỀ WHERE

VD5: TÌM GIÁO VIÊN CÓ LƯƠNG NẪM TRONG KHOẢNG 2000 ĐẾN 2300

BETWEEN

```
SELECT MAGV, HOTEN, LUONG
FROM GIAOVIEN
WHERE LUONG >= 2000 AND LUONG <= 2300
```

```
SELECT MAGV, HOTEN, LUONG
FROM GIAOVIEN
WHERE LUONG BETWEEN 2000 AND 2300
```

	MAGV	HOTEN	LUONG
1	001	Nguyễn Hoài An	2000
2	003	Nguyễn Ngọc Ánh	2200
3	004	Trương Nam Sơn	2300
4	007	Nguyễn Anh Trung	2100
5	009	Trần Hoàng Nam	2000

MỆNH ĐỀ WHERE

VD6: TÌM GIÁO VIÊN CÓ LƯƠNG NẪM TRONG NGOÀI KHOẢNG 2000 ĐẾN 2300

NOT BETWEEN

```
SELECT MAGV, HOTEN, LUONG
FROM GIAOVIEN
WHERE LUONG < 2000 OR LUONG > 2300
```

```
SELECT MAGV, HOTEN, LUONG
FROM GIAOVIEN
WHERE LUONG NOT BETWEEN 2000 AND 2300
```

	MAGV	HOTEN	LUONG
1	002	Trần Trà Hương	2500
2	005	Lý Hoàng Hà	2500
3	006	Trần Bạch Tuyết	1500
4	008	Trần Trung Hiếu	1800
5	010	Phạm Nam Thanh	1500

MỆNH ĐỀ WHERE

VD7: TÌM GIÁO VIÊN CÓ HỌ TÊN BẮT ĐẦU BẰNG CHỮ N

LIKE

```
SELECT MAGV, HOTEN  
FROM GIAOVIEN  
WHERE HOTEN LIKE 'N%'
```

CHUỖI BẤT KỲ

Results Messages

	MAGV	HOTEN
1	001	Nguyễn Hoài An
2	003	Nguyễn Ngọc Ánh
3	007	Nguyễn Anh Trung

MỆNH ĐỀ WHERE

NOT LIKE

```
SELECT MAGV, HOTEN  
FROM GIAOVIEN  
WHERE HOTEN LIKE 'N%'
```

↓ PHỦ ĐỊNH

```
SELECT MAGV, HOTEN  
FROM GIAOVIEN  
WHERE HOTEN NOT LIKE 'N%'
```

	MAGV	HOTEN
1	001	Nguyễn Hoài An
2	003	Nguyễn Ngọc Ánh
3	007	Nguyễn Anh Trung

	MAGV	HOTEN
1	002	Trần Trà Hương
2	004	Trương Nam Sơn
3	005	Lý Hoàng Hà
4	006	Trần Bạch Tuyết
5	008	Trần Trung Hiếu
6	009	Trần Hoàng Nam
7	010	Phạm Nam Thanh

MỆNH ĐỀ WHERE

VD8: TÌM GIÁO VIÊN TRONG KHOẢNG
1973 ĐẾN 1976

NGÀY GIỜ

SET DATEFORMAT YMD

```
SELECT MAGV, HOTEN, NGAYSINH  
FROM GIAOVIEN  
WHERE NGAYSINH BETWEEN '1973-01-01' AND '1976-01-01'
```

	MAGV	HOTEN	NGAYSINH
1	001	Nguyễn Hoài An	1973-02-15 00:00:00.000
2	003	Nguyễn Ngọc Ánh	1975-05-11 00:00:00.000
3	009	Trần Hoàng Nam	1975-11-12 00:00:00.000

MỆNH ĐỀ WHERE

VD9: TÌM GIÁO VIÊN KHÔNG CÓ GVQLCM

NULL

```
SELECT MAGV, GVQLCM  
FROM GIAOVIEN  
WHERE GVQLCM IS NULL
```

NOT NULL

```
SELECT MAGV, GVQLCM  
FROM GIAOVIEN  
WHERE GVQLCM IS NULL
```

	MAGV	GVQLCM
1	001	NULL
2	002	NULL
3	004	NULL
4	005	NULL
5	007	NULL
6	011	NULL

	MAGV	GVQLCM
1	003	002
2	006	004
3	008	007
4	009	001
5	010	007

MỆNH ĐỀ WHERE

VD10: TÌM GIÁO VIÊN THAM GIA ĐỀ TÀI 2 HOẶC 6

```
SELECT *  
FROM THAMGIADETAI  
WHERE MADT = '002' OR  
      MADT = '006'
```

```
SELECT * FROM THAMGIADETAI WHERE  
MADT = '002'  
UNION  
SELECT * FROM THAMGIADETAI WHERE  
MADT = '006'
```

$\sigma_{MADT='002' \vee MADT='006'}(THAMGIADETAI)$

$\sigma_{MADT='002'}(THAMGIADETAI) \cup \sigma_{MADT='006'}(THAMGIADETAI)$

	MAGV	MADT	STT	PHUCAP	KETQUA
1	001	002	1	0	NULL
2	001	002	2	2	NULL
3	003	002	2	0	NULL
4	004	006	2	1	ĐẠT
5	006	006	2	1,5	ĐẠT
6	009	002	3	0,5	NULL
7	009	002	4	1,5	NULL

MỆNH ĐỀ WHERE

VD10: TÌM GIÁO VIÊN THAM GIA ĐỀ TÀI 2 VÀ 6

```
SELECT *  
FROM THAMGIADETAI  
WHERE MADT = '002'  
AND MADT = '006'
```

```
SELECT * FROM THAMGIADETAI WHERE  
MADT = '002'  
INTERSECT  
SELECT * FROM THAMGIADETAI WHERE  
MADT = '006'
```

$\sigma_{MADT='002' \wedge MADT='006'}(THAMGIADETAI)$

$\sigma_{MADT='002'}(THAMGIADETAI) \cap \sigma_{MADT='006'}(THAMGIADETAI)$

MỆNH ĐỀ WHERE

VD11: TÌM GIÁO VIÊN SINH NĂM 1975

```
SELECT MAGV, HOTEN, NGAYSINH  
FROM GIAOVIEN  
WHERE YEAR(NGAYSINH)=1975
```

	MAGV	HOTEN	NGAYSINH
1	003	Nguyễn Ngọc Ánh	1975-05-11 00:00:00.000
2	009	Trần Hoàng Nam	1975-11-12 00:00:00.000

VD12: TÍNH TUỔI CỦA CÁC GIÁO VIÊN

```
SELECT MAGV, TUOI = YEAR(GETDATE())-YEAR(NGAYSINH)  
FROM GIAOVIEN
```

	MAGV	TUOI
1	001	50
2	002	63
3	003	48
4	004	64
5	005	69
6	006	43
7	007	47
8	008	46
9	009	48
10	010	43
11	011	NULL

MỆNH ĐỀ WHERE

VD13: TÌM HỌ TÊN CỦA NHỮNG GIÁO VIÊN ĐƯỢC TRẦN TRÀ HƯƠNG QUẢN LÝ CHUYÊN MÔN

```
SELECT GV2.*  
FROM GIAOVIEN GV1, GIAOVIEN GV2  
WHERE GV1.HOTEN= N'Trần Trà Hương' AND GV2.GVQLCM = GV1.MAGV
```

TRUY VẤN CƠ BẢN (2 BẢNG TRỞ LÊN)

Select a.*, b.*

From SINHVIEN a INNER JOIN NGANH b ON a.MaNganh = b.MaNganh

MaSV	HoSV	TenSV	Lop	Manganh
201001	Huỳnh Nhất	Luận	C10CK1	CS
201002	Nguyễn Anh	Tuấn	C55CK1	CS
201009	Lê Hồng	Khải	CS01TT	DS
201010	Phan Nhật Duy	Khanh	CS01TT	DS

MaNganh	TenNganh	SoTC
CS	Computer science	150
SE	Software Engineering	220
IS	Information System	190
IT	Information Technology	200

Inner join



Ứng với mỗi record của bảng bên trái (a), thực hiện phép **kết bảng** trên cột **MaNganh** với mỗi record của bảng bên phải (b)

MaSV	HoSV	TenSV	Lop	Manganh	MaNganh	TenNganh	SoTC
201001	Huỳnh Nhất	Luận	C10CK1	CS	CS	Computer science	150
201002	Nguyễn Anh	Tuấn	C55CK1	CS	CS	Computer science	150

TRUY VẤN CƠ BẢN (2 BẢNG TRỞ LÊN)

Select MaSV, TenSV, TenNganh

From SINHVIEN a **INNER JOIN** NGANH b **ON** a.MaNganh = b.MaNganh

Cách viết khác

Select MaSV, TenSV, TenNganh
From **SINHVIEN a** , **NGANH b**
Where **a.MaNganh = b.MaNganh**

MaSV	TenSV	TenNganh
201001	Luận	Computer science
201002	Tuấn	Computer science

Select MaSV, TenSV, TenNganh

From SINHVIEN a **INNER JOIN** NGANH b **ON** a.MaNganh = b.MaNganh

Where **Lop like '%55%'**

MaSV	TenSV	TenNganh
201002	Tuấn	Computer science

Select MaSV, TenSV, TenNganh

From SINHVIEN a **LEFT JOIN** NGANH b **ON** a.MaNganh = b.MaNganh

MaSV	TenSV	TenNganh
201001	Luận	Computer science
201002	Tuấn	Computer science
201009	Khải	
201010	Khanh	

Select MaSV, TenSV, TenNganh

From SINHVIEN a **RIGHT JOIN** NGANH b **ON** a.MaNganh = b.MaNganh

MaSV	TenSV	TenNganh
201001	Luận	Computer science
201002	Tuấn	Computer science
		Software Engineering
		Information System
		Information Technology

45

TRUY VẤN CƠ BẢN (2 BẢNG TRỞ LÊN)

-- KẾT NỘI --

```
SELECT GV.*, BM.* FROM GIAOVIEN GV INNER JOIN BOMON BM ON  
GV.MABM = BM.MABM
```

-- KẾT NGOẠI --

```
SELECT GV.*, BM.* FROM GIAOVIEN GV LEFT JOIN BOMON BM ON  
GV.MABM = BM.MABM
```

```
SELECT GV.*, BM.* FROM GIAOVIEN GV RIGHT JOIN BOMON BM ON  
GV.MABM = BM.MABM
```

```
SELECT GV.*, BM.* FROM GIAOVIEN GV FULL OUTER JOIN BOMON BM  
ON GV.MABM = BM.MABM
```

TRUY VẤN CƠ BẢN (2 BẢNG TRỞ LÊN)

-- KẾT NỘI --

```
SELECT GV.*, BM.* FROM GIAOVIEN GV INNER JOIN BOMON BM ON  
GV.MABM = BM.MABM
```

-- KẾT NGOẠI --

```
SELECT GV.*, BM.* FROM GIAOVIEN GV LEFT JOIN BOMON BM ON  
GV.MABM = BM.MABM
```

```
SELECT GV.*, BM.* FROM GIAOVIEN GV RIGHT JOIN BOMON BM ON  
GV.MABM = BM.MABM
```

```
SELECT GV.*, BM.* FROM GIAOVIEN GV FULL OUTER JOIN BOMON BM  
ON GV.MABM = BM.MABM
```

TRUY VẤN CƠ BẢN (2 BẢNG TRỞ LÊN)

-- TÌM MÃ GIÁO VIÊN CÓ NGƯỜI THÂN

```
SELECT MAGV  
FROM GIAOVIEN  
INTERSECT  
SELECT MAGV  
FROM NGUOITHAN
```

$\pi_{\text{MAGV}}(\text{GIAOVIEN}) \cap \pi_{\text{MAGV}}(\text{NGUOITHAN})$

--VD22: TÌM GIÁO VIÊN VỪA LÀ TRƯỞNG BM VỪA CHỦ NHIỆM ĐỀ TÀI

```
SELECT GVCNDT
FROM DETAI
INTERSECT
SELECT TRUONGBM
FROM BOMON
```

```
SELECT DT.GVCNDT
FROM BOMON BM, DETAI DT
WHERE BM.TRUONGBM = DT.GVCNDT
```

```
SELECT DT.GVCNDT
FROM BOMON BM INNER JOIN DETAI DT ON BM.TRUONGBM = DT.GVCNDT
```

TRUY VẤN CƠ BẢN (2 BẢNG TRỞ LÊN)

--VD14: TÌM MÃ GIÁO VIÊN KHÔNG CÓ NGƯỜI THÂN

```
SELECT MAGV  
FROM GIAOVIEN  
EXCEPT  
SELECT MAGV  
FROM NGUOITHAN
```

$\pi_{\text{MAGV}}(\text{GIAOVIEN}) - \pi_{\text{MAGV}}(\text{NGUOITHAN})$

--VD23: TÌM GIÁO VIÊN KHÔNG THAM GIA BẤT CỨ ĐỀ TÀI NÀO

MỆNH ĐỀ ORDER BY

Dùng để hiển thị kết quả câu truy vấn theo một thứ tự nào đó

SELECT <danh sách các cột>

FROM <danh sách các bảng>

WHERE <điều kiện>

GROUP BY <danh sách các cột gom nhóm>

HAVING <điều kiện>

ORDER BY <danh sách các cột> ASC | DESC

ASC: TĂNG (mặc định)

DESC: GIẢM

Sử dụng các hàm thống kê: **SUM, AVG, MIN, MAX, COUNT**

VD15: TÍNH SỐ LƯỢNG GIÁO VIÊN TOÀN TRƯỜNG

```
SELECT SLGV = COUNT(*)  
FROM GIAOVIEN
```

	SLGV
1	11

$$\rho_{\text{SLGV}}(\mathfrak{I}_{\text{count}(*)}(\text{GIAOVIEN}))$$

VD16: TÍNH SỐ LƯỢNG GIÁO VIÊN THUỘC BỘ MÔN HTTT

```
SELECT SLGV = COUNT(*)  
FROM GIAOVIEN  
WHERE MABM = 'HTTT'  
GROUP BY MABM
```

	SLGV
1	2

-- c1: chọn gv htttp sau đó đếm số lượng
 $\text{GVHTTT} \leftarrow \sigma_{\text{MABM}='HTTT'}(\text{GIAOVIEN})$
 $\text{KQ}(\text{SL}) \leftarrow \mathfrak{I}_{\text{COUNT}(\text{MAGV})}(\text{GVHTTT})$

--c2: gom nhóm MaBM rồi đếm MaGV
 $\text{KQ}(\text{MaBM}, \text{SL}) \leftarrow \mathfrak{I}_{\text{COUNT}(\text{MAGV})}(\text{GVHTTT})$

--VD18: CHO BIẾT SỐ LƯỢNG GIÁO VIÊN ĐÃ THAM GIA VÀO ĐỀ TÀI 'Ứng dụng hóa học xanh'

--VD19: CHO BIẾT TÊN NHỮNG GIÁO VIÊN ĐÃ THAM GIA TỪ 2 ĐỀ TÀI TRỞ LÊN

```
SELECT GV.HOTEN, COUNT(DISTINCT TGDT.MADT) as SL
FROM GIAOVIEN AS GV, THAMGIADETAI AS TGDT
WHERE GV.MAGV = TGDT.MAGV
GROUP BY GV.HOTEN
HAVING COUNT(DISTINCT TGDT.MADT)>1
```

	HOTEN	SL
1	Nguyễn Ngọc Ánh	2

$R1(\text{HoTen}, \text{SL}) \leftarrow \text{HOTEN} \bowtie \text{COUNT}(\text{MADT})(\text{GIAOVIEN} \bowtie \text{THAMGIADETAI})$

$R2(\text{HoTen}, \text{SL}) \leftarrow \sigma_{\text{SL} > 1}(R1)$

--VD20: CHO BIẾT MỨC LƯƠNG CAO NHẤT CỦA GIÁO VIÊN

```
SELECT MAX(LUONG)
FROM GIAOVIEN
```

--VD21: cho biết MAGV có họ tên bắt đầu bằng 'Nguyễn' và lương trên 20000 hoặc giáo viên là trưởng bộ môn có ngày nhận chức sau năm 1975

```
SELECT MAGV
FROM GIAOVIEN
WHERE HOTEN LIKE N'Nguyễn' AND LUONG > 20000
UNION
SELECT TRUONGBM
FROM BOMON
WHERE YEAR(NGAYNHANCHUC) >= 1975
```

-- VD24: CHO BIẾT MADT, STT CỦA CÔNG VIỆC, TÊN CÔNG VIỆC, KẾT QUẢ SẮP THEO MADT TĂNG, STT CÔNG VIỆC GIẢM

```
SELECT MADT, SOTT, TENCV
FROM CONGVIEC
ORDER BY MADT ASC, SOTT DESC
```

	MADT	SOTT	TENCV
1	001	5	Cài đặt thử nghiệm
2	001	4	Thiết kế hệ thống
3	001	3	Phân tích hệ thống
4	001	2	Xác định yêu cầu
5	001	1	Khởi tạo và Lập kế hoạch
6	002	5	Cài đặt thử nghiệm
7	002	4	Thiết kế hệ thống
8	002	3	Phân tích hệ thống
9	002	2	Xác định yêu cầu
10	002	1	Khởi tạo và Lập kế hoạch
11	006	2	Nuôi cấy
12	006	1	Lấy mẫu

TRUY VẤN LỒNG

Câu truy vấn cha (Outer Query)

SELECT <danh sách các cột>
FROM <danh sách các bảng>
WHERE <so sánh tập hợp> (

Câu truy vấn con (SubQuery)

SELECT <danh sách các cột>
FROM <danh sách các bảng>
WHERE <điều kiện>)

TRUY VẤN LỒNG

- Các câu lệnh SELECT có thể lồng nhau ở nhiều mức
- Câu truy vấn con thường trả về một tập các giá trị
- Các câu truy vấn con trong cùng một mệnh đề WHERE được kết hợp bằng phép nối logic
- Mệnh đề WHERE của câu truy vấn cha:
 - <biểu thức><so sánh tập hợp><Truy vấn cha>
 - So sánh tập hợp thường đi với toán tử
 - IN, NOT IN
 - ALL
 - ANY HOẶC SOME
 - Kiểm tra sự tồn tại
 - EXISTS
 - NOT EXISTS

TRUY VẤN LỒNG

Có 2 loại truy vấn lồng

- **LỒNG PHÂN CẤP**

- Mệnh đề **WHERE** câu truy vấn con không tham chiếu đến thuộc tính của các quan hệ trong mệnh đề FROM ở truy vấn cha
- Khi thực hiện, truy vấn con **thực hiện trước**

- **LỒNG TƯƠNG QUAN**

- Mệnh đề WHERE của truy vấn con tham chiếu **ít nhất một thuộc tính** của các quan hệ trong mệnh đề FROM ở truy vấn cha
- Khi thực hiện, truy vấn con sẽ được thực hiện **nhiều lần**, mỗi lần tương ứng với một bộ ở truy vấn cha

LỒNG PHÂN CẤP

--VD1: TÌM GIÁO VIÊN BỘ MÔN HỆ THỐNG THÔNG TIN

```
SELECT *  
FROM GIAOVIEN GV, BOMON BM  
WHERE GV.MABM= BM.MABM AND BM.TENBM = N'Hệ thống thông tin'
```

```
SELECT *  
FROM GIAOVIEN  
WHERE MABM IN (SELECT MABM  
                FROM BOMON  
                WHERE TENBM = N'Hệ thống thông tin')
```

	MAGV	HOTEN	Results grid	PHAI	NGAYSINH	DIACHI	GVQLCM	MABM
1	002	Trần Trà Hương	2500	Nữ	1960-06-20 00:00:00.000	125 Trần Hưng Đạo, Q1, TP HCM	NULL	HTTT
2	003	Nguyễn Ngọc Ánh	2200	Nữ	1975-05-11 00:00:00.000	12/21 Võ văn Kiệt, Thủ Đức, TP HCM	002	HTTT

LỒNG PHÂN CẤP

--VD2: TÌM GIÁO VIÊN LÀ TRƯỞNG BỘ MÔN

```
SELECT *  
FROM GIAOVIEN GV, BOMON BM  
WHERE GV.MAGV= BM.TRUONGBM
```

```
SELECT *  
FROM GIAOVIEN  
WHERE MAGV IN (SELECT TRUONGBM  
                FROM BOMON )
```

Mệnh đề WHERE của truy vấn con không tham chiếu đến thuộc tính của các quan hệ trong mệnh đề FROM của lớp cha

	MAGV	HOTEN	LUONG	PHAI	NGAYSINH	DIACHI	GVQLCM	MABM
1	001	Nguyễn Hoài An	2000	Nam	1973-02-15 00:00:00.000	25/3 Lạc Long Quân, Q.10, TP HCM	NULL	MMT
2	002	Trần Trà Hương	2500	Nữ	1960-06-20 00:00:00.000	125 Trần Hưng Đạo, Q1, TP HCM	NULL	HTTT
3	004	Trương Nam Sơn	2300	Nam	1959-06-20 00:00:00.000	215 Lí Thường Kiệt, TP Biên Hòa	NULL	VS
4	005	Lý Hoàng Hà	2500	Nam	1954-10-23 00:00:00.000	22/5 Nguyễn Xí, Q Bình Thạnh, TP HCM	NULL	VLDT
5	007	Nguyễn Anh Trung	2100	Nam	1976-06-05 00:00:00.000	234 3/2, TP Biên Hòa	NULL	HPT

LỒNG PHÂN CẤP

--VD3: cho biết MAGV, HOTEN có họ tên bắt đầu bằng 'Nguyễn' và lương trên 20000 hoặc giáo viên là trưởng bộ môn có ngày nhận chức sau năm 1975

```
SELECT MAGV, HOTEN
FROM GIAOVIEN
WHERE MAGV IN (SELECT MAGV
                FROM GIAOVIEN
                WHERE HOTEN LIKE N'Nguyễn' AND LUONG > 20000)
OR
MAGV IN (SELECT TRUONGBM
         FROM BOMON
         WHERE YEAR(NGAYNHANCHUC) >= 1975 )
```


LỒNG PHÂN CẤP

--VD4: TÌM NHỮNG GIÁO VIÊN KHÔNG THAM GIA BẤT CỨ ĐỀ TÀI NÀO

```
SELECT MAGV, HOTEN  
FROM GIAOVIEN  
WHERE MAGV NOT IN (SELECT MAGV  
                    FROM THAMGIADETAI)
```

```
SELECT MAGV, HOTEN  
FROM GIAOVIEN  
WHERE MAGV <> ALL(SELECT MAGV  
                  FROM THAMGIADETAI)
```

	MAGV	HOTEN
1	005	Lý Hoàng Hà
2	007	Nguyễn Anh Trung
3	008	Trần Trung Hiếu
4	010	Phạm Nam Thanh
5	011	Nguyễn Thành Chí Tâm

LỒNG PHÂN CẤP

--VD5: TÌM NHỮNG GIÁO VIÊN CÓ LƯƠNG LỚN HƠN LƯƠNG CỦA ÍT NHẤT MỘT GIÁO VIÊN BỘ MÔN 'CÔNG NGHỆ PHẦN MỀM'

```
SELECT GV1.*  
FROM GIAOVIEN GV1, GIAOVIEN GV2, BOMON BM  
WHERE GV2.MABM = BM.MABM AND  
      BM.TENBM=N'Hệ thống thông tin' AND  
      GV1.LUONG > GV2.LUONG
```

```
SELECT *  
FROM GIAOVIEN GV1  
WHERE GV1.LUONG > ANY (SELECT GV2.LUONG  
                        FROM GIAOVIEN GV2, BOMON BM  
                        WHERE GV2.MABM = BM.MABM AND  
                        BM.TENBM=N'Hệ thống thông tin')
```

LỒNG PHÂN CẤP

-VD6: TÌM NHỮNG GIÁO VIÊN CÓ LƯƠNG LỚN HƠN TẤT CẢ GIÁO VIÊN HỆ THỐNG THÔNG TIN

```
SELECT GV.*  
FROM GIAOVIEN GV, BOMON BM, THAMGIADETAI TG  
WHERE GV.MAGV= BM.TRUONGBM AND TG.MAGV = GV.MAGV
```

```
SELECT *  
FROM GIAOVIEN  
WHERE MAGV IN (SELECT TRUONGBM FROM BOMON)  
AND MAGV IN (SELECT MAGV FROM THAMGIADETAI)
```

LỒNG PHÂN CẤP

-VD7: TÌM NHỮNG GIÁO VIÊN KHÔNG CÓ NGƯỜI THÂN NÀO

```
SELECT *  
FROM GIAOVIEN  
WHERE MAGV NOT IN (SELECT MAGV  
                    FROM NGUOITHAN)
```

```
SELECT *  
FROM GIAOVIEN  
WHERE MAGV <> ALL (SELECT MAGV  
                   FROM NGUOITHAN)
```

LỒNG TƯỞNG QUAN

--VD1: TÌM NHỮNG GIÁO VIÊN LÀ TRƯỞNG BỘ MÔN

```
SELECT *  
FROM GIAOVIEN, BOMON  
WHERE MAGV = TRUONGBM
```

```
SELECT *  
FROM GIAOVIEN  
WHERE MAGV IN (SELECT TRUONGBM  
                FROM BOMON )
```

```
SELECT *  
FROM GIAOVIEN  
WHERE EXISTS (SELECT *  
              FROM BOMON  
              WHERE MAGV = TRUONGBM)
```

Mệnh đề where của **truy vấn con tham chiếu ít nhất một thuộc tính** của các quan hệ trong mệnh đề FROM ở truy vấn cha

LỒNG TƯƠNG QUAN

--VD2: TÌM NHỮNG GIÁO VIÊN CÓ LƯƠNG LỚN NHẤT

```
SELECT *  
FROM GIAOVIEN GV1  
WHERE GV1.LUONG >= ALL (SELECT MAX(LUONG)  
                        FROM GIAOVIEN)
```

```
SELECT *  
FROM GIAOVIEN GV1  
WHERE NOT EXISTS (SELECT *  
                  FROM GIAOVIEN GV2  
                  WHERE GV2.LUONG > GV1.LUONG)
```

GV lương lớn nhất khi không tồn tại 1 GV nào có lương lớn hơn lương GV đó

```
SELECT *  
FROM GIAOVIEN  
WHERE LUONG >= ALL (SELECT LUONG  
                    FROM GIAOVIEN)
```

GV có lương lớn nhất khi lương của GV đó lớn hơn hoặc bằng tất cả lương của các GV


LỒNG TƯỞNG QUAN

--VD2: TÌM GIÁO VIÊN TRÙNG TÊN, TRÙNG GIỚI TÍNH VỚI GV TRONG BỘ MÔN

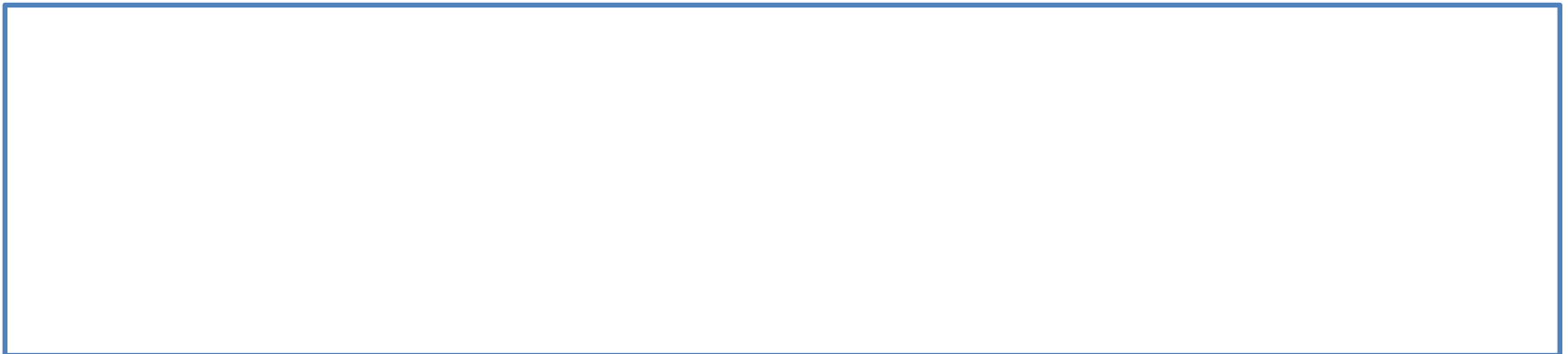
```
SELECT *  
FROM GIAOVIEN GV1  
WHERE EXISTS (SELECT *  
              FROM GIAOVIEN GV2  
              WHERE GV1.HOTEN LIKE GV2.HOTEN AND  
                    GV1.PHAI = GV2.PHAI AND  
                    GV1.MABM = GV2.MABM AND  
                    GV1.MAGV <> GV2.MAGV)
```

LỒNG TƯỞNG QUAN

--VD3: TÌM NHỮNG GIÁO VIÊN KHÔNG THAM GIA BẤT CỨ ĐỀ TÀI NÀO



-- VD4: TÌM NHỮNG GIÁO VIÊN CÓ LƯƠNG LỚN HƠN LƯƠNG CỦA ÍT NHẤT MỘT GIÁO VIÊN BỘ MÔN 'CÔNG NGHỆ PHẦN MỀM'

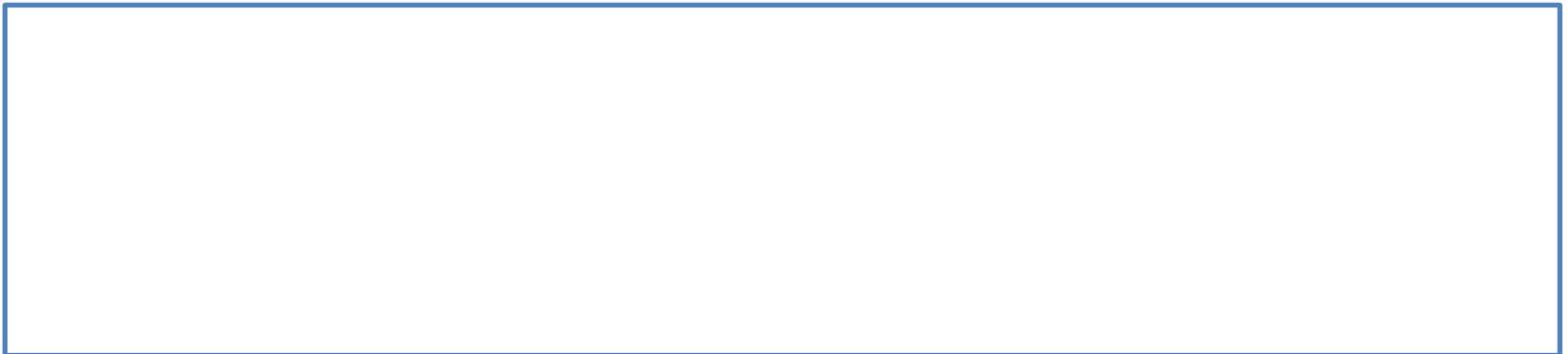


LỒNG TƯỞNG QUAN

--VD5: TÌM NHỮNG TRƯỞNG BỘ MÔN THAM GIA TỐI THIỂU 1 ĐỀ TÀI


A large, empty rectangular box with a blue border, intended for a diagram or answer related to the example VD5.

-- VD6: TÌM NHỮNG GIÁO VIÊN CÓ THAM GIA ĐỀ TÀI

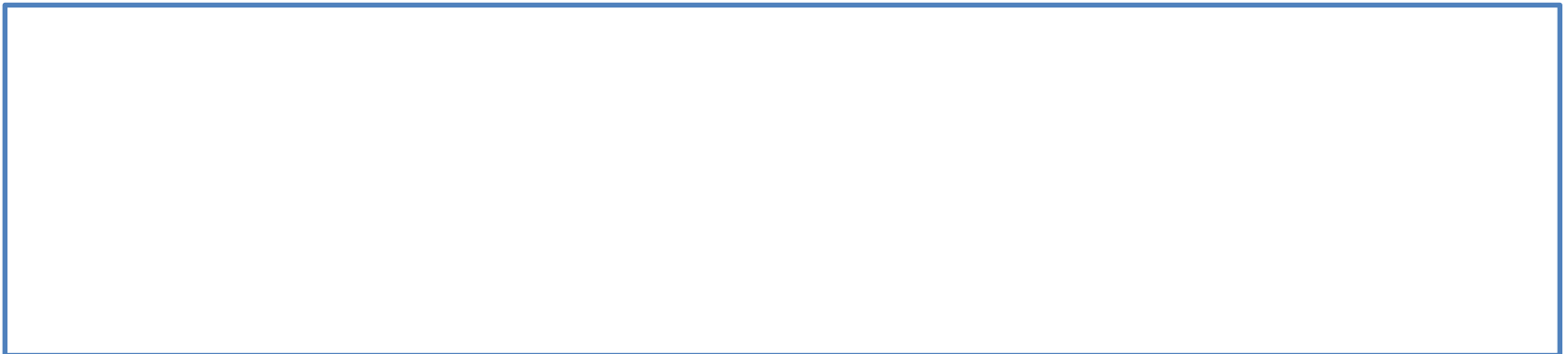
A large, empty rectangular box with a blue border, intended for a diagram or answer related to the example VD6.

LỒNG TƯỞNG QUAN

--VD7: LIỆT KÊ NHỮNG GIÁO VIÊN NHỎ TUỔI NHẤT (NĂM SINH LỚN NHẤT)



-- VD8: LIỆT KÊ NHỮNG GIÁO VIÊN LỚN TUỔI NHẤT (NĂM SINH NHỎ NHẤT)



NHẬN XÉT **IN** VÀ **EXISTS**

IN

- <tên cột> IN <câu truy vấn con>
- Thuộc tính ở mệnh đề SELECT của truy vấn con phải có cùng kiểu dữ liệu với thuộc tính mệnh đề WHERE của truy vấn cha

EXISTS

- Không cần có thuộc tính, hằng số hay biểu thức nào khác đứng trước
- Không nhất thiết liệt kê tên thuộc tính ở mệnh đề SELECT của truy vấn con
- Những truy vấn có = ANY hay IN đều có thể chuyển thành câu truy vấn EXISTS

NHẬN XÉT **IN** VÀ **EXISTS**

NHẬN XÉT 1

- Những câu truy vấn = **ANY** hay **IN** đều có thể chuyển thành câu truy vấn **EXISTS**
- Những câu truy vấn có **<>ALL** hay **NOT IN** đều có thể chuyển thành **NOT EXISTS**

NHẬN XÉT 2

- Subquery có thể xuất hiện ở mệnh đề FROM
- Subquery ở mệnh đề FROM trả về bảng trung gian (bảng tạm) sử dụng trong quá trình truy vấn

Truy vấn dữ liệu

--VD8: liệt kê những GIAOVIEN có lương trung bình cao nhất

```
SELECT MABM, MAX(AVG(LUONG))  
FROM GIAOVIEN  
GROUP BY MABM
```

```
SELECT MABM, AVG(LUONG)  
FROM GIAOVIEN  
GROUP BY MABM  
HAVING AVG(LUONG) >= ALL (SELECT AVG(LUONG)  
                           FROM GIAOVIEN  
                           GROUP BY MABM )
```

CẤU TRÚC CASE

- Cho phép kiểm tra điều kiện và xuất thông tin theo từng trường hợp
- Cú pháp

```
CASE <tên cột>
    WHEN <giá trị> THEN <biểu thức>
    WHEN <giá trị> THEN <biểu thức>
    ...
    [ELSE <biểu thức>]
END
```

CẤU TRÚC CASE

-- VD9: NẾU LÀ HTTT THÌ TĂNG 0.2 LƯƠNG, MMT TĂNG 0.1 LƯƠNG

```
SELECT MAGV, HOTEN, MABM, LUONG, (CASE MABM
                                   WHEN 'HTTT' THEN 0.2
                                   WHEN 'MMT' THEN 0.1
                                   END ) * LUONG AS BOSUNGLUONG
FROM GIAOVIEN
```

	MAGV	HOTEN	MABM	LUONG	BOSUNGLUONG
1	001	Nguyễn Hoài An	MMT	2000	200
2	002	Trần Trà Hương	HTTT	2500	500
3	003	Nguyễn Ngọc Ánh	HTTT	2200	440
4	004	Trương Nam Sơn	VS	2300	NULL
5	005	Lý Hoàng Hà	VLĐT	2500	NULL
6	006	Trần Bạch Tuyết	VS	1500	NULL
7	007	Nguyễn Anh Trung	HPT	2100	NULL
8	008	Trần Trung Hiếu	HPT	1800	NULL
9	009	Trần Hoàng Nam	MMT	2000	200
10	010	Phạm Nam Thanh	HPT	1500	NULL
11	011	Nguyễn Thành Chí Tâm	NULL	NULL	NULL

LỆNH SELECT ... INTO

- Tạo một table bằng cách “copy” từ một DB đã có

```
SELECT column1, column2, column3, ...  
INTO newtable [IN externaldb]  
FROM oldtable  
WHERE condition;
```

```
SELECT *  
INTO GV_Backup  
FROM GIAOVIEN
```


PHÂN LOẠI

SQL gồm 3 nhóm câu lệnh:

- Nhóm định nghĩa dữ liệu (DDL - Data Definition Language): **CREATE, ALTER, DROP**
- Ngôn ngữ thao tác dữ liệu (DML - Data Manipulation Language): **SELECT, INSERT, UPDATE, DELETE**
- Nhóm điều khiển dữ liệu (DCL – Data Control Language commands): **DENY, GRANT, REVOKE => HỌC Ở HỆ QTCSDL**

Các lệnh cập nhật dữ liệu

- Thêm dữ liệu: **Insert**
- Cập nhật dữ liệu: **Update**
- Xoá dữ liệu: **Delete**

- **Lưu ý: Dữ liệu** được thêm vào table, hay được update/delete **thành công** khi tuân thủ **các ràng buộc đã thiết lập**
 - Các ràng buộc trên table
 - Các ràng buộc giữa các table

Lệnh insert

- Thêm 1 hay nhiều dòng vào table (còn gọi là nhập dữ liệu)
- Cú pháp
 - (1) Nhập 1 hay nhiều record dữ liệu

```
INSERT INTO <tên bảng>(<danh sách các thuộc tính>)  
VALUES (<danh sách các giá trị>)
```

- (2) Nhập dữ liệu dựa vào dữ liệu của table khác

```
INSERT INTO <tên bảng>(<danh sách các thuộc tính>)  
SELECT ...
```

Lệnh insert

Nhận xét

- Thứ tự các giá trị trong <danh sách các giá trị> phải trùng với thứ tự các cột
- Có thể đưa NULL vào các cột cho phép giá trị NULL
- Dữ liệu nhập vào table đúng kiểu dữ liệu và thoả các ràng buộc đã định nghĩa cho table . Ngược lại lệnh insert bị lỗi

Lệnh update

- Thay đổi giá trị của thuộc tính cho các dòng của bảng (hay gọi là chỉnh sửa dữ liệu)
- Cú pháp

```
UPDATE <tên bảng>  
SET <tên thuộc tính>=<giá trị mới>,  
      <tên thuộc tính>=<giá trị mới>,  
      ...  
[WHERE <điều kiện>]
```

Lệnh update

- Những dòng thoả điều kiện ở mệnh đề where sẽ được thay đổi giá trị
- Nếu không dùng mệnh đề WHERE: thay đổi giá trị sẽ áp dụng cho tất cả các dòng trong bảng
- Lệnh UPDATE có thể gây ra vi phạm ràng buộc tham chiếu => xử lý bằng cách
 - Không cho sửa (No Action)
 - Sửa luôn những dòng có giá trị đang tham chiếu đến (CASCADE)

Lệnh update

- (1) `UPDATE SINHVIEN
SET Hocbong=Hocbong*1.1`
- (2) `UPDATE SINHVIEN
SET Lop = 'CS01TT'
WHERE MaSV='201001'`
- (3) `UPDATE SINHVIEN
SET Hocbong=Hocbong*1.1
WHERE MaNganh IN (Select MaNganh
From NGANH
Where SoTC >= 200)`

LỆNH DELETE

- Xoá các dòng của bảng
- Cú pháp

```
DELETE FROM <tên bảng>  
[WHERE <điều kiện>]
```

- Nhận xét
 - Những dòng thoả điều kiện ở mệnh đề WHERE sẽ bị xoá
 - Nếu không chỉ định điều kiện ở mệnh đề WHERE, tất cả các dòng trong bảng sẽ bị xoá
 - Lệnh DELETE có

LỆNH DELETE

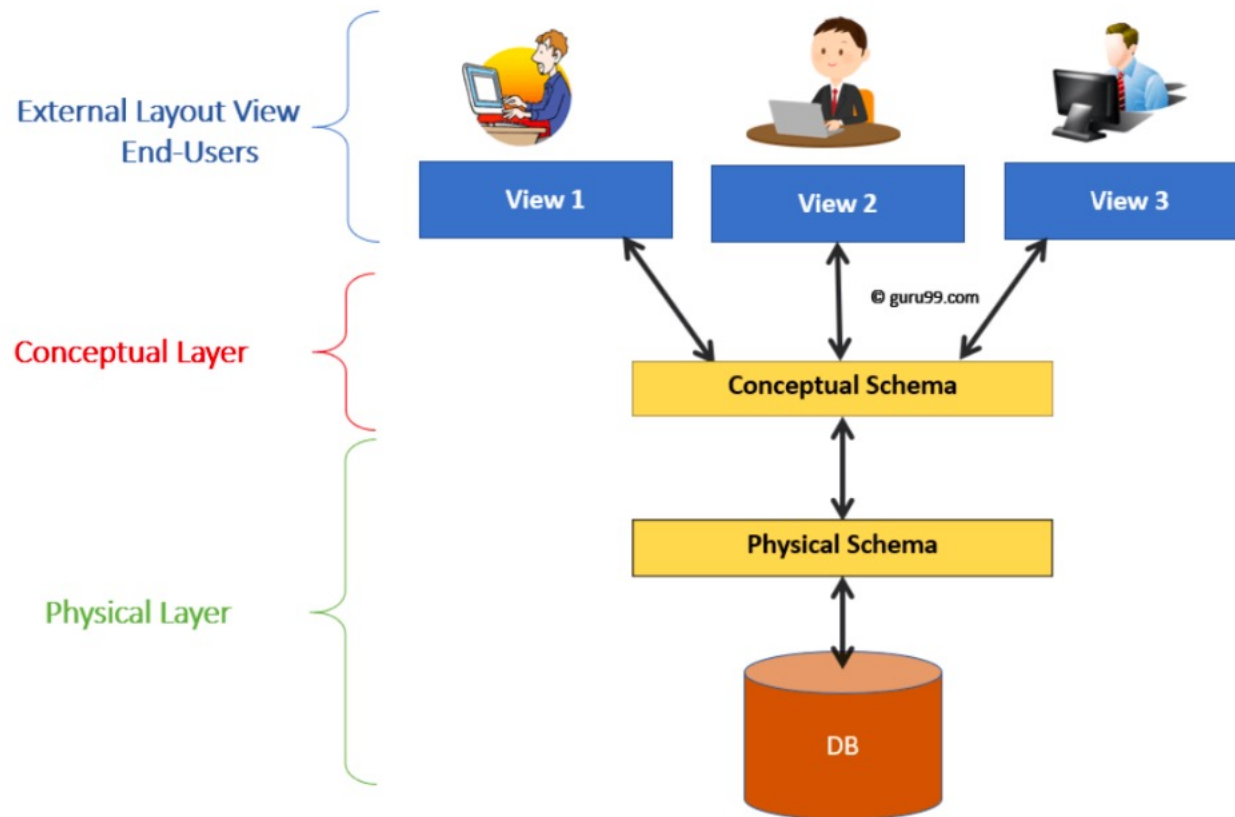
Nhận xét

- Những dòng thoả điều kiện ở mệnh đề WHERE sẽ bị xoá
- Nếu không chỉ định điều kiện ở mệnh đề WHERE, tất cả các dòng trong bảng sẽ bị xoá
- Lệnh DELETE có thể gây ra vi phạm RB tham chiếu => xử lý bằng cách :
 - Không cho xoá (NO ACTION)
 - Xoá luôn những dòng có giá trị đang tham chiếu đến (CASCADE)
 - Đặt NULL hay gán giá trị default cho những giá trị tham chiếu (SET NULL, SET DEFAULT)

GIÁ TRỊ NULL

- NULL được sử dụng để biểu diễn dữ liệu chưa biết, chưa xác định
- NULL khác với giá trị rỗng hay chuỗi rỗng ""
- Để so sánh với giá trị NULL bằng t-sql, sử dụng <operand> is null hay <operand> is not null

VIEW



TẠO VIEW

- DBMS hỗ trợ tạo view vì vai trò hữu ích của nó
- Vai trò của view ?
 - Tạo những khung nhìn dữ liệu khác nhau, phù hợp yêu cầu của người dùng
 - Cho phép trừu tượng dữ liệu – che dấu dữ liệu bên dưới
 - Kiểm soát truy suất dữ liệu
- Nhược điểm của view ?
 - Hiệu năng
 - Khả năng chỉnh sửa table qua view
 - Phụ thuộc : Khi table gốc thay đổi cấu trúc -> đòi hỏi thay đổi view

TẠO VIEW

- Các DBMS cho phép tạo và quản lý view bằng lệnh sql hay tool
 - Duy trì view như một object có sẵn
 - Cung cấp các thao tác tạo , hiệu chỉnh , xóa view, phân quyền theo view
- **Sử dụng view**
 - Sử dụng view giống như với table
 - Select
 - Insert / update / delete dữ liệu qua view
 - Insert into VIEW_NAME values (, , ,)

TẠO VIEW

Cú pháp tạo view (T-SQL)

- Cú pháp tạo view (t-sql)

```
CREATE VIEW [ schema_name . ] view_name [ ( column [ ,...n ] ) ]  
[ WITH <view_attribute> [ ,...n ] ]  
AS select_statement [ ; ]  
[ WITH CHECK OPTION ]
```

```
<view_attribute> ::=  
{  
[ ENCRYPTION ]  
[ SCHEMABINDING ]  
[ VIEW_METADATA ] }
```