

BÀI THỰC HÀNH 1: LÀM QUEN VỚI SQL SEVER VÀ CÁC THAO TÁC TRÊN BẢNG

I/ LÝ THUYẾT

1/ Ràng buộc toàn vẹn

+ **Ý nghĩa:** là những điều kiện bất biến mà tất cả các bộ của những quan hệ có liên quan trong CSDL đều phải thoả mãn ở mọi thời điểm

Ví dụ:

- Ngày sinh của sinh viên phải nhỏ hơn ngày nhập học
- Điểm của sinh viên phải từ 0 đến 10 là một qui định
- Giới tính của sinh viên chỉ có thể là nam hoặc nữ.
- Sinh viên đăng ký học với những môn học thuộc khoa mà sinh viên đó học
- Trong SQL Server có 6 ràng buộc toàn vẹn: CHECK, NOT NULL, DEFAULT, PRIMARY KEY, FOREIGN KEY, UNIQUE.
- Thuộc tính IDENTITY

+ Phân loại ràng buộc toàn vẹn

RBTV có thể chia thành hai loại:

- Ràng buộc mức cột
- Ràng buộc mức bảng

	Ràng buộc mức cột	Ràng buộc mức bảng
Ý nghĩa	Chỉ áp dụng ở mức cột	Có thể áp dụng ở một hoặc nhiều cột trong bảng
Phạm vi khai báo	Ngay sau khi khai báo cột (sau kiểu dữ liệu và phía trước dấu phẩy)	Khai báo sau khi đã khai báo sau cột cuối cùng của bảng và phía trước dấu đóng ngoặc)

Các loại ràng buộc	NOT NULL, PRIMARY KEY, UNIQUE, FOREIGN KEY, CHECK, DEFAULT	PRIMARY KEY, UNIQUE, FOREIGN KEY, CHECK, DEFAULT
Khai báo	<ul style="list-style-type: none"> - RB không đặt tên - RB có đặt tên 	- RB có đặt tên

2/ Tổng quan về SQL Server

+ Hệ quản trị CSDL: Hệ quản trị cơ sở dữ liệu (Database Management System - DBMS), là phần mềm hay hệ thống được thiết kế để quản trị một cơ sở dữ liệu. Cụ thể, các chương trình thuộc loại này hỗ trợ khả năng lưu trữ, sửa chữa, xóa và tìm kiếm thông tin trong một cơ sở dữ liệu (CSDL).

Các hệ quản trị CSDL phổ biến được nhiều người biết đến là MySQL, Oracle, PostgreSQL, SQL Server, DB2, Infomix...

+ SQL Server là hệ thống quản trị cơ sở dữ liệu quan hệ (Relational DataBase Management System- RDBMS) sử dụng các lệnh Transaction-SQL để trao đổi dữ liệu giữa Client Computer và Server Computer.

+ SQL Server có một số đặc tính sau:

- Cho phép quản trị một hệ CSDL lớn (lên đến vài tera byte), có tốc độ xử lý dữ liệu nhanh đáp ứng yêu cầu về thời gian.
- Cho phép nhiều người cùng khai thác trong một thời điểm đối với một CSDL và toàn bộ quản trị CSDL (lên đến vài chục ngàn user).
- Có hệ thống phân quyền bảo mật tương thích với hệ thống bảo mật của công nghệ NT (Network Technology), tích hợp với hệ thống bảo mật của Windows NT hoặc sử dụng hệ thống bảo vệ độc lập của SQL Server.
- Hỗ trợ trong việc triển khai CSDL phân tán và phát triển ứng dụng trên Internet
- Cho phép lập trình kết nối với nhiều ngôn ngữ lập trình khác dùng xây dựng các ứng dụng đặc thù (Visual Basic, C, C++, ASP, ASP.NET, XML,...)
- Sử dụng câu lệnh truy vấn dữ liệu Transaction-SQL (Access là SQL, Oracle là PL/SQL).

3/ Giới thiệu về SQL

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 chuẩn bao gồm khoảng 40 câu lệnh, tuy nhiên tùy vào HQT CSDL mà cú pháp sẽ có thay đổi nhưng nhìn chung hầu hết đều có cùng dạng và mục đích sử dụng. (Ví dụ SQL Server sử dụng ngôn ngữ T-SQL.)

T-SQL được chia ra làm 3 nhóm lệnh:

- **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: **Grand**, **Revoke**, **Deny**

4/ Các kiểu dữ liệu cơ bản trong SQL Server

Nhóm dữ liệu	Kiểu dữ liệu	Phạm vi	Số byte
Exact Numerics(Số chính xác)	smallint	$-2^{15} \rightarrow 2^{15}$	2 byte
	int	$-2^{15} \rightarrow 2^{15}$ hoặc $-2^{31} \rightarrow 2^{31}$	2byte hoặc 4byte
	tinyint	$0 \rightarrow 255(2^8 - 1)$	1 byte
	bit	gồm 0 và 1	
Approximate Numerics(Số xấp xỉ)	float	-1.79E + 308 $\rightarrow 1.79E + 308$	
Character Strings (Kiểu chuỗi)	char	Kiểu ký tự. Không hỗ trợ Unicode. Bộ nhớ cấp phát tĩnh.	<u>Ví dụ:</u> A, a, B, b, f, k, K. char(10) -> 10 ô nhớ này không ai được động vào. Luôn giữ 10 ô nhớ.
	nchar	Kiểu ký tự. Có hỗ trợ Unicode nên viết được tiếng Việt. Bộ nhớ cấp phát tĩnh.	<u>Ví dụ:</u> Khai báo: Truong Nếu khai báo bằng char vẫn chiếm 10 ô nhớ, varchar chỉ chiếm 5 ô nhớ
	varchar	Kiểu ký tự. Không hỗ trợ Unicode. Bộ nhớ cấp phát động.	
	nvarchar	Kiểu ký tự. Hỗ trợ Unicode nên viết tiếng Việt được. Bộ nhớ cấp phát động	
	text	lưu văn bản lớn	Cấp phát bộ nhớ động theo chiều dài kí tự nhập vào
	ntext	lưu văn bản lớn có tiếng Việt	
Data and Time	datetime	1/1/1753 $\rightarrow 31/12/9999$	
	smalldatetime	1/1/1900 $\rightarrow 6/6/2079$	
	date	Lưu giữ một date dạng June 30, 1991	Lưu trữ ngày, tháng, năm
	time	Lưu giữ một time dạng 12:30 P.M.	Lưu trữ giờ, phút, giây

5/ Các quy tắc viết lệnh trong SQL Server

- ✓ 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

2/ Các lệnh làm việc với CSDL

a/ Tạo CSDL

- 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|%]

<filespec>: để điều khiển các thuộc tính cho file được tạo

- Name: chỉ định tên logical cho file
- Filename: chỉ định tên, đường dẫn file hệ điều hành (file vật lý)
- Size: chỉ định kích thước của file, tối thiểu là 3MB
- Maxsize: chỉ định kích thước tối đa lớn nhất mà file có thể phát triển đến, có từ khóa unlimited chỉ định file được phát triển cho đến khi đĩa bị đầy
- Filegrowth: chỉ định độ tự động gia tăng của file

Ví dụ : Tạo cơ sở dữ liệu có tên là ‘QLSV’

```
CREATE DATABASE QLSV;
```

b/ Sử dụng CSDL

+ Cú pháp: **USE** <tên CSDL>;

Ví dụ: **USE QLSV;**

c/ Đổi tên CSDL

+ Cú pháp:

```
ALTER DATABASE <Tên_CSDL> MODIFY NAME = <tên mới>
```

Ví dụ: Thay đổi tên CSDL

```
ALTER DATABASE QLSV MODIFY NAME = QL_SV;
```

d/ Xóa CSDL: Khi sử dụng lệnh xóa, CSDL sẽ bị xóa khỏi vùng lưu trữ, muốn tạo thì phải thực thi lại lệnh

+ **Cú Pháp: DROP DATABASE <tên CSDL>;**

Ví dụ: DROP DATABASE QL_SV;

3/ Các lệnh làm việc với bảng

3.1/ Tạo cấu trúc bảng

+ **Cú pháp:**

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

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, UNIQUE, DEFAULT, PRIMARY KEY, FOREIGN KEY, CHECK.**
- **Constraint:** dùng khi có nhiều hơn một RBTV cùng loại, đặc biệt là với RBTV khóa chính.

Có các RBTV sau:

- **CHECK:** kiểm tra giá trị của cột thỏa mãn điều kiện sau CHECK
- **NOT NULL:** dữ liệu trên cột không được bỏ trống
- **PRIMARY KEY:** chỉ định khóa chính cho bảng
- **DEFAULT:** gán giá trị mặc định
- **UNIQUE:** giá trị trên cột là duy nhất

- FOREIGN KEY: ràng buộc khóa ngoại. Sử dụng để tham chiếu đến bảng dữ liệu nguồn.

✱ Thuộc tính IDENTITY: giá trị tự tăng, bắt đầu từ giá trị 1 và tự động tăng lên 1 đơn vị;

Nếu khai báo dạng IDENTITY (n,m) thì bắt đầu từ giá trị n và tự động tăng lên m đơn vị.

Ví dụ

Tạo cơ sở dữ liệu quản lý sinh viên gồm các bảng (table)

Sinhvien: các thông tin liên quan tới sinh viên

Tên trường	Kiểu dữ liệu	Chú thích
Masv	int (tự tăng)	Khóa
Tensv	nvarchar	Tên sinh viên, dữ liệu không được bỏ trống
Gioitinh	nvarchar	Giới tính, mặc định là giá trị Nam
Ngaysinh	date	Ngày tháng năm sinh, nhỏ hơn ngày hiện tại
Que	nvarchar	Quê, giá trị không được bỏ trống
Lop	nvarchar	Lớp

Monhoc: Thông tin về các môn học trong nhà trường

Tên trường	Kiểu dữ liệu	Chú thích
Mamh	int (tự tăng)	Khóa
Tenmh	nvarchar	Tên môn học, giá trị trên cột là duy nhất
DVHT	int	Số đơn vị học trình, nằm trong khoảng từ 2 đến 9

Ketqua: Điểm thi của sinh viên

Tên trường	Kiểu dữ liệu	Chú thích
Masv	int	khóa
Mamh	int	
Diem	float	Điểm từ 0-10

```

create database QLSV --Tạo CSDL sinh viên
use QLSV --Sử dụng CSDL
go
Create Table Sinhvien
(
    MaSV int identity primary key,
    TenSV Nvarchar(30) not null,
    GT NVarchar (5) default 'Nam',
    Ngaysinh Date check (Ngaysinh<getdate()),
    Que Nvarchar(50) not null,
    Lop Nvarchar (10)
)
drop table Sinhvien
--Tạo bảng Môn học
Create table Monhoc
(
    MaMH int identity primary key,
    TenMH Nvarchar (20) unique,
    DVHT int check (DVHT between 2 and 9)
)
--Tạo bảng điểm thi
create table Ketqua
(
    MaSV int,
    MaMH int,
    Diem float check (Diem between 0 and 10)
    constraint RB_Khoa primary key (MaSV,MaMH)--Định nghĩa khóa có hai thuộc tính
)

```

3.2 Các thao tác trên bảng

+ Xem thông tin về bảng: **sp_help** <tên bảng>

+ Đổi tên bảng: **exec sp_rename** <tên cũ> <tên mới>

+ Xóa bảng: **Drop Table** <tên bảng>

+ Thêm cột vào bảng:

```
ALTER Table <tên_bảng>
```

```
ADD <tên_Cột> Data_type [RBTv][,..]
```

+ Thay đổi kiểu dữ liệu của cột

```
ALTER Table <tên_bảng>
```

```
ALTER Column <tên_cột> <kiểu dl mới>
```

+ Hủy bỏ một cột trong bảng:

```
ALTER Table <tên_bảng>
```

```
DROP Column <ds tên_cột>
```

+ Thêm ràng buộc cho cột

```
ALTER Table <tên_bảng>
```

```
ADD Constraint <tên_ràng_buộc> <Loại_ràng_buộc> (tên cột)
```

Chi tiết các loại ràng buộc:

```
ALTER TABLE bang
ADD [CONSTRAINT ten_constraint]
    PRIMARY KEY (danh_sach_cot_khoa_chinh)
hoặc
    UNIQUE (danh_sach_cot)
hoặc
    FOREIGN KEY (danh_sach_cot_khoa_ngoai)
    REFERENCES bang_tham_chieu (ds_cot_tham_chieu)
hoặc
    CHECK (bieu_thuc_luan_ly)
hoặc
    DEFAULT gia_tri_mac_dinh FOR Ten_cot
```

+ Hủy ràng buộc đã đặt

```
ALTER Table <tên_bảng>
```

```
DROP Constraint <tên_RB>
```

+ Đổi tên cột:

```
EXEC SP_Rename '<tên bảng.tên cột>', 'tên mới', 'COLUMN'
```

+**Bật /tắt các ràng buộc**

- Tắt ràng buộc trong bảng:

ALTER TABLE Tên_bảng

NOCHECK CONSTRAINT ALL | Tên_constraint [...]

- Bật ràng buộc trong bảng

ALTER TABLE Tên_bảng

CHECK CONSTRAINT ALL | Tên_constraint [...]

Ví dụ:

--Xem thông tin về các bảng

sp_help sinhvien

sp_help monhoc

sp_help Ketqua

-- Chỉnh sửa bảng

-- Đổi tên bảng

exec sp_rename "sinhvien", "SV"

exec sp_rename "SV","sinhvien"

--

/*Chèn thêm 1 cột*/

Alter table Sinhvien

ADD

Dienthoai varchar(11),

CMTND nvarchar(40)

/*thay đổi kiểu dữ liệu của cột*/

Alter table sinhvien

Alter column CMTND int not null

/*thêm ràng buộc*/

alter table Monhoc

ADD

constraint RB_TenMH Unique(TenMH)

/*Từ khóa unique giúp trường đó có giá trị duy nhất VD như CMTND*/

/*Xóa cột*/

Alter table Sinhvien

Drop column CMTND, Dienthoai

/*Xóa ràng buộc*/

Alter table monhoc

Drop constraint RB_TenMH

3.3 Tạo liên kết giữa các bảng (Tạo ràng buộc khóa Ngoại)

Một khóa ngoại là một cột hoặc sự kết hợp của các cột được sử dụng để tạo một liên kết giữa dữ liệu trong hai bảng. Một số chú ý khi tạo liên kết giữa các bảng:

- Cột được tham chiếu trong bảng tham chiếu phải là khóa chính (hoặc là khóa phụ).
- Bảng tham chiếu phải được định nghĩa trước

+ Cú pháp: có hai cách tạo liên kết khóa ngoại

- Cách 1: Tạo lập ngay trong quá trình tạo bảng với điều kiện bảng khóa chính đã được định nghĩa trước

CREATE TABLE Tên_bảng

(

.....

FOREIGN KEY (Tên_cột) **REFERENCES** <Tên bảng tham chiếu_khoáchính>

)

- Cách 2: Tạo liên kết sau khi tạo các bảng (cách 2 hay được sử dụng để tránh tình trạng các bảng có mối quan hệ vòng)

ALTER TABLE Tên_bảng

ADD

CONSTRAINT <Tên_liên_kết> **FOREIGN KEY** (Tên_cột_khoá_chính)

REFERENCES <Tên bảng tham chiếu> (tên cột tham chiếu)

Ví dụ 1: Tạo liên kết ngay trong quá trình tạo bảng

```
create table Ketqua
(
  MaSV int,
  MaMH int,
  Diem int check (Diem between 0 and 10)
  constraint RB_Khoa primary key (MaSV,MaMH),
  Foreign key (MaSV) References sinhvien, --Tạo liên kết với bảng Sinhvien
  Foreign key (MaMH) References Monhoc    --Tạo liên kết với bảng môn học
)
```

Ví dụ 2: Tạo liên kết sau khi đã tạo bảng

```
Alter Table Ketqua
```

```
Add
```

```
Constraint R1 foreign key (MaSV) References Sinhvien(MaSV)
```

```
Alter Table Ketqua
```

```
Add
```

```
Constraint R2 Foreign Key (MaMH) References Monhoc(MaMH)
```

Chú ý: Sau khi tạo liên kết khóa ngoại, muốn xóa các bảng có trong liên kết ta phải thực hiện lệnh xóa ràng buộc khóa ngoại hoặc xóa bảng chứa ràng buộc khóa ngoại. Ví dụ, sau khi tạo ràng buộc giữa ba bảng Sinhvien, Monhoc và Ketqua, muốn xóa bảng Sinhvien ta phải xóa ràng buộc bằng lệnh sau:

```
Alter Table Ketqua
```

```
Drop constraint R1,R2
```

4. Tạo chỉ mục 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;

II. BÀI TẬP

Bài 1: Cho CSDL “Thu_vien” gồm các quan hệ sau:

BANDOC(maBD, hoteBD, ngaysinh, lop, quequan, sdt): mô tả thông tin về Bạn đọc bao gồm các thông tin: mã bạn đọc, họ tên bạn đọc, ngày sinh, lớp, quê quán và số điện thoại. Mỗi bạn đọc có một mã duy nhất, xác định các thông tin còn lại của bạn đọc.

SACH(maS, TenS, TheLoai, Tacgia, NamXB, NhaXB): mô tả thông tin về Sách, bao gồm: mã sách, tên sách, thể loại, tác giả, năm xuất bản, nhà xuất bản. Mã sách là thuộc tính khóa

PHIEUMUON(MaBD, MaS, Ngaymuon, Ngayhentra, TraSach): mô tả thông tin về quá trình mượn sách của bạn đọc thông qua phiếu mượn, bao gồm: mã bạn đọc, mã sách, ngày mượn sách, ngày hẹn trả, đã trả sách hay chưa (TraSach). Mỗi quyển sách chỉ được mượn tối đa 5 ngày, TraSach bằng 1 nếu đã trả hoặc bằng 0 nếu chưa trả. Khóa chính của bảng là (MaBD, MaS)

Yêu cầu: Tạo các bảng với các ràng buộc sau:

- Ràng buộc khóa chính, ràng buộc khóa ngoại
- Ràng buộc not null (đối với kiểu ký tự)
- *Ngaysinh* luôn nhỏ hơn ngày hiện tại của hệ thống.
- Giá trị mặc định của TraSach là 0.

Bài 2: Cho CSDL “QLMB” gồm các quan hệ sau:

CHUYENBAY(MaCB, GaDi, GaDen, DoDai, GioDi, GioDen, ChiPhi): mô tả thông tin về chuyến bay. Mỗi chuyến bay có một mã số duy nhất, đường bay, giờ đi và giờ đến. Thông tin về đường bay được mô tả bởi ga đi, ga đến, độ dài đường bay, chi phí phải trả cho phi công.

MAYBAY(MaMB, Hieu, TamBay): mô tả thông tin về máy bay. Mỗi máy bay có một mã số duy nhất, tên phân loại(Hieu) và tầm bay là khoảng cách xa nhất máy bay có thể bay mà không cần tiếp nhiên liệu. Một máy bay chỉ có thể thực hiện các chuyến bay có độ dài đường bay nhỏ hơn tầm bay của máy bay đó.

NHANVIEN(MaNV, Ten, Luong) mô tả thông tin về nhân viên phi hành đoàn gồm phi công và tiếp viên. Mỗi nhân viên có một mã số duy nhất, tên và mức lương.

CHUNGNHAN(MaNV, MaMB): mô tả thông tin về khả năng điều khiển máy bay của phi công nào đó. Một phi công chỉ có thể lái một chuyến bay nếu như phi công đó được công nhận có khả năng lái được loại máy bay có thể thực hiện chuyến bay đó.

+ **Mô tả các thuộc tính:**Tạo các ràng buộc khóa ngoại, khóa chính và not null.

Thuộc tính	Miền xác định
MaCB	char(5)
GaDi	varchar(50)
GaDen	varchar(50)
DoDai	int
GioDi	time
GioDen	time
ChiPhi	int
MaMB	int
Hieu	varchar(50)
TamBay	int
MaNV	char(9)
Ten	varchar(50)
Luong	int