

# Chương 4: Ngôn ngữ SQL

# Nội dung

1. Giới thiệu
2. Các ngôn ngữ giao tiếp
3. Ngôn ngữ định nghĩa dữ liệu
4. Ngôn ngữ thao tác dữ liệu
5. Ngôn ngữ truy vấn dữ liệu có cấu trúc

# 1. Giới thiệu

- ♦ Là ngôn ngữ chuẩn để truy vấn và thao tác trên CSDL quan hệ.
- ♦ Là ngôn ngữ phi thủ tục.
- ♦ Khởi nguồn của SQL là SEQUEL - *Structured English Query Language*, năm 1974).
- ♦ *Các chuẩn SQL*
  - SQL89
  - SQL92 (SQL2)
  - SQL99 (SQL3)

## 2. Các ngôn ngữ giao tiếp

- ◆ **Ngôn ngữ định nghĩa dữ liệu** (*Data Definition Language - DDL*): cho phép khai báo cấu trúc bảng, các mối quan hệ và **các ràng buộc**.
- ◆ **Ngôn ngữ thao tác dữ liệu** (*Data Manipulation Language - DML*): cho phép thêm, xóa, sửa dữ liệu.
- ◆ **Ngôn ngữ truy vấn dữ liệu** (*Structured Query Language – SQL*): cho phép truy vấn dữ liệu.
- ◆ **Ngôn ngữ điều khiển dữ liệu** (*Data Control Language – DCL*): khai báo bảo mật thông tin, cấp quyền và thu hồi quyền khai thác trên cơ sở dữ liệu.

# 3. Ngôn ngữ định nghĩa dữ liệu

## 3.1 Lệnh tạo bảng (CREATE)

3.1.1 Cú pháp

3.1.2 Một số kiểu dữ liệu

## 3.2 Lệnh sửa cấu trúc bảng (ALTER)

3.2.1 Thêm thuộc tính

3.2.2 Sửa kiểu dữ liệu của thuộc tính

3.2.3 Xoá thuộc tính

3.2.4 Thêm ràng buộc toàn vẹn

3.2.5 Xoá ràng buộc toàn vẹn

## 3.3 Lệnh xóa bảng (DROP)

## 3.1 Lệnh tạo bảng

### 3.1.1 Cú pháp

CREATE TABLE <tên\_bảng>

(

<tên\_cột1> <kiểu\_dữ\_liệu> [not null],

<tên\_cột2> <kiểu\_dữ\_liệu> [not null],

...

<tên\_cột1> <kiểu\_dữ\_liệu> [not null],

khai báo khóa chính, khóa ngoại, ràng buộc

)

## 3.1 Lệnh tạo bảng (2)

### 3.1.2 Một số kiểu dữ liệu

Kiểu dữ liệu	SQL Server
Chuỗi ký tự	varchar(n), char(n), nvarchar(n), nchar(n)
Số	tinyint, smallint, int, numeric(m,n), decimal(m,n), float, real, smallmoney, money
Ngày tháng	smalldatetime, datetime
Luận lý	bit

## 3.1 Lệnh tạo bảng (3)

*Lược đồ CSDL quản lý bán hàng gồm có các quan hệ sau:*

**KHACHHANG** (MAKH, HOTEN, DCHI, SODT, NGSINH,  
DOANHISO, NGDK, CMND)

**NHANVIEN** (MANV,HOTEN, NGVL, SODT)

**SANPHAM** (MASP,TENSP, DVT, NUOCSX, GIA)

**HOADON** (SOHD, NGHD, MAKH, MANV, TRIGIA)

**CTHD** (SOHD,MASP,SL)



## 3.1 Lệnh tạo bảng (4)

Create table KHACHHANG

(

MAKH char(4) primary key,

HOTEN varchar(40),

DCHI varchar(50),

SODT varchar(20),

NGSINH smalldatetime,

DOANHISO money,

NGDK smalldatetime,

CMND varchar(10)

)

## 3.1 Lệnh tạo bảng (5)

Create table CTHD

```
(  
    SOHD      int      foreign key  
                references HOADON(SOHD),  
    MASP      char(4) foreign key  
                references SANPHAM(MASP),  
    SL         int,  
    constraint PK_CTHD primary key (SOHD, MASP)  
)
```

## 3.2 Sửa cấu trúc bảng(1)

### 3.2.1 Thêm thuộc tính

*ALTER TABLE tênbảng ADD tên cột kiểu dữ liệu*

- Ví dụ: thêm cột Ghi\_chu vào bảng khách hàng  
*ALTER TABLE KHACHHANG ADD GHI\_CHU varchar(20)*

### 3.2.2 Sửa kiểu dữ liệu thuộc tính

*ALTER TABLE tênbảng ALTER COLUMN tên cột  
kiểu dữ liệu\_mới*

#### ♦ Lưu ý:

*Không phải sửa bất kỳ kiểu dữ liệu nào cũng được*

## 3.2 Sửa cấu trúc bảng(2)

- **Ví dụ:** Sửa Cột Ghi\_chu thành kiểu dữ liệu varchar(50)  
*ALTER TABLE KHACHHANG ALTER COLUMN GHI\_CHU varchar(50)*
- Nếu sửa kiểu dữ liệu của cột Ghi\_chu thành varchar(5), mà trước đó đã nhập giá trị cho cột Ghi\_chu có độ dài hơn 5 ký tự thì không được phép.
- Hoặc sửa từ kiểu chuỗi ký tự sang kiểu số, ...

### 3.2.3 Xóa thuộc tính

*ALTER TABLE tên\_bảng DROP COLUMN tên\_cột*

- Ví dụ: xóa cột Ghi\_chu trong bảng KHACHHANG  
*ALTER TABLE KHACHHANG DROP COLUMN Ghi\_chu*

## 3.2 Sửa cấu trúc bảng(3)

### 3.2.4 Thêm ràng buộc toàn vẹn

**ALTER TABLE <tên\_bảng>  
ADD CONSTRAINT  
<tên\_ràng\_buộc>**

**UNIQUE** tên\_cột

**PRIMARY KEY** (tên\_cột)

**FOREIGN KEY** (tên\_cột)

**REFERENCES** tên\_bảng  
(cột\_là\_khóa\_chính) **[ON  
DELETE CASCADE] [ON  
UPDATE CASCADE]**

**CHECK** (tên\_cột điều\_kiện)

## 3.2 Sửa cấu trúc bảng(4)

- ♦ Ví dụ
- ♦ Thêm ràng buộc khóa chính:
  - ALTER TABLE **NHANVIEN** ADD CONSTRAINT PK\_NV PRIMARY KEY (MANV)
- ♦ Thêm ràng buộc khóa ngoại
  - ALTER TABLE **CTHD** ADD CONSTRAINT FK\_CT\_SP FOREIGN KEY (MASP) REFERENCES **SANPHAM**(MASP)

## 3.2 Sửa cấu trúc bảng(5)

- ♦ Ví dụ
- ♦ Thêm ràng buộc check:
  - ALTER TABLE SANPHAM ADD CONSTRAINT CK\_GIA CHECK (GIA >=500)
- ♦ Thêm ràng buộc Unique:
  - ALTER TABLE KHACHHANG ADD CONSTRAINT UQ\_KH UNIQUE (CMND)

## 3.2 Sửa cấu trúc bảng(6)

### 3.2.5 Xóa ràng buộc toàn vẹn

*ALTER TABLE tên\_bảng DROP CONSTRAINT  
tên\_ràng\_buộc*

■ **Ví dụ:**

- Alter table CTHD drop constraint FK\_CT\_SP
- Alter table SANPHAM drop constraint ck\_gia

- ◆ **Lưu ý:** đối với ràng buộc khóa chính, muốn xóa ràng buộc này phải xóa hết các ràng buộc khóa ngoại tham chiếu tới nó



## 3.3 Lệnh xóa bảng

- ◆ **Cú pháp**

`DROP TABLE tên_bảng`

- ◆ **Ví dụ:** xóa bảng KHACHHANG.

`DROP TABLE KHACHHANG`

- ◆ **Lưu ý:** khi muốn xóa một bảng phải xóa tất cả những khóa ngoại tham chiếu tới bảng đó trước.

## 4. Ngôn ngữ thao tác dữ liệu

- ◆ Gồm các lệnh:

- 4.1 Lệnh thêm dữ liệu (INSERT)

- 4.2 Lệnh sửa dữ liệu (UPDATE)

- 4.3 Lệnh xóa dữ liệu (DELETE)

## 4.1 Thêm dữ liệu

### ◆ **Cú pháp**

INSERT INTO tên\_bảng (cột1,...,cộtn) VALUES  
(giá\_trị\_1,..., giá\_trị\_n)

INSERT INTO tên\_bảng VALUES (giá\_trị\_1,  
giá\_trị\_2,..., giá\_trị\_n)

### ◆ **Ví dụ:**

- insert into SANPHAM values('BC01','But chi', 'cay', 'Singapore', 3000)
- insert into SANPHAM(masp,tensp,dvt,nuocsx,gia)  
values ('BC01','But chi','cay','Singapore',3000)

## 4.2 Sửa dữ liệu

- ◆ **Cú pháp**

UPDATE tên\_bảng  
SET cột\_1 = giá\_trị\_1, cột\_2 = giá\_trị\_2 ....  
[WHERE điều\_kiện] không có thì có nghĩa là cập nhật hết bảng

- ◆ **Lưu ý:** cẩn thận với các lệnh xóa và sửa, nếu không có điều kiện ở WHERE nghĩa là xóa hoặc sửa tất cả.

- ◆ **Ví dụ:** Tăng giá 10% đối với những sản phẩm do “Trung Quoc” sản xuất

```
UPDATE SANPHAM  
SET Gia = Gia*1.1  
WHERE Nuocsx='Trung Quoc'
```

# Sửa dữ liệu

## NHANVIEN

MANV	HOTEN	DTHOAI	NGVL
NV01	Nguyen Nhu Nhut	0927345678	13/4/2006
NV02	Le Thi Phi Yen	0987567390	21/4/2006
NV03	Nguyen Van B	0997047382	27/4/2006
NV04	Ngo Thanh Tuan	0913758498	24/6/2006
NV05	Nguyen Thi Truc Thanh	0918590387	20/7/2006

## NHANVIEN

MANV	HOTEN	DTHOAI	NGVL
NV01	Nguyen Nhu Nhut	0927345678	13/4/2006
NV02	Le Thi Phi Yen	0987567390	21/4/2006
NV03	Nguyen Van B	0989999999	27/4/2006
NV04	Ngo Thanh Tuan	0913758498	24/6/2006
NV05	Nguyen Thi Truc Thanh	0918590387	20/7/2006

UPDATE NHANVIEN  
SET dthoai = '0989999999'  
WHERE manv = 'NV03'

# Sửa dữ liệu

## NHANVIEN

MANV	HOTEN	DTHOAI
NV01	Nguyen Nhu Nhut	0927345678
NV02	Le Thi Phi Yen	0987567390
NV03	Nguyen Van B	0997047382
NV04	Ngo Thanh Tuan	0913758498
NV05	Nguyen Thi Truc Thanh	0918590387

## NHANVIEN

MANV	HOTEN	DTHOAI
NV01	Nguyen Nhu Nhut	0927345678
NV02	Le Thi Phi Yen	0987567390
NV03	Nguyen Van Minh	0989999999
NV04	Ngo Thanh Tuan	0913758498
NV05	Nguyen Thi Truc Thanh	0918590387

UPDATE NHANVIEN

SET hoten='Nguyen Van Minh', dthoai = '0989999999'

WHERE manv = 'NV03'

# Sửa dữ liệu

NHANVIEN

MANV	HOTEN	DTHOAI
NV01	Nguyen Nhu Nhut	0927345678
NV02	Le Thi Phi Yen	0987567390
NV03	Nguyen Van B	0997047382
NV04	Ngo Thanh Tuan	0913758498
NV05	Nguyen Thi Truc Thanh	0918590387

NHANVIEN

MANV	HOTEN	DTHOAI
NV01	Nguyen Nhu Nhut	0989999999
NV02	Le Thi Phi Yen	0989999999
NV03	Nguyen Van Minh	0989999999
NV04	Ngo Thanh Tuan	0989999999
NV05	Nguyen Thi Truc Thanh	0989999999

UPDATE NHANVIEN  
SET dthoai = '0989999999'

## 4.3 Xóa dữ liệu

- ◆ **Cú pháp**

DELETE FROM tên\_bảng [WHERE điều\_kiện]

- ◆ **Ví dụ:**

- Xóa toàn bộ nhân viên

DELETE FROM NHANVIEN

- Xóa những sản phẩm do Trung Quốc sản xuất có giá thấp hơn 10000

DELETE FROM SANPHAM

WHERE (Gia < 10000) and (Nuocsx='Trung Quoc')



# Xóa dữ liệu

NHANVIEN

MANV	HOTEN	DTHOAI	LUONG
NV01	Nguyen Nhu Nhut	0927345678	2.800.000
NV02	Le Thi Phi Yen	0987567390	2.000.000
NV03	Nguyen Van B	0997047382	2.300.000
NV04	Ngo Thanh Tuan	0913758498	1.800.000
NV05	Nguyen Thi Truc Thanh	0918590387	2.500.000

NHANVIEN

DELETE FROM nhanvien  
WHERE manv = 'NV02'

MANV	HOTEN	DTHOAI	LUONG
NV01	Nguyen Nhu Nhut	0927345678	2.800.000
NV03	Nguyen Van B	0997047382	2.300.000
NV04	Ngo Thanh Tuan	0913758498	1.800.000
NV05	Nguyen Thi Truc Thanh	0918590387	2.500.000

# Xóa dữ liệu

NHANVIEN

MANV	HOTEN	DTHOAI	LUONG
NV01	Nguyen Nhu Nhut	0927345678	2.800.000
NV02	Le Thi Phi Yen	0987567390	2.000.000
NV03	Nguyen Van B	0997047382	2.500.000
NV04	Ngo Thanh Tuan	0913758498	1.800.000
NV05	Nguyen Thi Truc Thanh	0918590387	2.500.000

NHANVIEN

DELETE FROM nhanvien  
WHERE luong=2.500.000

MANV	HOTEN	DTHOAI	LUONG
NV01	Nguyen Nhu Nhut	0927345678	2.800.000
NV02	Le Thi Phi Yen	0987567390	2.000.000
NV04	Ngo Thanh Tuan	0913758498	1.800.000

# Xóa dữ liệu

## NHANVIEN

MANV	HOTEN	DTHOAI	LUONG
NV01	Nguyen Nhu Nhut	0927345678	2.800.000
NV02	Le Thi Phi Yen	0987567390	2.000.000
NV03	Nguyen Van B	0997047382	2.500.000
NV04	Ngo Thanh Tuan	0913758498	1.800.000
NV05	Nguyen Thi Truc Thanh	0918590387	2.500.000

## NHANVIEN

MANV	HOTEN	DTHOAI	LUONG
NV02	Le Thi Phi Yen	0987567390	2.000.000
NV04	Ngo Thanh Tuan	0913758498	1.800.000

DELETE FROM nhanvien  
WHERE luong>2.000.000

## 5. Ngôn ngữ truy vấn dữ liệu có cấu trúc

- 5.1 Toán tử truy vấn
- 5.2 Câu truy vấn tổng quát
- 5.3 Truy vấn đơn giản
- 5.4 Đặt bí danh, sử dụng \*, distinct
- 5.5 Phép kết
- 5.6 Hàm tính toán, gom nhóm
- 5.7 Truy vấn lồng
- 5.8 Phép toán tập hợp
- 5.9 Phép chia
- 5.10 Một số hàm toán học và xử lý chuỗi ký tự, ngày tháng, năm

## 5.1 Toán tử truy vấn

- ♦ Toán tử so sánh: =, >, <, >=, <=, <>
- ♦ Toán tử logic: AND, OR, NOT
- ♦ Phép toán: +, -, \*, /
- ♦ BETWEEN .... AND
- ♦ IS NULL, IS NOT NULL
- ♦ LIKE (\_ %)
- ♦ IN, NOT IN
- ♦ EXISTS , NOT EXISTS
- ♦ SOME, ALL

## 5.2 Câu truy vấn tổng quát

SELECT [DISTINCT] \*|tên\_cột | hàm  
FROM bảng  
[WHERE điều\_kiện]  
[GROUP BY tên\_cột]  
[HAVING điều\_kiện]  
[ORDER BY tên\_cột ASC | DESC]

## 5.3 Truy vấn đơn giản(1)

### ◆ SELECT

- Tương đương phép chiếu của ĐSQH
- Liệt kê các thuộc tính cần hiển thị trong kết quả

### ◆ WHERE

- Tương ứng với điều kiện chọn trong ĐSQH
- Điều kiện liên quan tới thuộc tính, sử dụng các phép nối luận lý AND, OR, NOT, các phép toán so sánh, BETWEEN

### ◆ FROM

- Liệt kê các quan hệ cần thiết, các phép kết

## 5.3 Truy vấn đơn giản(2)

- ♦ **SANPHAM** (MASP,TENSP, DVT, NUOCSX, GIA)
- ♦ Tìm masp, tensp do “Trung Quoc” sản xuất

```
Select      masp,tensp
From        SANPHAM
Where       nuocsx='Trung Quoc'
```



## 5.3 Truy vấn đơn giản(3)

- ◆ **SANPHAM** (MASP, TENS, DVT, NUOCSX, GIA)
- ◆ Tìm masp, tensp do “Trung Quoc” sản xuất có giá từ 2000 đến 3000

Select masp,tensp

From SANPHAM

Where nuocsx=‘Trung Quoc’

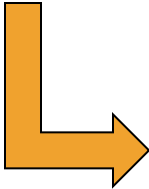
and gia >= 2000 and gia<= 3000

## 5.3 Truy vấn đơn giản(4)

### ◆ Sử dụng between SANPHAM

masp	tensp	dvt	nuocsx	gia
BB03	But bi	hop	Thai Lan	100000
BC01	But chi	cay	Singapore	3000
BC02	But chi	cay	Singapore	2500
ST04	So tay	quyen	Thai Lan	55000
ST05	So tay mong	quyen	Thai Lan	20000

Select \*  
from SANPHAM  
where gia between 2000 and 3000



masp	tensp	dvt	nuocsx	gia
BC01	But chi	cay	Singapore	3000
BC02	But chi	cay	Singapore	2500

## 5.3 Truy vấn đơn giản(5)

### Toán tử LIKE

- So sánh chuỗi tương đối
- Cú pháp: s LIKE p, p có thể chứa % hoặc \_
- % : thay thế một chuỗi ký tự bất kỳ
- \_ : thay thế một ký tự bất kỳ

## 5.3 Truy vấn đơn giản(6)

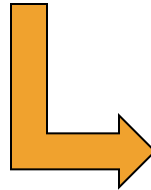
### ◆ Toán tử LIKE

- ◆ Hiển thị sản phẩm có masp bắt đầu là B, kết thúc là 1

#### SANPHAM

masp	tensp	dvt	nuocsx	gia
BB21	But bi	hop	Thai Lan	100000
BC01	But chi	cay	Singapore	3000
BC02	But chi	cay	Singapore	2500
ST04	So tay	quyen	Thai Lan	55000
ST01	So tay mong	quyen	Thai Lan	20000

Select masp, tensp  
From SANPHAM  
Where masp like 'B%1'



masp	tensp
BB21	But bi
BC01	But chi

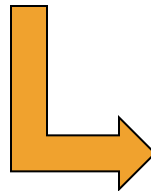
## 5.3 Truy vấn đơn giản(6)

### ◆ Toán tử LIKE

- ◆ Select masp,tensp from SANPHAM where masp like 'B\_01'

SANPHAM

masp	tensp	dvt	nuocsx	gia
BB21	But bi	hop	Thai Lan	100000
BC01	But chi	cay	Singapore	3000
BC02	But chi	cay	Singapore	2500
ST04	So tay	quyen	Thai Lan	55000
ST01	So tay mong	quyen	Thai Lan	20000



masp	tensp
BC01	But chi

## 5.3 Truy vấn đơn giản(7)

### ◆ IS NULL, IS NOT NULL

- Select \* from HOADON where makh is Null
- Select SOHD from HOADON where makh is Not Null

HOADON

SOHD	NGHD	MAKH	TRIGIA
1017	02/01/2007	KH08	35,000
1018	13/01/2007	KH08	330,000
1019	13/01/2007	KH01	30,000
1020	14/01/2007	KH09	70,000
1021	16/01/2007	KH10	67,500
1022	16/01/2007	Null	7,000
1023	17/01/2007	Null	330,000



SOHD	NGHD	MAKH	TRIGIA
1022	16/01/2007	Null	7,000
1023	17/01/2007	Null	330,000

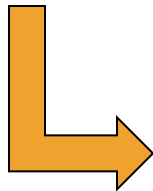
Select \* from HOADON  
where makh is Null

## 5.3 Truy vấn đơn giản(8)

### ♦ Toán tử IN, NOT IN

SANPHAM

masp	tensp	dvt	nuocsx	gia
BB03	But bi	hop	Thai Lan	100000
BC01	But chi	cay	Singapore	3000
BC02	But chi	cay	Singapore	2500
ST04	So tay	quyen	Thai Lan	55000
ST05	So tay mong	quyen	Thai Lan	20000



Select \*

From SANPHAM

Where masp **NOT IN**

('BB01','BC02','BB03')

masp	tensp	dvt	nuocsx	gia
BC01	But chi	cay	Singapore	3000
ST04	So tay	quyen	Thai Lan	55000
ST05	So tay mong	quyen	Thai Lan	20000

## 5.4 Đặt bí danh (1)

- ◆ Đặt bí danh – Alias: cho thuộc tính và quan hệ:  
tên\_cũ AS tên\_mới (hoặc tên\_cũ    tên\_mới)

```
Select manv, hoten as [ho va ten]  
From NHANVIEN
```

NHANVIEN

manv	hoten	dthoai
NV01	Nguyen Nhu Nhut	0927345678
NV02	Le Thi Phi Yen	0987567390
NV03	Nguyen Van B	0997047382
NV04	Ngo Thanh Tuan	0913758498
NV05	Nguyen Thi Truc Thanh	0918590387



manv	ho va ten
NV01	Nguyen Nhu Nhut
NV02	Le Thi Phi Yen
NV03	Nguyen Van B
NV04	Ngo Thanh Tuan
NV05	Nguyen Thi Truc Thanh



## 5.4 Đặt bí danh (2)

```
Select  hv.manv,  hv.hoten  [ho va ten]
From    NHANVIEN hv
```

NHANVIEN

manv	hoten	dthoai
NV01	Nguyen Nhu Nhut	0927345678
NV02	Le Thi Phi Yen	0987567390
NV03	Nguyen Van B	0997047382
NV04	Ngo Thanh Tuan	0913758498
NV05	Nguyen Thi Truc Thanh	0918590387



manv	ho va ten
NV01	Nguyen Nhu Nhut
NV02	Le Thi Phi Yen
NV03	Nguyen Van B
NV04	Ngo Thanh Tuan
NV05	Nguyen Thi Truc Thanh

## 5.4 Đặt bí danh (3)

Select nuocsx, gia\*1.1 as [gia ban]  
from SANPHAM  
where nuocsx<>'Thai Lan'

### SANPHAM

masp	tensp	dvt	nuocsx	gia
BB03	But bi	hop	Thai Lan	100000
BC01	But chi	cay	Singapore	3000
BC02	But chi	cay	Singapore	5000
ST04	So tay	quyen	Thai Lan	55000
ST05	So tay mong	quyen	Thai Lan	20000



nuocsx	gia ban
Singapore	3300
Singapore	5500

## 5.4 Sử dụng \* (3)

- ◆ Liệt kê tất cả các thuộc tính của quan hệ:
  - Select \* from Nhanvien where luong > 2400000
  - Select NHANVIEN.\* from NHANVIEN where luong > 2400000
  - Select hv.\* from NHANVIEN hv where luong > 2400000

NHANVIEN

MANV	HOTEN	LUONG
NV01	Nguyen Nhu Nhut	2.800.000
NV02	Le Thi Phi Yen	2.000.000
NV03	Nguyen Van B	2.500.000
NV04	Ngo Thanh Tuan	1.800.000
NV05	Nguyen Thi Truc Thanh	2.500.000



NHANVIEN

MANV	HOTEN	LUONG
NV01	Nguyen Nhu Nhut	2.800.000
NV03	Nguyen Van B	2.500.000
NV05	Nguyen Thi Truc Thanh	2.500.000

## 5.4 Sử dụng distinct (4)

- ◆ Distinct: trùng chỉ lấy một lần
  - SELECT DISTINCT lương FROM nhanvien

NHANVIEN

MANV	HOTEN	LUONG
NV01	Nguyen Nhu Nhut	2.800.000
NV02	Le Thi Phi Yen	2.000.000
NV03	Nguyen Van B	2.500.000
NV04	Ngo Thanh Tuan	2.800.000
NV05	Nguyen Thi Truc Thanh	2.500.000



LUONG
2.800.000
2.000.000
2.500.000

## 5.4 Sử dụng Order by (5)

- ◆ Order by: sắp xếp kết quả hiển thị
  - Sắp xếp tăng dần: ASC
  - Sắp xếp giảm dần: DESC
  - `Select * from NHANVIEN order by LUONG DESC`

NHANVIEN

MANV	HOTEN	LUONG
NV01	Nguyen Nhu Nhut	2.800.000
NV02	Le Thi Phi Yen	2.000.000
NV03	Nguyen Van B	2.500.000
NV04	Ngo Thanh Tuan	2.800.000
NV05	Nguyen Thi Truc Thanh	2.500.000

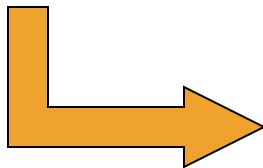


MANV	HOTEN	LUONG
NV01	Nguyen Nhu Nhut	2.800.000
NV04	Ngo Thanh Tuan	2.800.000
NV03	Nguyen Van B	2.500.000
NV05	Nguyen Thi Truc Thanh	2.500.000
NV02	Le Thi Phi Yen	2.000.000

## 5.4 Sử dụng Order by (6)

### SANPHAM

masp	tensp	dvt	nuocsx	gia
BB03	But bi	hop	Thai Lan	100000
BC01	But chi	cay	Singapore	3000
BC02	But chi	cay	Singapore	5000
ST04	So tay	quyen	Thai Lan	55000
ST05	So tay mong	quyen	Thai Lan	20000



```
SELECT *  
FROM sanpham  
ORDER BY nuocsx DESC, gia ASC
```

masp	tensp	dvt	nuocsx	gia
ST05	So tay mong	quyen	Thai Lan	20000
ST04	So tay	quyen	Thai Lan	55000
BB03	But bi	hop	Thai Lan	100000
BC01	But chi	cay	Singapore	3000
BC02	But chi	cay	Singapore	5000

## 5.5 Phép kết: Inner Join (1)

MONHOC	
Mamon	Tenmh
CSDL	Co so du lieu
CTRR	Cau truc roi rac
THDC	Tin hoc dai cuong

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	8.5
HV01	CTRR	8.5
HV03	CTRR	9.0
HV01	THDC	7.0
HV02	THDC	5.0
HV03	THDC	7.5
HV03	CSDL	6.0

In ra mã học viên, tên môn học và điểm thi của học viên

```
SELECT mahv, tenmh, diem
FROM monhoc INNER JOIN ketquathi
ON mamon =mamh
```

Lưu ý: có thể thay inner join bằng join

Cách khác

```
SELECT mahv, tenmh, diem
FROM monhoc , ketquathi
WHERE mamon =mamh
```

## 5.5 Phép kết: Inner Join (2)

HOCVIEN	
Mahv	Hoten
HV01	Nguyen Van Lan
HV02	Tran Hong Son
HV03	Nguyen Le
HV04	Le Minh

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	8.5
HV01	CTRR	8.5
HV03	CTRR	9.0

In ra mã học viên, tên học viên, mã môn học và điểm thi của học viên

```
SELECT hocvien.mahv, hoten, mamh, diem
FROM   hocvien JOIN ketquathi
      ON  hocvien.mahv=ketquathi.mahv
```

Đặt lại tên cho quan hệ hocvien và ketquathi:

```
SELECT hv.mahv, hoten, mamh, diem
FROM   hocvien hv JOIN ketquathi kq
      ON  hv.mahv=kq.mahv
```

Cách khác:

```
SELECT hv.mahv, hoten, mamh, diem
FROM   hocvien hv, ketquathi kq
WHERE  hv.mahv=kq.mahv
```



## 5.5 Phép kết: Inner Join (3)

HOCVIEN	
Mahv	Hoten
HV01	Nguyen Van Lan
HV02	Tran Hong Son
HV03	Nguyen Le
HV04	Le Minh

In ra mã học viên, tên học viên, mã môn học và điểm thi của học viên có mã số HV01

```
SELECT hv.mahv, hoten, mamh, diem
FROM   hocvien hv JOIN   ketquathi kq
      ON hv.mahv=kq.mahv
WHERE  hv.mahv = 'HV01'
```

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	8.5
HV01	CTRR	8.5
HV03	CTRR	9.0

Cách khác:

```
SELECT hv.mahv, hoten, mamh, diem
FROM   hocvien hv,   ketquathi kq
WHERE  hv.mahv=kq.mahv
      AND hv.mahv = 'HV01'
```

## 5.5 Phép kết: Inner Join (4)

HOCVIEN	
Mahv	Hoten
HV01	Nguyen Van Lan
HV02	Tran Hong Son
HV03	Nguyen Le
HV04	Le Minh

Tìm những học viên có điểm thi môn CSDL nhỏ hơn 8

```
SELECT  hv.mahv, hoten  -- Hoặc hv.*
FROM    hocvien hv JOIN  ketquathi kq
      ON hv.mahv=kq.mahv
WHERE   kq.mamh='CSDL' AND diem<8
```

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	8.5
HV01	CTRR	8.5
HV03	CTRR	9.0

Cách khác:

```
SELECT  hv.mahv, hoten  -- Hoặc hv.*
FROM    hocvien hv, ketquathi kq
WHERE   hv.mahv=kq.mahv
      AND kq.mamh='CSDL' AND diem<8
```

## 5.5 Phép kết: Inner Join (5)

### ◆ Ví dụ:

- In ra danh sách các khách hàng (MAKH, HOTEN) đã mua hàng trong ngày 1/1/2007.

```
select  KHACHHANG.makh,hoten
from    KHACHHANG inner join HOADON on
        KHACHHANG.makh=HOADON.makh
where   nghd='1/1/2007'
```

## 5.5 Phép kết: left join (6)

HOCVIEN	
Mahv	Hoten
HV01	Nguyen Van Lan
HV02	Tran Hong Son
HV03	Nguyen Le
HV04	Le Minh

In ra danh sách tất cả các học viên và mã môn học, điểm thi của học viên đó (nếu có)

```
SELECT hv.mahv, hoten, mamh, diem
FROM   hocvien hv LEFT JOIN ketquathi kq
ON     hv.mahv=kq.mahv
```

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	8.5
HV01	CTRR	8.5
HV03	CTRR	9.0



Mahv	hoten	Mamh	Diem
HV01	Nguyen Van Lan	CSDL	7.0
HV01	Nguyen Van Lan	CTRR	8.5
HV02	Tran Hong Son	CSDL	8.5
HV03	Nguyen Le	CTRR	9.0
HV04	Le Minh	NULL	NULL

## 5.5 Phép kết: left join (7)

- ◆ **Ví dụ:** In ra danh sách tất cả các hóa đơn và họ tên của khách hàng mua hóa đơn đó (nếu có)
  - Select    sohd, hoten  
From      HOADON left join KHACHHANG on  
            HOADON.makh=KHACHHANG.makh
  - Select    sohd, hoten  
From      HOADON ,KHACHHANG  
where      HOADON.makh\*=KHACHHANG.makh

## 5.5 Phép kết: right join (8)

HOCVIEN	
Mahv	Hoten
HV01	Nguyen Van Lan
HV02	Tran Hong Son
HV03	Nguyen Le
HV04	Le Minh

In ra danh sách tất cả các học viên và mã môn học, điểm thi của học viên đó (nếu có)

```
SELECT  mamh,    diem, hv.mahv,    hoten
FROM    ketquathi kq RIGHT JOIN    hocvien hv
ON      hv.mahv=kq.mahv
```

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	8.5
HV01	CTRR	8.5
HV03	CTRR	9.0



mamh	diem	mahv	hoten
CSDL	7.0	HV01	Nguyen Van Lan
CTRR	8.5	HV01	Nguyen Van Lan
CSDL	8.5	HV02	Tran Hong Son
CTRR	9.0	HV03	Nguyen Le
NULL	NULL	HV04	Le Minh

## 5.6 Gom nhóm dữ liệu (1)

### Các hàm tính toán cơ bản

- COUNT: Đếm số bộ dữ liệu của thuộc tính
- MIN: Tính giá trị nhỏ nhất
- MAX: Tính giá trị lớn nhất
- AVG: Tính giá trị trung bình
- SUM: Tính tổng giá trị các bộ dữ liệu

## 5.6 Gom nhóm dữ liệu (2)

Hiển thị điểm thi cao nhất

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	9.0
HV01	CTRR	8.0
HV03	CTRR	9.0

```
SELECT MAX(Diem) Max_D  
FROM ketquathi
```



Max_D
9.0



## 5.6 Gom nhóm dữ liệu (3)

Hiển thị điểm thi cao nhất, thấp nhất, trung bình

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	9.0
HV01	CTRR	8.0
HV03	CTRR	9.0



max_D	min_D	avg_D
9.0	7	8.25

```
SELECT MAX(Diem) max_D, MIN(Diem) min_D, AVG(Diem) avg_D  
FROM ketquathi
```

## 5.6 Gom nhóm dữ liệu (4)

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	8.0
HV01	CTRR	8.0
HV03	CTRR	9.0

Hiển thị điểm thi cao nhất của môn CSDL



max_D
8.0

```
SELECT MAX(Diem) max_D
FROM ketquathi
WHERE mamh = 'CSDL'
```

## 5.6 Gom nhóm dữ liệu (5)

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	9.0
HV01	CTRR	8.0
HV03	CTRR	9.0
HV03	CSDL	3.0
HV01	THDC	8.0

Có bao nhiêu học viên thi môn  
CSDL đạt điểm lớn hơn 5



SL
2

```
SELECT COUNT(Mahv) SL
FROM ketquathi
WHERE mamh= 'CSDL' AND diem>5
```

## 5.6 Gom nhóm dữ liệu (6)

### **Gom nhóm: mệnh đề GROUP BY**

- ◆ Sử dụng hàm gom nhóm trên các bộ trong quan hệ.
- ◆ Mỗi nhóm bộ bao gồm tập hợp các bộ có cùng giá trị trên các thuộc tính gom nhóm
- ◆ Hàm gom nhóm áp dụng trên mỗi bộ độc lập nhau.
- ◆ SQL có mệnh đề GROUP BY để chỉ ra các thuộc tính gom nhóm, các thuộc tính này phải xuất hiện trong mệnh đề SELECT

## 5.6 Gom nhóm dữ liệu (7)

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	9.0
HV01	CTRR	8.0
HV03	CTRR	9.0
HV03	CSDL	3.0
HV01	THDC	8.0

Hiển thị điểm thi cao nhất của từng học viên



mahv	max_D
HV01	8.0
HV02	9.0
HV03	9.0

```
SELECT mahv, MAX(Diem) max_D  
FROM ketquathi  
GROUP BY mahv
```

## 5.6 Gom nhóm dữ liệu (8)

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	9.0
HV01	CTRR	8.0
HV03	CTRR	9.0
HV03	CSDL	3.0
HV01	THDC	8.0

Hiển thị điểm thi cao nhất, thấp nhất của từng học viên



mahv	max_D	min_D
HV01	8.0	7.0
HV02	9.0	9.0
HV03	9.0	3.0

```
SELECT mahv, MAX(Diem) max_D, MIN (Diem) min_D  
FROM ketquathi  
GROUP BY mahv
```

## 5.6 Gom nhóm dữ liệu (9)

HOCVIEN	
Mahv	Hoten
HV01	Nguyen Van Lan
HV02	Tran Hong Son
HV03	Nguyen Le

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	9.0
HV01	CTRR	8.0
HV03	CTRR	9.0
HV03	CSDL	3.0
HV01	THDC	8.0

Hiển thị điểm thi cao nhất của từng học viên (mahv, hoten, max\_D)



mahv	Hoten	max_D
HV01	Nguyen Van Lan	8.0
HV02	Tran Hong Son	9.0
HV03	Nguyen Le	9.0

```
SELECT hv.mahv, hoten, MAX(Diem) max_D
FROM hocvien hv, ketquathi kq
WHERE hv.mahv = kq.mahv
GROUP BY hv.mahv, hoten
```

Ghi chú: Nếu không gom nhóm theo họ tên mà select theo họ tên thì câu lệnh sẽ gây ra lỗi.

## 5.6 Gom nhóm dữ liệu (10)

HOCVIEN	
Mahv	Hoten
HV01	Tran Hong Son
HV02	Tran Hong Son
HV03	Nguyen Le

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	9.0
HV01	CTRR	8.0
HV03	CTRR	9.0
HV03	CSDL	3.0
HV01	THDC	8.0

Hiển thị điểm thi cao nhất của từng học viên (mahv, hoten, max\_D). (Cách khác)



mahv	Hoten	max_D
HV01	Tran Hong Son	8.0
HV02	Tran Hong Son	9.0
HV03	Nguyen Le	9.0

```
SELECT h.mahv, hoten, k.max_D
FROM hocvien h JOIN (SELECT mahv, MAX(Diem) max_D
                     FROM ketquathi
                     GROUP BY mahv) k
ON h.mahv = k.mahv
```



## 5.6 Gom nhóm dữ liệu (11)

HOCVIEN	
Mahv	Hoten
HV01	Tran Hong Son
HV02	Tran Hong Son
HV03	Nguyen Le

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	9.0
HV01	CTRR	8.0
HV03	CTRR	9.0
HV03	CSDL	3.0
HV01	THDC	8.0

Hiển thị điểm thi cao nhất của từng học viên (hoten, max\_D)



Hoten	max_D
Tran Hong Son	8.0
Tran Hong Son	9.0
Nguyen Le	9.0

```
SELECT hoten, MAX(Diem) max_D
FROM hocvien hv, ketquathi kq
WHERE hv.mahv = kq.mahv
GROUP BY hv.mahv, hoten
```

## 5.6 Điều kiện trên nhóm dữ liệu - HAVING (12)

### - Điều kiện sau gom nhóm: mệnh đề HAVING

- Lọc kết quả theo điều kiện, sau khi đã gom nhóm
  - Điều kiện ở HAVING được thực hiện sau khi gom nhóm, các điều kiện có liên quan đến thuộc tính Group By
- ♦ **Ví dụ:** tìm phòng có số lượng nhân viên “Nữ” trên 5 người

SELECT	phong
FROM	NhanVien
WHERE	phai = 'Nữ'
GROUP BY	phong
HAVING	count(manv) > 5

## 5.6 Điều kiện trên nhóm dữ liệu - HAVING (13)

Hiển thị những học viên có điểm thi  
cao nhất lớn hơn 8.5

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	9.0
HV01	CTRR	8.0
HV03	CTRR	9.0
HV03	CSDL	3.0
HV01	THDC	8.0



mahv	MAX(Diem)
HV01	8.0
HV02	9.0
HV03	9.0



mahv	max_D
HV02	9.0
HV03	9.0

```
SELECT mahv, MAX(Diem) max_D
FROM ketquathi
GROUP BY mahv
HAVING MAX(Diem) >8.5
```

## 5.6 Điều kiện trên nhóm dữ liệu - HAVING (14)

HOCVIEN	
Mahv	Hoten
HV01	Nguyen Van Lan
HV02	Tran Hong Son
HV03	Nguyen Le

Những học viên nào có điểm trung bình >6  
(mahv, hoten, diemTB)



Mahv	Hoten	diemTB
HV01	Nguyen Van Lan	7.0
HV02	Tran Hong Son	9.0

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	9.0
HV01	CTRR	7.5
HV03	CTRR	9.0
HV03	CSDL	3.0
HV01	THDC	8.0

```
SELECT hv.mahv, hoten, AVG(Diem) diemTB
FROM   hocvien hv, ketquathi kq
WHERE  hv.mahv = kq.mahv
GROUP BY hv.mahv, hoten
HAVING AVG(Diem)>6
```

## 5.6 Ví dụ

NHANVIEN					
MANV	HOTEN	PHAI	MANQL	PHONG	LUONG
NV001	Nguyễn Ngọc Linh	Nữ	Null	NC	2.800.000
NV002	Đinh Bá Tiến	Nam	NV002	DH	2.000.000
NV003	Nguyễn Văn Mạnh	Nam	NV001	NC	2.300.000
NV004	Trần Thanh Long	Nam	NV002	DH	1.800.000
NV005	Nguyễn Thị Hồng Vân	Nữ	NV001	NC	2.500.000
NV006	Nguyễn Minh	Nam	NV002	DH	2.000.000
NV007	Hà Duy Lập	Nam	NV003	NC	1.800.000
NV008	Trần Kim Duyên	Nữ	NV003	NC	1.800.000
NV009	Nguyễn Kim Anh	Nữ	NV003	NC	2.000.000

## 5.6 Ví dụ

1. Tính lương thấp nhất, cao nhất, trung bình và tổng lương của tất cả các nhân viên.
2. Có tất cả bao nhiêu nhân viên.
3. Bao nhiêu nhân viên có người quản lý.
4. Có tất cả bao nhiêu người quản lý.
5. Có bao nhiêu nhân viên không có người quản lý trực tiếp.
6. Bao nhiêu phòng ban có nhân viên trực thuộc.
7. Tính lương trung bình của tất cả các nhân viên.
8. Tính lương trung bình của các nhân viên theo từng phòng ban.

## 5.6 Ví dụ

1. Tính lương thấp nhất, cao nhất, trung bình và tổng lương của tất cả các nhân viên.

```
SELECT  min(luong) as thapnhat,  
        max(luong) as caonhat,  
        avg(luong) as trungbinh,  
        sum(luong) as tongluong  
  
FROM    NhanVien
```

## 5.6 Ví dụ

2. Có tất cả bao nhiêu nhân viên.

```
SELECT count(*) FROM NhanVien
```



## 5.6 Ví dụ

3. Bao nhiêu nhân viên có người quản lý.

- `Select count(*) FROM NhanVien WHERE manql is not null.`
- `SELECT count(Manql) FROM NhanVien.`

## 5.6 Ví dụ

4. Có tất cả bao nhiêu người quản lý.

```
SELECT count(DISTINCT ma_nql) as SLNQL  
FROM Nhanvien
```

## 5.6 Ví dụ

5. Có bao nhiêu nhân viên không có người quản lý trực tiếp.

```
SELECT count(*) as SLNV  
FROM Nhanvien  
WHERE ma_nql IS NULL
```

## 5.6 Ví dụ

6. Bao nhiêu phòng ban có nhân viên trực thuộc.

```
SELECT count(distinct phong) FROM NhanVien
```

## 5.6 Ví dụ

7. Tính lương trung bình của tất cả các nhân viên.

```
SELECT      avg(LUONG) as LUONGTB  
FROM        NhanVien
```

## 5.6 Ví dụ

8. Tính lương trung bình của các nhân viên theo từng phòng ban.

```
SELECT      phong, avg(LUONG) as LUONGTB
FROM        NhanVien
GROUP BY    phong
```

# Bài tập

## NHANVIEN

MANV	HOTEN	PHAI	MANQL	PHONG	LUONG
NV001	Nguyễn Ngọc Linh	Nữ	Null	NC	2.800.000
NV002	Đinh Bá Tiến	Nam	NV002	DH	2.000.000
NV003	Nguyễn Văn Mạnh	Nam	NV001	NC	2.300.000
NV004	Trần Thanh Long	Nam	NV002	DH	1.800.000
NV005	Nguyễn Thị Hồng Vân	Nữ	NV001	NC	2.500.000
NV009	Nguyễn Kim Anh	Nữ	NV003	TC	2.000.000

## PHONGBAN

MAPH	TENPH	TrgPh
NC	Nghiên cứu	NV005
DH	Điều hành	NV002
TC	Tài chính	NULL

# Bài tập

1. Hiển thị maph, tenph, trgph, hoten (họ tên trưởng phòng).
2. Phòng ban NC (maph) có bao nhiêu nhân viên.
3. Phòng “Nghiên cứu” có bao nhiêu nhân viên.
4. Hiển thị những phòng ban (maph) có số nhân viên nhiều hơn 4.
5. Hiển thị những phòng ban (maph, tenph) có số nhân viên nữ nhiều hơn 2.



# Top N

- ◆ Trả về N dòng kết quả đầu tiên của câu truy vấn.
- ◆ Cú pháp: **TOP N**  
*với N là số nguyên dương*
- ◆ Nên sử dụng ORDER BY để sắp xếp kết quả khi lấy TOP N:  
Vd: Tìm mức lương cao nhất: distinct : không trùng  
..

```
SELECT TOP 1 luong as CN  
FROM Nhanvien  
ORDER BY luong DESC
```

# Top N

- ◆ `SELECT TOP N [PERCENT] [WITH TIES]`  
`FROM bang [WHERE dieu_kien]`  
`[ORDER BY bieu_thuc [ASC | DESC]];`
- ◆ Ví dụ TOP 10 sẽ chèn 10 hàng đầu tiên từ bộ kết quả.
- ◆ **PERCENT** : Ví dụ như TOP 10 PERCENT sẽ trả về 10% giá trị đầu trong bộ kết quả.
- ◆ **WITH TIES** : Các hàng có giá trị giống với hàng cuối cùng trong bộ kết quả sẽ được trả về. Điều này có thể gây ra tình huống số hàng trả về nhiều hơn biến TOP cho phép.

## 5.7 Truy vấn lồng (1)

- ◆ **EXIST và NOT EXIST** chỉ trả về giá trị **True** hoặc **False**.
- ◆ Lượng từ *Exists (tồn tại)*
  - ◆ **WHERE Exists** (<Tập hợp| truy vấn con>)
    - ◆ **True**: nếu tập hợp (truy vấn con) khác rỗng, **có** bộ giá trị trả về.
    - ◆ **False**: ngược lại, **không** có bộ giá trị nào trả về, rỗng.
- ◆ Giá trị của **NOT EXIST**:
  - **True**: khi câu truy vấn con **không** trả về bộ giá trị nào.
  - **False**: khi câu truy vấn con **có** bộ giá trị trả về.

# Tìm những khách hàng (makh, hoten) đã mua hàng

```
SELECT makh, hoten
FROM khachhang kh
WHERE EXISTS (SELECT *
              FROM hoadon hd
              WHERE hd.makh=kh.makh)
```

## Khachhang

MAKH	HOTEN	DCHI
KH01	Nguyen Van A	731 Tran Hung Dao, Q5, TpHCM
KH02	Tran Ngoc Han	23/5 Nguyen Trai, Q5, TpHCM
KH03	Tran Ngoc Linh	45 Nguyen Canh Chan, Q1, TpHCM
KH04	Tran Minh Long	50/34 Le Dai Hanh, Q10, TpHCM
KH05	Le Nhat Minh	34 Truong Dinh, Q3, TpHCM

## Hoadon

SOHD	NGHD	MAKH	TRIGIA
1001	23/07/2006	KH01	320,000
1002	12/08/2006	KH01	840,000
1003	23/08/2006	KH02	100,000
1004	01/09/2006	KH02	180,000
1008	28/10/2006	KH01	440,000
1009	28/10/2006	KH05	200,000

EXISTS (SELECT \*  
FROM hoadon hd  
WHERE hd.makh = 'KH01')

→ true

MAKH	HOTEN
KH01	Nguyen Van A

# Tìm những khách hàng (makh, hoten) đã mua hàng

```
SELECT makh, hoten
FROM khachhang kh
WHERE EXIST (SELECT *
             FROM hoadon hd
             WHERE hd.makh=kh.makh)
```

## Khachhang

MAKH	HOTEN	DCHI
KH01	Nguyen Van A	731 Tran Hung Dao, Q5, TpHCM
KH02	Tran Ngoc Han	23/5 Nguyen Trai, Q5, TpHCM
KH03	Tran Ngoc Linh	45 Nguyen Canh Chan, Q1, TpHCM
KH04	Tran Minh Long	50/34 Le Dai Hanh, Q10, TpHCM
KH05	Le Nhat Minh	34 Truong Dinh, Q3, TpHCM

## Hoadon

SOHD	NGHD	KH	TRIGIA
1001	23/07/2006	KH01	320,000
1002	12/08/2006	KH01	840,000
1003	23/08/2006	KH02	100,000
1004	01/09/2006	KH02	180,000
1008	28/10/2006	KH01	440,000
1009	28/10/2006	KH05	200,000

MAKH	HOTEN
KH01	Nguyen Van A
KH02	Tran Ngoc Han

EXIST (SELECT \*  
FROM hoadon hd  
WHERE hd.makh = 'KH02')

true

# Tìm những khách hàng (makh, hoten) đã mua hàng

```
SELECT makh, hoten
FROM khachhang kh
WHERE EXISTS (SELECT *
               FROM hoadon hd
               WHERE hd.makh=kh.makh)
```

thuoc tinh ket = dieu kien

## Khachhang

MAKH	HOTEN	DCHI
KH01	Nguyen Van A	731 Tran Hung Dao, Q5, TpHCM
KH02	Tran Ngoc Han	23/5 Nguyen Trai, Q5, TpHCM
KH03	Tran Ngoc Linh	45 Nguyen Canh Chan, Q1, TpHCM
KH04	Tran Minh Long	50/34 Le Dai Hanh, Q10, TpHCM
KH05	Le Nhat Minh	34 Truong Dinh, Q3, TpHCM

## Hoadon

SOHD	NGHD	KH	TRIGIA
1001	23/07/2006	KH01	320,000
1002	12/08/2006	KH01	840,000
1003	23/08/2006	KH02	100,000
1004	01/09/2006	KH02	180,000
1008	28/10/2006	KH01	440,000
1009	28/10/2006	KH05	200,000

EXISTS (SELECT \*  
FROM hoadon hd  
WHERE hd.makh = 'KH03')



false

MAKH	HOTEN
KH01	Nguyen Van A
KH02	Tran Ngoc Han

# Tìm những khách hàng (makh, hoten) đã mua hàng

```
SELECT makh, hoten
FROM khachhang kh
WHERE EXISTS (SELECT *
              FROM hoadon hd
              WHERE hd.makh=kh.makh)
```

## Khachhang

MAKH	HOTEN	DCHI
KH01	Nguyen Van A	731 Tran Hung Dao, Q5, TpHCM
KH02	Tran Ngoc Han	23/5 Nguyen Trai, Q5, TpHCM
KH03	Tran Ngoc Linh	45 Nguyen Canh Chan, Q1, TpHCM
KH04	Tran Minh Long	50/34 Le Dai Hanh, Q10, TpHCM
KH05	Le Nhat Minh	34 Truong Dinh, Q3, TpHCM

## Hoadon

SOHD	NGHD	KH	TRIGIA
1001	23/07/2006	KH01	320,000
1002	12/08/2006	KH01	840,000
1003	23/08/2006	KH02	100,000
1004	01/09/2006	KH02	180,000
1008	28/10/2006	KH01	440,000
1009	28/10/2006	KH03	200,000

EXISTS (SELECT \*  
FROM hoadon hd  
WHERE hd.makh = "KH04')

→ false

MAKH	HOTEN
KH01	Nguyen Van A
KH02	Tran Ngoc Han

# Tìm những khách hàng (makh, hoten) đã mua hàng

```
SELECT makh, hoten
FROM khachhang kh
WHERE EXISTS (SELECT *
              FROM hoadon hd
              WHERE hd.makh=kh.makh)
```

## Khachhang

MAKH	HOTEN	DCHI
KH01	Nguyen Van A	731 Tran Hung Dao, Q5, TpHCM
KH02	Tran Ngoc Han	23/5 Nguyen Trai, Q5, TpHCM
KH03	Tran Ngoc Linh	45 Nguyen Canh Chan, Q1, TpHCM
KH04	Tran Minh Long	50/34 Le Dai Hanh, Q10, TpHCM
KH05	Le Nhat Minh	34 Truong Dinh, Q3, TpHCM

## Hoadon

SOHD	NGHD	KH	TRIGIA
1001	23/07/2006	KH01	320,000
1002	12/08/2006	KH01	840,000
1003	23/08/2006	KH02	100,000
1004	01/09/2006	KH02	180,000
1008	28/10/2006	KH01	440,000
1009	28/10/2006	KH05	200,000

MAKH	HOTEN
KH01	Nguyen Van A
KH02	Tran Ngoc Han
KH05	Le Nhat Minh

EXISTS (SELECT \*  
FROM hoadon hd  
WHERE hd.makh = "KH05") →  
→ true



## 5.7 Truy vấn lồng (2)

### In hoặc Exists

HOCVIEN	
Mahv	Hoten
HV01	Nguyen Van Lan
HV02	Tran Hong Son
HV03	Nguyen Le
HV04	Le Minh

KETQUATHI		
Mahvien	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	8.5
HV01	CTRR	8.5
HV03	CTRR	9.0

Tìm những học viên thi được 8.5 điểm

```
SELECT *  
FROM   hocvien  
WHERE  mahv IN (SELECT mahvien  
                FROM ketquathi  
                WHERE diem=8.5)
```

```
SELECT *  
FROM   hocvien  
WHERE  EXISTS (SELECT mahvien  
                FROM   ketquathi  
                WHERE  mahvien=mahv  
                AND diem=8.5)
```

## 5.7 Truy vấn lồng (3)

### In hoặc Exists

- ◆ **Ví dụ:** Tìm các số hóa đơn (sohd) mua cùng lúc 2 sản phẩm có mã số “BB01” và “BB02”.
  - select distinct sohd  
from CTHD where masp='BB01' and sohd **IN**  
(select sohd from CTHD where masp='BB02')
  - select distinct A.sohd  
from CTHD A where A.masp='BB01' and  
**EXISTS** (select \* from CTHD B  
where B.masp='BB02' and A.sohd=B.sohd)

## 5.7 Truy vấn lồng (4)

### Not In hoặc Not Exists

- ♦ **Ví dụ:** Tìm các số hóa đơn (sohd) có mua sản phẩm mã số 'BB01' nhưng không mua sản phẩm mã số 'BB02'.
  - `select distinct sohd  
from CTHD where masp='BB01' and sohd NOT IN  
(select sohd from CTHD where masp='BB02')`
  - `select distinct A.sohd  
from CTHD A where A.masp='BB01' and  
NOT EXISTS (select * from CTHD B  
where B.masp='BB02' and A.sohd=B.sohd)`

# Nhận xét IN và EXISTS

## ◆ IN

- Thuộc tính ở mệnh đề SELECT của truy vấn con phải có cùng kiểu dữ liệu với thuộc tính ở 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 câu truy vấn có =ANY hay IN đều có thể chuyển thành câu truy vấn có EXISTS

## 5.7 Truy vấn lồng (5)

- Lượng từ **All**
- **Where** <Tên thuộc tính> <Phép so sánh>  
**All** (<Tập hợp giá trị | truy vấn con> )
  - **TRUE**: khi **một giá trị** của thuộc tính thỏa mãn **phép so sánh** với **mọi** phần tử trong tập hợp giá trị (hoặc tập giá trị do truy vấn con trả ra).
  - **FALSE**: Ngược lại

## 5.7 Truy vấn lồng (6)

Tìm học viên (Mahv) có điểm thi cao nhất

KETQUATHI		
Mahv	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	9.0
HV01	CTRR	8.5
HV03	CTRR	9.0

```
SELECT Mahv
FROM ketquathi
WHERE diem >= ALL (SELECT diem
                   FROM ketquathi)
```

Hoặc có thể viết như sau

```
SELECT Mahv
FROM ketquathi
WHERE diem = (SELECT MAX(diem)
             FROM ketquathi)
```

## 5.7 Truy vấn lồng (7)

HOCVIEN	
Mahv	Hoten
HV01	Nguyen Van Lan
HV02	Tran Hong Son
HV03	Nguyen Le
HV04	Le Minh

KETQUATHI		
Mahvien	Mamh	Diem
HV01	CSDL	7.0
HV02	CSDL	8.5
HV01	CTRR	8.5
HV03	CTRR	9.0

Tìm những học viên thi được 8.5 điểm

```
SELECT *  
FROM hocvien  
WHERE mahv = ANY (SELECT mahv  
                   FROM ketquathi  
                   WHERE diem=8.5)
```

=ANY tương đương với toán tử IN  
Cách khác: sử dụng phép kết, ...

## 5.8 Phép toán tập hợp (1) Phép hội (union).

### NHANVIEN

MANV	HOTEN	LUONG
NV001	Nguyễn Ngọc Linh	2.800.000
NV002	Đinh Bá Tiến	2.000.000
NV003	Nguyễn Văn Mạnh	2.300.000
NV004	Trần Thanh Long	1.800.000
NV005	Nguyễn Thị Hồng Vân	2.500.000
NV009	Nguyễn Kim Anh	2.000.000

### PHONGBAN

MAPH	TENPH	TrgPH
NC	Nghiên cứu	NV05
DH	Điều hành	NV02
TC	Tài chính	NULL

Hiển thị những nhân viên (manv) có lương lớn hơn 2.500.000 hoặc là trưởng phòng của phòng của phòng 'DH'

```
SELECT manv
FROM nhanvien
WHERE luong>2500000
UNION
SELECT trgPH
FROM phongban
WHERE maph='DH'
```



## 5.8 Phép toán tập hợp (2)

### Phép trừ (**EXCEPT**).

CTHD

SOHD	MASP	SL
1001	BB01	10
1001	BC04	20
1002	BB01	20
1002	BB02	20
1003	BB03	10
1004	TV01	20
1004	BB01	5

Hiển thị những hóa đơn (sohd)  
mua BB01 nhưng không mua  
BB02

```
SELECT sohd
FROM cthd
WHERE masp='BB01'
EXCEPT
SELECT sohd
FROM cthd
WHERE masp='BB02'
```

## 5.8 Phép toán tập hợp (3)

### Phép giao (**INTERSECT**).

CTHD

SOHD	MASP	SL
1001	BB01	10
1001	BC04	20
1002	BB01	20
1002	BB02	20
1003	BB03	10
1004	TV01	20
1004	BB01	5

Hiển thị những hóa đơn (sohd)  
vừa mua BB01 vừa mua BB02

```
SELECT sohd
FROM cthd
WHERE masp='BB01'
INTERSECT
SELECT sohd
FROM cthd
WHERE masp='BB02'
```

## 5.9 Phép chia

### Sử dụng NOT EXISTS

- ◆ Cho lược đồ CSDL:
  - SANPHAM (MASP, TENS<sub>P</sub>, DVT, NUOCS<sub>X</sub>, GIA)
  - HOADON (SOHD, NGHD, TRIGIA)
  - CTHD (SOHD, MASP, SL)
- ◆ Ví dụ: Tìm số hóa đơn đã mua tất cả các sản phẩm.

## 5.9 Phép chia

- ◆ Tìm hóa đơn đã mua tất cả các sản phẩm.
- ◆ Biến đổi thành:
  - Tìm hóa đơn mà **không tồn tại** sản phẩm mà hóa đơn này **không mua**.
  - Hoặc: hiển thị hóa đơn với điều kiện là không tồn tại sản phẩm mà hóa đơn này không mua.
- ◆ Để đơn giản, xét số hóa đơn HD01. Ta có thể viết lại: Hiển thị hóa đơn HD01 với điều kiện là không tồn tại /sản phẩm mà hóa đơn HD01 không mua.

Hiển thị hóa đơn HD01 với điều kiện là không tồn tại /sản phẩm mà hóa đơn HD01 không mua

- ◆ sản phẩm mà hóa đơn HD01 không mua

**SANPHAM** (MASP,TENSP, DVT, NUOCSX, GIA)  
**HOADON** (SOHD, NGHD, MAKH, MANV, TRIGIA)  
**CTHD** (SOHD,MASP,SL)

Hiển thị hóa đơn HD01 với điều kiện là không tồn tại /sản phẩm mà hóa đơn HD01 không mua

- ◆ sản phẩm mà hóa đơn HD01 không mua

```
SELECT * FROM sanpham
WHERE masp NOT IN
    (SELECT masp FROM cthd
     WHERE cthd.sohd= 'HD01' )
```

Hiển thị số hóa đơn HD01 với điều kiện là  
không tồn tại /sản phẩm mà hóa đơn  
HD01 không mua

```
SELECT *  
FROM hoadon  
WHERE sohđ= 'HD01' AND  
NOT EXISTS
```

```
(SELECT * FROM sanpham  
WHERE masp NOT IN  
(SELECT masp FROM cthđ  
WHERE cthđ.sohđ= 'HD01' ))
```

# Hiển thị số hóa đơn HD01 với điều kiện là không tồn tại /sản phẩm mà hóa đơn HD01 không mua

Viết lại trong truy vấn con:

```
SELECT *  
FROM hoadon  
WHERE sohd= 'HD01' AND  
      NOT EXISTS  
      (SELECT * FROM sanpham  
      WHERE masp NOT IN  
      (SELECT masp FROM cthd  
      WHERE cthd.sohd=hoadon.sohd ))
```



Tổng quát: hiển thị số hóa đơn với điều kiện là không tồn tại sản phẩm mà hóa đơn này không mua.

```
SELECT *  
FROM hoadon  
WHERE
```

```
NOT EXISTS
```

```
(SELECT * FROM sanpham  
WHERE masp NOT IN  
  (SELECT masp FROM cthd  
   WHERE cthd.sohd=hoadon.sohd ))
```

Tổng quát: hiển thị số hóa đơn với điều kiện là không tồn tại sản phẩm mà hóa đơn này không mua.

Thay NOT IN bằng NOT EXISTS

```
SELECT *  
FROM hoadon  
WHERE
```

```
NOT EXISTS
```

```
(SELECT * FROM sanpham  
WHERE NOT EXISTS  
    (SELECT masp FROM cthd  
    WHERE sanpham.masp = cthd.masp AND  
          cthd.sohd=hoadon.sohd ))
```

## 5.9 Phép chia

### Sử dụng NOT EXISTS

- ♦ **Ví dụ:** Tìm số hóa đơn đã mua tất cả những sản phẩm do “Trung Quoc” sản xuất.
- ♦ `Select sohd from HOADON where not exists  
(select * from SANPHAM  
where nuocsx='Trung Quoc' and not exists  
(select * from CTHD where  
HOADON.sohd=CTHD.sohd and  
CTHD.masp=SANPHAM.masp))`

## 5.9 Phép chia (tt)

PHICONG	
MAPC	TENPC
A	Peter
B	John
C	Red

PHICO	
MAPHICO	TENPHICO
X	Boeing-777
Y	Boeing-787
Z	Airbus-A380

CHUNGNHAN	
MAPC	MAPHICO
A	X
A	Y
B	Z
C	X
C	Y
C	Z

- ♦ **Ví dụ:** Tìm Phi công có thể lái tất cả phi cơ?

## 5.9 Phép chia (tt)

### Sử dụng NOT EXISTS 2 lần:

- ♦ Ví dụ: Tìm Phi công có thể lái tất cả phi cơ:
- ♦ Select MAPC from PHICONG where **not exists** (1)  
(select \* from PHICO where **not exists** (2)  
(select \* from CHUNGNHAN CN  
where PHICONG.MAPC=CN.MAPC and  
CN.MAPHICO= PHICO. MAPHICO))

## 5.9 Phép chia (tt)

PHICONG	
MAPC	TENPC
A	Peter
B	John
C	Red

PHICO	
MAPHICO	TENPHICO
X	Boeing-777
Y	Boeing-787
Z	Airbus-A380

CHUNGNHAN	
MAPC	MAPHICO
A	X
A	Y
B	Z
C	X
C	Y
C	Z

[A,X] -> (2) sai -> ko chọn  
 [A,Y] -> (2) sai -> ko chọn  
 [A,Z] -> (2) đúng -> chọn  
 => (1) sai do có Z => ko chọn A

[B,X] -> (2) đúng -> chọn  
 [B,Y] -> (2) đúng -> chọn  
 [B,Z] -> (2) sai -> ko chọn  
 => (1) sai do có X,Y => ko chọn B

[C,X] -> (2) Sai -> ko chọn  
 [C,Y] -> (2) sai -> ko chọn  
 [C,Z] -> (2) sai -> ko chọn  
 => (1) đúng do ko có gì cả => chọn C  
 C là phi công có thể lái tất cả phi cơ

## 5.10 Một số hàm toán học và xử lý chuỗi ký tự, ngày tháng, năm

### ◆ Hàm toán học:

- Hàm ABS: trả về giá trị tuyệt đối của một số.
  - `ABS(-1234.56)`
- Hàm PI() trả về số pi trong toán học.
  - `PI()` Kết quả: 3.14159265358979
- Hàm POWER trả về phép tính lũy thừa
  - `POWER(3,2)`. Kết quả là: 9

## 5.10 Một số hàm toán học và xử lý chuỗi ký tự, ngày tháng, năm

### ◆ Hàm toán học:

- Hàm ROUND trả về số được làm tròn lên
  - $\text{ROUND}(123.4567, 2) = 123.46$
  - $\text{ROUND}(123.4567, -1) = 120$
- Hàm SQRT trả về kết quả căn bậc hai
  - $\text{SQRT}(9) = 3$



## 5.10 Một số hàm toán học và xử lý chuỗi ký tự, ngày tháng, năm

### ◆ Hàm toán học:

- Hàm FLOOR trả về số nguyên nhỏ hơn hoặc bằng số được làm tròn:  $\text{FLOOR}(12.3) = 12$
- Hàm CEILING trả về số nguyên lớn hơn hoặc bằng giá trị làm tròn:  $\text{CEILING}(23.45) = 24$

## 5.10 Một số hàm toán học và xử lý chuỗi ký tự, ngày tháng, năm

- ◆ Hàm xử lý chuỗi ký tự:
  - Hàm UPPER: trả về chữ in hoa
    - `UPPER('Hello') = 'HELLO'`
  - Hàm LOWER: trả về chữ in thường
    - `LOWER ('Hello') = 'hello'`
  - Hàm LEN: trả về số ký tự trong chuỗi
    - `LEN ('Hello') = 5`

## 5.10 Một số hàm toán học và xử lý chuỗi ký tự, ngày tháng, năm

- ◆ Hàm xử lý chuỗi ký tự:
  - Hàm LEFT, RIGHT, SUBSTRING: hàm cắt chuỗi bên trái, phải, giữa
    - LEFT ('Hello world', 3) = 'HEL'
    - RIGHT ('Hello world', 5) = 'world'
    - SUBSTRING ('Hello world', 3, 2) = 'll'
  - Hàm LTRIM, RTRIM: loại bỏ khoảng trắng bên trái, bên phải.

## 5.10 Một số hàm toán học và xử lý chuỗi ký tự, ngày tháng, năm

- ◆ Hàm xử lý ngày tháng năm
  - Hàm GETDATE() trả về ngày tháng hiện tại.
  - Hàm DATEPART(datepart, date):
    - date là ngày truyền vào để lấy các phần tương ứng với tham số datepart.
    - datepart đại diện một phần của tham số date:
      - ◆ yy, yyyy: năm của date
      - ◆ q, qq quý của date
      - ◆ mm, m: tháng của date
      - ◆ .....

## 5.10 Một số hàm toán học và xử lý chuỗi ký tự, ngày tháng, năm

- ◆ Hàm xử lý ngày tháng năm
  - Hàm DAY(date): trả về ngày của tham số date.
  - Hàm MONTH (date): trả về tháng của tham số date.
  - Hàm YEAR(date): trả về năm của tham số date.