



TÀI LIỆU HƯỚNG DẪN THỰC HÀNH

MÔN CƠ SỞ DỮ LIỆU – IT004

BUỔI THỰC HÀNH 02

Hướng dẫn thực hành

Lê Võ Đình Kha – 18520872@gm.uit.edu.vn

Lê Thị Trà My – 19521861@gm.uit.edu.vn

GIỚI THIỆU NỘI DUNG BUỔI THỰC HÀNH SỐ 2

NỘI DUNG



Phần 1: Ôn tập lại buổi Thực hành số 1.

Phần 2: Các lệnh về Database.

Phần 3: Các lệnh về Table (Create, Drop, Alter).

Phần 4: Các lệnh tạo ràng buộc toàn vẹn.

Phần 5: Các lệnh thao tác dữ liệu (Insert, Update, Delete).

ÔN TẬP LẠI BUỔI THỰC HÀNH SỐ 1

- Phần mềm **Microsoft SQL Server**.
- Các kiểu dữ liệu trong ngôn ngữ SQL

ÔN TẬP LẠI BUỔI THỰC HÀNH SỐ 1

Trong Microsoft SQL mỗi cột (column), biến cục bộ (local variable) biểu thức (expression) và tham số (parameter) đều có một kiểu dữ liệu liên quan. Sau đây là 6 kiểu dữ liệu trong SQL.

1. Kiểu dữ liệu **Exact Numeric (Số chính xác, không sai số)** trong SQL
2. Kiểu dữ liệu **Approximate Numeric** trong SQL
3. Kiểu dữ liệu **Date và Time** trong SQL
4. Kiểu dữ liệu **Character String (Kiểu chuỗi)** trong SQL
5. Kiểu dữ liệu **Unicode Character String (Kiểu chuỗi có chứa Unicode)** trong SQL
6. Kiểu dữ liệu **Binary** trong SQL
7. Các kiểu dữ liệu khác

Tài liệu tham khảo về các kiểu dữ liệu:

<https://learn.microsoft.com/vi-vn/sql/t-sql/data-types/data-types-transact-sql?view=sql-server-ver15>

CÁC LỆNH VỀ DATABASE

- **Tạo Database:** Để tạo một **Database** mới ta sử dụng cú pháp:

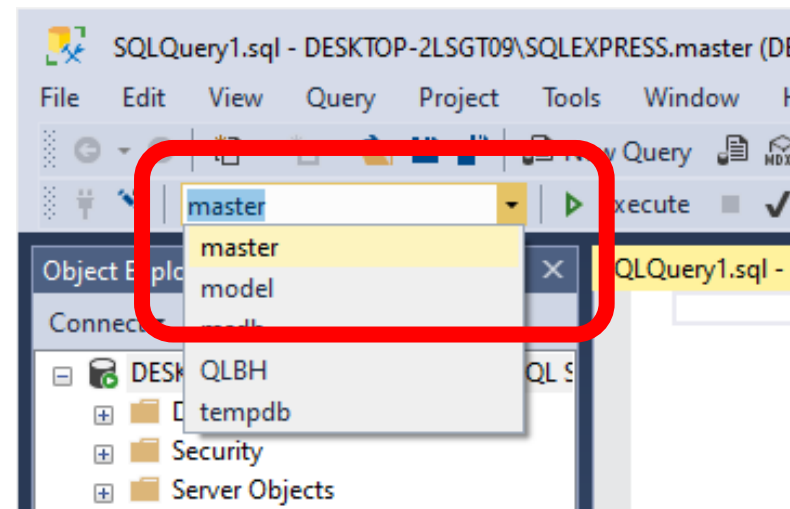
CREATE DATABASE <Tên Database>

Ví dụ: CRETAE DATABASE **QuanLyGiaoVu**

- **Sử dụng Database:** Sau khi tạo mới một Database ta cần dùng lệnh **USE** để chỉ định Database sẽ thao tác trên đó. Cú pháp như sau:

USE <Tên Database>

Ví dụ: USE **QuanLyGiaoVu**



- Chú ý "Available Database" có đúng Database cần thao tác chưa

CÁC LỆNH VỀ DATABASE

- **Xóa Database:** Để xóa một **Database** ta sử dụng cú pháp:

DROP DATABASE *<Tên Database>*

Ví dụ: DROP DATABASE **QuanLyGiaoVu**

- **Xóa nhiều Database:** Để xóa nhiều database cùng một lúc:

DROP DATABASE *<Tên Database 1>, ..., <Tên Database N>*

Ví dụ: DROP DATABASE **QuanLyGiaoVu, QuanLyCaSi, QuanLyBanHang**

- *Chú ý những database đang được sử dụng (Available Database) sẽ không xóa được.*

CÁC LỆNH VỀ TABLE

- SQL sử dụng các thuật ngữ tương đương với thuật ngữ về cơ sở dữ liệu như sau:

SQL	Cơ sở dữ liệu
Bảng	Quan hệ
Cột	Thuộc tính
Dòng dữ liệu	Bộ giá trị

- Để định nghĩa một bảng trong SQL ta cần: tên bảng, các cột, kiểu dữ liệu của cột và các ràng buộc toàn vẹn (RBTV – Integrity constraint) trên nó.

CÁC LỆNH VỀ TABLE

- Để tạo **Table**, ta sử dụng cú pháp sau:

CREATE TABLE *<Tên Table>*

(

<Tên column 1> <Kiểu dữ liệu> (integrity_constraint),

<Tên column 2> <Kiểu dữ liệu> (integrity_constraint),

)

CÁC LỆNH VỀ TABLE

- Ví dụ:

CREATE TABLE KHACHHANG

(

MaKH char(4) PRIMARY KEY,

HoTen varchar(40) NOT FULL,

DiaChi varchar(50) NOT FULL,

NgSinh smalldatetime,

DoanhSo money,

)

CÁC LỆNH VỀ TABLE

- Để xóa **Table**, ta sử dụng cú pháp sau:

DROP TABLE <Tên Table>

Ví dụ: DROP TABLE **KHACHHANG**

Lưu ý: Không thể **drop** được **table** khi nó đang có liên kết khóa ngoại.
Muốn xóa **table** đó trước hết phải xóa khóa ngoại đi.

Mở rộng các lệnh DROP khác:

- Để xóa một bảng View ta sử dụng cú pháp

DROP VIEW <Tên View>

- Để xóa một Procedure ta sử dụng cú pháp

DROP (PROCEDURE | PROC) procedure_name

- **Ngoài ra** còn các lệnh DROP khác như là 'DROP INDEX', 'DROP ROLE', 'DROP SCHEMA', 'DROP RULE'...

CÁC LỆNH VỀ TABLE

- **SQL** hỗ trợ **ALTER TABLE** cho phép chúng ta thêm, xóa hay sửa các thành phần của table.
- Để sửa **Table**, ta sử dụng cú pháp sau:

Thêm cột

ALTER TABLE *<Tên Table>* **ADD** *<Tên cột>* *<Kiểu dữ liệu>*

Xóa cột

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

Sửa cột (Thay đổi kiểu dữ liệu)

ALTER TABLE *<Tên Table>* **ALTER COLUMN** *<Tên cột>* *<Kiểu dữ liệu mới>*

CÁC LỆNH VỀ TABLE

- Để thực hiện đổi tên (Rename) một bảng ta sử dụng Stored Procedure **sp_name**:

Sửa tên bảng

EXEC sp_rename <Tên Table>, <Tên Table mới>

Sửa tên cột

EXEC sp_rename <Tên Table.Tên Column>, <Tên Column mới>, <COLUMN>

- Các lệnh **sp_help**, **sp_tables**, **sp_columns**, **sp_pkeys**, **sp_fkeys**, **sp_helpconstraint**, **sp_helpsql**

- Lưu ý: Sinh viên có thể tham khảo thêm các lệnh sp_ khác tại: [TẠI ĐÂY](#)

RÀNG BUỘC TOÀN VỆN

- Việc ràng buộc toàn vẹn trong SQL Server được chia làm 2 loại chính
 - ❖ **Loại đơn giản:** Sử dụng CONSTRAINT để mô tả
 - ❖ **Loại phức tạp:** Sử dụng TRIGGER để thực hiện

RÀNG BUỘC TOÀN VỆN

- Một số ràng buộc thường sử dụng:
 - ✓ **NOT NULL:** Dữ liệu không được để trống.
 - ✓ **UNIQUE:** Dữ liệu không được trùng lặp.
 - ✓ **PRIMARY KEY:** Ràng buộc khóa chính (Not null + Unique)
 - ✓ **FOREIGN KEY:** Ràng buộc khóa ngoại.
 - ✓ **CHECK:** Kiểm tra dữ liệu nhập vào phải theo định dạng.
 - ✓ **DEFAULT:** Tạo giá trị mặc định.

- Các nguyên tắc chung

- ❖ Một **constraint** luôn gắn với một bảng.
- ❖ Nếu không đặt tên thì hệ thống sẽ tự động phát sinh tên cho **constraint**.
- ❖ Có thể tạo **constraint**:
 - o Cùng với thời điểm tạo bảng.
 - o Sau khi đã tạo bảng xong (dùng câu lệnh Alter).
- ❖ Có thể khai báo **constraint** ở mức cột hoặc mức bảng.
- ❖ Có thể xem các **constraint** hiện có trong database.

RÀNG BUỘC TOÀN VỆN

Cú pháp chi tiết từng loại constraint

(CONSTRAINT ten_constraint)

PRIMARY KEY (danh_sach_cot_khoa_chinh)

- Hoặc UNIQUE (danh_sach_cot)

- Hoặc

FOREIGN KEY (danh_sach_cot_khoangoai)

REFERENCES bang_tham_chieu (ds_cot_tham_chieu)

- Hoặc CHECK (bieu_thuc_luan_ly)

- Hoặc DEFAULT gia_tri_mac_dinh FOR Ten_cot

RÀNG BUỘC TOÀN VỆN

Cách 1: Cách viết Constraint: Định nghĩa ràng buộc ngay sau kiểu dữ liệu

```
CREATE TABLE KHACHHANG
```

```
(
```

```
    MAKH char(4) PRIMARY KEY,
```

```
    HOTEN varchar(40) NOT NULL,
```

```
    SDT varchar(20) UNIQUE,
```

```
    DOANHISO money DEFAULT 0,
```

```
)
```

- **Lưu ý:** Nên dùng với Primary Key, Not Null, Unique, Default.

RÀNG BUỘC TOÀN VỆN

Cách 2: Cách viết Constraint: Định nghĩa ràng buộc ngay tại dòng cuối cùng của lệnh tạo bảng.

```
CREATE TABLE CTHD
(  
    SOHD int,  
    MASP char(4),  
    SL int,  
    constraint pk_cthd primary key(SOHD,MASP),  
)
```

- Lưu ý: Nên dùng với Primary Key (Khóa chính có nhiều thuộc tính).
- Có thể viết tắt: Primary key(SOHD,MASP)

RÀNG BUỘC TOÀN VỆN

Cách 3: Cách viết Constraint: Thêm ràng buộc sau khi tạo bảng.

- Sử dụng lệnh sửa bảng (ALTER TABLE) và thêm (ADD) ràng buộc.

Ví dụ:

```
ALTER TABLE HOADON ADD FOREIGN KEY(MAKH) REFERENCES  
KHACHHANG(MAKH)
```

```
ALTER TABLE HOADON ADD CONSTRAINT fk02_HD FOREIGN KEY(MANV)  
REFERENCES NHANVIEN(MANV)
```

- Lưu ý:** Nên dùng với Foreign Key, Check
- Khóa ngoại ở bảng nào thì sửa bảng đó

RÀNG BUỘC TOÀN VỆN

Ràng buộc CHECK:

- Dùng kiểm tra dữ liệu nhập vào thỏa điều kiện nhất định.
- Áp dụng cho : Ràng buộc Miền giá trị, Ràng buộc Liên thuộc tính (Xem lý thuyết Chương 5).

Cách viết:

ALTER TABLE <Tên bảng> ADD CHECK (Điều kiện)

Ví dụ:

```
ALTER TABLE KHACHHANG ADD CHECK (DOANHSON >= 0)
ALTER TABLE SANPHAM ADD CONSTRAINT CHK_DVT
CHECK(( DVT = 'CAY' ) OR ( DVT = 'HOP' ) OR (DVT = 'CAI'))
```

RÀNG BUỘC TOÀN VỆN

Toán tử **LIKE**:

- Dùng kiểm tra dữ liệu kiểu chuỗi theo một định dạng nhất định.
- Dấu gạch dưới '_' : thay thế cho 1 ký tự.
- Dấu phần trăm '%': thay thế cho 0, 1, hoặc n ký tự.

• Ví dụ:

LIKE 'a__'	Chuỗi bắt đầu là a , sau a có 2 ký tự.
LIKE 'a%'	Chuỗi bắt đầu là a , sau a có thể có 0 hoặc n ký tự.
LIKE '_a___'	Ký tự thứ 2 là a, sau a có 3 ký tự

ALTER TABLE **SINHVIENT** ADD CHECK (MSSV LIKE '__52__')

CÁC LỆNH THAO TÁC DỮ LIỆU

- Các lệnh thao tác dữ liệu Data Manipulation Language (DML):

Thêm mới dữ liệu.

INSERT INTO <Tên bảng> **VALUES**('Giá trị 1', 'Giá trị 2', Giá trị 3)

Ví dụ:

INSERT INTO NHANVIEN VALUES

('NV01','Nguyen Nhu Nhat', '927345678', '13/04/2006')

INSERT INTO NHANVIEN (MANV, HOTEN, SODT, NGVL)

VALUES ('NV01', 'Nguyen Nhu Nhat', '927345678','13/04/2006')

CÁC LỆNH THAO TÁC DỮ LIỆU

- Các lệnh thao tác dữ liệu Data Manipulation Language (DML):

Xóa dữ liệu.

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

Ví dụ:

DELETE FROM NHANVIEN **WHERE** MANV='NV01'

DELETE FROM SINHVIEN **WHERE** MSSV LIKE '1852____'

CÁC LỆNH THAO TÁC DỮ LIỆU

- Các lệnh thao tác dữ liệu Data Manipulation Language (DML):

Cập nhật, chỉnh sửa dữ liệu.

UPDATE <Tên bảng> SET <Tên cột 1> = <Giá trị mới>,
<Tên cột 2> = <Giá trị mới> WHERE <Điều kiện>

Ví dụ:

UPDATE NHANVIEN SET HOTEN='Nguyen Van Tien', SODT='083568711'
WHERE MANV='NV02'

CÁC LỆNH THAO TÁC DỮ LIỆU

- Các lệnh thao tác dữ liệu Data Manipulation Language (DML):

Sao chép bảng dữ liệu

SELECT * INTO <Tên bảng mới> FROM <Tên bảng cũ>

Ví dụ: Sao chép bảng KHACHHANG thành KHACHHANG_NEW

SELECT * INTO KHACHHANG_NEW FROM KHACHHANG

BÀI TẬP

- Sinh viên hoàn thành:
 - Phần I bài tập **QuanLyBanHang** từ câu 1 đến câu 10.
 - Phần II bài tập **QuanLyBanHang** từ câu 2 đến câu 5.
 - Phần III bài tập **QuanLyBanHang** từ câu 1 đến câu 11.
 - Phần I bài tập **QuanLyGiaoVu** từ câu 3 đến câu 8.
 - Phần I bài tập **QuanLyGiaoVu** từ câu 11 đến câu 14.
 - Phần III bài tập **QuanLyGiaoVu** từ câu 1 đến câu 5.

HỎI - ĐÁP

