

# TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

## KHOA HỆ THỐNG THÔNG TIN

*Tài liệu bài giảng:*

# MÔN CƠ SỞ DỮ LIỆU

Chương 4:

## NGÔN NGỮ SQL

ThS. Dương Phi Long – Email: [longdp@uit.edu.vn](mailto:longdp@uit.edu.vn)

# NỘI DUNG BÀI HỌC

01



Giới thiệu

02



Ngôn ngữ định nghĩa dữ liệu

03



Ngôn ngữ thao tác dữ liệu

04



Ngôn ngữ truy vấn dữ liệu



# 1

## Giới thiệu Ngôn ngữ SQL

1. Ngôn ngữ SQL
2. Phân loại ngôn ngữ SQL

# 1. Ngôn ngữ SQL (Structured Query Language)

- Ngôn ngữ cấp cao
- Người sử dụng chỉ cần đưa ra nội dung cần truy vấn
- Được phát triển bởi IBM (1970s), khởi nguồn của SQL là SEQUEL - Structured English Query Language (1974)
- Các chuẩn SQL
  - SQL86 (SQL87)
  - SQL89
  - SQL92 (SQL2)
  - SQL99 (SQL3)
  - SQL2003

# 1. Ngôn ngữ SQL (Structured Query Language)

- Sử dụng các thuật ngữ:
  - Bảng (Table) ~ quan hệ
  - Cột (Column) ~ thuộc tính
  - Dòng (Row) ~ bộ

TABLE				
		Rows		
	Columns			

## 2. Phân loại Ngôn ngữ SQL

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



# 2 Ngôn ngữ định nghĩa dữ liệu

1. Một số kiểu dữ liệu
2. Tạo bảng
3. Sửa cấu trúc bảng
4. Xóa bảng

# Ngôn ngữ định nghĩa dữ liệu

- Là ngôn ngữ mô tả
  - Lược đồ cho mỗi quan hệ
  - Miền giá trị tương ứng của từng thuộc tính
  - Ràng buộc toàn vẹn
  - Chỉ mục trên mỗi quan hệ
- Bao gồm: Tạo, sửa và xóa bảng





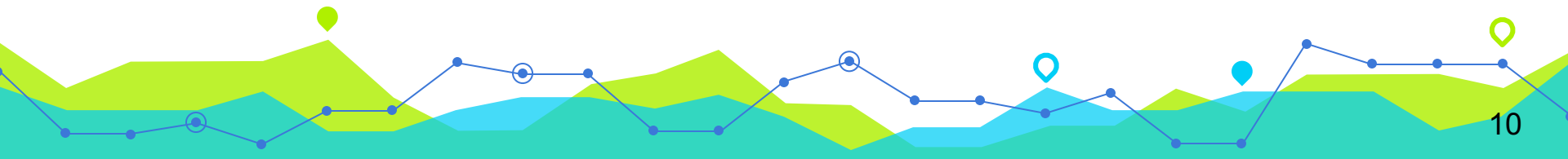
# 1. Một số kiểu dữ liệu

Kiểu dữ liệu	MS SQL Server
Chuỗi ký tự	varchar(n), char(n), nvarchar(n), nchar(n), text, ...
Số	tinyint, smallint, int, bigint, numeric(m, n), decimal(m,n), float, real, smallmoney, money
Ngày tháng	smalldatetime, datetime, ...
Luận lý	Bit (0, 1, NULL)

Tham khảo thêm: [https://www.w3schools.com/sql/sql\\_datatypes.asp](https://www.w3schools.com/sql/sql_datatypes.asp)

## 2. Tạo bảng

- Để định nghĩa một bảng
  - Tên bảng
  - Các thuộc tính
    - Tên thuộc tính
    - Kiểu dữ liệu
    - Các RBTV trên thuộc tính
- Từ khóa: **CREATE**



## 2. Tạo bảng

### - Cú pháp:

Not null, unique, khóa chính...

**CREATE TABLE** <Tên\_bảng>

(

<Tên\_cột 1> <Kiểu\_dữ\_liệu> [<RBTV>],

<Tên\_cột 2> <Kiểu\_dữ\_liệu> [<RBTV>],

...

[<RBTV>] ← Khóa chính, khóa ngoại, ...

)

## 2. Tạo bảng

### - **RBTV:**

- Null, Not null
- Unique
- Default
- Check
- Primary key
- Foreign key

### - **Khai báo RBTV**

CONSTRAINT <Ten RBTV> <RBTV>

### **VD1:**

constraint PK\_SP primary key (MaSP)

constraint FK\_CTHD\_HD foreign key  
(SoHD) references HOADON (SoHD)

## 2. Tạo bảng

Cho Lược đồ CSDL Quản lý bán hàng gồm những quan hệ sau:

**KHACHHANG** (MaKH, HoTen, DChi, SoDT, NgSinh, DoanhSo, NgDK, CMND)

**NHANVIEN** (MaNV, HoTen, NgVL, SoDT)

**SANPHAM** (MaSP, TenSP, DVT, NuocSX, Gia)

**HOADON** (SoHD, NgHD, MaKH, MaNV, TriGia)

**CTHD** (SoHD, MaSP, SoLuong)

## 2. Tạo bảng

**VD2:** Tạo **KHACHHANG** (MaKH, HoTen, DChi, SoDT, NgSinh, DoanhSo, NgDK, CMND)



**CREATE TABLE KHACHHANG (**

MaKH

char(4)

primary key,

HoTen

nvarchar(30)

not null,

DChi

nvarchar(50),

SoDT

varchar(15),

NgSinh

smalldatetime,

DoanhSo

int,

NgDK

smalldatetime,


CMND

varchar(9)


)

## 2. Tạo bảng

**VD3:** Tạo **CTHD** (SoHD, MaSP, SoLuong)



**CREATE TABLE** CTHD (  
SOHD int Primary Key,  
MASP char(4) Primary Key,  
SL int  
)



## 2. Tạo bảng

**VD3:** Tạo **CTHD** (SoHD, MaSP, SoLuong)

**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)

)



## 2. Sửa cấu trúc bảng

- Phân loại:
  - Thêm thuộc tính vào bảng
  - Sửa kiểu dữ liệu của thuộc tính
  - Xóa thuộc tính
  - Thêm RBTV
  - Xóa RBTV
- Từ khóa: **ALTER**

## 2.1. Thêm thuộc tính

- **Cú pháp:**

***ALTER TABLE** <Tên bảng> **ADD** <Tên cột> <Kiểu DL>*

- **VD4:** Thêm cột Ghi\_chu vào bảng KHACHHANG

➡ **ALTER TABLE** KHACHHANG **ADD** GHI\_CHU varchar(20)

## 2.2. Sửa kiểu dữ liệu thuộc tính

- **Cú pháp:** Không sửa được tên thuộc tính

***ALTER TABLE*** <Tên bảng> ***ALTER COLUMN*** <Tên cột> <Kiểu DL mới>

- **VD5:** Sửa Kiểu DL cột Ghi\_chu của bảng KHACHHANG thành nvarchar(50)

**ALTER TABLE** KHACHHANG

**ALTER COLUMN** GHI\_CHU nvarchar(50)

- **Lưu ý:** Không phải sửa thành kiểu dữ liệu nào cũng được (Độ dài dữ liệu, dữ liệu ký tự chuỗi thành ký tự số, ...)

## 2.3. Xóa thuộc tính

- **Cú pháp:**

***ALTER TABLE** <Tên bảng> **DROP COLUMN** <Tên cột>*

- **VD6:** Xóa cột Ghi\_chu trong bảng KHACHHANG



**ALTER TABLE** KHACHHANG **DROP COLUMN** GHI\_CHU

## 2.4. Thêm Ràng buộc Toàn vẹn

- Cú pháp:

**ALTER TABLE** <tên bảng>  
**ADD CONSTRAINT** <tên RBTV>

**UNIQUE** <tên cột>

**PRIMARY KEY** <tên cột>

**FOREIGN KEY** <tên cột>

**REFERENCES** <tên bảng>  
(<tên cột khóa chính>)

**CHECK** (<điều kiện>)

## 2.4. Thêm Ràng buộc Toàn vẹn

- **VD7:** Thêm RB Khóa chính cho NHANVIEN



**ALTER TABLE** NHANVIEN **ADD CONSTRAINT** PK\_NV  
PRIMARY KEY (MaNV)

- **VD8:** Thêm RB Khóa ngoại giữa SANPHAM và CTHD



**ALTER TABLE** CTHD **ADD CONSTRAINT** FK\_CTHD\_SP  
FOREIGN KEY (MaSP) REFERENCES SANPHAM (MaSP)

## 2.4. Thêm Ràng buộc Toàn vẹn

- **VD9:** Thêm RB Giá trong SANPHAM từ 500 trở lên



**ALTER TABLE** SANPHAM

**ADD CONSTRAINT** Check\_Gia CHECK (Gia >= 500)

- **VD10:** Thêm RB CMND của các khách hàng là khác nhau



**ALTER TABLE** KHACHHANG

**ADD CONSTRAINT** Unique\_CMND UNIQUE (CMND)

## 2.5. Xóa Ràng buộc Toàn vẹn

- **Cú pháp:**

***ALTER TABLE** <Tên bảng> **DROP CONSTRAINT** <Tên RBTV>*

- **VD11:** Xóa RB Check\_Gia

 **ALTER TABLE SANPHAM DROP CONSTRAINT** Check\_Gia

- **Lưu ý:** Khi xóa RB Khóa chính phải xóa hết các RB Khóa ngoại tham chiếu tới nó



### 3. Xóa bảng

- Từ khóa: **DROP**
- **Cú pháp:**

***DROP TABLE** <Tên bảng>*

- **VD12:** Xóa bảng SANPHAM



**DROP TABLE** SANPHAM

- **Lưu ý:** Khi xóa một bảng phải xóa hết các RB Khóa ngoại tham chiếu tới bảng đó

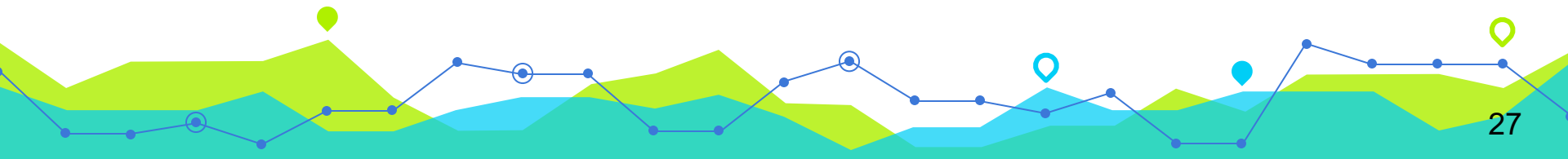


# 3 Ngôn ngữ thao tác dữ liệu

1. Thêm dữ liệu
2. Sửa dữ liệu
3. Xóa dữ liệu

# Ngôn ngữ thao tác dữ liệu

- Là ngôn ngữ cho phép thao tác trên dữ liệu của bảng
- Bao gồm:
  - Thêm dữ liệu vào bảng
  - Cập nhật, sửa dữ liệu của bảng
  - Xóa dữ liệu trong bảng



# 1. Thêm dữ liệu

- Thêm 1 hay nhiều dòng dữ liệu vào bảng
- Cần xác định
  - Tên bảng cần chèn dữ liệu
  - Danh sách các thuộc tính cần thêm dữ liệu
  - Danh sách các giá trị tương ứng
- Từ khóa: **INSERT**

## 1.1. Thêm 01 dòng dữ liệu

- **Cú pháp:**

**INSERT INTO** <Tên bảng> [ (<danh sách các thuộc tính>) ]  
**VALUES** (<danh sách các giá trị>)

- **VD13:**

**INSERT INTO** SANPHAM (MaSP, TenSP, DVT, NuocSX, Gia)  
**VALUES** ('BC01', 'But chi', 'cay', 'Singapore', 3000)

**INSERT INTO** SANPHAM  
**VALUES** ('BC01', 'But chi', 'cay', 'Singapore', 3000)

## 1.2. Thêm nhiều dòng dữ liệu

- **Cú pháp:** (SQL Server 2008+)

**INSERT INTO** <Tên bảng> [ (<danh sách các thuộc tính>) ]  
**VALUES** {(<danh sách các giá trị>) }

- **VD14:**

**INSERT INTO** SANPHAM (MaSP, TenSP, DVT, NuocSX, Gia)  
**VALUES** ('BC01', 'But chi', 'cay', 'Singapore', 3000),  
('BB01', 'But bi', 'cay', 'Thai Lan', 3500),  
('BC02', 'But chi', 'cay', 'Thai Lan', 2500)



## 1.2. Thêm nhiều dòng dữ liệu

- **Cú pháp:**

***INSERT INTO*** <Tên bảng> <Câu truy vấn>

- **VD15:**

**INSERT INTO** SANPHAM\_Sing **Select** \* **from** SANPHAM  
**where** NuocSX= 'Singapore'

Lưu ý: Bảng SANPHAM\_Sing phải được tạo trước khi thực thi câu lệnh trên.

## 1.2. Thêm nhiều dòng dữ liệu

- **Cú pháp:**

**SELECT \* INTO** <Tên bảng mới> **FROM** <Tên bảng có sẵn>

- **VD16:**

**SELECT \* INTO** SANPHAM\_Sing **FROM** SANPHAM  
**WHERE** NuocSX= 'Singapore'

Lưu ý: Bảng SANPHAM\_Sing là một bảng copy từ SANPHAM, không tạo trước khi thực thi câu lệnh trên.



# 1. Thêm dữ liệu

## - Một số lưu ý:

- Thứ tự các cột phải tương ứng với thứ tự các giá trị
- Có thể thêm giá trị NULL ở những thuộc tính không là khóa chính và không là NOT NULL
- Câu lệnh INSERT sẽ **gặp lỗi khi vi phạm RBTV**:
  - Khóa chính
  - Khóa ngoại
  - NOT NULL - các thuộc tính có ràng buộc NOT NULL bắt buộc phải có giá trị

## 2. Sửa dữ liệu

- Để cập nhật 01 hay nhiều dòng dữ liệu trong bảng
- Từ khóa: **UPDATE**
- **Cú pháp:**

**UPDATE** <Tên bảng >

**SET** { <thuộc tính> = <giá trị mới> }

[ **WHERE** <Điều kiện> ]

## 2. Sửa dữ liệu

- **VD17:** Tăng 5% giá tất cả các sản phẩm



**UPDATE** SANPHAM

**SET** Gia = Gia \* 1.05

- **VD18:** Giảm 10% giá của các sản phẩm của Trung Quốc



**UPDATE** SANPHAM

**SET** Gia = Gia \* 0.9

**WHERE** NuocSX= 'Trung Quoc'



## 2. Sửa dữ liệu

### - Một số lưu ý:

- Nếu **có WHERE**, những dòng thỏa điều kiện WHERE sẽ được cập nhật giá trị mới
- Nếu **không có WHERE**, cập nhật tất cả các dòng trong bảng
- Khi thực thi có thể **vi phạm các RBTV khóa ngoại**
  - Không cho cập nhật dữ liệu
  - Cập nhật tự động các dòng có giá trị tham chiếu đến

CASCADE

### 3. Xóa dữ liệu

- Để xóa 01 hay nhiều dòng dữ liệu trong bảng
- Từ khóa: **DELETE**
- **Cú pháp:**

***DELETE FROM*** <Tên bảng >  
***[ WHERE*** <Điều kiện> ***]***

### 3. Xóa dữ liệu

- **VD19:** Xóa tất cả các khách hàng



**DELETE FROM KHACHHANG**

- **VD20:** Xóa các sản phẩm của Trung Quốc và có giá dưới 5000



**DELETE FROM SANPHAM**

**WHERE** NuocSX= 'Trung Quoc'

**AND** Gia < 5000

### 3. Xóa dữ liệu

#### - Một số lưu ý:

- Nếu **có WHERE**, những dòng thỏa điều kiện WHERE sẽ bị xóa
- Nếu **không có WHERE**, xóa tất cả các dòng trong bảng
- Khi thực thi có thể **vi phạm các RBTV khóa ngoại**
  - Không cho xóa
  - Xóa các dòng có giá trị tham chiếu đến CASCADE
  - Cập nhật giá trị Null cho giá trị tham chiếu



# 4 Ngôn ngữ truy vấn dữ liệu

1. Các toán tử, các hàm
2. Truy vấn đơn giản
3. Truy vấn sử dụng toán tử tập hợp
4. Truy vấn lồng
5. Truy vấn sử dụng các hàm tính toán trên nhóm
6. Truy vấn sử dụng kết ngoài



# Ngôn ngữ truy vấn dữ liệu

- Là ngôn ngữ chuẩn, có cấu trúc dùng để truy vấn và thao tác trên CSDL quan hệ.
- **Cú pháp tổng quát:**

**SELECT** [DISTINCT] <Danh sách các thuộc tính / hàm >

**FROM** <Danh sách các bảng>

[ **WHERE** <Điều kiện> ]

[ **GROUP BY** <Danh sách thuộc tính gom nhóm> ]

[ **HAVING** <Điều kiện trên nhóm> ]

[ **ORDER BY** {<Thuộc tính sắp xếp> ASC / DESC } ]

# Ngôn ngữ truy vấn dữ liệu

- **<Danh sách các thuộc tính / hàm >**
  - Tên các thuộc tính/ hàm cần được hiển thị ở kết quả.
  - Các thuộc tính/ hàm cách nhau bởi dấu ` , `
- **<Danh sách các bảng>:**
  - Tên các bảng
  - Các bảng cách nhau bởi dấu ` , `
- **<Điều kiện>:**
  - Sử dụng toán tử so sánh
  - Là biểu thức Boolean để xác định dòng nào sẽ được chọn
  - Sử dụng toán tử logic để nối các điều kiện

# Ngôn ngữ truy vấn dữ liệu

## - Các toán tử:

- Các phép toán cơ bản
- Các toán tử so sánh
- Các toán tử logic

## - Các biểu thức, hàm:

- Các hàm xử lý chuỗi
- Các hàm xử lý số
- Các hàm xử lý ngày
- Các hàm tính toán trên nhóm

# Ngôn ngữ truy vấn dữ liệu

## - Các dạng truy vấn:

- Truy vấn cơ bản, đơn giản
- Truy vấn sử dụng phép toán tập hợp
- Truy vấn lồng
- Truy vấn sử dụng các hàm tính toán trên nhóm
- Truy vấn sử dụng các hàm tính toán trên nhóm và có điều kiện

# 1. Các toán tử, các hàm

- **Các phép toán cơ bản:**  $+$ ,  $-$ ,  $\times$ ,  $\div$
- **Các toán tử so sánh:**
  - $=$ ,  $>$ ,  $<$ ,  $>=$ ,  $<=$ ,  $<>$
  - BETWEEN
  - IS NULL, IS NOT NULL
  - LIKE % hoặc \_: % đại diện nhiều ký tự, \_ đại diện 01 ký tự
  - IN, NOT IN
  - EXISTS, NOT EXISTS
  - SOME, ALL, ANY
- **Toán tử logic:** AND, OR

# 1. Các toán tử, các hàm

## - Một số hàm xử lý chuỗi:

- SUBSTRING ( )
- LEFT ( )
- UPPER ( )
- REPLACE ( )
- CONCAT ( )
- RIGHT ( )
- LOWER ( )
- ...

## - Các hàm xử lý số:

- ABS ( )
- SQRT ( )
- ROUND ( )
- SIGN ( )
- POWER ( )
- ....

## - Các hàm xử lý ngày:

- DAY ( )
- MONTH ( )
- YEAR ( )

## - Các hàm tính toán trên nhóm:

- MIN ( )
- MAX ( )
- AVG ( )
- SUM ( )
- COUNT ( )

## 2. Truy vấn cơ bản

**SELECT** [DISTINCT] <Danh sách các thuộc tính / hàm >  
**FROM** <Danh sách các bảng>  
[ **WHERE** <Điều kiện> ]  
[ **ORDER BY** { <Thuộc tính sắp xếp> ASC / DESC } ]

**VD21:** Liệt kê mã, họ tên của tất cả các nhân viên



**SELECT** MaNV, HoTEN **FROM** NHANVIEN

**VD22:** Liệt kê mã, họ tên các nhân viên thuộc phòng có mã 'PB01'



**SELECT** MaNV, HoTen **FROM** NHANVIEN  
**WHERE** Phong = 'PB01'

## 2. Truy vấn cơ bản

- Sử dụng **\***: Lấy tất cả các thuộc tính

**VD23:** Liệt kê thông tin của tất cả các nhân viên



**SELECT \* FROM NHANVIEN**

- Sử dụng **AS**: Đặt bí danh

**VD24:** Liệt kê mã, họ tên các nhân viên nam



**SELECT** MaNV **AS** 'Mã NV', HoTen **AS** 'Họ và Tên NV'  
**FROM** NHANVIEN  
**WHERE** GT = 'Nam'



## 2. Truy vấn cơ bản

- Sử dụng **Hàm, biểu thức tính toán**

**VD25:** Liệt kê mã, họ tên các nhân viên thuộc phòng có mã 'PB01' và lương tạm tính sau khi tăng 20%



**SELECT** MaNV, HoTen, Luong \*1.2 'Lương tạm tính'

**FROM** NHANVIEN

**WHERE** Phong = 'PB01'

## 2. Truy vấn cơ bản

- Sử dụng **Hàm, biểu thức tính toán**

**VD26:** Liệt kê mã, họ tên và năm sinh của các nhân viên nam



```
SELECT MaNV, HoTen, Year (NgSinh) 'Nam Sinh'  
FROM NHANVIEN  
WHERE GT = 'Nam'
```

## 2. Truy vấn cơ bản

- Sử dụng **DISTINCT**: loại bỏ các dòng trùng nhau trong kết quả

**VD27:** Liệt kê các mức lương của nhân viên. Sắp xếp giảm dần



**SELECT** DISTINCT Lương 'Mức Lương'

**FROM** NHANVIEN

**ORDER BY** Lương DESC  
**ORDER BY** (tên thuộc tính cũ)

Lương
5000
4500
6200
4500
9000
5500
6400



Mức Lương
9000
6400
6200
5500
5000
4500

## 2. Truy vấn cơ bản

- Sử dụng **Toán tử so sánh** và **Toán tử logic**:

**VD28:** Liệt kê các nhân viên nam có lương trên 5.000.000



**SELECT** MaNV, HoTen

**FROM** NHANVIEN

**WHERE** GT = 'Nam' **AND** Luong > 5000000

## 2. Truy vấn cơ bản

- Sử dụng **Toán tử so sánh** và **Toán tử logic**:

**VD29:** Tìm các nhân viên có họ là Nguyen, tên là An



**SELECT** MaNV, HoTen

**FROM** NHANVIEN

**WHERE** HoTen **LIKE** 'Nguyen%An'

## 2. Truy vấn cơ bản

**VD30:** Liệt kê các nhân viên thuộc phòng có mã là 'PB01' hoặc 'PB02'



**Cách 1:** **SELECT** MaNV, HoTen

**FROM** NHANVIEN

**WHERE** Phong = 'PB01' **OR** Phong = 'PB02'

**Cách 2:** **SELECT** MaNV, HoTen

**FROM** NHANVIEN

**WHERE** Phong **IN** ('PB01', 'PB02')

## 2. Truy vấn cơ bản

- Sử dụng **Phép kết bằng**: Để kết 02 hay nhiều bảng
  - Cách 1: Sử dụng **inner join on** <điều kiện kết> ở **FROM**
  - Cách 2: Đưa <điều kiện kết> xuống **WHERE**

**VD31:** Liệt kê các nhân viên và tên phòng ban của nhân viên



**Cách 1:** **SELECT** MaNV, HoTen, TenPH

**FROM** NHANVIEN **inner join** PHONGBAN

**on** NHANVIEN.Phong = PHONGBAN.MaPH

## 2. Truy vấn cơ bản

- Sử dụng **Phép kết bằng**:

**VD31:** Liệt kê các nhân viên và tên phòng ban của nhân viên




**Cách 2:** **SELECT** MaNV, HoTen, TenPH  
**FROM** NHANVIEN, PHONGBAN  
**WHERE** NHANVIEN.Phong = PHONGBAN.MaPH

**Hoặc:** **SELECT** MaNV, HoTen, TenPH  
**FROM** NHANVIEN NV, PHONGBAN PB  
**WHERE** NV.Phong = PB.MaPH



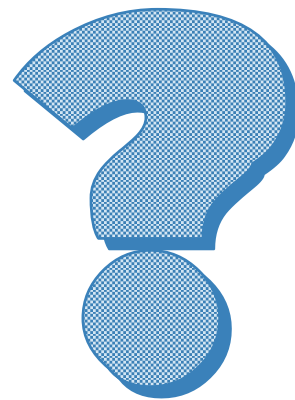
## 2. Truy vấn cơ bản

**VD32:** Liệt kê các nhân viên đang làm việc tại phòng Kế toán



```
SELECT MaNV, HoTen, TenPH
FROM NHANVIEN NV, PHONGBAN PB
WHERE NV.Phong = PB.MaPH
AND TenPH = 'Kế toán'
```

```
SELECT MaNV, HoTen, TenPH
FROM NHANVIEN NV, PHONGBAN PB
WHERE NV.Phong = PB.MaPH
AND TenPH = n'Kế toán'
```



**NHẬN XÉT**

## 2. Truy vấn cơ bản

**VD33:** Liệt kê các nhân viên có lương từ 5.000.000 đến 8.000.000



**Cách 1:** **SELECT** MaNV, HoTen, Luong  
**FROM** NHANVIEN  
**WHERE** Luong >= 5000000 **AND** Luong <=8000000

**Cách 2:** **SELECT** MaNV, HoTen, Luong  
**FROM** NHANVIEN  
**WHERE** Luong **Between** 5000000 **and** 8000000

## 2. Truy vấn cơ bản

**VD34:** Liệt kê các nhân viên có người quản lý trực tiếp



**SELECT** MaNV, HoTen

**FROM** NHANVIEN

**WHERE** MaNQL IS NOT NULL

## 2. Truy vấn cơ bản

**VD35:** Liệt kê các nhân viên sinh năm 1990

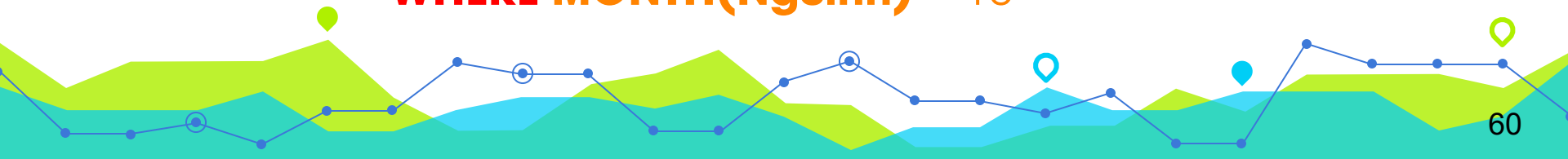


```
SELECT MaNV, HoTen  
FROM NHANVIEN  
WHERE YEAR(NgSinh) = 1990
```

**VD36:** Liệt kê các nhân viên có ngày sinh trong tháng 10



```
SELECT MaNV, HoTen  
FROM NHANVIEN  
WHERE MONTH(NgSinh) = 10
```



## 2. Truy vấn cơ bản

**VD37:** Liệt kê các nhân viên nữ và sắp xếp theo lương giảm dần,  
Mã nhân viên tăng dần



**SELECT** MaNV, HoTen, Luong

**FROM** NHANVIEN

**WHERE** GT = 'Nu'

**ORDER BY** Luong **DESC**, MaNV **ASC**

### 3. Truy vấn sử dụng phép toán tập hợp

- Sử dụng phép toán tập hợp để kết hợp kết quả của 02 câu truy vấn

(Câu truy vấn 1)

<PHÉP TOÁN TẬP HỢP>

(Câu truy vấn 2)

- Các phép toán tập hợp:
  - Hội: **UNION**
  - Giao: **INTERSECT**
  - Trừ: **EXCEPT**

### 3. Truy vấn sử dụng phép toán tập hợp

**VD38:** Liệt kê các mã nhân viên thực hiện đề án DA01 hoặc DA02



```
(SELECT MaNV  
FROM PHANCONG  
WHERE MaDA = 'DA01')  
UNION  
(SELECT MaNV  
FROM PHANCONG  
WHERE MaDA = 'DA02')
```

### 3. Truy vấn sử dụng phép toán tập hợp

**VD39:** Liệt kê các mã nhân viên thực hiện cả 2 đề án DA01 và DA02



**Cách 1:** (**SELECT** MaNV

**FROM** PHANCONG

**WHERE** MaDA = 'DA01')

**INTERSECT**

(**SELECT** MaNV

**FROM** PHANCONG

**WHERE** MaDA = 'DA02')



### 3. Truy vấn sử dụng phép toán tập hợp

**VD40:** Liệt kê các mã nhân viên thực hiện đề án DA01 nhưng không thực hiện đề án DA02



**Cách 1:** (**SELECT** MaNV  
**FROM** PHANCONG  
**WHERE** MaDA = 'DA01')  
**EXCEPT**  
(**SELECT** MaNV  
**FROM** PHANCONG  
**WHERE** MaDA = 'DA02')

### 3. Truy vấn sử dụng phép toán tập hợp

**VD41:** Cho biết những nhân viên không tham gia đề án nào



**Cách 1:** (**SELECT** MaNV, HoTen

**FROM** NHANVIEN)

**EXCEPT**

(**SELECT** NV.MaNV, HoTen

**FROM** NHANVIEN NV, PHANCONG PC

**WHERE** NV.MaNV = PC. MaNV)

## 4. Truy vấn lồng

- Gồm 2 hay nhiều câu truy vấn lồng vào nhau

**SELECT** [DISTINCT] <Danh sách các thuộc tính / hàm >  
**FROM** <Danh sách các bảng>  
**WHERE** <Điều kiện> (**Câu truy vấn con**)

## 4. Truy vấn lồng

- **<Điều kiện>**: Truy vấn con trả về giá trị tập hợp
  - <biểu thức> **[NOT] IN** (<truy vấn con>)
  - <biểu thức> <phép toán so sánh> **ANY** (<truy vấn con>)
  - <biểu thức> <phép toán so sánh> **ALL** (<truy vấn con>)
- **<Điều kiện>**: Kiểm tra sự tồn tại trong kết quả của truy vấn con
  - **[NOT] EXISTS** (<truy vấn con>):
    - Trả về True nếu có ít nhất một bộ trong truy vấn con
    - Trả về False nếu ngược lại.

## 4. Truy vấn lồng

**VD39:** Liệt kê các mã nhân viên thực hiện cả 2 đề án DA01 và DA02



Cách 2:

**SELECT** MaNV

**FROM** PHANCONG

**WHERE** MaDA = 'DA01'

**AND** MaNV **IN** (**SELECT** MaNV

**FROM** PHANCONG

**WHERE** MaDA = 'DA02')

## 4. Truy vấn lồng

**VD39:** Liệt kê các mã nhân viên thực hiện cả 2 đề án DA01 và DA02



Cách 3: **SELECT** MaNV

**FROM** PHANCONG PC\_DA1

**WHERE** MaDA = 'DA01'

**AND EXISTS (SELECT \***

**FROM** PHANCONG PC\_DA2

**WHERE** MaDA = 'DA02'

**AND PC\_DA2.MaNV = PC\_DA1.MaNV)**

## 4. Truy vấn lồng

**VD39:** Liệt kê các mã nhân viên thực hiện cả 2 đề án DA01 và DA02

**SELECT** MaNV

**FROM** PHANCONG PC\_DA1

**WHERE** MaDA = 'DA01'

**AND EXISTS (SELECT \***

**FROM** PHANCONG PC\_DA2 → **FALSE**

**WHERE** MaDA = 'DA02'

**AND PC\_DA2.MaNV = PC\_DA1.MaNV)**

PC_DA1	
MaNV	MaDA
NV01	DA01
NV03	DA01

PC_DA2	
MaNV	MaDA
NV01	DA02
NV04	DA02

'NV01'

'NV03'

## 4. Truy vấn lồng

**VD40:** Liệt kê các mã nhân viên thực hiện đề án DA01 nhưng không thực hiện đề án DA02



Cách 2:

**SELECT** MaNV

**FROM** PHANCONG

**WHERE** MaDA = 'DA01'

**AND** MaNV **NOT IN** (**SELECT** MaNV

**FROM** PHANCONG

**WHERE** MaDA = 'DA02')



## 4. Truy vấn lồng

**VD40:** Liệt kê các mã nhân viên thực hiện đề án DA01 nhưng không thực hiện đề án DA02



Cách 3:

```
SELECT MaNV
FROM PHANCONG PC_DA1
WHERE MaDA = 'DA01'
AND NOT EXISTS (SELECT *
FROM PHANCONG PC_DA2
WHERE MaDA = 'DA02'
AND PC_DA2.MaNV = PC_DA1.MaNV)
```

## 4. Truy vấn lồng

**VD41:** Cho biết những nhân viên không tham gia đề án nào



**Cách 2:** **SELECT** MaNV, HoTen

**FROM** NHANVIEN

**WHERE** MaNV **NOT IN** (**SELECT** MaNV

**FROM** PHANCONG)

## 4. Truy vấn lồng

**VD41:** Cho biết những nhân viên không tham gia đề án nào



Cách 3: **SELECT** MaNV, HoTen

**FROM** NHANVIEN NV

**WHERE NOT EXISTS (SELECT \***

**FROM** PHANCONG PC

**WHERE** NV.MaNV = PC. MaNV)

## 4. Truy vấn lồng

**VD42:** Tìm nhân viên có lương cao hơn lương của tất cả nhân viên thuộc phòng Kế toán



**(SELECT** MaNV, HoTen

**FROM** NHANVIEN

**WHERE** Luong > **ALL** (**SELECT** Luong

**FROM** NHANVIEN NV, PHONGBAN PB

**WHERE** NV.Phong = PB.MaPH

AND TenPH = 'Kế Toán')

## 4. Truy vấn lồng

**VD43:** Tìm nhân viên tham gia tất cả các đề án của công ty

NHANVIEN	
MaNV	HoTen
NV01	Nguyễn Tuyết An
NV02	Trần Ngọc Minh
NV03	Phạm Tiến Dũng

DEAN	
MaDA	TenDA
DA01	Hệ thống quản lý sinh viên
DA02	Hệ thống quản lý tín dụng ngân hàng

PHANCONG		
MaNV	MaDA	ThoiGian
NV01	DA01	20
NV01	DA02	30
NV03	DA01	20

## 4. Truy vấn lồng

**VD43:** Tìm nhân viên tham gia tất cả các đề án của công ty

 **Cách 1:**

```
SELECT MaNV, HoTen  
FROM NHANVIEN NV  
WHERE NOT EXISTS
```

```
(SELECT *
```

```
FROM DEAN DA
```

```
WHERE NOT EXISTS (SELECT *
```

```
FROM PHANCONG PC
```

```
WHERE PC.MaDA = DA.MaDA
```

```
AND PC.MaNV = NV.MaNV))
```

Tìm những đề án  
mà NV01 không  
thực hiện?

Tìm nhân viên **không** thuộc  
danh sách những nhân viên  
**không** thực hiện  
ít nhất 1 đề án?

## 4. Truy vấn lồng

**VD43:** Tìm nhân viên tham gia tất cả các đề án của công ty



Cách 1:

**SELECT** MaNV, HoTen

**FROM** NHANVIEN NV

**WHERE NOT EXISTS (SELECT \***

**FROM** DEAN DA

**WHERE NOT EXISTS (SELECT \***

**FROM** PHANCONG PC

**WHERE** PC.MaDA = DA.MaDA

**AND** PC.MaNV = NV.MaNV ))

## 5. Truy vấn sử dụng hàm tính toán trên nhóm

- Sử dụng các hàm tính toán, có thể có gom nhóm dữ liệu và tính toán trên nhóm
- Các hàm có đầu vào là một tập giá trị và trả về một giá trị đơn:  
**Min()**, **Max()**, **Avg()**, **Sum()**, **Count()**

**SELECT** [DISTINCT] <Danh sách các thuộc tính / hàm >  
**FROM** <Danh sách các bảng>  
[ **WHERE** <Điều kiện> ]  
[ **GROUP BY** <Danh sách thuộc tính gom nhóm> ]  
[ **HAVING** <Điều kiện trên nhóm> ]  
[ **ORDER BY** {<Thuộc tính sắp xếp> ASC / DESC }]



## 5. Truy vấn sử dụng hàm tính toán trên nhóm

**VD44:** Tìm lương cao nhất, thấp nhất, trung bình của công ty



**SELECT** Max (Luong), Min (Luong), Avg (Luong)  
**FROM** NHANVIEN

Phong	Luong
PH01	5000
PH01	4500
PH01	6200
PH02	4500
PH03	9000
PH03	5500
PH03	6400

Max(Luong)	Min(Luong)	Avg(Luong)
9000	4500	5781.4

## 5. Truy vấn sử dụng hàm tính toán trên nhóm

**VD45:** Tìm lương cao nhất, thấp nhất, trung bình của phòng có mã là PH01



**SELECT** Phong, **Max** (Luong), **Min** (Luong), **Avg** (Luong)  
**FROM** NHANVIEN  
**WHERE** Phong = 'PB01'

Phong	Luong
PH01	5000
PH01	4500
PH01	6200
<del>PH02</del>	<del>4500</del>
<del>PH03</del>	<del>9000</del>
<del>PH03</del>	<del>5500</del>
<del>PH03</del>	<del>6400</del>

Phong	Max(Luong)	Min(Luong)	Avg(Luong)
PH01	5000	4500	5233

## 5. Truy vấn sử dụng hàm tính toán trên nhóm

**VD46:** Tìm mức lương cao nhất, thấp nhất, trung bình của mỗi phòng ban trong công ty



**SELECT** Phong, **Max** (Luong), **Min** (Luong), **Avg** (Luong)  
**FROM** NHANVIEN  
**GROUP BY** Phong

Phong	Luong
PH01	5000
PH01	4500
PH01	6200
PH02	4500
PH03	9000
PH03	5500
PH03	6400

Phong	Max(Luong)	Min(Luong)	Avg(Luong)
PH01	5000	4500	5233
PH02	4500	4500	4500
PH03	9000	5500	6966

## 5. Truy vấn sử dụng hàm tính toán trên nhóm

**VD47:** Thống kê số lượng nhân viên của mỗi phòng. Lấy ra thông tin mã phòng, tên phòng, số lượng nhân viên



```
SELECT PB.MaPH, TenPH, Count (MaNV) 'SLNV'  
FROM NHANVIEN NV, PHONGBAN PB  
WHERE NV.Phong = PB.MaPH  
GROUP BY MaPH, TenPH
```

MaPH	TenPH	SLNV
PB01	Nhan Su	20
PB02	Ke Toan	12
PB03	CNTT	20
PB04	Truyen thong	9
PB05	Hanh Chinh	14

## 5. Truy vấn sử dụng hàm tính toán trên nhóm

**VD48:** Tìm phòng ban có số lượng nhân viên lớn hơn 10. Lấy ra thông tin mã phòng, tên phòng, số lượng nhân viên. Sắp xếp theo số lượng nhân viên giảm dần.




```
SELECT PB.MaPH, TenPH, Count (MaNV) 'SLNV'  
FROM NHANVIEN NV, PHONGBAN PB  
WHERE NV.Phong = PB.MaPH  
GROUP BY MaPH, TenPH  
HAVING Count (MaNV) > 10  
ORDER BY Count (MaNV) DESC
```

MaPH	TenPH	SLNV
PB01	Nhan Su	20
PB03	CNTT	20
PB05	Hanh Chinh	14
PB02	Ke Toan	12
<del>PB04</del>	<del>Truyen thong</del>	<del>9</del>

## 5. Truy vấn sử dụng hàm tính toán trên nhóm

**VD49:** Tìm 3 phòng ban có số lượng nhân viên nhiều nhất. Lấy ra thông tin mã phòng, tên phòng, số lượng nhân viên.



```
SELECT TOP 3 PB.MaPH, TenPH, Count (MaNV) 'SLNV'  
FROM NHANVIEN NV, PHONGBAN PB  
WHERE NV.Phong = PB.MaPH  
GROUP BY MaPH, TenPH  
ORDER BY Count (MaNV) DESC
```

MaPH	TenPH	SLNV
PB01	Nhan Su	20
PB03	CNTT	20
PB05	Hanh Chinh	14
<del>PB02</del>	<del>Ke Toan</del>	<del>12</del>
<del>PB04</del>	<del>Truyen thong</del>	<del>9</del>

## 5. Truy vấn sử dụng hàm tính toán trên nhóm

**VD50:** Tìm các phòng ban có số lượng nhân viên nhiều nhất. Lấy ra thông tin mã phòng, tên phòng, số lượng nhân viên.

 **Cách 1:**

```
SELECT PB.MaPH, TenPH, Count (MaNV) 'SLNV'  
FROM NHANVIEN NV, PHONGBAN PB  
WHERE NV.Phong = PB.MaPH  
GROUP BY MaPH, TenPH  
HAVING Count (MaNV) >= ALL (SELECT Count (MaNV)  
                                FROM NHANVIEN  
                                GROUP BY Phong)
```

MaPH	TenPH	SLNV
PB01	Nhan Su	20
PB03	CNTT	20
<del>PB05</del>	<del>Hanh Chinh</del>	<del>14</del>
<del>PB02</del>	<del>Ke Toan</del>	<del>12</del>
<del>PB04</del>	<del>Truyen thong</del>	<del>9</del>

## 5. Truy vấn sử dụng hàm tính toán trên nhóm

**VD50:** Tìm các phòng ban có số lượng nhân viên nhiều nhất. Lấy ra thông tin mã phòng, tên phòng, số lượng nhân viên.

 **Cách 2:**

**SELECT TOP 1 WITH TIES** PB.MaPH, TenPH, **Count** (MaNV) 'SLNV'

**FROM** NHANVIEN NV, PHONGBAN PB

**WHERE** NV.Phong = PB.MaPH

**GROUP BY** MaPH, TenPH

**ORDER BY Count** (MaNV) DESC

MaPH	TenPH	SLNV
PB01	Nhan Su	20
PB03	CNTT	20
<del>PB05</del>	<del>Hanh Chinh</del>	<del>14</del>
<del>PB02</del>	<del>Ke Toan</del>	<del>12</del>
<del>PB04</del>	<del>Truyen thong</del>	<del>9</del>




## 5. Truy vấn sử dụng hàm tính toán trên nhóm

**VD51:** Thống kê số lượng đề án mà từng nhân viên đã tham gia

NHANVIEN	
MaNV	HoTen
NV01	Nguyễn Tuyết An
NV02	Trần Ngọc Minh
NV03	Phạm Tiến Dũng

PHANCONG		
MaNV	MaDA	ThoiGian
NV01	DA01	20
NV01	DA02	30
NV03	DA01	20



```
SELECT NV.MaNV, HoTen, Count (MaDA) 'SLDATG'  
FROM NHANVIEN NV, PHANCONG PC  
WHERE NV.MaNV = PC.MaNV  
GROUP BY NV.MaNV, HoTen
```

KẾT QUẢ		
MaNV	HoTen	SLDATG
NV01	Nguyễn Tuyết An	2
NV02	Trần Ngọc Minh	0
NV03	Phạm Tiến Dũng	1

# 5. Truy vấn sử dụng hàm tính toán trên nhóm

**VD43:** Tìm nhân viên tham gia tất cả các đề án của công ty

NHANVIEN	
MaNV	HoTen
NV01	Nguyễn Tuyết An
NV02	Trần Ngọc Minh
NV03	Phạm Tiến Dũng

DEAN	
MaDA	TenDA
DA01	Hệ thống quản lý sinh viên
DA02	Hệ thống quản lý tín dụng ngân hàng

PHANCONG		
MaNV	MaDA	ThoiGian
NV01	DA01	20
NV01	DA02	30
NV03	DA01	20

**Cách 2:**

**SELECT** NV.MaNV, HoTen, **Count** (MaDA) 'SLDATG'  
**FROM** NHANVIEN NV, PHANCONG PC  
**WHERE** NV.MaNV = PC.MaNV  
**GROUP BY** NV.MaNV, HoTen  
**HAVING** **Count** (MaDA) = (**SELECT** **Count** (MaDA)  
**FROM** DEAN)

KẾT QUẢ		
MaNV	HoTen	SLDATG
NV01	Nguyễn Tuyết An	2
<del>NV02</del>	<del>Trần Ngọc Minh</del>	<del>0</del>
<del>NV03</del>	<del>Phạm Tiến Dũng</del>	<del>1</del>

## 6. Truy vấn sử dụng phép kết ngoài

- Sử dụng kết ngoài trong một số trường hợp để hạn chế việc bỏ sót dữ liệu
- Cách thức sử dụng tương tự inner join trong kết bằng.

- **Phép kết trái:**

Table1 **left outer join** Table2 **on** <điều kiện kết>

- **Phép kết phải:**

Table1 **right outer join** Table2 **on** <điều kiện kết>

- **Phép kết đầy đủ:**

Table1 **full outer join** Table2 **on** <điều kiện kết>

## 6. Truy vấn sử dụng phép kết ngoài

**VD41:** Cho biết những nhân viên **không** tham gia đề án nào

NHANVIEN				
MaNV	HoTen	NgaySinh	GT	Phong
NV01	Nguyễn Tuyết An	01/10/1978	Nữ	PB01
NV02	Trần Ngọc Minh	25/01/1987	Nữ	PB01
NV03	Phạm Tiến Dũng	12/12/1982	Nam	PB02

PHANCONG		
MaNV	MaDA	ThoiGian
NV01	DA01	20
NV01	DA02	30
NV03	DA01	20

Cách 4:



<del>NHANVIEN</del> <del>PHANCONG</del> MaNV							
MaNV	HoTen	NgaySinh	GT	Phong	MaNV	MaDA	ThoiGian
NV01	Nguyễn Tuyết An	01/10/1978	Nữ	PB01	NV01	DA01	20
NV01	Nguyễn Tuyết An	01/10/1978	Nữ	PB01	NV01	DA02	30
NV03	Phạm Tiến Dũng	12/12/1982	Nam	PB02	NV03	DA01	20
NV02	Trần Ngọc Minh	25/01/1987	Nữ	PB01	NULL	NULL	NULL

## 6. Truy vấn sử dụng phép kết ngoài

**VD41:** Cho biết những mã nhân viên **không** tham gia đề án nào



**Cách 4:**

**SELECT** NV.MaNV, HoTen

**FROM** NHANVIEN NV **left outer join** PHANCONG PC

**on** NV.MaNV = PC.MaNV

**WHERE** MaDA **IS NULL**

# Tổng kết chương



## Giới thiệu

1. Ngôn ngữ SQL
2. Phân loại ngôn ngữ SQL



## Ngôn ngữ định nghĩa dữ liệu

1. Một số kiểu dữ liệu
2. Tạo bảng
3. Sửa cấu trúc bảng
4. Xóa bảng



# Tổng kết chương



## Ngôn ngữ thao tác dữ liệu

1. Thêm dữ liệu
2. Sửa dữ liệu
3. Xóa dữ liệu



## Ngôn ngữ Truy vấn dữ liệu

1. Các toán tử, các hàm
2. Truy vấn cơ bản
3. Truy vấn sử dụng toán tử tập hợp
4. Truy vấn lồng
5. Truy vấn sử dụng hàm tính toán trên nhóm
6. Truy vấn sử dụng kết ngoài



# THANKS!

**Any questions?**

