

TỔNG QUAN VỀ SQL

*SQL là gì?

- SQL (Structured Query Language - ngôn ngữ truy vấn mang tính cấu trúc) là một loại ngôn ngữ máy tính phổ biến để tạo, sửa, và lấy dữ liệu từ một hệ quản trị cơ sở dữ liệu quan hệ.
- SQL (Structured Query Language) là tập lệnh truy xuất CSDL quan hệ.

Giới thiệu

- SQL (Structured Query Language) là một loại ngôn ngữ máy tính phổ biến để tạo, sửa, và lấy dữ liệu từ một hệ quản trị cơ sở dữ liệu quan hệ.
- **Data Definition Language (DDL):** Các lệnh mô tả CSDL, gồm: tạo (**Create**), sửa (**Alter**), xoá (**Drop**) các bảng và ràng buộc
- **Data Manipulation Language (DML):** Các lệnh thao tác truy vấn dữ liệu, gồm: **Chèn(Insert)**, **Cập nhật(Update)**, **Xóa(Delete)**, Lựa chọn (**Select**)
- **Data Control Language (DCL):** Các lệnh điều khiển CSDL, dùng để quản lý quyền hạn của user, Gồm: **grant**, **revoke**, **deny**

Quy tắc viết lệnh

- Không phân biệt chữ hoa, chữ thường
- Nội dung 1 lệnh SQL có thể viết trên nhiều dòng.
- Từ khoá không viết tắt hay phân cách trên nhiều dòng
- Các mệnh đề thường được đặt trên nhiều dòng khác nhau
- Ta có thể sử dụng các ký tự đặc biệt như: +, -, /, *, ... để biểu diễn giá trị trong câu lệnh.

Chú thích và tiếng việt

- Dòng đơn ⇔ --
- Nhóm dòng ⇔ /* ... */
- Sử dụng tiếng việt trong truy vấn
 - Chọn kiểu dữ liệu hỗ trợ Unicode (nchar, nvarchar, ntext)
 - Thêm tiền tố N (National Characters) vào trước chuỗi cần nhập để báo cho SQL Server đây là chuỗi Unicode
- VD:
 - **Insert into** PHONGBAN(MAPHG, TENPHG)
values('QL', N'Quản lý ')

Chương 2

DDL

-ĐỊNH NGHĨA DỮ LIỆU-

2.1.2. Tạo Database

- Cú pháp cơ bản:

CREATE DATABASE *<database-name>*
[**ON** *<filespec>*]

- Trong đó:

- *<database-name>*

- *<filespec>* ::= (**NAME** = logical_file_name,
 FILENAME = os_file_name
 [, **SIZE** = *n* [KB|MB|GB|TB]]
 [, **MAXSIZE** = *max* [KB|MB|GB|TB] | Unlimited]
 [, **FILEGROWTH** = *grow* [KB|MB|GB|TB|%]
)

-
- Ví dụ 1: Tạo CSDL có tên là “Quan_ly_SV”

Create Database Quan_ly_SV;

Ví dụ 2:

- Tạo Database **MyDB** với tập tin dữ liệu chính là **MyDB_Data.mdf**, đặt tại thư mục **D:\Databasse** với dung lượng khởi tạo là **1MB**, tối đa là **10MB** và độ gia tăng kích thước cho phép là **10%**.

Create Database MyDB

```
On(  
    Name = MyDB_Data,  
    FileName = 'D:\Database\MyDB_Data.mdf' ,  
    Size = 1MB,  
    Maxsize = 10MB,  
    FileGrowth = 10%  
)
```

- Xóa cơ sở dữ liệu

DROP DATABASE <tên CSDL>;

Ví dụ:

```
DROP Database Quan_ly_SV;
```

- Sử dụng cơ sở dữ liệu

USE <tên CSDL>;

Ví dụ:

```
USE Quan_ly_SV;
```

2.2. Bảng

2.2.1 Giới thiệu

- Được sử dụng để định nghĩa một bảng dữ liệu mới
- Khi định nghĩa bảng mới cần xác định:
 - *Bảng mới được tạo ra với mục đích gì? Vài trò gì?*
 - *Cấu trúc bảng gồm trường nào? Mỗi trường có ý nghĩa gì? Kiểu dữ liệu của trường là gì? Có cho phép nhận giá trị Null không?*
 - *Trường nào làm khóa chính?*
 - *Bảng có quan hệ với bảng khác không? Quan hệ như thế nào?*
 - *Có ràng buộc – yêu cầu gì cho từng cột không?*

2.2.2 Cú pháp tạo bảng

```
CREATE TABLE <table-name>  
(  
    column1    data-type    [RBTV],  
    [column2    data-type    [RBTV],]  
    ...  
    [columnn    data-type    [RBTV],]  
    [RBTV]  
)
```

■ Trong đó:

- **Table-name:** là tên bảng cần tạo, tuân thủ nguyên tắc định danh, không quá 128 ký tự
- **Column:** tên cột cần tạo trong bảng, mỗi bảng có ít nhất một cột
- **Data-type:** xác định kiểu dữ liệu được lưu trữ trong cột,
Kiểu dữ liệu là thuộc tính bắt buộc
- **RBTV:** gồm các ràng buộc về khuôn dạng dữ liệu hay các ràng buộc về bảo toàn dữ liệu, có thể: **NOT NULL, NULL, UNIQUE, DEFAULT, PRIMARY KEY, IDENTITY, CHECK,...**

Các kiểu dữ liệu

- Phụ thuộc vào hệ quản trị cơ sở dữ liệu (Access)

Kiểu dữ liệu	Mô tả	Kích thước
Text	Lưu trữ thông tin dạng ký tự và ký số	1byte lưu trữ tối đa 255 ký tự
Number	Thông tin dạng số dùng để tính toán	1 – 2 - 4 – 8 – 12 byte
Date/Time	Lưu dữ liệu dạng ngày giờ	8 byte
Currency	Dữ liệu dạng tiền tệ có phần thập phân từ 1 đến 4 số	8 byte
Autonumber	Dữ liệu dạng số tự động tăng với bước tăng là 1, ta sẽ không thể cập nhật lại	4 byte
Memo	Dữ liệu dạng ký tự với lượng thông tin lớn	2 byte
Yes/No	Dữ liệu logic dạng True/False	1 bit
OLE Object	Dữ liệu dạng âm thanh, hình ảnh	1 GB (tùy vào dung lượng đĩa)
Hyperlink	Dữ liệu dạng siêu liên kết	

Kiểu dữ liệu SQL – Server

Kiểu dữ liệu	Mô tả	Kích thước
Char	Kiểu kí tự với độ dài không thay đổi	Lưu trữ tối đa 8000 kí tự
Nchar	Giống Char nhưng hỗ trợ Unicode	Tối đa 4000 kí tự
Varchar	Dạng kí tự với độ dài thay đổi tùy theo độ dài thực của dữ liệu	Tối đa 4000 kí tự
Nvarchar	Giống Varchar nhưng hỗ trợ Unicode	Tối đa 4000 kí tự
Text	Kiểu văn bản gồm cả ký tự xuống dòng	Cỡ GB tùy dung lượng máy
Ntext	Giống Text hỗ trợ Unicode	Cỡ GB tùy dung lượng máy
Date/Time	Dữ liệu ngày giờ chia 2 dạng: Date/Time đầy đủ ngày và thời gian . SmallDateTime chỉ ngày hoặc thời gian	8 byte
Numeric	Dữ liệu dạng số nguyên: int, smallint, bigint, số thực: float, real, decimal, numeric	2 – 4 – 8 - 12 byte tùy theo kiểu số được chọn
Monetary	Kiểu tiền tệ: Money và SmallMoney	8 byte

Ví dụ:

```
CREATE TABLE NHANVIEN (  
    MANV INT  
    TENNV VARCHAR(50),  
    NS DATETIME,  
    DCHI VARCHAR(50),  
    GT VARCHAR (3),  
    LUONG INT,  
    MAP INT  
);
```


2.2.3 Các ràng buộc khi tạo bảng

- Ràng buộc được sử dụng theo 2 cách:
 - Đặt ràng buộc ngay tại vị trí khai báo của cột \Leftrightarrow ***ràng buộc trực tiếp cho thuộc tính***
 - Đặt ràng buộc xuống phần cuối cùng của câu lệnh sau khi đã liệt kê hết các cột của bảng \Leftrightarrow ***thường dùng với ràng buộc gộp cho nhiều thuộc tính***

*Ràng buộc CHECK

- *Sử dụng nhằm chỉ định điều kiện hợp lệ đối với dữ liệu.*
- **Cú pháp:**
[CONSTRAINT *tên-ràng-buộc*] **CHECK** (*điều kiện*)
 - **Điều kiện:** là một biểu thức so sánh
- **Ví dụ:**

```
CREATE TABLE SINHVIEN (  
    MASV VARCHAR(9)  
    TENS VARCHAR(50),  
    D_TOAN INT CHECK (D_Toan between 0 and 10),  
    D_VAN INT CHECK (D_Van>=0 and D_Van<=10)  
)
```

```
CREATE TABLE SINHVIEN (  
    MASV  VARCHAR(9),  
    TENSX  VARCHAR(50) NOT NULL,  
    D_TOAN INT,  
    D_VAN  INT,  
    constraint chk_Diem CHECK ( (D_Van>=0 and  
    D_Van<=10) and ( D_Toan between 0 and 10) )  
)
```

- Ví dụ:

```
CREATE TABLE CTHOADON(  
    SOHD Varchar(3) Primary Key,  
    MAVTU Varchar(3) Primary Key,  
    Soluong int CHECK (Soluong>0),  
    Dongia float CHECK (Dongia>0)  
)
```

Hoặc

```
CREATE TABLE CTHOADON(  
    SOHD Varchar(3) not null,  
    MAVTU Varchar(3) not null,  
    Soluong int,  
    Dongia float,  
CONSTRAINT CK_SL_DG CHECK (Soluong>0 and Dongia>0))
```

*Ràng buộc NOT NULL

- *Bắt buộc phải nhập giá trị cho cột khi thêm dữ liệu vào bảng*
- Cú pháp:
<column> <type> NOT NULL

Ví dụ:

```
CREATE TABLE SINHVIEN (  
    MASV  VARCHAR(9)  
    TENSX  VARCHAR(50) NOT NULL,  
    D_TOAN INT,  
    D_VAN  INT )
```

*Ràng buộc PRIMARY KEY

- Được dùng để định nghĩa khóa chính của bảng
- Cú pháp:

<column> <type> **PRIMARY KEY**

hoặc

[CONSTRAINT tên-ràng-buộc] **PRIMARY KEY**([d/s cột])

- Ví dụ:

```
CREATE TABLE SINHVIEN (  
    MASV VARCHAR(9) PRIMARY KEY,  
    TENSX VARCHAR(50),  
    D_TOAN INT,  
    D_VAN INT )
```

■ Ví dụ:

CREATE TABLE PHANCONG(

MADA **varchar(4),**

MANV **varchar(100),**

Constraint PK_Ma PRIMARY KEY(MADA, MANV)

)

*Ràng buộc DEFAULT

- Xác định giá trị mặc định ban đầu cho từng cột
- Cú pháp:

<Ten cột> datatype DEFAULT <gt>

Hoặc

[CONSTRAINT tên-ràng-buộc] **DEFAULT** (giá-trị)

- Ví dụ:

```
CREATE TABLE VATTU(  
    MAVTU varchar(4) primary key,  
    TenVtu varchar(100),  
    SoLuong int Default (100)  
)
```


- Ví dụ:

```
CREATE TABLE SINHVIEN(  
    MASV varchar(4) primary key ,  
    TenSV varchar(100),  
    GioiTinh varchar(3) Default ( 'Nam' )  
)
```

*Ràng buộc UNIQUE

- Dùng khi quy định một cột nào đó cho phép chỉ nhập một giá trị duy nhất cho từng dòng
- Cú pháp:
<column> <type> **UNIQUE**

[**CONSTRAINT** Tên_RB] **UNIQUE** (DS cột)

- Trong một bảng chỉ có một ràng buộc khoá chính nhưng có thể có nhiều ràng buộc dữ liệu duy nhất

- Ví dụ:

CREATE TABLE VATTU

(MAVTU varchar(4) **primary key**,
TenVtu varchar(100) **UNIQUE**)

CREATE TABLE VATTU

(MAVTU varchar(4) **not null**,
TenVtu varchar(100),

Constraint UN_Vtu UNIQUE (MaVtu,TenVtu))

*Ràng buộc FOREIGN KEY

- Đảm bảo việc nhập dữ liệu cho một cột thuộc tính nào đó phù hợp tham chiếu tới một bảng quan hệ khác
- Cú pháp:

REFERENCES *tên-bảng-tham-chiếu(d/s cột tham chiếu)*
hoặc

[**CONSTRAINT** *tên-ràng-buộc*] **FOREIGN KEY**([d/s cột])

REFERENCES *tên-bảng-tham-chiếu(d/s cột tham chiếu)*

[ON DELETE CASCADE | NO ACTION | SET NULL | SET DEFAULT]

[ON UPDATE CASCADE | NO ACTION | SET NULL | SET DEFAULT]

- Việc định nghĩa ràng buộc Foreign Key gồm:
 - Tên cột và danh sách cột của bảng tham gia vào khóa ngoại
 - Tên bảng được tham chiếu bởi khóa ngoại và danh sách cột được tham chiếu đến trong bảng tham chiếu

- Cách xử lý bản ghi được sử dụng trong các trường hợp:
 - **CASCADE**: tự động xóa (cập nhật) nếu bản ghi được tham chiếu bị xóa (cập nhật)
 - **NO ACTION**: (*mặc định*) nếu bản ghi trong bảng tham chiếu đang được tham chiếu bởi một bản ghi bất kỳ trong bảng được định nghĩa thì bản ghi đó không được phép xóa hoặc cập nhật
 - **SET NULL**: cập nhật lại khóa ngoại của bản ghi thành giá trị NULL (*nếu cột phép nhận giá trị NULL*)
 - **SET DEFAULT**: cập nhật lại khóa ngoại của bản ghi nhận giá trị mặc định (*nếu có giá trị mặc định*)

■ Ví dụ

CREATE TABLE CTHOADON

(SOHD **Varchar(3)** **REFERENCES** HOADON(SOHD),
MAVTU **Varchar(3)** **REFERENCES** VATTU(MAVTU),
Soluong **int**,
Dongia **float**)

CREATE TABLE CTHOADON

(SOHD **Varchar(3)**,
MAVTU **Varchar(3)**,
Soluong **int**,
Dongia **float**,
FOREIGN KEY (SOHD) **REFERENCES** HOADON(SOHD),
FOREIGN KEY (MAVTU) **REFERENCES** VATTU(MAVTU)
)

*Ràng buộc IDENTITY

■ Cú pháp:

[constraint name] **IDENTITY** [(*start*, *step*)],

- ❑ Trong một bảng chỉ có tối đa một cột được chỉ định làm cột định danh.
- **start**: là số mà SQL Server sử dụng để cấp phát cho mẫu tin đầu tiên. Mặc định là 1.
- **step**: là chỉ số mà SQL Server cộng lên để cấp phát cho từng mẫu tin kế tiếp. Mặc định là 1.

Ví dụ áp dụng

- Tạo CSDL “Quản lý nhân sự”
- Gồm các bảng:
 - **NHAN VIEN** (MaNV, HoTen, GT, NS, QQ, DT, MaPB)
 - **PHONG BAN** (MaPB, TenPB, DienThoai) *
 - **CHUC VU** (MaCV, TenCV, HSPC)
 - **TD_HV** (MaTDHV, TenTD, ChuyenNganh)
 - **BACLUONG** (MaBL, HS, HSPC)
 - **DC_PB** (MaPB, DiaChi)
 - **NV_CHUCVU** (MaNV, MaCV, NgayQD)
 - **NV_TD HV** (MaNV, MaTD, NgayBD)
 - **NV_BACLUONG** (MaNV, MaBL, NgayQD)
- Yêu cầu ràng buộc:
 - Giới tính chỉ nhận Nam và Nữ
 - Chức vụ chỉ có “GD”, “PGD”, “TP”, “PP”, “NV”
 - HSL mặc định là 2.54
 - HSL từ 2.54 đến 12
 - HSPC từ 0.4 đến 1.2

4. Sửa cấu trúc bảng

- Sử dụng câu lệnh **ALTER TABLE**.
- Câu lệnh này cho phép thực hiện được các thao tác sau:
 - *Bổ sung một cột mới vào bảng.*
 - *Xoá một cột khỏi bảng.*
 - *Thay đổi định nghĩa kiểu của một cột trong bảng.*
 - *Xoá bỏ hoặc bổ sung các ràng buộc cho bảng*

*Thêm cột mới vào trong bảng

- Cú pháp:

ALTER TABLE Tên_bảng

ADD Tên_cột Kiểu_dữ_liệu [RBTV] [...]

- *Luôn thêm cột mới vào cuối bảng*

- Ví dụ:

ALTER TABLE VATTU

ADD Soluong **int**

***Hủy bỏ một cột trong bảng**

❑Cú pháp

ALTER TABLE Tên_bảng

DROP COLUMN Tên_cột [...]

❑Ví dụ:

ALTER TABLE VATTU

DROP COLUMN TenVtu

*Sửa đổi kiểu dữ liệu của cột

❑ Cú pháp:

ALTER TABLE Tên_bảng

ALTER COLUMN Tên_cột Kiểu_dữ_liệu_mới

❑ *Kiểu dữ liệu mới phải lớn hơn kiểu dữ liệu cũ đã có*

❑ Ví dụ:

ALTER TABLE VATTU

ALTER COLUMN TenVtu **Nvarchar(30)**

*Thêm ràng buộc cho cột

❑ Cú pháp:

ALTER TABLE Tên_bảng

ADD CONSTRAINT Tên_ràng_buộc Loại_ràng_buộc

❑ Ví dụ:

ALTER TABLE VATTU

ADD CONSTRAINT CK_NgayNhap

CHECK (Ngaynhap <= GetDate())

***Hủy ràng buộc đã đặt**

❑ Cú pháp

ALTER TABLE Tên_bảng

DROP CONSTRAINT Tên_ràng_buộc

❑ Ví dụ:

ALTER TABLE VATTU

DROP CONSTRAINT CK_NgayNhap

■ Lưu ý:

- Khi thêm cột vào bảng đã có ít nhất 1 bản ghi thì cột mới thêm phải cho phép nhận giá trị NULL hoặc phải có giá trị mặc định
- Muốn xóa cột có tồn tại ràng buộc/có tham chiếu khóa ngoài thì phải xóa ràng buộc hoặc khóa ngoài
- Nếu thêm ràng buộc cho bảng đã có dữ liệu mà ràng buộc không thỏa mãn với các dữ liệu đã có thì nó sẽ thêm vào được.

*Bật tắt các ràng buộc

❑ Cú pháp

ALTER TABLE Tên_bảng

NOCHECK CONSTRAINT ALL | Tên_constraint [...]

ALTER TABLE Tên_bảng

CHECK CONSTRAINT ALL | Tên_constraint [...]

*Đổi tên cột

Cú pháp

EXEC SP_Rename ‘ *Tên_bảng.Tên_cột* ’ , ‘ *Tên_mới* ’ ,
‘ **COLUMN** ’

- Ví dụ:

EXEC	SP_RENAME	'VATTU.MAVTU',	'MAVATTU',
	'COLUMN'		

***Đổi tên bảng**

❑ Cú pháp

EXEC sp_rename 'Tên_bảng', 'Tên_mới'

❑ Ví dụ:

EXEC SP_RENAME 'VATTU', 'VT'

Ví dụ - Thay đổi cấu trúc bảng

- *Thêm cột NGHENGHIEP có kiểu CHAR với độ rộng cho phép 20*
- *Thay đổi độ rộng cột thành 50 ký tự*
- *Thêm ràng buộc NOT NULL với trường NGHENGHIEP*
- *Xóa cột NGHENGHIEP*

3. Xóa bảng

- Để xóa một bảng khỏi CSDL

- Cú pháp:

DROP TABLE Danh_sách_tên_các_bảng

- Ví dụ:

DROP TABLE VatTu, CTHoaDon

- Lưu ý:

- Câu lệnh Drop Table không thể thực hiện nếu bảng cần xóa được tham chiếu bởi một **Foreign Key**
- Các ràng buộc, chỉ mục, trigger,.. đều bị xóa, nếu tạo lại bảng thì cũng phải tạo lại các đối tượng này
- Sau khi xóa không thể khôi phục lại bảng và dữ liệu bảng

IV. INDEX

- Index là chỉ mục quan trọng trong CSDL đặc biệt với CSDL lớn.
- Index có thể thiết lập cho 1 hoặc nhiều cột của bảng
- Index được sắp xếp nhằm hỗ trợ việc tìm kiếm, truy vấn dữ liệu một cách nhanh chóng.

CREATE [UNIQUE] [CLUSTERED] [NONCLUSTERED]
INDEX <tên index>
ON <tên bảng>(tên cột,..)

- Unique**: dữ liệu cột Index là duy nhất không lặp lại
- Clustered**: dữ liệu được sắp xếp vật lý trên ổ đĩa
- Nonclustered**: dữ liệu được sắp xếp logic, nhanh trong nhập liệu

Ví dụ: tạo index trên cột MaNV của bảng Nhân viên

```
CREATE INDEX ID_MANV ON NHANVIEN(MANV)
```

Xóa INDEX

- **DROP INDEX**<tên index> ;
- Ví dụ: xóa index vừa thiết lập

```
DROP INDEX ID_MANV;
```

Ví dụ: Tạo CSDL

- NHANVIEN (MANV, HOTen, NS, GT, HSL, DC, MAPB, NgayVL)
- PHONGBAN (MAPB, TENDV , MaTP, NgayBD, **SoNV**)
- DIADIEM_PB (MaPB, DiaDiem)
- DUAN (MADA, TENDA, DIADIEM, **NgBD**, MAPB)
- THANNHAN (HOTEN, NS, GT, QuanHe, MANV)
- *PHANCONG* (MADA, MANV, SoGio, **NgLamDA**)
- **Thỏa mãn các ràng buộc:**
 - Giới tính chỉ nhận 2 giá trị ‘Nam’ và ‘Nữ’
 - Quan hệ của Thân nhân với Nhân viên chỉ có thể: ‘Chồng’, ‘Vợ’, ‘Bố’, ‘Mẹ’, ‘Con’
 - Số giờ làm của nhân viên trong dự Dự án từ 2 đến 10 giờ
 - Hệ số lương mặc định là 3
 - Mã đơn vị được tham chiếu từ bảng ĐƠN VỊ
 - Mã nhân viên được tham chiếu từ bảng NHÂN VIÊN
 - Mã dự án được tham chiếu từ bảng DỰ ÁN