
Chương 3

XÂY DỰNG CSDL QUAN HỆ TRONG SQL SERVER

Mục tiêu

- Sau khi học xong chương này, SV có thể:
 - Xây dựng được CSDL quan hệ trong SQL Server.
 - Tạo được các bảng dữ liệu.
 - Định nghĩa được các ràng buộc toàn vẹn.
 - Biết cách tạo dữ liệu mẫu.

Nội dung chi tiết

- Tạo Cơ sở dữ liệu (Database)
- Tạo Bảng (Table)
- Định nghĩa các ràng buộc toàn vẹn (Constraint)
- Tạo dữ liệu mẫu
- Bài tập ứng dụng

Tạo Cơ sở dữ liệu (Database)

- Là bước thiết kế CSDL ở mức vật lý.
- Từ mô hình dữ liệu quan hệ, xây dựng CSDL quan hệ trong 1 Hệ quản trị CSDL cụ thể.
- Cần phải bổ sung các ràng buộc và đảm bảo các yêu cầu về toàn vẹn dữ liệu.

Tạo Cơ sở dữ liệu (Database)

■ Thiết kế các bảng:

- Là bước đầu tiên trong thiết kế CSDL vật lý.
- Mỗi bảng biểu diễn một quan hệ đã được chuẩn hóa.
- VD: Quan hệ MONHOC (MAMH, TENMH, SOTIET) được biểu diễn dưới dạng bảng như sau:

MONHOC

<u>MAMH</u>	TENMH	SOTIET
-------------	-------	--------

Tạo Cơ sở dữ liệu (Database)

■ Thiết kế các bảng:

- Các quan hệ đã chuẩn hóa trở thành các bảng.
- Các thuộc tính trong quan hệ trở thành các cột trong bảng. Mỗi thuộc tính có tên, kiểu dữ liệu, miền giá trị và ràng buộc cho các thuộc tính (nếu có).
- Khóa của quan hệ trở thành khóa chính (Primary Key) của bảng và có giá trị NOT NULL.
- Các liên kết giữa các quan hệ trở thành ràng buộc tham chiếu (ràng buộc khóa ngoại).

Tạo Cơ sở dữ liệu (Database)

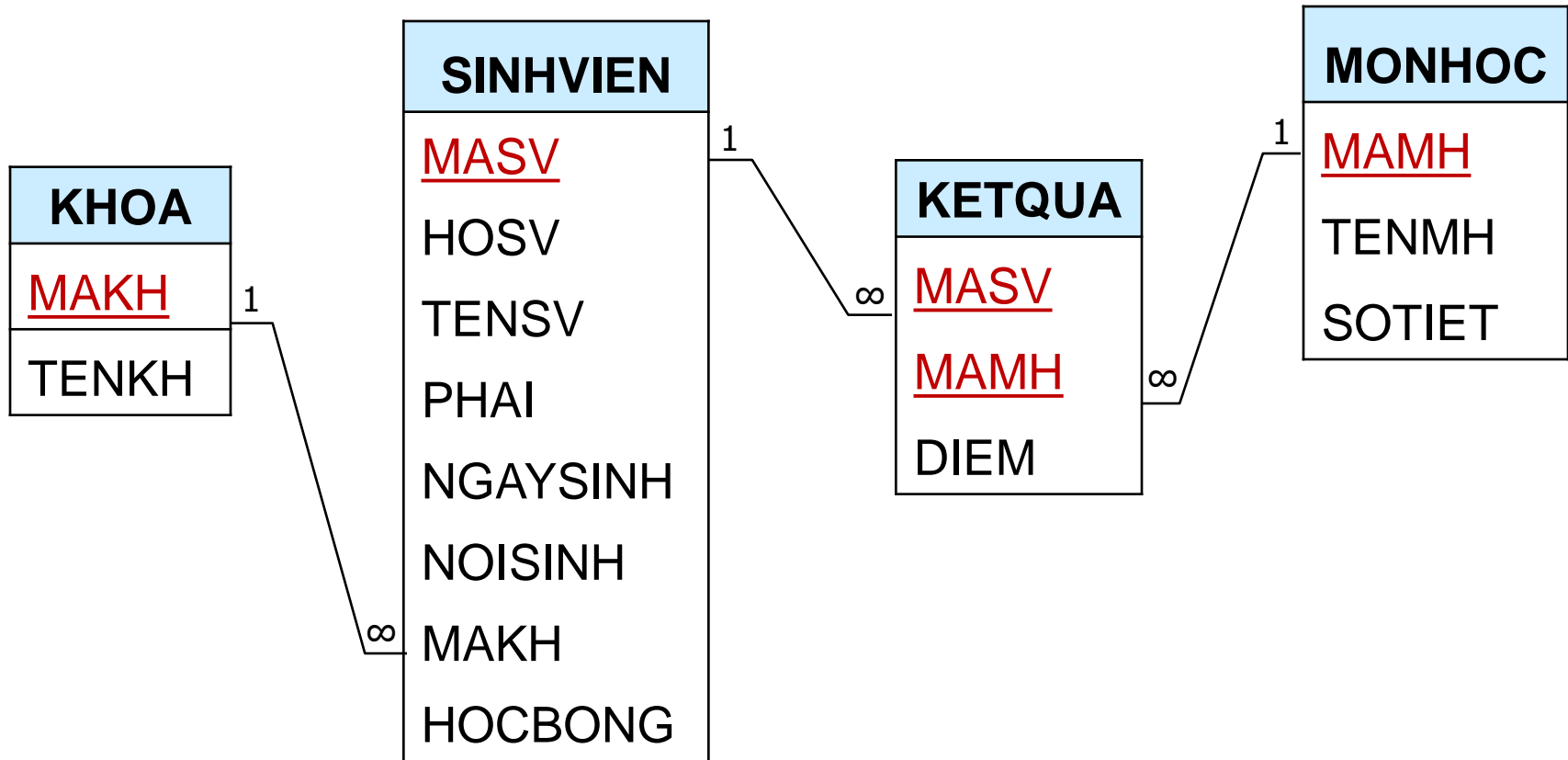
■ Thiết kế các bảng:

- VD: CSDL “Quản lý sinh viên” gồm 4 quan hệ như sau:
 - **KHOA** (MAKH, TENKH)
 - **SINHVIEN** (MASV, HOSV, TENSX, PHAI, NGAYSINH, NOISINH, MAKH, HOCBONG)
 - **KETQUA** (MASV, MAMH, DIEM)
 - **MONHOC** (MAMH, TENMH, SOTIET)

Tạo Cơ sở dữ liệu (Database)

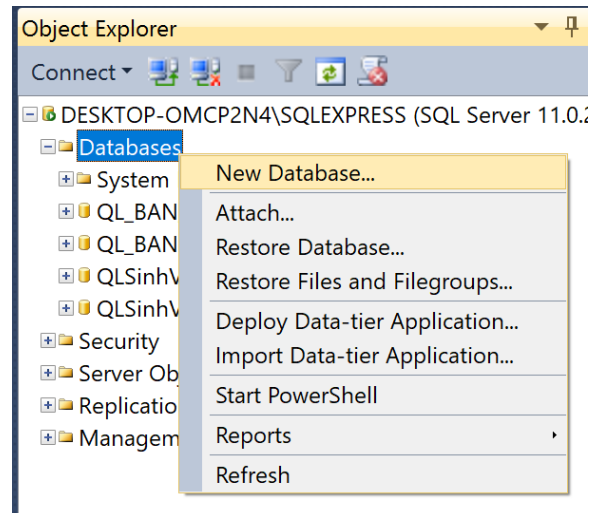
■ Thiết kế các bảng:

- 4 quan hệ tương ứng với 4 bảng và ràng buộc giữa các bảng như sau:



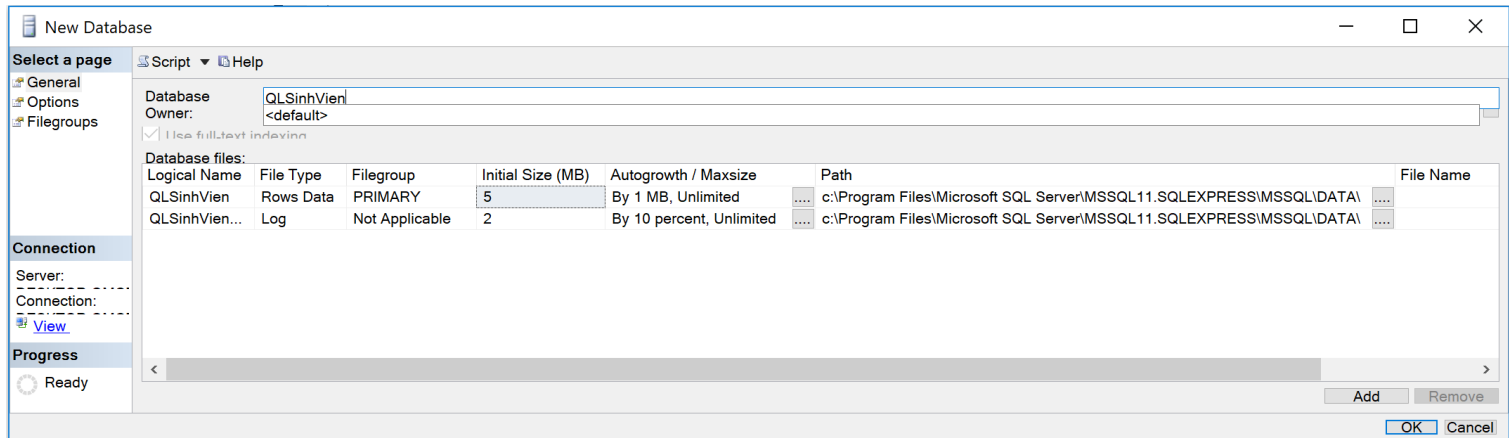
Tạo Cơ sở dữ liệu (Database)

- Trong SQL Server, có 2 cách để thực hiện thao tác tạo CSDL:
 - Sử dụng giao diện
 - Sử dụng câu lệnh (script)
- **Cách 1: Sử dụng tiện ích Management Studio:**
 - Click chuột phải vào đối tượng **Databases**, chọn **New Database...**



Tạo Cơ sở dữ liệu (Database)

- Cách 1: Sử dụng tiện ích Management Studio:
 - Nhập tên CSDL và nhấn nút OK.



- Database files:
 - ✓ **Logical Name:** tên tập tin Data Files / Log Files
 - ✓ **Initial Size (MB):** kích thước ban đầu
 - ✓ **Autogrowth/ Maxsize:** tự động tăng kích thước / kích thước tối đa
 - **By ? MB, Limited to ? MB:** theo MB, giới hạn tối đa ? MB
 - **By ? Percent, Unlimited:** theo phần trăm, không giới hạn kích thước.
 - ✓ **Path:** vị trí lưu trữ
 - ✓ **File Name:** tên tập tin vật lý sẽ được tạo ra và lưu trữ trên server.

Tạo Cơ sở dữ liệu (Database)

■ Cách 2: Sử dụng tiện ích Query Editor:

- Viết câu lệnh sau:

```
create database Tên_CSDL
on primary
(name = 'Tên_CSDL',
filename = 'Đường_dẫn\ Tên_CSDL.mdf',
size = Kích_thước_ban_đầu_MB,
maxsize = Kích_thước_tối_đa_MB,
filegrowth = Kích_thước_tăng)
log on
(name = 'Tên_CSDL_Log',
filename = 'Đường_dẫn\ Tên_CSDL_Log.ldf',
size = Kích_thước_ban_đầu_MB,
maxsize = Kích_thước_tối_đa_MB,
filegrowth = Kích_thước_tăng)
go
```

```
create database QLSinhVien
on primary
(name = 'QLSinhVien',
filename = 'D:\ QLSinhVien.mdf',
size = 50MB,
maxsize = 200MB,
filegrowth = 10MB)
log on
(name = 'QLSinhVienLog',
filename = 'D:\ QLSinhVienLog.ldf',
size = 10MB,
maxsize = unlimited,
filegrowth = 5MB)
go
```

Tạo Cơ sở dữ liệu (Database)

■ Cách 2: Sử dụng tiện ích Query Editor:

- Hoặc viết câu lệnh ngắn gọn với các thiết lập mặc định như sau:

```
create database Tên_CSDL  
go
```

- VD: Tạo CSDL QLSinhVien.

```
create database QLSinhVien  
go
```

Tạo Cơ sở dữ liệu (Database)

■ Cách 2: Sử dụng tiện ích Query Editor:

- Viết câu lệnh sử dụng CSDL như sau:

```
use Tên_CSDL  
go
```

- VD: Sử dụng CSDL QLSinhVien.

```
use QLSinhVien  
go
```

Tạo Bảng (Table)

- Một số kiểu dữ liệu thông dụng trong SQL Server:
 - Kiểu chuỗi:

Kiểu dữ liệu	Miêu tả
Kiểu chuỗi không chứa Unicode	
char(n)	độ dài cố định, tối đa là n ký tự ($n \leq 8000$).
varchar(n)	độ dài tùy biến, tối đa là n ký tự ($n \leq 8000$).
text	độ dài tùy biến, tối đa là 2.147.483.647 ký tự.
Kiểu chuỗi có chứa Unicode	
nchar(n)	độ dài cố định, tối đa là n ký tự ($n \leq 4000$).
nvarchar(n)	độ dài tùy biến, tối đa là n ký tự ($n \leq 4000$).
ntext	độ dài tùy biến, tối đa là 1.073.741.823 ký tự.

Tạo Bảng (Table)

- Một số kiểu dữ liệu thông dụng trong SQL Server:
 - Kiểu số:

Kiểu dữ liệu	Miêu tả
Số nguyên	
tinyint	Chứa các số từ 0 đến 255 .
smallint	Chứa các số từ -32,768 đến 32,767 .
int	Chứa các số từ -2,147,483,648 đến 2,147,483,647 .
bit	Chứa 2 số: 0 và 1
money	Chứa các số từ -922,337,203,685,477.5808 đến 922,337,203,685,477.5807 .
Số thực	
float	Chứa các số từ -1.79E + 308 đến 1.79E + 308 .
real	Chứa các số từ -3.40E + 38 đến 3.40E + 38 .

Tạo Bảng (Table)

■ Một số kiểu dữ liệu thông dụng trong SQL Server:

- Kiểu ngày tháng:

Kiểu dữ liệu	Miêu tả
datetime	Từ Jan 1, 1753 đến Dec 31, 9999
date	Từ Jan 1, 0001 đến Dec 31, 9999

- Kiểu dữ liệu khác:

Kiểu dữ liệu	Miêu tả
cursor	tham chiếu tới một đối tượng con trỏ (Cursor)
table	lưu giữ một tập hợp kết quả để xử lý vào lần sau

Tạo Bảng (Table)

- Câu lệnh tạo bảng có 1 cột là khoá chính:

```
create table Tên_bảng (  
    Tên_cột_thứ_1 Kiểu_dữ_liệu primary key,  
    Tên_cột_thứ_2 Kiểu_dữ_liệu,  
    ...  
    Tên_cột_thứ_n Kiểu_dữ_liệu  
)
```

- Câu lệnh tạo bảng có nhiều cột là khoá chính:

```
create table Tên_bảng (  
    Tên_cột_thứ_1 Kiểu_dữ_liệu,  
    Tên_cột_thứ_2 Kiểu_dữ_liệu,  
    ...  
    Tên_cột_thứ_n Kiểu_dữ_liệu,  
    primary key (danh_sách_tên_các_cột_là_khoá_chính)  
)
```

Tạo Bảng (Table)

■ Ví dụ 1:

- Tạo bảng KHOA gồm có: mã khoa (khoá chính) và tên khoa.

Cách 1:

```
create table KHOA (  
    MAKH char(2) primary key,  
    TENKH nvarchar(30)  
)
```

Cách 2:

```
create table KHOA (  
    MAKH char(2),  
    TENKH nvarchar(30),  
    primary key (MAKH)  
)
```

Cách 3:

```
create table KHOA (  
    MAKH char(2),  
    TENKH nvarchar(30),  
    constraint PK_KHOA_MAKH  
    primary key (MAKH)  
)
```

Tạo Bảng (Table)

- Câu lệnh tạo bảng có khoá chính và khoá ngoại:

```
create table Tên_bảng (  
    Tên_cột_thứ_1 Kiểu_dữ_liệu,  
    Tên_cột_thứ_2 Kiểu_dữ_liệu,  
    ...  
    Tên_cột_thứ_n Kiểu_dữ_liệu,  
    primary key (danh_sách_tên_các_cột_là_khoá_chính),  
    foreign key(tên_cột_khoá_ngoại) references Bảng_tham_chiếu (tên_cột_khoá_chính)  
)
```

Tạo Bảng (Table)

■ Ví dụ 2:

- Tạo bảng SINHVIEN gồm có: mã sinh viên (khoá chính), họ sinh viên, tên sinh viên, phái, ngày sinh, nơi sinh, mã khoa (khoá ngoại tham chiếu đến cột mã khoa trong bảng KHOA) và học bổng.

```
create table SINHVIEN (  
    MASV char(3) primary key,  
    HOSV nvarchar(15),  
    TENSX nvarchar(7),  
    PHAI bit,  
    NGAYSINH datetime,  
    NOISINH nvarchar(15),  
    MAKH char(2),  
    HOCBONG int,  
    foreign key(MAKH) references KHOA(MAKH)  
)
```

Tạo Bảng (Table)

■ Ví dụ 3:

- Tạo bảng KETQUA gồm có: mã sinh viên và mã môn học (khoá chính đồng thời) và điểm. Mã sinh viên là khoá ngoại tham chiếu đến cột mã sinh viên trong bảng SINHVIEN và mã môn học cũng là khoá ngoại tham chiếu đến cột mã môn học trong bảng MONHOC.

```
create table KETQUA(  
    MASV char(3),  
    MAMH char(2),  
    DIEM real,  
    primary key(MASV,MAMH),  
    foreign key(MASV) references SINHVIEN(MASV),  
    foreign key(MAMH) references MONHOC(MAMH)  
)
```

Tạo Bảng (Table)

■ Thiết lập cột định danh tăng tự động:

- Giá trị của cột định danh là số nguyên không trùng lặp do hệ thống tự động phát sinh tuần tự cho mỗi hàng trong một bảng.
- Một bảng chỉ được một cột có thuộc tính Identity.
- Cú pháp:

IDENTITY[(seed, increment)]

- Trong đó:
 - **seed**: giá trị khởi đầu.
 - **increment**: giá trị tăng.
- VD:
 - **IDENTITY** hay **IDENTITY(1, 1)**: giá trị khởi đầu là 1 và các giá trị sẽ được tự động tạo ra là 1, 2, 3, ...
 - **IDENTITY(3, 5)**: giá trị khởi đầu là 3 và các giá trị sẽ được tự động tạo ra là 3, 8, 13, ...

```
create table MONHOC(  
    MAMH int identity primary key,  
    TENMH nvarchar(25),  
    SOTIET int  
)
```

Định nghĩa các ràng buộc toàn vẹn (Constraint)

■ Khái niệm:

- Thiết lập ràng buộc trong SQL Server là thao tác thiết lập những qui định khi thêm mới hay thay đổi dữ liệu.
- Các ràng buộc trong SQL Server được quản lý bởi hai đối tượng: **Constraint** và **Trigger**.
 - **Constraint**: quản lý các ràng buộc dữ liệu duy nhất, dữ liệu tồn tại, miền giá trị, ...
 - **Trigger**: quản lý các ràng buộc phức tạp, liên quan đến dữ liệu trên nhiều bảng.

Định nghĩa các ràng buộc toàn vẹn (Constraint)

■ Các loại ràng buộc dữ liệu:

- **Ràng buộc khoá chính (primary key constraint):** dữ liệu phải khác rỗng và duy nhất, có thể được thiết lập trong quá trình tạo bảng hoặc sau khi tạo bảng với cú pháp như sau:

```
alter table Tên_bảng  
add constraint Tên_constraint primary key (danh_sách_tên_các_cột_là_khoá_chính)
```

VD:

```
create table KHOA (  
    MAKH char(2) not null,  
    TENKH nvarchar(30)  
)  
alter table KHOA  
add primary key(MAKH)
```

Hoặc:

```
create table KHOA (  
    MAKH char(2) not null,  
    TENKH nvarchar(30)  
)  
alter table KHOA  
add constraint PK_KHOA_MAKH primary key(MAKH)
```


Định nghĩa các ràng buộc toàn vẹn (Constraint)

■ Các loại ràng buộc dữ liệu:

- **Ràng buộc khoá ngoại (foreign key constraint)**: tham chiếu đến cột làm khóa chính trong 1 bảng khác, phải tạo bảng có cột làm khóa chính trước khi tạo bảng có khóa ngoại. Trong một bảng có thể định nghĩa nhiều khoá ngoại.
- Cũng có thể được thiết lập trong quá trình tạo bảng hoặc sau khi tạo bảng với cú pháp như sau:

```
alter table Tên_bảng  
add constraint Tên_constraint foreign key (Tên_cột_khoá_ngoại)  
references Bảng_tham_chiếu (Tên_cột_khoá_chính)
```

Định nghĩa các ràng buộc toàn vẹn (Constraint)

■ Các loại ràng buộc dữ liệu:

- Ràng buộc khoá ngoại (foreign key constraint):
- VD1:

```
create table SINHVIEN (  
    MASV char(3) not null,  
    HOSV nvarchar(15),  
    TENSX nvarchar(7),  
    PHAI bit,  
    NGAYSINH datetime,  
    NOISINH nvarchar(15),  
    MAKH char(2),  
    HOCBONG int
```

```
)  
alter table SINHVIEN  
add constraint PK_SINHVIEN_MASV primary key(MASV),  
    constraint FK_SINHVIEN_MAKH foreign key(MAKH) references KHOA(MAKH)
```

Định nghĩa các ràng buộc toàn vẹn (Constraint)

- Các loại ràng buộc dữ liệu:
 - Ràng buộc khoá ngoại (foreign key constraint):
 - VD2:

```
create table KETQUA(
```

```
    MASV char(3) not null,
```

```
    MAMH char(2) not null,
```

```
    DIEM real
```

```
)
```

```
alter table KETQUA
```

```
add constraint PK_KETQUA_MASV_MAMH primary key(MASV,MAMH),
```

```
    constraint FK_KETQUA_MASV foreign key(MASV) references SINHVIEN(MASV) ,
```

```
    constraint FK_KETQUA_MAMH foreign key(MAMH) references MONHOC(MAMH)
```

Định nghĩa các ràng buộc toàn vẹn (Constraint)

■ Các loại ràng buộc dữ liệu:

- Ràng buộc dữ liệu duy nhất (unique constraint): dữ liệu có thể rỗng, nhưng nếu có dữ liệu, thì dữ liệu phải duy nhất, không cho phép dữ liệu bị trùng lặp.

- VD:

```
create table KHOA (  
    MAKH char(2) primary key,  
    TENKH nvarchar(30) unique  
)
```

Hoặc:

```
create table KHOA (  
    MAKH char(2),  
    TENKH nvarchar(30),  
    constraint PK_KHOA_MAKH primary key (MAKH),  
    constraint UQ_KHOA_TENKH unique (TENKH)  
)
```

Định nghĩa các ràng buộc toàn vẹn (Constraint)

■ Các loại ràng buộc dữ liệu:

- Ràng buộc dữ liệu duy nhất (unique constraint): có thể được thiết lập trong quá trình tạo bảng hoặc sau khi tạo bảng với cú pháp như sau:

`alter table Tên_bảng`

`add constraint Tên_constraint unique(Tên_cột_duy_nhất)`

- VD:

```
create table KHOA (  
    MAKH char(2) not null,  
    TENKH nvarchar(30)  
)
```

```
alter table KHOA
```

```
add constraint PK_KHOA_MAKH primary key(MAKH),  
    constraint UQ_KHOA_TENKH unique (TENKH)
```

Định nghĩa các ràng buộc toàn vẹn (Constraint)

■ Các loại ràng buộc dữ liệu:

- **Ràng buộc miền giá trị (check constraint):** quy định dữ liệu được thêm mới hoặc thay đổi phải thoả điều kiện được chỉ định trong lúc thiết lập. Trong một bảng có thể định nghĩa nhiều ràng buộc miền giá trị.
 - Các phép so sánh gồm: > (lớn hơn), < (nhỏ hơn), >= (lớn hơn hoặc bằng), <= (nhỏ hơn hoặc bằng), <> (hoặc !=) (khác), in (giá trị 1, giá trị 2, ...) (là 1 phần tử trong tập hợp các giá trị).
 - Các phép kết hợp gồm: and (và), or (hay, hoặc), between giá trị 1 and giá trị 2 (nằm trong khoảng từ giá trị 1 đến giá trị 2)

Định nghĩa các ràng buộc toàn vẹn (Constraint)

■ Các loại ràng buộc dữ liệu:

- Ràng buộc miền giá trị (check constraint):
- VD:

```
create table KETQUA (  
    MASV char(3),  
    MAMH char(2),  
    DIEM real check(DIEM >=0 and DIEM<=10),  
    primary key(MASV,MAMH),  
    foreign key(MASV) references SINHVIEN(MASV),  
    foreign key(MAMH) references MONHOC(MAMH)  
)
```

Định nghĩa các ràng buộc toàn vẹn (Constraint)

- Các loại ràng buộc dữ liệu:
 - Ràng buộc miền giá trị (check constraint):
 - VD:

```
create table KETQUA (  
    MASV char(3),  
    MAMH char(2),  
    DIEM real,  
    constraint PK_KETQUA_MASV_MAMH primary key (MASV, MAMH),  
    constraint FK_KETQUA_MASV foreign key (MASV) references SINHVIEN(MASV),  
    constraint FK_KETQUA_MAMH foreign key (MAMH) references MONHOC(MAMH),  
    constraint CK_KETQUA_DIEM check (DIEM>=0 AND DIEM<=10)  
)
```


Định nghĩa các ràng buộc toàn vẹn (Constraint)

■ Các loại ràng buộc dữ liệu:

- Ràng buộc miền giá trị (check constraint): có thể được thiết lập trong quá trình tạo bảng hoặc sau khi tạo bảng với cú pháp như sau:

```
alter table Tên_bảng
```

```
add constraint Tên_constraint check(biểu_thức_điều_kiện)
```

- VD:

```
create table KETQUA (  
  MASV char(3) not null,  
  MAMH char(2) not null,  
  DIEM real  
)
```

```
alter table KETQUA
```

```
add constraint PK_KETQUA_MASV_MAMH primary key(MASV,MAMH),  
  constraint FK_KETQUA_MASV foreign key(MASV) references SINHVIEN(MASV),  
  constraint FK_KETQUA_MAMH foreign key(MAMH) references MONHOC(MAMH),  
  constraint CK_KETQUA_DIEM check(DIEM between 0 and 10)
```

Định nghĩa các ràng buộc toàn vẹn (Constraint)

■ Các loại ràng buộc dữ liệu:

- Ràng buộc dữ liệu mặc định (default constraint): gán giá trị mặc định trong trường hợp dữ liệu của cột không được nhập vào hay không được xác định.

- VD:

```
create table MONHOC (  
    MAMH char(2) not null,  
    TENMH nvarchar(25),  
    sotiet int default 30,  
    constraint PK_MONHOC_MAMH primary key(MAMH),  
    constraint UQ_MONHOC_TENMH unique(TENMH),  
    constraint CK_MONHOC_SOTIET check (sotiet>=30)  
)
```

Định nghĩa các ràng buộc toàn vẹn (Constraint)

■ Các loại ràng buộc dữ liệu:

- Ràng buộc dữ liệu mặc định (default constraint): có thể được thiết lập trong quá trình tạo bảng hoặc sau khi tạo bảng với cú pháp như sau:

`alter table Tên_bảng`

`add constraint Tên_constraint default Giá_trị_mặc_định for Tên_cột`

- VD:

```
create table MONHOC (  
    MAMH char(2) not null,  
    TENMH nvarchar(25),  
    SOTIET int  
)
```

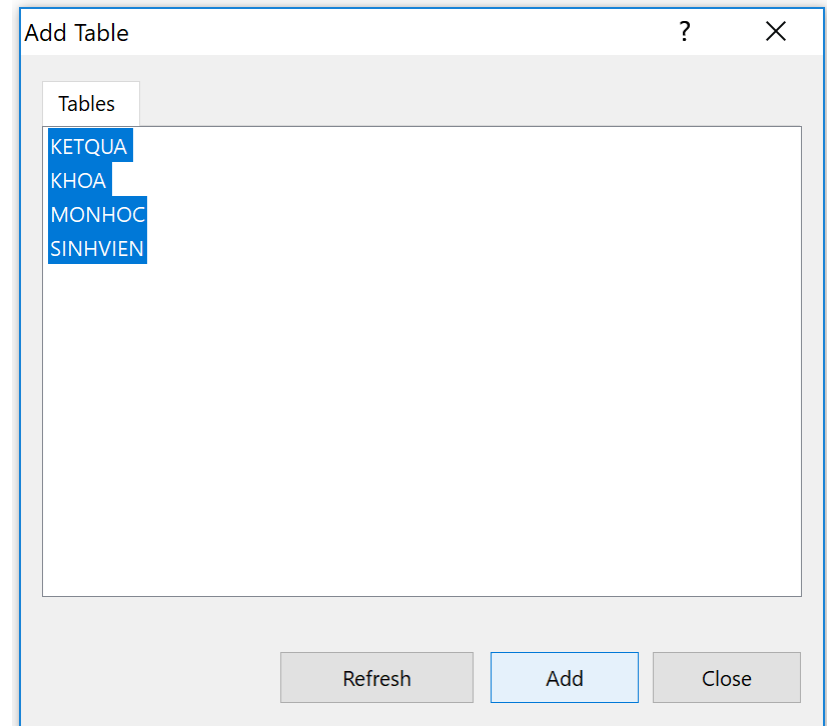
```
alter table MONHOC
```

```
add constraint PK_MONHOC_MAMH primary key(MAMH),  
constraint UQ_MONHOC_TENMH unique(TENMH),  
constraint CK_MONHOC_SOTIET check(SOTIET>=30),  
constraint DF_MONHOC_SOTIET default 30 for SOTIET
```

Định nghĩa các ràng buộc toàn vẹn (Constraint)

■ Tạo mô hình quan hệ (Database Diagram):

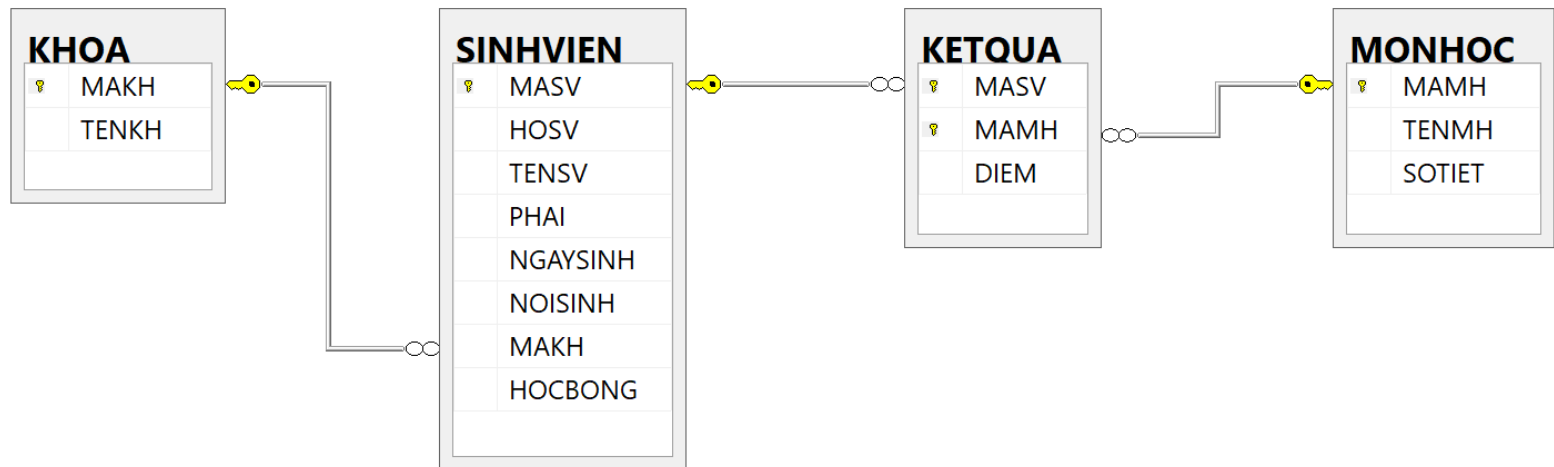
- Nhấp chuột phải vào đối tượng Database Diagrams trong CSDL, chọn lệnh New Database Diagram.
- Nhấn nút Yes.
- Chọn các bảng dữ liệu muốn tạo mô hình quan hệ.
- Nhấn nút Add.



Định nghĩa các ràng buộc toàn vẹn (Constraint)

■ Tạo mô hình quan hệ (Diagram):

- Chương trình tự động tạo mô hình quan hệ theo các ràng buộc khóa ngoại đã khai báo.



Tạo dữ liệu mẫu

- Là thao tác nhập dữ liệu vào cho các bảng.
- Cú pháp đầy đủ:

**insert into Tên_bảng (Tên_cột_1, Tên_Cột_2, ..., Tên_Cột_n)
values (Giá_trị_1, Giá_trị_2, ..., Giá_trị_n)**

- Có thể lược bỏ (Tên_cột_1, Tên_Cột_2, ..., Tên_Cột_n) nếu nhập liệu đúng với thứ tự và cấu trúc các cột đã khai báo trong lúc tạo bảng.

- Cú pháp ngắn gọn:

insert into Tên_bảng values (Giá_trị_1, Giá_trị_2, ..., Giá_trị_n)

Tạo dữ liệu mẫu

■ Quy cách nhập liệu:

- **Kiểu chuỗi:** đặt trong cặp dấu nháy đơn ' '. Nếu nhập liệu có dấu (kiểu nchar hoặc nvarchar) thì thêm ký tự N phía trước.

- **VD1:**

insert into KHOA values ('AV', N'Anh Văn')

- **Kiểu số:** nhập giá trị số, không đặt trong cặp dấu nháy đơn ' '.

- **VD2: Trường hợp mã môn học là cột có kiểu chuỗi**

insert into MONHOC values('01', N'Nhập môn máy tính', 30)

- **VD3: Trường hợp mã môn học là cột định danh tăng tự động:**

không nhập liệu cột mã môn học, chỉ nhập liệu cho 2 cột TENMH và SOTIET.

insert into MONHOC(TENMH, SOTIET) values (N'Nhập môn máy tính', 30)

- **Kiểu ngày:** nhập giá trị tháng trước, ngày sau (yyyy-mm-dd) và đặt trong cặp dấu nháy đơn ' '.

- **VD4:**

insert into SINHVIEN values ('A01', N'Nguyễn Thu', N'Hải', 0, '1980-02-23', N'TP.HCM', 'AV', 100000)

Bài tập ứng dụng

- Tạo CSDL QLSINHVIEN như sau (các field được gạch dưới và in đậm là khóa chính của bảng):

KHOA – Danh sách khoa

Field Name	Field Type	Field Size	Description
<u>MAKH</u>	char	2	Mã khoa
TENKH	nvarchar	30	Tên khoa

- Tên khoa phải duy nhất.

MONHOC – Danh sách môn học

Field Name	Field Type	Field Size	Description
<u>MAMH</u>	char	2	Mã môn học
TENMH	nvarchar	25	Tên môn học
SOTIET	int	≥ 30	Số tiết

- Tên môn học phải duy nhất.
- Số tiết phải từ 30 trở lên.
- Giá trị mặc định cho cột Số tiết là 30.

Bài tập ứng dụng

- Tạo CSDL QLSINHVIEN như sau (các field được gạch dưới và in đậm là khóa chính của bảng):

SINHVIEN – Danh sách sinh viên

Field Name	Field Type	Field Size	Description
<u>MASV</u>	char	3	Mã SV
HOSV	nvarchar	35	Họ SV
TENSV	nvarchar	15	Tên SV
PHAI	bit		Phái (1: Nam, 0: Nữ)
NGAYSINH	datetime		Ngày sinh
NOISINH	nvarchar	15	Nơi sinh
MAKH	char	2	Mã khoa
HOCBONG	int	≥ 0	Học bổng

- Mã khoa là khóa ngoại tham chiếu đến bảng KHOA.
- Học bổng phải ≥ 0 .
- Giá trị mặc định cho cột Học bổng là 0.
- Giá trị mặc định cho cột Phái là 1 (nam)

Bài tập ứng dụng

- Tạo CSDL QLSINHVIEN như sau (các field được gạch dưới và in đậm là khóa chính của bảng):

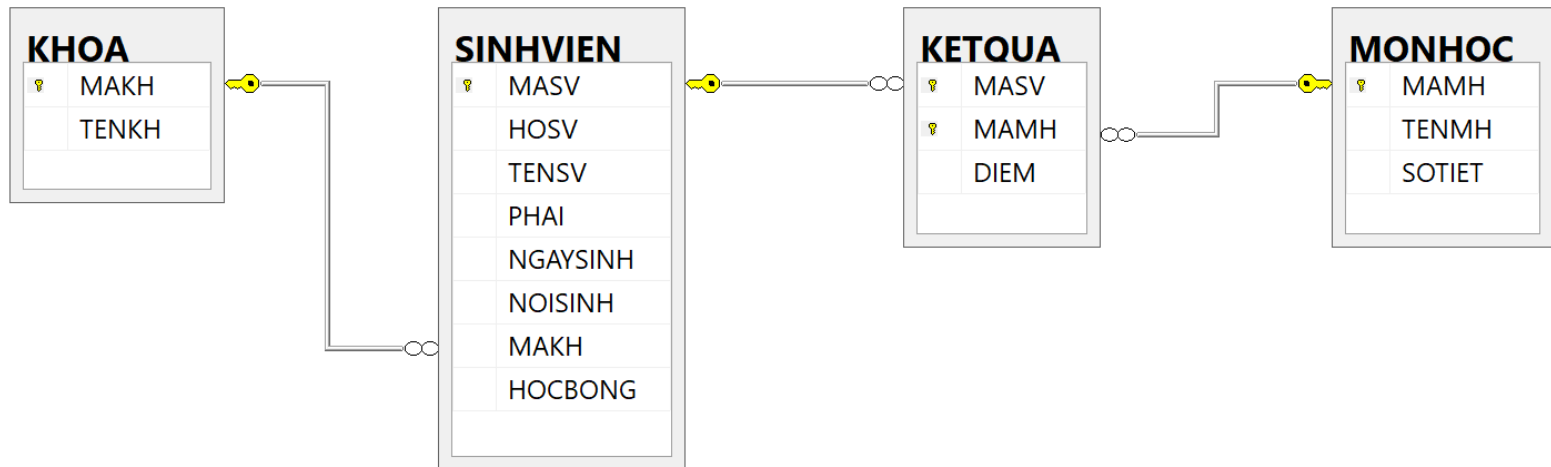
KETQUA – Kết quả học tập

Field Name	Field Type	Field Size	Description
<u>MASV</u>	char	3	Mã SV
<u>MAMH</u>	char	2	Mã môn học
DIEM	real	>=0 và <=10	Điểm

- Mã SV là khóa ngoại tham chiếu đến bảng SINHVIEN.
- Mã môn học là khóa ngoại tham chiếu đến bảng MONHOC.
- Điểm phải từ 0 đến 10.

Bài tập ứng dụng

- Tạo CSDL QLSINHVIEN như sau (các field được gạch dưới và in đậm là khóa chính của bảng):
 - Quan hệ giữa các bảng:



Bài tập ứng dụng

■ Dữ liệu mẫu:

SINHVIEN							
MASV	HOSV	TENS	PHAI	NGAYSINH	NOISINH	MAKH	HOCBONG
A01	Nguyễn Thu	Hải	0	23/02/1980	TP.HCM	AV	100,000
A02	Trần Văn	Chính	1	24/12/1982	TP.HCM	TH	100,000
A03	Lê Thu Bạch	Yến	0	21/02/1982	Hà Nội	AV	140,000
A04	Trần Anh	Tuấn	1	08/12/1984	Long An	LS	80,000
A05	Trần Thanh	Triều	1	01/02/1980	Hà Nội	VL	80,000
B01	Trần Thanh	Mai	0	20/12/1981	Bến Tre	TH	200,000
B02	Trần Thị Thu	Thủy	0	13/02/1982	TP.HCM	TH	30,000
B03	Trần Thị	Thanh	0	31/12/1982	TP.HCM	TH	50,000

KETQUA		
MASV	MAMH	DIEM
A01	01	10
A01	02	4
A01	05	9
A01	06	3
A02	01	5
A03	02	5
A03	04	10
A03	06	1
A04	02	4
A04	04	6
B01	01	0
B01	04	8
B02	03	6
B02	04	8
B03	02	10
B03	03	9

MONHOC		
MAMH	TENMH	SOTIET
01	Nhập môn máy tính	30
02	Trí tuệ nhân tạo	45
03	Truyền tin	45
04	Đồ họa	50
05	Văn phạm	40
06	Đàm thoại	30
07	Vật lý nguyên tử	30

KHOA	
MAKH	TENKH
AV	Anh Văn
LS	Lịch sử
TH	Tin học
TR	Triết
VL	Vật lý
SH	Sinh học

Câu hỏi trắc nghiệm

1. Chọn phát biểu ĐÚNG về việc tạo CSDL:

- a. Là bước thiết kế CSDL ở mức vật lý.
- b. Từ mô hình dữ liệu quan hệ, xây dựng CSDL quan hệ trong 1 Hệ quản trị CSDL cụ thể.
- c. Cần phải bổ sung các ràng buộc và đảm bảo các yêu cầu về toàn vẹn dữ liệu.
- d. **Tất cả đều đúng.**

Câu hỏi trắc nghiệm

2. Chọn lệnh đúng để tạo CSDL có tên là QLSV:

- a. create table QLSV
- b. delete table QLSV
- c. **create database QLSV**
- d. drop database QLSV

Câu hỏi trắc nghiệm

3. Cho bảng PHONGBAN với cấu trúc như sau:

Field name	Field Type	Field size
<u>MAPHONG</u>	char	2
TENPHONG	nvarchar	50

Trong đó, MAPHONG là khóa chính, TENPHONG phải duy nhất
Câu lệnh tạo bảng PHONGBAN dưới đây là đúng hay sai?

```
create table PHONGBAN (  
    MAPHONG char(2) primary key,  
    TENPHONG nvarchar(50) unique  
)
```

- a. Đúng
- b. Sai

Câu hỏi trắc nghiệm

4. Cho bảng PHONGBAN với cấu trúc như sau:

Field name	Field Type	Field size
<u>MAPHONG</u>	char	2
TENPHONG	nvarchar	50

Trong đó, MAPHONG là khóa chính, TENPHONG phải duy nhất
Dữ liệu mẫu:

MAPHONG	TENPHONG
NS	Phòng Nhân sự
KT	Phòng Kế toán
KD	Phòng Kinh doanh

Hãy chọn lệnh ĐÚNG để nhập liệu phòng ban mới:

- a. insert into PHONGBAN values ('KT', 'Phòng Kỹ thuật')
- b. insert into PHONGBAN values ('KT', N'Phòng Kỹ thuật')
- c. insert into PHONGBAN values ('KK', 'Phòng Kỹ thuật')
- d. **insert into PHONGBAN values ('KK', N'Phòng Kỹ thuật')**

Câu hỏi trắc nghiệm

5. Có các loại constraint (ràng buộc dữ liệu) nào?

- a. primary key constraint, foreign key constraint, unique constraint và check constraint.
- b. primary key constraint, foreign key constraint, unique constraint, check constraint và default constraint.
- c. primary key constraint, foreign key constraint và unique constraint.
- d. primary key constraint và foreign key constraint.