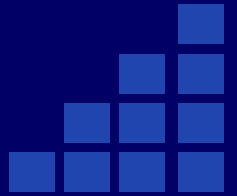
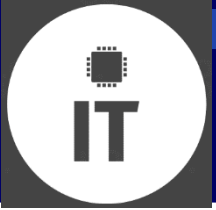


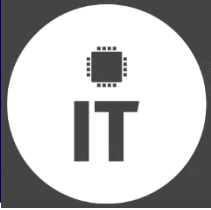


HỌC PHẦN: CƠ SỞ DỮ LIỆU

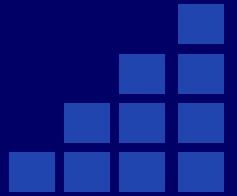
Trần Thị Dung



Ngôn ngữ truy vấn dữ liệu (Structured Query Language)

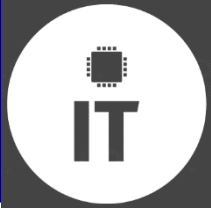


Mục tiêu

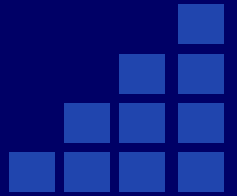


Giúp học viên có thể:

- Tạo bảng dữ liệu cho hệ CSDL
- Thay đổi cấu trúc bảng, thực hiện các tính toán thống kê và lấy dữ liệu từ một hệ quản trị cơ sở dữ liệu quan hệ
- Truy vấn dữ liệu



Khái quát về ngôn ngữ dữ liệu SQL

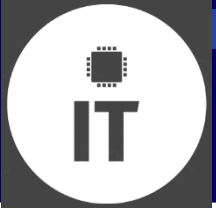


❖ Ngôn ngữ ĐSQH

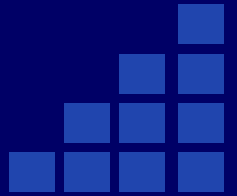
- Cách thức truy vấn dữ liệu
- Khó khăn cho người sử dụng

❖ 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)
- Được gọi là SEQUEL
- Được ANSI công nhận và phát triển thành chuẩn
 - SQL-86, SQL-92, SQL-99



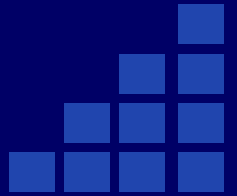
Khái quát về ngôn ngữ dữ liệu SQL



❖ SQL sử dụng thuật ngữ

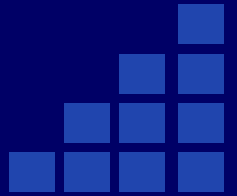
- Bảng ~ quan hệ
- Cột ~ thuộc tính
- Dòng ~ bộ

SQL không phân biệt chữ hoa hay chữ thường



❖ Số (numeric)

- INTEGER: TINYINT (-127 -> 128), SMALLINT, INT, MEDIUMINT, BIGINT (UNSIGNED)
- SMALLINT
- NUMERIC, NUMERIC(p), NUMERIC(p,s)
- DECIMAL, DECIMAL(p), DECIMAL(p,s)
- REAL
- DOUBLE PRECISION
- FLOAT, FLOAT(p)



❖ Chuỗi ký tự (character string)

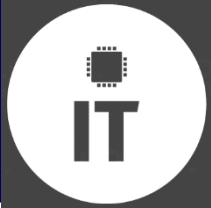
- CHARACTER, CHARACTER(n)
- CHARACTER VARYING(x)
- CHAR, VARCHAR, NVARCHAR
- ENUM

❖ Chuỗi bit (bit string)

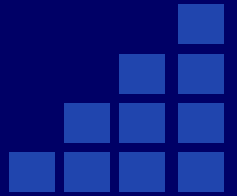
- BIT, BIT(x)
- BIT VARYING(x)

❖ Ngày giờ (datetime)

- DATE gồm ngày, tháng và năm
- TIME gồm giờ, phút và giây
- TIMESTAMP gồm ngày và giờ



Các toán tử



1. Các toán tử số học:

$^$ (lũy thừa), $*$ (nhân), $/$ (chia), mod (phần dư), $+$ (cộng), $-$ (trừ)

2. Các toán tử luận lý:

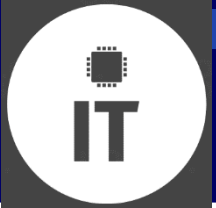
not (phủ định), and (phép hội), or (phép tuyển)

3. Các toán tử tập hợp:

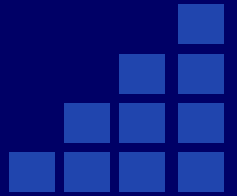
- ✓ In (danh sách các giá trị),
- ✓ LIKE, NOT LIKE,
- ✓ union (phép hợp), intersect (phép giao), minus (phép trừ)

4. Các toán tử so sánh :

$=$, $<>$, $>$, $<$, $>=$, $<=$



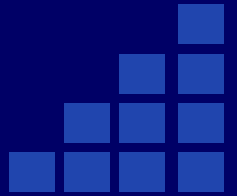
Lệnh tạo cơ sở dữ liệu



CREATE DATABASE [IF NOT EXISTS] tenDB;

❖ Ví dụ:

CREATE DATABASE QuanLyBanHang;

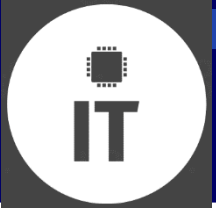


❖ Để định nghĩa một bảng

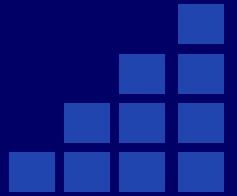
- Tên bảng
- Các thuộc tính
 - Tên thuộc tính
 - Kiểu dữ liệu
 - Các RBTV trên thuộc tính

❖ Cú pháp

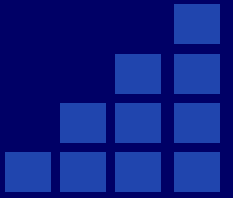
```
CREATE TABLE <Tên_bảng> (  
    <Tên_cột 1> <Kiểu_dữ_liệu> [<RBTV>],  
    <Tên_cột 2> <Kiểu_dữ_liệu> [<RBTV>],  
    ...  
    [<RBTV>]  
)
```



Ví dụ - Tạo bảng



```
Create table ChitietHD(  
    Sohd char(5)  
    Mah char(5)  
    Slban real  
);
```

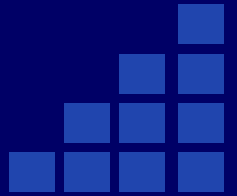
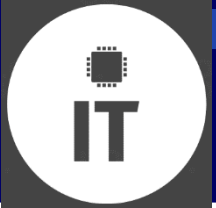


❖ <RBTV>

- NOT NULL
- NULL
- UNIQUE
- DEFAULT
- PRIMARY KEY
- FOREIGN KEY / REFERENCES
- CHECK

❖ Đặt tên cho RBTV

CONSTRAINT <Ten_RBTV> <RBTV>

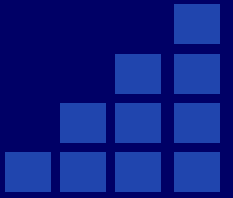


Tạo bảng HANG:

```
Create table HANG(  
    Mah char(5) not Null,  
    Tenh char(30),  
    Slton integer,  
    Primary Key (Mah),  
    Check (Slton>=0)) ;
```



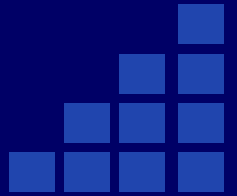
Ví dụ - RBTV



Create table ChitietHD(
Sohd char(5) not null,
Mah char(5) not null,
Slban real,
Constraint PK primary key (Sohd, Mah),
Foreign Key (soHD) References HOADON,
Check (Slban>=0)) ;



Lệnh sửa bảng



❖ Thêm cột

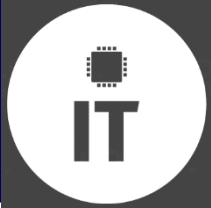
```
ALTER TABLE <Tên_bảng> ADD  
    <Tên_cột> <Kiểu_dữ_liệu> [<RBTV>]
```

❖ Xóa cột

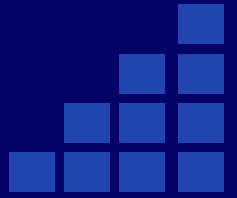
```
ALTER TABLE <Tên_bảng> DROP COLUMN <Tên_cột>
```

❖ Mở rộng cột

```
ALTER TABLE <Tên_bảng> ALTER COLUMN  
    <Tên_cột> <Kiểu_dữ_liệu_mới>
```



Lệnh sửa bảng (tt)



Ví dụ:

- Thêm cột Giới tính vào bảng KHACH

Alter table KHACH **Add** GT char(3);

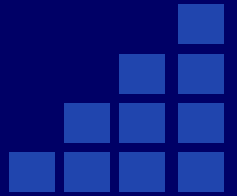
- Thay đổi độ rộng của cột địa chỉ trong bảng khachhang

Alter table KHACH

Alter column Diachi char(40);

- Xoá bỏ cột GT trong bảng KHACH

Alter table KHACH **Drop column** GT;



❖ Được dùng để xóa cấu trúc bảng

- Tất cả dữ liệu của bảng cũng bị xóa

❖ Cú pháp

```
DROP TABLE <Tên_bảng>
```

❖ Ví dụ

```
DROP TABLE NHANVIEN
```

```
DROP TABLE PHONGBAN
```

Lệnh xóa bảng (tt)

NHANVIEN

HONV	TENLOT	TENNV	<u>MANV</u>	NGSINH	DCHI	PHAI	LUONG	MA_NQL	PHG
------	--------	-------	-------------	--------	------	------	-------	--------	-----

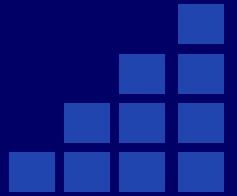
PHONGBAN

TENPHG	<u>MAPHG</u>	TRPHG	NG_NHANCHUC
--------	--------------	-------	-------------





Các lệnh cập nhật CSDL



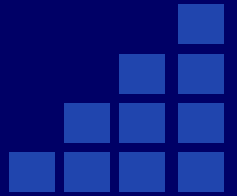
- ❖ **Bổ sung bộ mới**
- ❖ **Sửa nội dung của bộ**
- ❖ **Xoá bộ**

Lưu ý:

- Kiểu dữ liệu
- Chiều dài
- Các ràng buộc toàn vẹn: khóa chính, khóa ngoại, kiểm tra, duy nhất



Lệnh INSERT (bổ sung bộ mới)



- ❖ Dùng để thêm 1 hay nhiều dòng vào bảng
- ❖ **Cú pháp** (thêm 1 dòng)

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

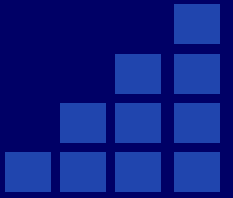


Lệnh INSERT (tt)

❖ Ví dụ

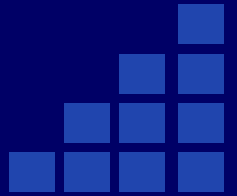
```
INSERT INTO NHANVIEN(HONV, TENLOT, TENNV, MANV)  
VALUES ('Le', 'Van', 'Tuyen', '635635635')
```

```
INSERT INTO NHANVIEN(HONV, TENLOT, TENNV, MANV, DCHI)  
VALUES ('Le', 'Van', 'Tuyen', '635635635', NULL)
```



❖ Nhận xét

- Thứ tự các giá trị phải trùng với thứ tự các cột
- Có thể thêm giá trị NULL ở những thuộc tính không là khóa chính và NOT NULL
- Câu lệnh INSERT sẽ gặp lỗi nếu vi phạm RBTV
 - Khóa chính
 - Tham chiếu
 - NOT NULL - các thuộc tính có ràng buộc NOT NULL bắt buộc phải có giá trị



❖ Cú pháp (thêm nhiều dòng)

INSERT INTO <tên bảng>(<danh sách các thuộc tính>)
<câu truy vấn con>

Hoặc

INSERT INTO <tên bảng>
<câu truy vấn con>

(Nếu giá trị của các biểu thức sau từ khoá SELECT hoàn toàn phù hợp về số lượng, miền giá trị và thứ tự của các cột trong bảng)

Ví dụ: *Thêm Sinh viên mới từ 1 danh sách:*

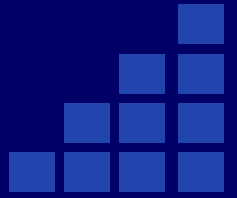
```
INSERT INTO SINHVIEN  
SELECT *  
FROM DSSV_NEW ;
```

Lưu ý:

DSSV_New có cùng cấu trúc với SINHVIEN



Lệnh DELETE (Xóa bộ)

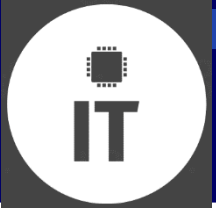


- ❖ Dùng để xóa các dòng của bảng

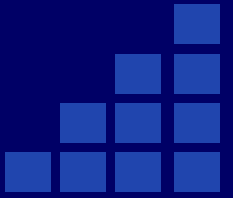
- ❖ **Cú pháp:**

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

- ❖ Các bản ghi thỏa mãn điều kiện Where sẽ bị xóa khỏi bảng



Lệnh DELETE (tt)

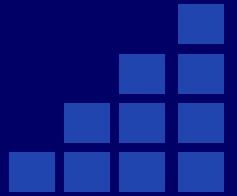


❖ **Ví dụ:** Xóa đi những nhân viên ở phòng 'Nghien cuu'

```
DELETE FROM NHANVIEN  
WHERE MAPHG IN (  
    SELECT MAPHG  
    FROM PHONGBAN  
    WHERE TENPHG='Nghien cuu')
```



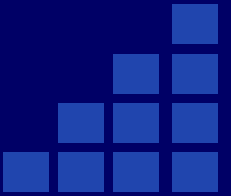
Lệnh DELETE (tt)



❖ Nhận xét

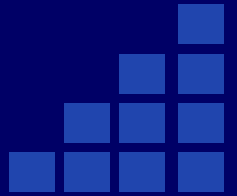
- Số lượng số dòng bị xóa phụ thuộc vào điều kiện ở mệnh đề WHERE
- Nếu không chỉ định điều kiện ở mệnh đề WHERE, tất cả các dòng trong bảng sẽ bị xóa
- Lệnh DELETE có thể gây ra vi phạm RB tham chiếu
 - Không cho xóa
 - Xóa luôn những dòng có giá trị đang tham chiếu đến
 - CASCADE
 - Đặt NULL cho những giá trị tham chiếu

Lệnh DELETE (tt)



MANV	HONV	TENLOT	TENNV	NGSINH	DCHI	PHAI	LUONG	MA_NQL	PHG
333445555	Nguyen	Thanh	Tung	12/08/1955	638 NVC Q5	Nam	40000	888665555	5
987987988	Nguyen	Manh	Hung	09/15/1962	Ba Ria VT	Nam	38000	333445555	5
453453453	Tran	Thanh	Tam	07/31/1972	543 MTL Q1	Nu	25000	333445555	5
999887777	Bui	Ngoc	Hang	07/19/1968	33 NTH Q1	Nu	38000	987654321	4
987654321	Le	Quynh	Nhu	07620/1951	219 TD Q3	Nu	43000	888665555	4
987987987	Tran	Hong	Quang	04/08/1969	980 LHP Q5	Nam	25000	987654321	4
888665555	Pham	Van	Vinh	11/10/1945	450 TV HN	Nam	55000	NULL	1

MA_NVIENT	SODA	THOIGIAN
333445555	10	10.0
888665555	20	20.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
453453453	1	20.0



- ❖ Dùng để thay đổi giá trị của thuộc tính cho các dòng của bảng
- ❖ **Cú pháp**

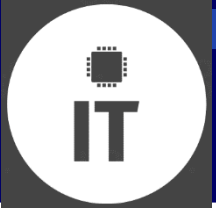
UPDATE <tên bảng>

SET <tên thuộc tính>=<giá trị mới>,

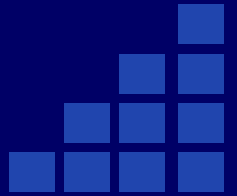
<tên thuộc tính>=<giá trị mới>,

...

[**WHERE** <điều kiện>]



Lệnh UPDATE (tt)



❖ Ví dụ

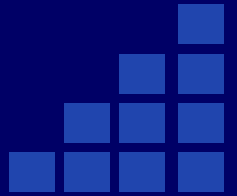
```
UPDATE NHANVIEN
```

```
SET NGSINH='08/12/1965'
```

```
WHERE MANV='333445555'
```

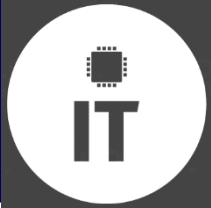
```
UPDATE NHANVIEN
```

```
SET LUONG=LUONG*1.1
```

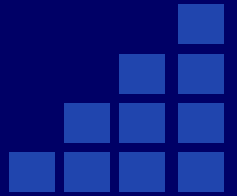


❖ Nhận xét

- Những dòng thỏa điều kiện tại mệnh đề WHERE sẽ được cập nhật giá trị mới
- Nếu không chỉ định điều kiện ở mệnh đề WHERE, tất cả các dòng trong bảng sẽ bị cập nhật
- Lệnh UPDATE có thể gây ra vi phạm RB tham chiếu
 - Không cho sửa
 - Sửa luôn những dòng có giá trị đang tham chiếu đến
 - CASCADE



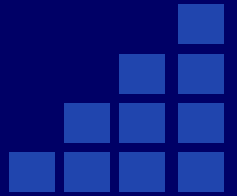
Các lệnh liên quan đến cấu trúc



- CREATE TABLE (tạo bảng)
- DROP TABLE (xóa bảng)
- ALTER TABLE (sửa bảng)
- CREATE DOMAIN (tạo miền giá trị)
- CREATE DATABASE
- ...



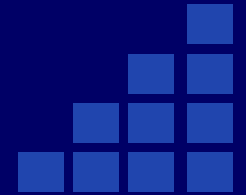
Các lệnh truy vấn CSDL



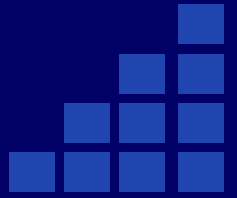
- ❖ Câu lệnh **SELECT** tổng quát
- ❖ Mệnh đề **SELECT**
- ❖ Mệnh đề **FROM**
- ❖ Mệnh đề **WHERE**
- ❖ Mệnh đề **ORDER BY**
- ❖ Mệnh đề **GROUP BY**
- ❖ Truy vấn từ nhiều bảng
- ❖ Truy vấn lồng nhau



Câu lệnh truy vấn tổng quát



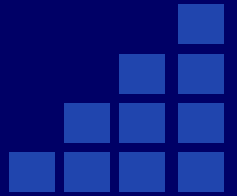
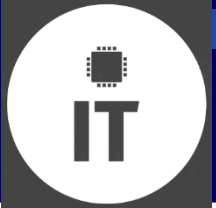
SELECT distinct /*/danh sách thuộc tính/
 <biểu thức>,...
FROM danh_sách_các_quan_hệ
[WHERE điều_kiện]
[GROUP BY danh_sách_cột_cần_nhóm]
[HAVING BY điều_kiện_nhóm]
[ORDER BY danh_sách_cột_cần_sắp_xếp]



❖ Gồm 3 mệnh đề

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

- <danh sách các cột>
 - Tên các cột cần được hiển thị trong kết quả truy vấn
- <danh sách các bảng>
 - Tên các bảng liên quan đến câu truy vấn
- <điều kiện>
 - Biểu thức boolean xác định dòng nào sẽ được rút trích
 - Nối các biểu thức: AND, OR, và NOT
 - Phép toán: < , > , ≤ , ≥ , ≠ , =, LIKE và BETWEEN



- MỆNH ĐỀ SELECT, FROM

SELECT distinct /*/

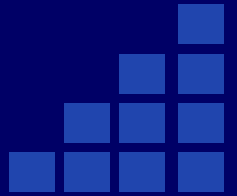
 danh sách thuộc tính/

 <biểu thức>,...

FROM <danh sách các quan hệ>



MỆNH ĐỀ SELECT, FROM



- ❖ Ví dụ: Hãy đưa ra Tên của tất cả các khách hàng

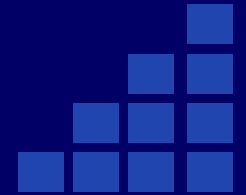
```
Select  TENKH  
From    KHACH;
```

- ❖ Lưu ý: khi mệnh đề From chỉ ra từ hai quan hệ trở lên, nếu có một thuộc tính ở mệnh đề select là thuộc tính của nhiều hơn một quan hệ thì cần phải chỉ rõ thuộc tính đó thuộc về quan hệ nào theo cú pháp

tênquan hệ.tênthuộc tính



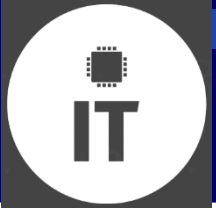
MỆNH ĐỀ SELECT, FROM (tt)



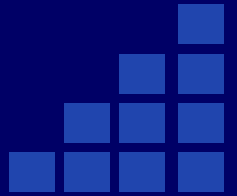
❖ **Phát biểu Select với AS:** Đặt lại tên Field khi hiển thị kết quả

Cú pháp: *Tên cột của bảng AS tên mới*

(Nếu tên có chứa các ký tự đặc biệt và/hoặc khoảng trắng thì viết tên đó trong cặp dấu ngoặc vuông)

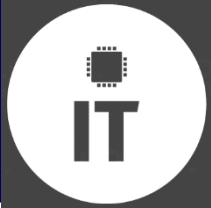


MỆNH ĐỀ SELECT, FROM (tt)

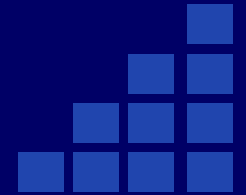


❖ **Trong ví dụ trên ta có thể viết lại:**

```
Select  TENKH as [Tên Khách hàng]  
From    KHACH;
```



MỆNH ĐỀ SELECT, FROM (tt)

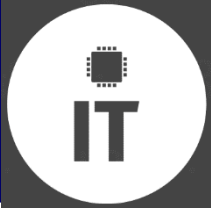


❖ Phát biểu Select với TOP N

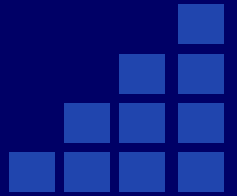
Cho phép lấy ra chỉ một số mẫu tin nào đó theo 1 tiêu chuẩn nhất định.

Ví dụ: Đưa ra danh sách 10 mặt hàng đầu tiên.

```
SELECT TOP 10 *  
FROM Hang;
```

MỆNH ĐỀ SELECT, FROM (tt)

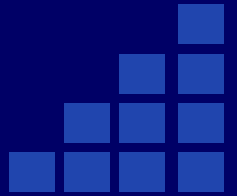


- ❖ Có thể dùng các phép toán số học +, -, *, /, ^, %, các hàm tính toán đối với các cột kiểu số trong mệnh đề SELECT

Ví dụ:

```
Select MaH, Slton*10
```

```
From  HANG;
```



- MỆNH ĐỀ WHERE

SELECT distinct /*/

 danh sách thuộc tính/

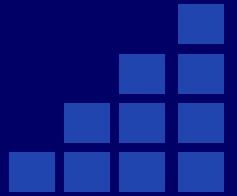
 <biểu thức>,...

FROM <danh sách các quan hệ>

WHERE <biểu thức điều kiện>



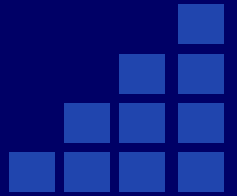
MỆNH ĐỀ WHERE



- <biểu thức điều kiện>
 - Biểu thức boolean xác định dòng nào sẽ được rút trích
 - Nối các biểu thức: AND, OR, và NOT
 - Phép toán: $<$, $>$, \leq , \geq , \neq , $=$, LIKE và BETWEEN,...



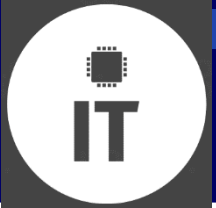
MỆNH ĐỀ WHERE (tt)



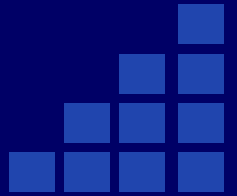
- ❖ Ví dụ: Đưa ra các thông tin MANV, HONV, TENLOT, TENNV của những Nhân viên Nam thuộc Phòng 5

```
SELECT MANV, HONV, TENLOT, TENNV  
FROM NHANVIEN  
WHERE PHG=5 AND PHAI='Nam';
```

MANV	HONV	TENLOT	TENNV
333445555	Nguyen	Thanh	Tung
987987987	Nguyen	Manh	Hung



MỆNH ĐỀ WHERE (tt)

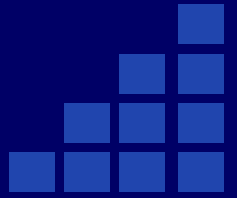


Ví dụ: Hiển thị các mặt hàng có số lượng tồn lớn hơn 100

```
Select      *  
From  HANG  
Where    Slton>100;
```



MỆNH ĐỀ WHERE (tt)



- ❖ **Toán tử Between:** nằm trong 1 miền
- ❖ Ví dụ: Đưa ra danh sách các mặt hàng có số lượng tồn từ 50 đến 100 sản phẩm

Khach (MAKH, TENKH, DIACHI, DIENTHOAI)

Loaihang(MALOI, TENLOAI)

Hang(MAH, TENH, SLTON, MALOI)

HoaDon(SOHD, NGAYHD, MAKH)

ChitietHD(SOHD, MAH, SLB, DGIA)

Select *

From HANG

Where Slton Between 50 and 100;

Hoặc Where (Slton>=50 and Slton<=100)

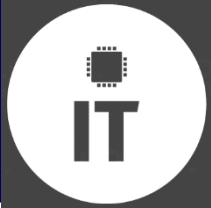


MỆNH ĐỀ WHERE (tt)

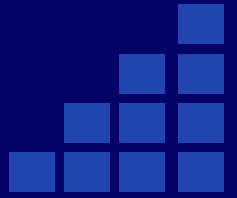
❖ **Toán tử LIKE:** phép toán so sánh gần giống, sử dụng dấu các ký tự đại diện như _ (1 ký tự), % (1 chuỗi bất kỳ)

Ví dụ: Cho hiển thị các khách hàng ở địa chỉ bắt đầu bằng chữ H

```
Select      *  
From        KHACH  
Where       diachi    like 'H%';
```



MỆNH ĐỀ WHERE (tt)



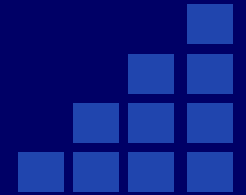
- ❖ **Toán tử IN:** phép toán so sánh trong 1 tập hợp, 1 danh sách
- ❖ Ví dụ: Đưa ra danh sách Khách hàng quê quán ở các tỉnh ‘Cần Thơ’, ‘An Giang’, ‘Kiên Giang’

SELECT *

FROM Khach

WHERE Diachi IN ('Cần Thơ', 'An Giang', 'Kiên Giang');

Hoặc Where (Diachi="Cần Thơ" or Diachi="An Giang" or Diachi = "Kiên Giang")



- ❖ Dùng để hiển thị kết quả câu truy vấn theo một thứ tự nào đó
- ❖ **Cú pháp**

SELECT <danh sách các cột>

FROM <danh sách các bảng>

WHERE <điều kiện>

ORDER BY <danh sách các cột> [ASC/DESC]

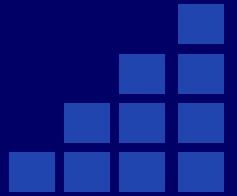
- ASC: tăng (mặc định)
- DESC: giảm

❖ Ví dụ:

```
SELECT MA_NVIENT, SODA  
FROM PHANCONG  
ORDER BY MA_NVIENT DESC, SODA
```

MA_NVIENT	SODA
999887777	10
999887777	30
987987987	10
987987987	30
987654321	10
987654321	20
987654321	30

Mệnh đề GROUP BY



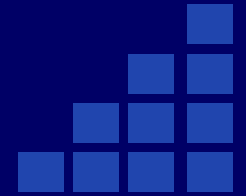
❖ Dùng để nhóm các bản ghi có giá trị giống nhau trên các cột được liệt kê sau từ khóa GROUP BY.

❖ Cú pháp:

```
SELECT <danh sách các cột>  
FROM <danh sách các bảng>  
[WHERE <điều kiện>  
GROUP BY <tên cột 1>,<tên cột 2>,...  
[HAVING <điều kiện>
```

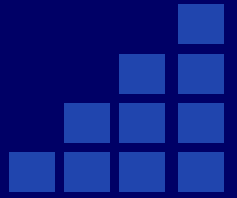


Mệnh đề GROUP BY (tt)



❖ Một số hàm thông dụng tính toán trên nhóm bản ghi

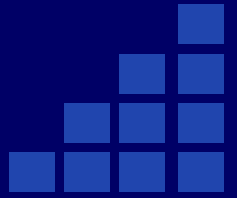
- COUNT (*| <tên cột>): đếm số bản ghi có giá trị xác định tại cột được cho bởi <tên cột>
- SUM (<biểu thức>): tìm tổng giá trị các biểu thức
- MIN (<biểu thức>): tìm giá trị nhỏ nhất
- MAX (<biểu thức>): tìm giá trị lớn nhất
- AVG (<biểu thức>): tính giá trị trung bình của biểu thức dựa trên các bản ghi của các nhóm.



Nhận xét:

❖ Mệnh đề HAVING

- HAVING theo sau GROUP BY dùng để kiểm tra điều kiện nhóm. Nhóm nào thoả mãn điều kiện sau HAVING thì mới được hiển thị
- Chỉ kiểm tra điều kiện trên nhóm, không là điều kiện lọc trên từng bộ
- Dữ liệu được nhóm trước rồi mới kiểm tra điều kiện sau HAVING.



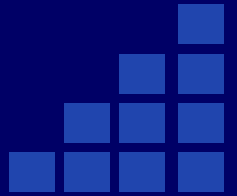
❖ Thứ tự thực hiện câu truy vấn có mệnh đề GROUP BY và HAVING

- (1) Chọn ra những dòng thỏa điều kiện trong mệnh đề WHERE
- (2) Những dòng này sẽ được gom thành nhiều nhóm tương ứng với mệnh đề GROUP BY
- (3) Áp dụng các hàm kết hợp cho mỗi nhóm
- (4) Bỏ qua những nhóm không thỏa điều kiện trong mệnh đề HAVING
- (5) Rút trích các giá trị của các cột và hàm kết hợp trong mệnh đề SELECT

FROM → WHERE → GROUP BY → HAVING → SELECT → ORDER BY



Mệnh đề GROUP BY (tt)

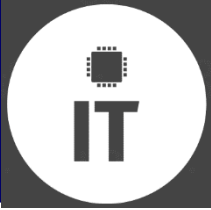


- ❖ **Ví dụ 1:** Tính tổng số lượng hàng của mỗi hoá đơn bán.

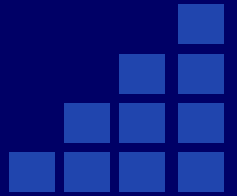
```
Select SoHD, sum(Slban) as So_luong_hang_ban  
From chitietHD  
Group By Sohd;
```

- ❖ **Ví dụ 2:** Cho biết các khách hàng có nhiều hơn 15 lần mua hàng

```
Select Mak, count(MaKH) AS So_Lan_mua_hang  
From HOADON  
Group By MaKH  
Having count(MaKH)>15;
```



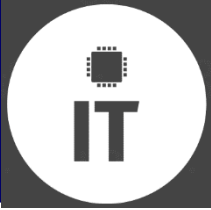
BÀI TẬP ÁP DỤNG



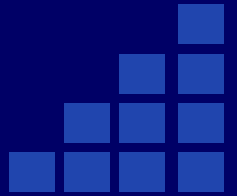
- ❖ **Sinhvien**(MASV, HOTENSV, GT, NGAYSINH, NOISINH, TINH, MALOP)
- ❖ **Lop**(MALOP,TENLOP,MAKHOA)
- ❖ **Khoa**(MAKHOA,TENKHOA)
- ❖ **Monhoc**(MAMH,TENMH,SoTC)
- ❖ **Giangvien**(MAGV,HOTENGV,HOCVI,CHUYENNGANH, MAKHOA)
- ❖ **Ketqua**(MASV, MAMH, LANTHI,DIEMTHI)
- ❖ **Phancong**(MALOP,MAMH,MAGV)

❖ Yêu cầu:

1. Lập danh sách những môn học có số tín chỉ ≥ 4 . Danh sách cần MAMH, TENMH, SOTC.
2. Lập danh sách các sinh viên có mã lớp là CDTH2A, CDTH2B, CDTH2C.
3. Lập danh sách các sinh viên có họ là Nguyễn đang theo học tại lớp có mã lớp là CDTH2A
4. Lập danh sách các lớp có mã khoa là “CNTT” danh sách cần các thông tin MALOP, TENLOP, TENKHOA
5. Lập bảng điểm trung bình lần 1 các môn học của các sinh viên thuộc lớp có mã lớp là CDTH2A. Danh sách cần: MASV, HOTENSV, DIEMTB((trong đó DIEMTB là thuộc tính tự đặt).



Truy vấn từ nhiều bảng



- ❖ Khi thông tin cần lấy ra có từ nhiều bảng khác nhau, cần thực hiện truy vấn từ nhiều bảng

SELECT <Danh sách các cột>

FROM <Danh sách các Bảng>

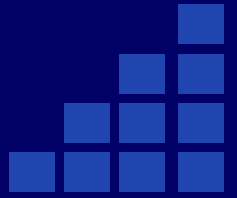
WHERE <Các điều kiện>

- ❖ Lưu ý:

- ☞ Cần kết nối các bảng lại với nhau. Với n bảng cần có n-1 điều kiện kết.
- ☞ Các tên cột cùng có ở nhiều bảng cần ghi rõ theo dạng:
[Tên Bảng].[Tên cột]



Truy vấn từ nhiều bảng (tt)



- ❖ Các bảng được liên kết với nhau qua phép kết nối của từ khóa FROM hoặc thông qua điều kiện của từ khóa WHERE của câu lệnh SELECT

➤ Kết nối bảng qua điều kiện của WHERE:

Tenquanhe1.TTKN=Tenquanhe2.TTKN

– Ví dụ: Cho biết tên các khách hàng mua hàng trong năm 2019

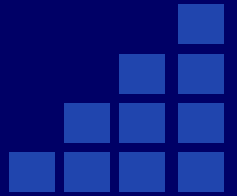
Select KHACH.Tenk

From KHACH, HOADON

Where (KHACH.MaKH=HOADON.MaKH) and
(year(NgayHD)=2019);



Truy vấn từ nhiều bảng (tt)



- Liên kết bảng với nhau qua phép kết nối của từ khóa FROM → sử dụng phép kết nối nội **Inner join**:

Quanhe1 **Inner join** *Quanhe2* **On** *Quanhe1.TT1 = Quanhe2.TT2*

Select KHACH.Tenk

From KHACH **Inner Join** HOADON **On**
KHACH.MaKH=HOADON.MaKH

Where year(NgayHD)=2019;

Truy vấn lồng nhau

- Là những câu lệnh mà trong mệnh đề WHERE hoặc HAVING (thường là trong mệnh đề WHERE) có chứa thêm một câu lệnh Select khác nữa.
- Câu lệnh này thường gặp khi cần sử dụng kết quả của 1 câu truy vấn để làm điều kiện cho 1 câu truy vấn khác

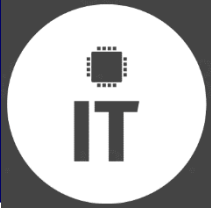
Câu truy vấn cha
(Outer query)

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

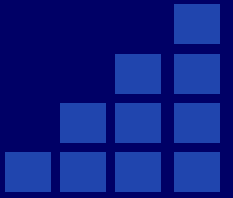
```
SELECT <danh sách các cột>  
FROM <danh sách các bảng>  
WHERE <điều kiện>)
```

Câu truy vấn con
(Subquery)

- Khi thực hiện, truy vấn con sẽ được thực hiện trước, rồi lấy kết quả để thực hiện truy vấn cha



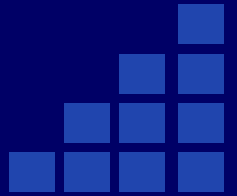
Truy vấn lồng nhau (tt)



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



Xét CSDL:



Sinhvien(MASV, HOTENSV, GT, NGAYSINH, NOISINH, TINH, MALOP)

Lop(MALOP,TENLOP,MAKHOA)

Khoa(MAKHOA,TENKHOA)

Monhoc(MAMH,TENMH,SoTC)

Giangvien(MAGV,HOTENGV,HOCVI,CHUYENNGANH, MAKHOA)

Ketqua(MASV, MAMH, LANTHI,DIEMTHI)

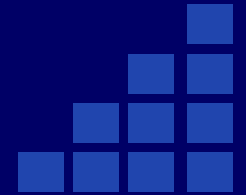
Phancong(MALOP,MAMH,MAGV)

Ví dụ 1: Lập danh sách những sinh viên lớp CDTH2A có điểm thi lần 1 môn học CSDL cao nhất.

```
SELECT sinhvien.MASV, HOTENSV, NU, NGAYSINH,  
       ketqua.DIEMTHI  
FROM   sinhvien, ketqua  
WHERE  Sinhvien.MASV=Ketqua.MASV AND  
       DIEMTHI>=ALL(SELECT DIEMTHI  
                    FROM ketqua  
                    WHERE MAMH='CSDL' AND Lanthi=1)
```


Ví dụ 2: Lập danh sách những giảng viên cùng khoa với giảng viên NGUYEN VAN THANH ?

```
SELECT *  
FROM giangvien  
WHERE makhoa in  
    (SELECT makhoa  
     FROM giangvien  
     WHERE Hotengv= "Nguyen Van Thanh");
```



Cho CSDL gồm quan hệ:

Nhanvien (MANV, HOTEN, NS, GIOITINH, DIACHI, TDNN, LƯƠNG, MAPHONG)

Phong (MAPHONG, TENPHONG, DIADIEM)

Ví dụ 3: Đưa ra danh sách các nhân viên có lương lớn hơn lương mọi nhân viên làm ở phòng Kinh doanh?

*Select **

From NHANVIEN

Where luong > (Select max(luong)

From NHANVIEN

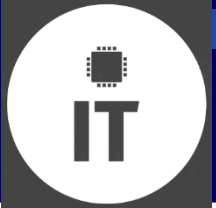
Where Maphong in (Select Maphong

From Phong

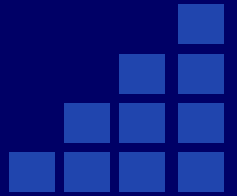
Where Tenphong = 'Kinh doanh'))

Ví dụ 4: Cho biết danh sách các nhân viên đã học ít nhất một ngoại ngữ?

```
Select * From NHANVIEN nv  
Where Exists (Select * From TDNN td  
Where nv.Manv=td.Manv)
```



BÀI TẬP ỨNG DỤNG



Cho CSDL:

Congtrinh(MACT, TENCT, ĐIADIEM, NGÀYCAPGP,
NGAYKC,NGAYHT)

Nhanvien(MANV, HOTEN, NGÀYSINH, PHAI, ĐIACHI, MAPB)

Phongban(MAPB, TENPB)

Phancong(MACT, MANV, SLNGAYCONG)

❖ Yêu cầu:

1. Đưa ra danh sách những nhân viên có tham gia vào công trình có mã công trình(MACT) là X.

Yêu cầu các thông tin: MANV,HOTEN, SLNGAYCONG, trong đó MANV được sắp tăng dần

2. Đếm số lượng ngày công của mỗi công trình.

Yêu cầu các thông tin: MACT, TENCT, TONGNGAYCONG (TONGNGAYCONG là thuộc tính tự đặt)

3. Danh sách những nhân viên có sinh nhật trong tháng 08.

Yêu cầu các thông tin: MANV, TENNV, NGAYSINH, ĐIACHI,TENPB, sắp xếp quan hệ kết quả theo thứ tự tuổi giảm dần.

4. Đếm số lượng nhân viên của mỗi phòng ban.

Yêu cầu các thông tin: MAPB, TENPB, SOLUONG. (SOLUONG là thuộc tính tự đặt.)

**Nhanvien (MANV,HOTEN,NGAYSINH,ĐIACHI,
MAPB, MACV)**

Phongban (MAPB,TENPB)

Chucvu (MACV,TENCV,PHUCAP)

❖ **Yêu cầu:**

- 1. Tạo các bảng trên**
- 2. Thêm vào bảng Nhanvien bản ghi có thông tin sau: 052364, Nguyen Huy Hoang, 05/12/1998**
- 3. Thêm vào bảng nhanvien cột SDT**
- 4. Xóa cột SDT khỏi bảng Nhanvien**

❖ Sử dụng CSDL quản lý bán hàng để minh họa cho các câu lệnh:

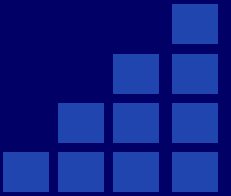
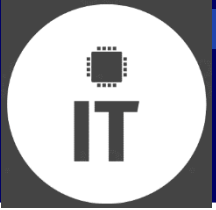
Khach (MAKH, TENKH, DIACHI, DIENTHOAI)

Loaihang(MALOAI, TENLOAI)

Hang(MAH, TENH, SLTON, MALOAI)

HoaDon(SOHD, NGÀYHD, MAKH)

ChitietHD(SOHD, MAH, SLB, DGIA)



- ❖ **SQL**
- ❖ **Trigger**
- ❖ **Function**
- ❖ **View**
- ❖ **Procedures**