

TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT
KHOA KỸ THUẬT CÔNG NGHỆ

HỆ QUẢN TRỊ CSDL

Chương 3 TABLE & VIEW

Phone: 0274. 3834930

Website: www.et.tdmu.edu.vn



Nội dung

❖ Các kiểu dữ liệu trong SQL

❖ Câu lệnh định nghĩa dữ liệu

- Tạo bảng
- Câu lệnh cập nhật dữ liệu
- Câu lệnh thay đổi cấu trúc bảng
- Xóa bảng

❖ VIEW



Các kiểu dữ liệu trong SQL

- ❖ Kiểu dữ liệu của một thuộc tính chỉ định kiểu và dung lượng có thể lưu trữ của một đối tượng
- ❖ SQL Server hỗ trợ một số kiểu dữ liệu được cài đặt sẵn như sau:



Các kiểu dữ liệu trong SQL (tt)

Exact numbers	int	- Sử dụng 4 byte trong bộ nhớ máy tính. - Thường được dùng để lưu trữ giá trị số nguyên
	smallint	- Sử dụng 2 byte trong bộ nhớ máy tính. - Lưu trữ các số nguyên từ -32768 đến 32767.
	Tinyint	- Chiếm 1 byte trong bộ nhớ - Có giá trị từ 0 đến 255
	Bigint	- Sử dụng 8 bytes trong bộ nhớ máy tính - Lưu trữ các số nguyên từ -2^{63} (-223372036854775807) đến $2^{63}-1$
	numeric(p,d)	- Kiểu số với độ chính xác cố định - Biểu diễn số gồm p chữ số và 1 dấu chấm, có d chữ số bên phải dấu chấm thập phân
	money	- Sử dụng 8 byte trong bộ nhớ máy tính. - Biểu diễn giá trị dữ liệu tiền tệ từ $(-2^{63}/10000)$ đến $(2^{63}-1)$.

 Các kiểu dữ liệu trong SQL (tt)		
Approximate numerics	float(n)	<ul style="list-style-type: none"> - Sử dụng 8 byte trong bộ nhớ máy tính. - Biểu diễn các số dấu chấm động từ -1.79E+308 đến 1.79E+308
	real	<ul style="list-style-type: none"> - Sử dụng 4 byte trong bộ nhớ máy tính. - Biểu diễn các số dấu chấm động có độ chính xác từ -3.4E+38 đến 3.4E+38
Date and time	datetime	<ul style="list-style-type: none"> - Biểu diễn ngày và giờ - Được lưu trữ như là 2 số integer, chiếm 2 bytes, chính xác đến phần trăm của giây.
	Small datetime	<ul style="list-style-type: none"> - Biểu diễn ngày và giờ - Chính xác đến phút

Bài giảng Hệ quản trị CSDL

5

 Các kiểu dữ liệu trong SQL (tt)		
Character String	char	- Lưu trữ dữ liệu ký tự, được cố định kích thước và không hỗ trợ Unicode
	varchar	- Lưu trữ dữ liệu ký tự, độ dài có thể thay đổi và không hỗ trợ Unicode
Unicode Types	text	- Lưu trữ dữ liệu chuỗi, độ dài lớn và không hỗ trợ Unicode
	ntext	- Lưu trữ dữ liệu ký tự, độ dài lớn và có hỗ trợ Unicode
	nchar	- Lưu trữ dữ liệu ký tự, được cố định kích thước và có hỗ trợ Unicode
	nvarchar	- Lưu trữ dữ liệu ký tự, độ dài có thể thay đổi và có hỗ trợ Unicode

Bài giảng Hệ quản trị CSDL

6



Các kiểu dữ liệu trong SQL (tt)

❖ Kiểu Date/ Time

- Date: Kiểu ngày
- Time: Kiểu giờ



Nội dung

❖ Các kiểu dữ liệu trong SQL

❖ Câu lệnh định nghĩa dữ liệu

- **Tạo bảng**
- Câu lệnh cập nhật dữ liệu
- Câu lệnh thay đổi cấu trúc bảng
- Xóa bảng



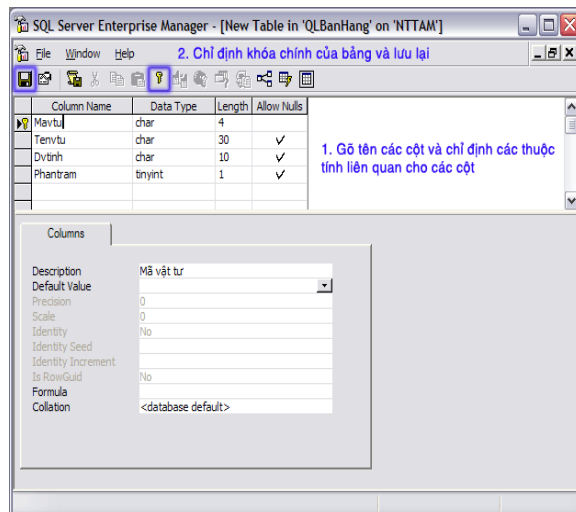
Tạo bảng

- ❖ Bảng dùng để lưu trữ thông tin của một đối tượng trong thực tế
 - Gồm có dòng và cột
 - Bảng trong CSDL thường có khóa chính
 - Các bảng thường có mối quan hệ với nhau bằng các quan hệ
- ❖ Trong SQL Server, bảng có thể có các ràng buộc, Trigger



Tạo bảng – các thuộc tính của bảng

- Tên bảng
- Tên cột
- Kiểu dữ liệu
 - Độ dài dữ liệu
 - Số ký số lưu trữ
 - Số số lẻ lưu trữ
- Thuộc tính trên cột
 - Allow null
 - Identity Default
 - Value





Cú pháp tạo bảng

```
CREATE TABLE <tên bảng>
(
  <tên cột 1> <kiểu dữ liệu>[CONSTRAINT<tên 1>],
  <tên cột 2> <kiểu dữ liệu>[CONSTRAINT<tên 2>],
  ...
  <tên cột n><kiểu dữ liệu>[CONSTRAINT <tên n>]
  [, CONSTRAINT <tên 1>] [, CONSTRAINT <tên 2>]
  ...
  [,CONSTRAINT <tên n>]
)
```



Cú pháp tạo bảng (tt)

- Trong đó, cú pháp khai báo của các ràng buộc toàn vẹn như sau:
 - [CONSTRAINT <tên RBTV>] NULL|NOT NULL|UNIQUE[(<tên cột i>,<tên cột j>...)]|PRIMARY KEY[(<tên cột i>,<tên cột j>...)]|FOREIGN KEY [(<tên cột i>,<tên cột j>...)] REFERENCES <tên bảng>(<tên cột i>,<tên cột j>...)|CHECK (<điều kiện>)**



Cú pháp tạo bảng (tt)

- ❖ Ví dụ: Tạo bảng với lược đồ quan hệ sau:
- ❖ **HANGHOA (MaHG, TenHG, DVT)**
- ❖ Mã hàng là khóa chính, tên hàng và đơn vị tính.
Tất cả các thuộc tính không được rỗng

```
CREATE TABLE Hanghoa
(
  MaHG varchar(10) Not Null Primary Key,
  TenHG nvarchar(50) Not Null,
  DVT varchar(5) Not Null
)
```



Các loại ràng buộc trong bảng

- ❖ SQL Server cho phép tạo ràng buộc giá trị trong một cột xác định để dữ liệu không thích hợp sẽ không được thêm vào.
- ❖ Kiểu dữ liệu của cột cũng được phép ràng buộc giá trị trong cột, nhưng nó không được xem là ràng buộc.



Các loại ràng buộc (tt)

- ❖ **NULL/ NOT NULL**: Khi ràng buộc không rỗng được chỉ ra, cột bắt buộc phải có giá trị khi bạn thêm dữ liệu vào bảng.
- ❖ **DEFAULT**: Ràng buộc mặc định được tạo ra cho cột
- ❖ **CHECK**: Ràng buộc kiểm tra chỉ ra miền giá trị được phép.
- ❖ **UNIQUE**: Ràng buộc duy nhất chỉ ra rằng giá trị nhập vào một cột phải duy nhất.
- ❖ **PRIMARY KEY**: Ràng buộc khóa chính dùng để xác định duy nhất một dòng dữ liệu.
- ❖ **FOREIGN KEY**: Ràng buộc khóa ngoại dùng để tham chiếu đến một nhận dạng duy nhất trong một bảng khác trong CSDL.

Bài giảng Hệ quản trị CSDL

15



Ràng buộc NULL/ NOT NULL

- ❖ Đặc trưng về khả năng Null của một thuộc tính quyết định các hàng trong bảng có thể chứa giá trị null cho cột đó.
- ❖ Khả năng Null của một cột có thể được định nghĩa khi tạo một bảng
 - Từ khóa NULL được sử dụng để chỉ ra rằng giá trị NULL là được phép trong một cột
 - Từ khóa NOT NULL được sử dụng để chỉ ra rằng giá trị NULL là không được phép trong một cột

Bài giảng Hệ quản trị CSDL

16



Ràng buộc NULL/ NOT NULL (tt)

- Ví dụ:

```
CREATE TABLE NhanVien  
(  
    MaNV char(5) Not Null,  
    TenNV nvarchar (50) ,  
    MaNQL char(5) Null  
)
```



Ràng buộc Default

- Ràng buộc default có thể được tạo ra tại thời điểm tạo bảng hoặc thêm sau khi bảng được tạo.
- Với một cột, chỉ có thể tạo được một giá trị default.
- Giá trị default có thể là một hằng, một hàm hệ thống, một biến toàn cục, hoặc một hàm do người dùng định nghĩa.



Ràng buộc Default (tt)

- Định nghĩa default trong khi tạo bảng:

```
CREATE TABLE <tên bảng> (<tên cột> <KDL>
[NULL|NOT NULL]
[CONSTRAINT <tên ràng buộc>] DEFAULT
<biểu thức>
```

Ví dụ: Tạo bảng StoreProduct (ProductID, Name, Price).
Mặc định giá trị 100 cho thuộc tính Price.

```
CREATE TABLE StoreProduct
(
    ProductID int NOT NULL
    Name varchar(40) NOTT NULL
    Price money NOT NULL DEFAULT (100)
)
```

Bài giảng Hệ quản trị CSDL

19



Ràng buộc Default (tt)

- Định nghĩa default đối với một bảng đã tồn tại:

```
ALTER TABLE <tên bảng>
ADD [CONSTRAINT <tên ràng buộc>] DEFAULT
<biểu thức> FOR <tên cột> ]
```

Ví dụ: Trong bảng Hàng Hóa, thêm ràng buộc
Default cho Cột số lượng là 0.

**Alter Table HangHoa
Add Constraint HH_SL_DF Default (0) for
SoLuong**

Bài giảng Hệ quản trị CSDL

20



Ràng buộc dạng IDENTITY

- ❖ Thuộc tính IDENTITY của SQL Server được sử dụng để tạo ra các cột nhận dạng, chúng chứa các giá trị tự động phát sinh tuần tự để nhận dạng duy nhất mỗi hàng trong một bảng
- ❖ Một thuộc tính nhận dạng có 2 thành phần:
 - Giá trị khởi đầu
 - Giá trị tăng



Ràng buộc dạng IDENTITY (tt)

- ❖ Cú pháp:


```
CREATE TABLE <table name>  
(Column_name data_type [identity [(seed_value,  
increment_value)]] NOT NULL)
```
- ❖ Trong đó:
 - seed_value: giá trị khởi đầu
 - increment_value: giá trị tăng
- Ví dụ:


```
CREATE TABLE ContactPhone (  
Person_ID int IDENTITY(500,1) NOT NULL,  
MobileNumber bigint NOT NULL  
)
```



Ràng buộc Primary Key (Khoá chính)

- Ràng buộc PRIMARY KEY được sử dụng để tạo một khóa chính và đảm bảo toàn vẹn thực thể bảng.
- Cú pháp để thêm một khóa chính trong khi tạo bảng:
- **CREATE TABLE <tên bảng> (**
 <tên cột> <kiểu dữ liệu> PRIMARY KEY [ds cột]
)
- **CREATE TABLE <tên bảng>**
 <tên cột> <kdl> [ds cột] CONSTRAINT <tên RB>
 PRIMARY KEY)



Ràng buộc Primary Key (tt)

Ví dụ: Tạo bảng PhongBan với khóa chính là MaPhong

```
CREATE TABLE PhongBan
( MaPhong char(5) PRIMARY KEY,
  TenPhong nvarchar(30) NOT NULL,
  TrPhong char(5),
  DiaDiem nvarchar(20)
)
```

Hoặc có thể viết:

```
CREATE TABLE PhongBan
( MaPhong char(5) CONSTRAINT pk_pb PRIMARY
  KEY,
```

...



Ràng buộc duy nhất

- Ràng buộc khóa duy nhất:
 - được sử dụng để bảo đảm rằng chỉ các giá trị duy nhất được nhập vào trong cột hoặc một tập hợp các cột (→ cho phép nhà phát triển chắc chắn rằng không có các giá trị trùng lặp được nhập vào)
 - Được sử dụng trong câu lệnh create table để định nghĩa khóa phụ cho bảng
 - Ràng buộc UNIQUE cho phép null



Ràng buộc duy nhất (tt)

- Cú pháp:
 - `CREATE TABLE <tên bảng> ([ds cột,] <tên cột> <kdl> [CONSTRAINT <tên RB>] UNIQUE [,ds cột])`
- Ví dụ:


```
CREATE TABLE KháchHang (
    makh int PRIMARY KEY,
    tenkh nvarchar(30),
    diachi nvarchar(30),
    sodt char(10) UNIQUE )
```



Ràng buộc khóa ngoại

- Một khóa ngoại trong một bảng là một cột, nó chỉ đến một khóa chính trong một bảng khác
- Ràng buộc khóa ngoại được sử dụng để đảm bảo toàn vẹn tham chiếu
- Cú pháp:
 - **CREATE TABLE <tên bảng 1>(**
 [ds cột,]
 <tên cột> <kdl> [CONSTRAINT <tên RB>] FOREIGN KEY REFERENCES
 <tên bảng 2>(cột làm khóa chính> [, ds cột])



Ràng buộc khóa ngoại (tt)

- Ví dụ:


```
CREATE TABLE Nhanvien(
    manv char(5) NOT NULL,
    tennv nvarchar(30),
    manqly char(5),
    phong int foreign key references
    PhongBan (maphong)
)
```
- Hoặc có thể viết như sau:



Ràng buộc khóa ngoại (tt)

- Ví dụ: (tt)
- Hoặc có thể viết như sau:


```
CREATE TABLE Nhanvien(
    manv      char(5)      NOT      NULL, tennv
    nvarchar(30),
    manqly char(5), phong int
    constraint fk_nv_pb foreign key(phong)
    references PhongBan(maphong)
)
```



Ràng buộc kiểm tra (check)

- **Ràng buộc check**
 - Được sử dụng để chỉ định điều kiện hợp lệ đối với dữ liệu
- Cú pháp:
 - CREATE TABLE <Tên bảng> (
 - [ds cột,] <tên cột> <kdl> [CONSTRAINT
 - <tên RB>] CHECK (<điều kiện>)
- Ví dụ:

```
Create table KetQua(
    masv int, mamh char(4),
    diem float,
    lanthi int check(lanthati<3)
```




Nội dung

- ❖ **Các kiểu dữ liệu trong SQL**
- ❖ **Câu lệnh định nghĩa dữ liệu**
 - **Tạo bảng**
 - **Câu lệnh cập nhật dữ liệu**
 - Câu lệnh thay đổi cấu trúc bảng
 - Xóa bảng
- ❖ **VIEW**

31

Bài giảng Hệ quản trị CSDL



Câu lệnh cập nhật dữ liệu

- Là câu lệnh dùng để cập nhật dữ liệu.
- Bao gồm các câu lệnh: INSERT, UPDATE, DELETE.
 - **Lệnh thêm dữ liệu vào bảng:**

```
INSERT INTO <Tên bảng> [(<DS các trường>)] VALUES (<DS các giá trị tương ứng>)
```

32

Bài giảng Hệ quản trị CSDL



Câu lệnh cập nhật dữ liệu (tt)

Vd: MON (MaMH, TenMH, DVHT)

Thêm vào bảng MON với mã môn TH345, tên môn là Hệ Quản trị CSDL, đơn vị học trình là 5

```
INSERT INTO MON (MaMH, TenMH, DVHT)
VALUES ('TH345', 'Hệ quản trị CSDL', 5)
```

Hoặc:

```
INSERT INTO MON VALUES ('TH345', 'Hệ quản
trị CSDL', 5)
```



Câu lệnh cập nhật dữ liệu (tt)

- Một số lưu ý:

- Thêm ký tự N trước chuỗi Unicode

- Ví dụ:

```
Insert Into NHANVIEN
Values ('NV01', N'Nguyễn Văn Trường', 'Nam')
```

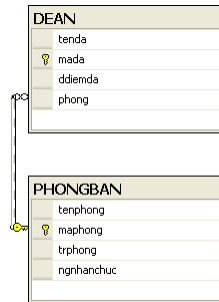
- Nếu thuộc tính được khai báo là NOT NULL thì bắt buộc phải có giá trị khi nhập một dòng vào bảng



Câu lệnh cập nhật dữ liệu (tt)

- Một số lưu ý (tt)

- Nhập dữ liệu khi đã có ràng buộc khóa ngoại



- Trường hợp 1:

- Cách 1:

- B1: Nhập PHONGBAN
- B2: Nhập DEAN

- Cách 2:

- B1: Nhập DEAN, nhập phong = null
- B2: Nhập PHONGBAN
- B3: Cập nhật DEAN

Bài giảng Hệ quản trị CSDL

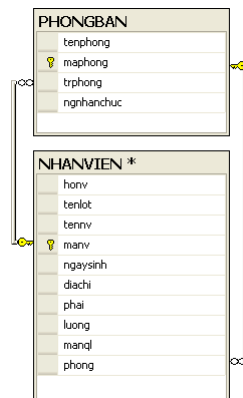
35



Câu lệnh cập nhật dữ liệu (tt)

- Một số lưu ý (tt)

- Nhập dữ liệu khi đã có ràng buộc khóa ngoại



- Trường hợp 2:

- Cách 1:

- B1: Nhập NHANVIEN, đặt phong = null
- B2: Nhập PHONGBAN
- B3: Cập nhật thuộc tính phong của NHANVIEN

- Cách 2:

- B1: Nhập PHONGBAN, đặt trphong = null
- B2: Nhập NHANVIEN
- B3: Cập nhật PHONGBAN

Bài giảng Hệ quản trị CSDL

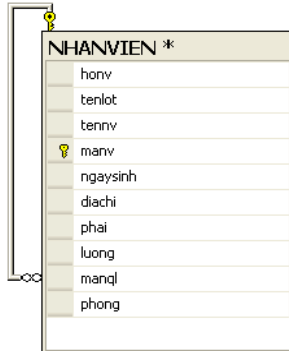
36



Câu lệnh cập nhật dữ liệu (tt)

- Một số lưu ý (tt)

- Nhập dữ liệu khi đã có ràng buộc khóa ngoại



- Trường hợp 3:

- Cách 1:

- Những nhân viên có manql là null thì nhập trước
- Sau đó nhập những nhân viên mà đã nhập thông tin người quản lý nhân viên đó.

- Cách 2:

- B1: Nhập NHANVIEN đặt manql = null
- B2: Cập nhật manql của NHANVIEN

Bài giảng Hệ quản trị CSDL

37



Câu lệnh cập nhật dữ liệu (tt)

- Lệnh cập nhật dữ liệu trong bảng:

```
UPDATE <Tên bảng> SET <Tên trường1> =
    <GT1>, ..., <Tên Trường N> =
    <GTN> [WHERE <điều kiện>]
```

- Vd: Cập nhật số DVHT cho môn học 'TH345' tăng 1

```
UPDATE MON SET DVHT = DVHT + 1
WHERE MaMH = 'TH345'
```

Bài giảng Hệ quản trị CSDL

38



Câu lệnh cập nhật dữ liệu (tt)

- **Lệnh xóa dữ liệu khỏi bảng:**

DELETE FROM <Tên bảng> [WHERE <điều kiện chọn>]

- Vd: Xóa các môn học có số DVHT nhỏ hơn 2

DELETE FROM MON WHERE DVHT < 2

- Lưu ý: Câu lệnh delete chỉ xóa dữ liệu của bảng, cấu trúc bảng tạo bởi câu lệnh create table vẫn được giữ nguyên.



Nội dung

❖ Các kiểu dữ liệu trong SQL

❖ Câu lệnh định nghĩa dữ liệu

- Tạo bảng
- Câu lệnh cập nhật dữ liệu
- Câu lệnh thay đổi cấu trúc bảng
- Xóa bảng

❖ VIEW



Câu lệnh thay đổi cấu trúc bảng

- Thêm một cột vào bảng có sẵn

- Cú pháp

```
ALTER TABLE <tên bảng> ADD <tên cột 1>
<kd11>[,<tên cột 2> <kd12>, ...]
```

- Ví dụ: Thêm cột ngaysinh vào bảng NhanVien

```
ALTER TABLE NhanVien ADD ngaysinh datetime
```



Lệnh thay đổi cấu trúc bảng (tt)

- Chỉnh sửa cột trong bảng

- Cú pháp

```
ALTER TABLE <table_name>
ALTER COLUMN <tên cột 1> <kd11>
[,<tên cột 2> <kd12> ...]
```

- Ví dụ: Sửa cột tennv trong bảng NhanVien thành cột có kiểu dữ liệu là nvarchar(20)

```
ALTER TABLE NhanVien ALTER column tennv
nvarchar(20)
```



Lệnh thay đổi cấu trúc bảng (tt)

- Xóa cột trong bảng

- Cú pháp

```
ALTER TABLE <table_name> DROP COLUMN
<column_name1> [, <column_name2>
```

- Ví dụ: Xóa cột diadiem trong bảng PhongBan

```
ALTER TABLE PhongBan DROP COLUMN
diadiem
```



Lệnh thay đổi cấu trúc bảng (tt)

- Thêm RB khóa chính

- Cú pháp

```
ALTER TABLE <tên bảng> ADD CONSTRAINT
<tên RB> PRIMARY KEY (<DS các cột làm
khóa chính>)
```

- Ví dụ: Thêm khóa chính cho bảng KetQua

```
ALTER TABLE KetQua ADD CONSTRAINT pk_kq
PRIMARY KEY (masv, mamh)
```



Lệnh thay đổi cấu trúc bảng (tt)

- Thêm RB khóa ngoại

- Cú pháp

```
ALTER TABLE <tên bảng> ADD CONSTRAINT
<tên RB> FOREIGN KEY (<DS các cột làm
khóa ngoại>) REFERENCES <tên bảng tham
chiếu>(<ds cột>)
```

- Ví dụ: Thêm RB khóa ngoại cho bảng KetQua

```
ALTER TABLE KetQua ADD CONSTRAINT
fk_kq_sv FOREIGN KEY(masv) REFERENCES
SinhVien(masv)
```



Lệnh thay đổi cấu trúc bảng (tt)

- Thêm RB kiểm tra

- Cú pháp

```
ALTER TABLE <tên bảng> ADD CONSTRAINT
<tên RB> CHECK(<điều kiện>)
```

- Ví dụ: Thêm RB kiểm tra đảm bảo thuộc tính phái trong bảng NhanVien chỉ nhận giá trị “nam” hoặc “nữ”

```
ALTER TABLE NhanVien ADD CONSTRAINT
ck_phai CHECK(phai in ('nam', 'nu'))
```



Lệnh xóa ràng buộc

- **Cú pháp**

```
ALTER TABLE <tên_bảng> DROP CONSTRAINT
<tên_RB>
```

- **Ví dụ:**

- **Xóa RB kiểm tra trong bảng NhanVien**

```
ALTER TABLE NhanVien
DROP CONSTRAINT ck_phai
```

- **Xóa RB khóa ngoại trong bảng KetQua**

```
ALTER TABLE KetQua
DROP CONSTRAINT fk_kq_sv
```




Nội dung

- ❖ **Các kiểu dữ liệu trong SQL**

- ❖ **Câu lệnh định nghĩa dữ liệu**

- **Tạo bảng**
- **Câu lệnh cập nhật dữ liệu**
- **Câu lệnh thay đổi cấu trúc bảng**
- **Xóa bảng**


- ❖ **VIEW**



Xóa bảng

- Cú pháp:
`Drop table <tên bảng>`
- Ví dụ: Xóa bảng NhanVien
`Drop table NhanVien`

49



Xóa bảng (tt)

- Lưu ý khi xóa bảng có liên quan đến khóa ngoại:
 - Nếu không có tham chiếu vòng thì tiến hành xóa bảng chứa khóa ngoại trước sau đó xóa bảng còn lại, hoặc xóa khóa ngoại rồi sau đó tiến hành xóa các bảng.
 - Nếu có tham chiếu vòng thì xóa một khóa để làm mất tham chiếu vòng rồi tiến hành như trường hợp 1.
 - Có thể sử dụng câu lệnh sau làm vô hiệu hóa các RB trên bảng trước khi xóa bảng:

```
Alter table <tên bảng> nocheck
constraint all
```

50



Ví dụ

- Tạo csdl QLDT gồm các lược đồ sau:
 - **SINHVIEN** (MaSV, Hoten, Namsinh, QQ, Hocluoc)
 - **DETAI** (MaDT, TenDT, Chunhiem, Kinhphi)
 - **SV_DT** (MaSV, MaDT, NoiAD, KQ)

- Ta có các câu lệnh SQL sau:

```
Create database QLDT
```

```
Use QLDT
```

```
Create table SinhVien(
    masv int primary key, hoten nvarchar(30),
    namsinh datetime, qq nvarchar(20), hocluoc
    float)
```



Ví dụ (tt)

```
Create table DeTai(
    madt int primary key, tendt
    nvarchar(30),
    chunhiem nvarchar(30), kinhphi int)
```

```
Create table SV_DT(
    masv int foreign key references
    SinhVien(masv),
    madt int foreign key references
    DeTai(madt),
    noiAD nvarchar(20), kq float,
    constraint pk_svdt primary
    key(masv, madt))
```



Khung nhìn (view)

- ❖ Bảng là một quan hệ được tổ chức lưu trữ vật lý trong CSDL
- ❖ Khung nhìn cũng là một quan hệ
 - Không được lưu trữ vật lý (bảng ảo)
 - Không chứa dữ liệu
 - Được định nghĩa từ những bảng khác
 - Có thể truy vấn hay cập nhật thông qua khung nhìn



Khung nhìn (view)

- ❖ View là một bảng ảo – bảng logic được hình thành nên từ kết quả truy vấn của câu lệnh SELECT.
- ❖ Ưu điểm
 - Đơn giản hóa các truy vấn phức tạp
 - Hạn chế truy cập dữ liệu theo phân loại người sử dụng
 - Tăng tính bảo mật dữ liệu
 - Là bước trung gian giữa ứng dụng và CSDL



Khung nhìn (view)

❖ Tạo khung nhìn

```
CREATE VIEW <tên khung nhìn> AS <câu truy vấn>
```

❖ Bảng ảo này có:

- Danh sách thuộc tính trùng với các thuộc tính trong mệnh đề SELECT
- Số dòng phụ thuộc vào điều kiện ở mệnh đề WHERE
- Dữ liệu được lấy từ các bảng ở mệnh đề FROM



Khung nhìn (tt)

- ❖ Câu lệnh dưới đây tạo khung nhìn có tên DSSV từ câu lệnh SELECT truy vấn dữ liệu từ hai bảng SINHVIEN và LOP

```
CREATE VIEW DSSV
AS
SELECT Masv, Hodem, Ten,
       DATEDIFF(YY,Ngaysinh,GETDATE()) AS Tuoai,
       Tenlop
FROM Sinhvien,Lop
WHERE Sinhvien.Malop=Lop.Malop
```



Khung nhìn (tt)

❖ Nếu thực hiện lệnh:

SELECT * FROM DSSV

Ta được kết quả:

MASV	HODEM	TEN	TUOI	TENLOP
0241010001	Ngô Thị Nhật	Anh	22	Toán K24
0241010002	Nguyễn Thị Ngọc	Anh	21	Toán K24
0241010003	Ngô Việt	Bắc	22	Toán K24
0241010004	Nguyễn Đình	Bình	22	Toán K24
0241010005	Hồ Đăng	Chiến	22	Toán K24
0241020001	Nguyễn Tuấn	Anh	25	Tin K24
0241020002	Trần Thị Kim	Anh	22	Tin K24
0241020003	Võ Đức	Ấn	22	Tin K24
0241020004	Nguyễn Công	Bình	25	Tin K24
0241020005	Nguyễn Thanh	Bình	22	Tin K24
...

Bài giảng Hệ quản trị CSDL

57



Khung nhìn (tt)

❖ Câu lệnh dưới đây tạo khung nhìn từ câu truy vấn tương tự như ví dụ trên nhưng có đặt tên cho các cột trong khung nhìn:

```
CREATE VIEW DSSV(Ma, Ho,Ten,Tuoi, Lop) AS
SELECT Masv, Hodem, Ten,
DATEDIFF(YY, Ngaysinh,GETDATE()), Tenlop
FROM Sinhvien, Lop
WHERE Sinhvien.Malop = Lop.Malop
```

Bài giảng Hệ quản trị CSDL

58



Khung nhìn (tt)

❖ Lúc này, nếu thực hiện lệnh:

SELECT * FROM DSSV

Ta được kết quả:

MÃ	HỌ	TÊN	TUỔI	LOP
0241010001	Ngô Thị Nhật	Anh	22	Toán K24
0241010002	Nguyễn Thị Ngọc	Anh	21	Toán K24
0241010003	Ngô Việt	Bắc	22	Toán K24
0241010004	Nguyễn Đình	Bình	22	Toán K24
0241010005	Hồ Đăng	Chiến	22	Toán K24
0241020001	Nguyễn Tuấn	Anh	25	Tin K24
0241020002	Trần Thị Kim	Anh	22	Tin K24
0241020003	Võ Đức	Ấn	22	Tin K24
0241020004	Nguyễn Công	Bình	25	Tin K24
0241020005	Nguyễn Thanh	Bình	22	Tin K24
...

Bài giảng Hệ quản trị CSDL

59



Khi nào thì dùng View?

- ❖ Khi muốn user chỉ có thể truy cập giới hạn một số lượng cột, hoặc số lượng dòng nhất định.
- ❖ Khi một câu truy vấn phức tạp trong tương lai có thể phải sử dụng lại nhiều lần.
 - Tạo View để lưu câu truy vấn.
 - Trường hợp hay xảy ra nhất là các câu truy vấn tính toán, tổng hợp dữ liệu.
- ❖ Không muốn user biết được tên bảng, cấu trúc thật sự của bảng.

Bài giảng Hệ quản trị CSDL

60



Khung nhìn – Lưu ý

- ❖ Phải đặt tên cho các cột của khung nhìn trong các trường hợp sau đây:
 - Trong kết quả của câu lệnh SELECT có ít nhất một cột được sinh ra bởi một biểu thức (tức là không phải là một tên cột trong bảng cơ sở) và cột đó không được đặt tiêu đề.
 - Tồn tại hai cột trong kết quả của câu lệnh SELECT có cùng tiêu đề cột.



Khung nhìn – Lưu ý

❖ Ví dụ:

CREATE VIEW Tuoisinhvien

AS

```
SELECT Masv, Hodem, Ten,
       DATEDIFF(Y, Ngaysinh, GETDATE())
FROM Sinhvien
```



Sai do cột thứ 4 không xác định được tên cột



Khung nhìn (tt)

❖ Tạo khung nhìn

```
CREATE VIEW <tên khung nhìn> [(danh_sách_tên_cột)]
AS <câu truy vấn>
```

❖ Sửa khung nhìn

```
ALTER VIEW <tên khung nhìn> [(danh_sách_tên_cột)]
AS <câu truy vấn>
```

❖ Xóa bỏ khung nhìn:

```
DROP VIEW <tên khung nhìn>
```



Cập nhật, bổ sung và xóa dữ liệu qua view

```
INSERT INTO <tên_view>[(<danh sách tên cột>)]
VALUES (<danh sách giá trị>);
```

```
UPDATE <tên_view>
SET <tên_cột_1> = <biểu_thức_1>
    [, <tên_cột_2> = <biểu_thức_2>, . . .]
[WHERE <điều_kiện>]
```

```
DELETE FROM <tên_view>
[WHERE <điều_kiện>]
```




Cập nhật, bổ sung và xóa ...(tt)

- ❖ Về mặt lý thuyết, để có thể thực hiện thao tác bổ sung, cập nhật và xóa, một khung nhìn phải thoả mãn các điều kiện sau đây:



Cập nhật, bổ sung và xóa ...(tt)

- ❖ Mệnh đề SELECT không thể chứa từ khóa DISTINCT.
- ❖ Mệnh đề SELECT không thể chứa các hàm tổng.
- ❖ Mệnh đề SELECT không thể chứa các toán tử tập hợp.
- ❖ Mệnh đề SELECT không thể chứa một mệnh đề ORDER BY.
- ❖ Mệnh đề FROM không thể chứa nhiều bảng.
- ❖ Mệnh đề WHERE không thể chứa các truy vấn phụ.
- ❖ Truy vấn có thể không chứa GROUP BY hoặc HAVING.
- ❖ Các cột được tính có thể không được cập nhật.
- ❖ Tất cả các cột NOT NULL từ bảng cơ sở phải được bao gồm trong view để truy vấn INSERT hoạt động



Cập nhật, bổ sung và xóa ...(tt)

❖ Ví dụ 1

CREATE TABLE **DONVI**

```
(
  Madv INT PRIMARY KEY,
  Tendv NVARCHAR(30) NOT NULL,
  Dienthoai NVARCHAR(10) NULL,
)
```

CREATE TABLE **NHANVIEN**

```
(
  Manv NVARCHAR(10) PRIMARY KEY,
  Hoten NVARCHAR(30) NOT NULL,
  Ngaysinh DATETIME NULL,
  Diachi NVARCHAR(50) NULL,
  Madv INT FOREIGN KEY
  REFERENCES Donvi (Madv)
)
```

Bài giảng Hệ quản trị CSDL

67



Cập nhật, bổ sung và xóa ...(tt)

❖ Ví dụ 1 (tt)

Giả sử trong hai bảng này đã có dữ liệu như sau:

❖ Bảng DONVI

MADV	TENDV	DIENTHOAI
1	P. Kinh doanh	822321
2	P. Tiếp thị	822012

❖ Bảng NHANVIEN

MANV	HOTEN	NGAYSINH	DIACHI	MADV
NV01	Tran Van A	1975-02-03 00:00:00	77 Tran Phu	1
NV02	Mai Thi B	1977-05-04 00:00:00	34 Nguyen Hue	2
NV03	Nguyen Van C	NULL	NULL	2

Bài giảng Hệ quản trị CSDL

68



Cập nhật, bổ sung và xóa ...(tt)

❖ Ví dụ 1 (tt)

Định nghĩa khung nhìn NV1 như sau:

```
CREATE VIEW NV1
```

```
AS
```

```
SELECT Manv, Hoten, Madv FROM NHANVIEN
```

❖ Nếu ta thực hiện câu lệnh:

```
INSERT INTO NV1 VALUES('NV04','Le Thi D',1)
```



Cập nhật, bổ sung và xóa ...(tt)

❖ Ví dụ 1 (tt)

❖ Bảng NHANVIEN

MANV	HOTEN	NGAYSINH	DIACHI	MADV
NV01	Tran Van A	1975-02-03 00:00:00	77 Tran Phu	1
NV02	Mai Thi B	1977-05-04 00:00:00	34 Nguyen Hue	2
NV03	Nguyen Van C	NULL	NULL	2
NV04	Le Thi D	NULL	NULL	1

Bản ghi mới



Cập nhật, bổ sung và xóa ...(tt)

❖ Ví dụ 2

CREATE VIEW NV2

AS

SELECT Manv, Hoten, YEAR(Ngaysinh) AS Namsinh, Madv
FROM NHANVIEN

❖ Xét các câu lệnh sau:

1. INSERT INTO NV2 (Manv, Hoten, Madv)
VALUES ('NV03', 'Le Van E', 1)
2. UPDATE NV2 SET Hoten='Le Thi X' WHERE Manv='NV04'
3. DELETE FROM NV2 WHERE Manv='NV04'

Bài giảng Hệ quản trị CSDL

71



Khung nhìn

Ví dụ:

❖ UPDATE NHANVIEN_VIEW SET TUOI = 30
WHERE TEN=N'Hoàng'

❖ DELETE FROM NHANVIEN_VIEW WHERE (TUOI
>55 and PHAI = N'Nữ') or (TUOI >60 and PHAI =
N'Nam')

Bài giảng Hệ quản trị CSDL

72



Bài tập 1

- Cho lược đồ CSDL quản lý hàng hóa như sau:
 - HangHoa(MaHG, TenHG, DVT, Dongia, Cohang)
 - Khách(MaKH, Hoten, Diachi, Daily, sodienthoai)
 - HoaDon(SoHD, Ngaylap, Ngaygiao, Trigia, MaKH)
 - Chitiet_HD(SoHD, MaHG, Soluong, Giaban)
- Thực hiện các công việc sau:
 1. Viết câu lệnh để tạo tất cả các bảng dựa vào các LĐQH trên (có đầy đủ RB khóa chính, khóa ngoại và not null).
 2. Trong bảng Khách, viết câu lệnh sửa cột diachi thành nvarchar(30), giả sử trước đó cột này có kiểu char(20).
 3. Viết câu lệnh kiểm tra RB trên bảng Chitiet_HD.
 4. Viết câu lệnh không kiểm tra RB trên bảng HoaDon.
 5. Thêm RB kiểm tra cho thuộc tính cohang trong bang HangHoa là 0 hoặc 1.
 6. Thêm RB duy nhất cho thuộc tính sodienthoai trong bảng Khách.

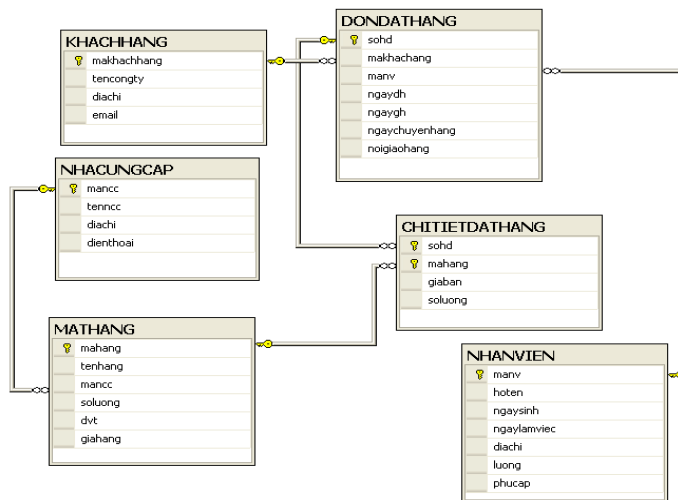
Bài giảng Hệ quản trị CSDL

73



Bài tập 2

- Cho lược đồ csdl QLBanHang được mô tả như sau:



Bài giảng Hệ quản trị CSDL

74



Bài tập 2

- Thực hiện các yêu cầu sau bằng SQL
 - Tạo CSDL và tạo các bảng
 - Cho biết mã và tên của các mặt hàng có giá lớn hơn 10 và số lượng hiện có ít hơn 20
 - Cho biết thông tin những khách hàng nào đã mua mặt hàng áo Việt Tiến
 - Cho biết thông tin những mặt hàng nào chưa từng được khách hàng đặt mua?
 - Cho biết tổng số lượng bán được của mỗi mặt hàng
 - Bổ sung ràng buộc cho bảng DONDATHANG ràng buộc kiểm tra ngày giao hàng và ngày chuyển hàng phải sau hoặc bằng với ngày đặt hàng.
 - Thống kê số lượng hóa đơn đã lập của mỗi nhân viên

Bài giảng Hệ quản trị CSDL

75



Bài tập 3

Tạo CSDL
và các
bảng sau
bằng lệnh:

Tacgia(matg, hotengt, namsinh, quequan, madv)

Donvi(madv, tendv, soch)

Tapchi(matc, tentc, noixb)

Chitiettc(macttc, matc, sora, sotrang)

Baibao(mabb, tenbb, macttc, ngaydang)

Tacgia_baibao(matg, mabb)

- Cho biết ds bài báo đăng vào ngày 19/2/2017
- Cho biết ds các chi tiết tạp chí của tạp chí "Khoa học công nghệ"
- Cho biết danh sách tác giả (matg, hoten, namsinh, tendv) là tác giả của bài báo có mã "bbcنت01"
- Cho biết ds tác giả (matg, hoten) của những tác giả không có bài báo nào

Bài giảng Hệ quản trị CSDL

76



Tài liệu tham khảo

- ❖ Slide Bài giảng Hệ quản trị Cơ sở dữ liệu, khoa CNTT Đại học Thủ Dầu Một
- ❖ Mike Chapple, Microsoft SQL Server 2012 for Dummies, Wiley, 2013.
- ❖ Tạ Thị Thu Phượng, Hệ quản trị cơ sở dữ liệu (Bài giảng tóm tắt), Đại học Đà Lạt.

Bài giảng Hệ quản trị CSDL

77



Phone: 0650. 3834930

Website: www.fit.tdmu.edu.vn