

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HCM
KHOA CÔNG NGHỆ THÔNG TIN



BÀI TẬP THỰC HÀNH MÔN: HỆ CƠ SỞ DỮ LIỆU

Bài tập được thiết kế theo từng TUẦN mỗi TUẦN là 3 tiết có sự hướng dẫn của GV.

Cuối mỗi buổi thực hành, sinh viên nộp lại phần bài tập mình đã thực hiện cho GV hướng dẫn.

Những câu hỏi mở rộng/khó giúp sinh viên trau dồi thêm kiến thức của môn học. Sinh viên phải có trách nhiệm nghiên cứu, tìm câu trả lời nếu chưa thực hiện xong trong giờ thực hành.

TP. Hồ Chí Minh
Năm 2021

BÀI TẬP TUẦN 1

Số tiết: 3

Mục tiêu:

- ✎ Thao tác được với giao diện của SQL Server
- ✎ Tạo Database - cơ sở dữ liệu (CSDL) và thực hiện các thao tác cơ bản trên CSDL bằng lệnh và bằng công cụ design
- ✎ Tạo các Table (Bảng dữ liệu) và nhập dữ liệu bằng công cụ design
- ✎ Tạo lược đồ quan hệ (Relationship Diagram)
- ✎ Biết các kiểu dữ liệu (DataType) trong SQL Server
- ✎ Biết tạo, sửa, xóa và áp dụng các kiểu dữ liệu trong SQL Server
- ✎ Biết sử dụng một số thủ tục trợ giúp về database và Datatype

PHẦN 1: TÌM HIỂU SQL SERVER MANAGEMENT STUDIO

1. Giới thiệu SQL Server Management Studio:

- SQL Server Management Studio là một môi trường tích hợp cho phép truy cập, cấu hình, quản lý, quản trị và phát triển tất cả các công cụ của SQL Server. Nó kết hợp một nhóm công cụ đồ họa cho phép soạn thảo một lượng lớn mã lệnh tương tác với SQL SERVER và đến người quản trị cũng như người lập trình.
- SQL Server Management Studio kết hợp các đặc trưng của Enterprise Manager, Query Analyzer, and Analysis Manager vào một môi trường thống nhất. Bên cạnh đó, SQL Server Management Studio làm việc với tất cả công cụ của SQL Server như Reporting Services and Integration Services. Người lập trình cũng như người quản trị dễ dàng thao tác trên một môi trường đồng nhất và thân thiện.

2. Khởi động SQL Server Management Studio:

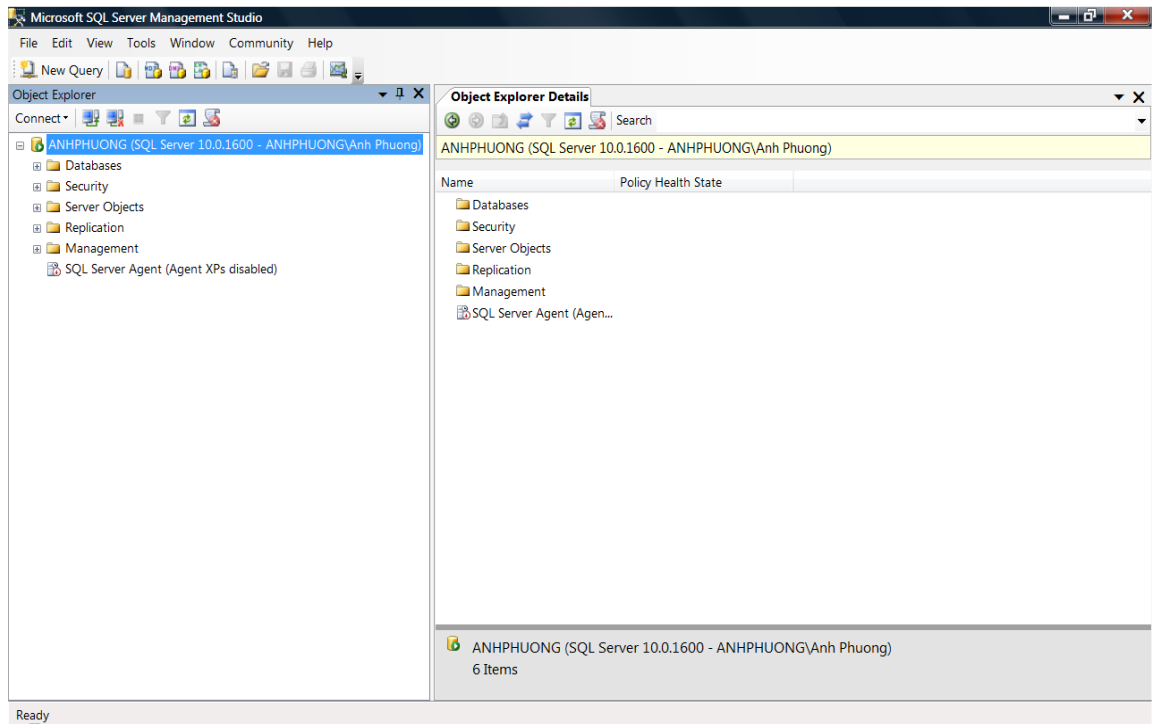
Vào start → chọn program → chọn Microsoft SQL Server 2012 → chọn SQL Server Management Studio



Hình 1.1 Kết nối vào SQL Server

Chú ý những thành phần trên hộp thoại sau:

- **Server Type:** các subsystems của SQL Server mà người dùng có thể đăng nhập vào, gồm:
 - Database engine.
 - Analysis Services.
 - Report Server.
 - Integration Services
- **Server Name:** tên của Server mà người dùng muốn đăng nhập:
 - “.”: đăng nhập vào một thể hiện mặc định của SQL Server trên cùng máy tính đang đăng nhập.
 - “.” (local): định danh tự động và cách đăng nhập đến server đó.
 - Nếu bạn mở hộp Server name bạn có thể tìm kiếm nhiều server local hoặc network connection bằng cách chọn <Browse for more...>.
- **Authentication:** xác định các loại hình kết nối bạn muốn sử dụng. Có 2 cách đăng nhập:
 - **Windows Authentication:** thông tin đăng nhập Windows được chuyển thành tài khoản đăng nhập SQL Server.
 - **SQL Server Authentication:** Người dùng cung cấp username và password để đăng nhập vào SQL Server.
- Sau khi nhấn nút Connect sẽ xuất hiện màn hình sau:



3. Chọn **Connect**: Kết nối
Cancel: Hủy bỏ thao tác
Option: Các lựa chọn khác
4. Bạn hãy cho khởi động dịch vụ SQL Server, SQL Server Agent.
5. Vào menu View, Chọn Object Explorer Details
 - Lần lượt mở các nhánh của cây MicroSoft SQL Servers.
 - Tìm hiểu sơ lược cửa sổ, thực đơn, thanh công cụ của **SQL Server Management Studio**.
6. Tại cửa sổ Object Explorer, thực hiện:
 Quan sát các thành phần đối tượng trên cửa sổ và hãy cho biết:
Có bao nhiêu SQL Server Group, mỗi Server tên là gì? Đang connect hay disconnect?

.....

Liệt kê các thành phần trong Server hiện hành, cho biết chức năng của mỗi thành phần

.....

Trong server hiện hành, có các Database nào?

.....
 (Hãy so sánh tên của các database với các database của máy bên cạnh)

Trong mỗi Database có những đối tượng nào?

.....

.....
(Các database khác nhau thì các đối tượng có khác nhau không?)

Mở database Master, khảo sát các đối tượng:

- Vào đối tượng Table, tìm hiểu cấu trúc và dữ liệu của bảng (lưu ý: chỉ được chọn xem không nên xóa hay sửa dữ liệu): Sysdatabases, SysObjects, systypes, syslogins, sysusers, sysmessages, syspermissions...
 - Vào đối tượng Stored Procedures, tìm hiểu nội dung của các thủ tục sau (lưu ý: chỉ được chọn xem không nên xóa hay sửa): sp_help, sp_helpdb, sp_helpconstraint, sp_rename, sp_renamedb, sp_table, sp_addlogin, sp_addmessage, sp_addrole ...
 - Lần lượt vào đối tượng còn lại User, Role, ...
7. **Tìm hiểu các mục trong menu Help.** Lần lượt tìm hiểu các lệnh Create DataBase, Create Table, Alter Table, Select Statement, Select into, Update Statement, Insert Statement, DataType, Triggers... (Hướng dẫn: Gõ tên lệnh/từ khóa cần tìm và nhấn Enter)
8. **Khởi động màn hình Query Editor:**

Nhập dòng lệnh sau trên cửa sổ Query Editor:

USE master

SELECT * from dbo.MSreplication_options

Nhấn F5 để thực thi và quan sát kết quả hiển thị.

Tìm hiểu các mục trong menu Tools→Option

PHẦN 2: TẠO VÀ QUẢN LÝ CƠ SỞ DỮ LIỆU

PHẦN LÝ THUYẾT

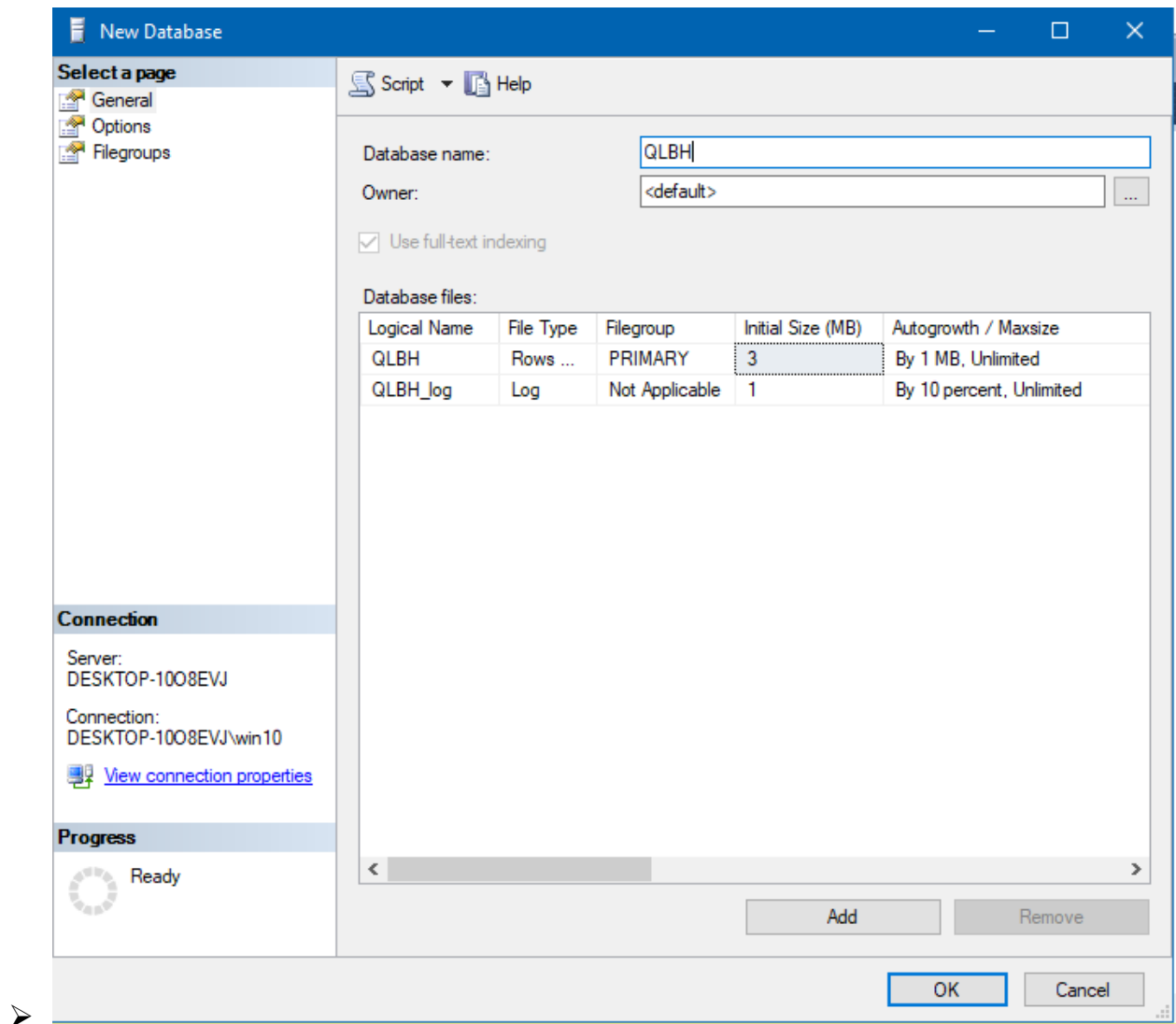
I. Giới thiệu Database

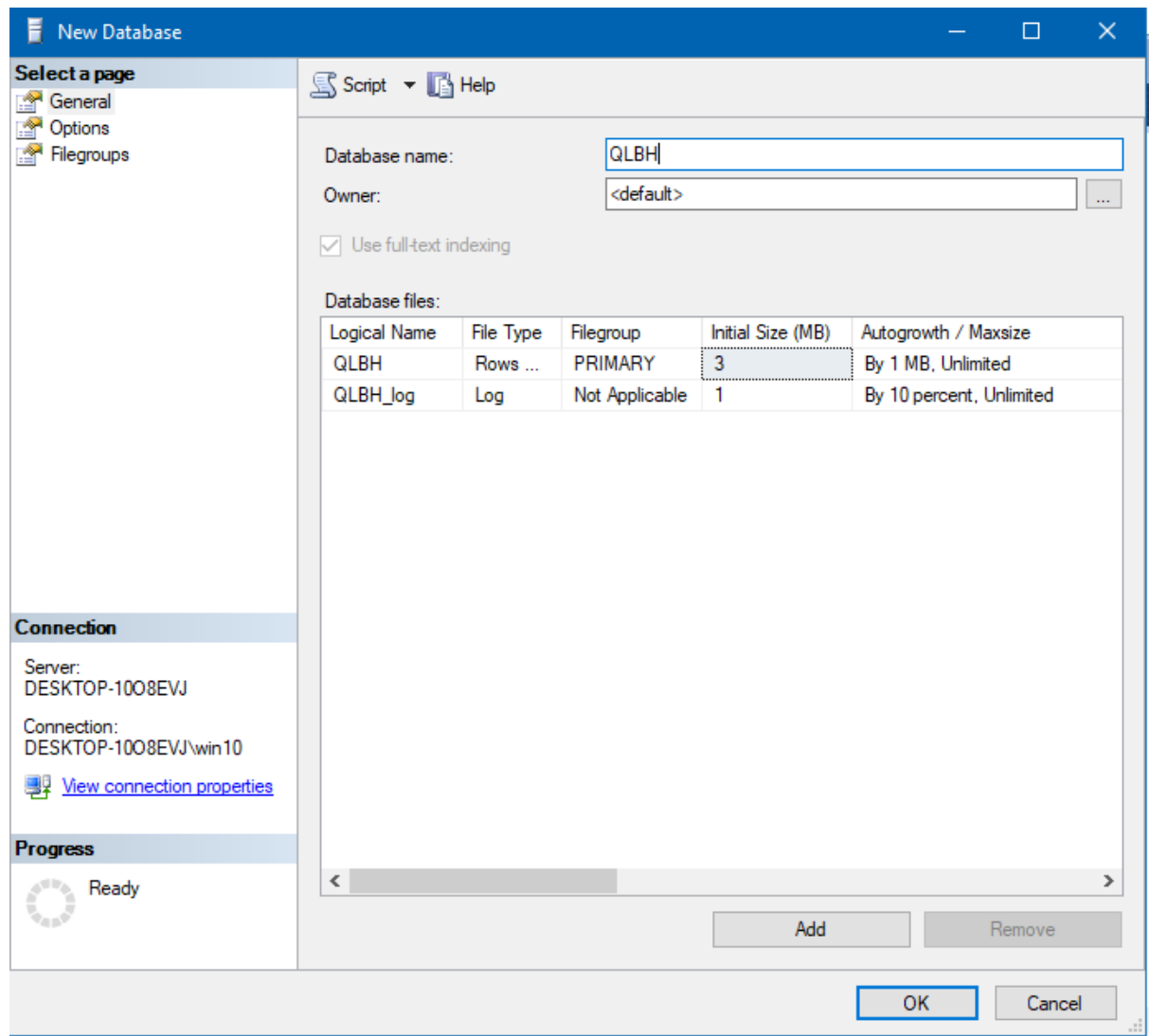
1. **Databases:** chứa tất cả các cơ sở dữ liệu hệ thống và cơ sở dữ liệu người dùng trong SQL Server. Cơ sở dữ liệu trong SQL server là cơ sở dữ liệu quan hệ, bao gồm một tập các quan hệ, mỗi quan hệ là một bảng dữ liệu bao gồm các dòng và cột.
2. Trong một CSDL có tối thiểu 2 tập tin:
 - **File dữ liệu cơ bản (Primary data file) (.mdf):** mỗi CSDL chỉ có duy nhất 1 file cơ bản (mặc định), dùng để ghi nhận lại tất cả những tập tin khác trong CSDL và lưu trữ dữ liệu.
 - **Các file thứ cấp (Secondary data files) (.ndf) (tùy chọn):** một CSDL có thể có hay không có nhiều file thứ cấp, dùng để lưu các đối tượng của CSDL.
 - **File nhật ký giao dịch (Transaction log file) (.ldf):** mỗi CSDL có từ 1 hay nhiều file nhật ký, dùng để chứa những thông cần thiết cho việc phục hồi tất cả những giao tác (transaction) trong CSDL.
 - **Về mặt vật lý:** một Database bao gồm hai hay nhiều hơn hai tập tin trên một hay nhiều đĩa. Chỉ thấy được bởi nhà quản trị và nó trong suốt đối với người sử dụng

- **Về mặt Logic:** một database được xây dựng thành các thành phần mà được hiển thị với người dùng như Table, View, Procedure, ...
 - Khi tạo 1 CSDL, thì các file dữ liệu và log được tạo ra tại vị trí do ta xác định.
 - Các file này có nằm trên những đĩa vật lý khác nhau để cải thiện việc thực thi của hệ thống.
3. **Filegroup** có thể chứa 1 hay nhiều file. Một CSDL có thể được chứa trong 1 hay 1 số filegroup. Có 3 loại: Primary filegroup, user-define filegroups và default filegroup
- **Primary FileGroup:** chứa file dữ liệu chính (.mdf) và bất cứ file thứ cấp nào (.ndf). Tất cả các bảng hệ thống phải nằm trong primary filegroup.
 - **User-defined filegroup:** do người dùng xác định trong lệnh CREATE/ALTER DATABASE
 - **Default filegroup:** là bất kỳ filegroup nào trong DB. Thường thì primary filegroup chính là default filegroup nhưng owner có quyền thay đổi. Tất cả bảng và index mặc định đều được tạo ra trong default filegroup.

II. Các lệnh định nghĩa dữ liệu (DDL - Data Definition Language)

1. Tạo cơ sở dữ liệu bằng các sử dụng công cụ SQL Server Management Studio (SSMS)
 - Click chuột phải trên thư mục Database → Chọn New Database
 - Trong ô Database name: Nhập tên Database
 - Trong Database file:
 - Logical Name: Nhập tên luận lý
 - Initial Size: qui định kích thước khởi tạo cho các tập tin
 - Autogrowth/Maxsize: qui định kích thước tối đa và kích cỡ gia tăng
 - Tại mục Path: chọn vị trí lưu file
 - File Name: đặt tên file lưu trên ổ đĩa
 - Nhấn nút Add nếu muốn tạo thêm file thứ cấp, nhấn nút remove để xóa file
 - Nhấn nút OK





2. Tạo cơ sở dữ liệu:

Cú pháp:

CREATE DATABASE *database_name*

[**ON**

[< filespec > [,...n]]

[, < filegroup > [,...n]]

]

[**LOG ON** { < filespec > [,...n] }]

Cú pháp Filespec:

(**NAME** = *logical_name*,

FILENAME = '*path\filename*',

SIZE = *size_in_MB*,

MAXSIZE = *size_in_MB* | **UNLIMITED**,

FILEGROWTH = %*or_MB*)

Ví dụ 1 : Tạo CSDL QLSV

CREATE DATABASE QLSV

ON PRIMARY


```

    (NAME=QLSV_Data,
    FILENAME='T:\BTSQ\QLSV_Data.mdf',
    SIZE=10MB,
    MAXSIZE=15MB,
    FILEGROWTH=20%)
LOG ON
    ( NAME=QLSV_Log,
    FILENAME= 'T:\BTSQ\QLSV_Log.ldf',
    SIZE=3MB,
    MAXSIZE=5MB,
    FILEGROWTH=1MB)

```

Ví dụ 2: Tạo CSDL QLBH

```

CREATE DATABASE QLBH
ON PRIMARY
( NAME = QLBH_data1, FILENAME = 'T:\BTSQ\QLBH_data1.mdf',
  SIZE = 10, MAXSIZE = 50, FILEGROWTH = 15% ),
( NAME = QLBH_data2, FILENAME = 'T:\BTSQ\QLBH_data2.ndf',
  SIZE = 10, MAXSIZE = 50, FILEGROWTH = 15% ),
FILEGROUP QLBHGroup1
( NAME = QLBH_data3, FILENAME = 'T:\BTSQ\QLBH_data3.ndf',
  SIZE = 10, MAXSIZE = 50, FILEGROWTH = 5 )
( NAME = QLBH_data4,
  FILENAME = 'T:\BTSQ\QLBH_data4.ndf',
  SIZE = 10, MAXSIZE = 50, FILEGROWTH = 5 )
LOG ON
( NAME = 'QLBH_log',
  FILENAME = 'T:\BTSQ\QLBH_log.ldf',
  SIZE = 5MB,
  MAXSIZE = 25MB,
  FILEGROWTH = 5MB )

```

3. Mở CSDL

Cú pháp: USE TenCSDL

Ví dụ: use QLSV

4. Xem thông của tất cả các CSDL trong Server hiện hành

Cú pháp: sp_helpdb

sp_helpdb

5. Xem thông tin của một CSDL

Cú pháp: sp_helpdb TenCSDL

sp_helpdb QLSV

6. Kiểm tra không gian sử dụng của CSDL

Cú pháp: sp_spaceused

sp_spaceused

7. Thay đổi cấu trúc CSDL

Cú pháp:

ALTER DATABASE *database_name*

```

ADD FILE filespec [TO FILEGROUP filegroup_name]
ADD LOG FILE filespec
| REMOVE FILE logical_filename
| ADD FILEGROUP filegroup_name
| REMOVE FILEGROUP filegroup_name
| MODIFY FILE filespec
| MODIFY FILEGROUP filegroup_name
filegroup_property
|SET optionspec [WITH termination]
    
```

Ví dụ:

- a) Thêm file group GroupOrder vào CSDL QL BH

Ví dụ: `ALTER DATABASE QL BH ADD FILEGROUP QL BHGroup1`

- b) Chỉnh sửa Size của tập tin

```

ALTER DATABASE QL BH
MODIFY FILE (NAME = 'QL BH_log', size =10MB)
    
```

- c) Bổ sung thêm một tập tin dữ liệu thứ cấp QL BH_data2

```

ALTER DATABASE QL BH
ADD File (Name =QL BH_data2,
Filename = 'T:\BTSQL\QL BH_data2.mdf',
SIZE =10 MB,
Maxsize =20MB)
    
```

- d) Xóa file thứ cấp QL BH_data2

```

ALTER DATABASE QL BH REMOVE FILE QL BH_data2
    
```

- e) Xóa file group GroupOrder: chú ý file group muốn xóa phải trống

```

ALTER DATABASE QL BH REMOVE FILE QL BHGroup1
    
```

- f) Thay đổi thuộc tính CSDL

Cú pháp:

```

ALTER DATABASE database_name
SET option [, status]
Option
AUTO_SHRINK
CURSOR_CLOSE_ON_COMMIT
RECOVERY FULL | BULK_LOGGED | SIMPLE
SINGLE_USER | RESTRICTED_USER | MULTI_USER
READ_ONLY | READ_WRITE
    
```

Ví dụ:

```

ALTER DATABASE QL BH
SET Read_Only
    
```

- g) Đổi tên cơ sở dữ liệu:

Cú pháp: `sp_renamedb [@dbname =] 'old_name', [@newname =] 'new_name'`

Ví Dụ: `Sp_ReNamedb 'QL BH', 'Banhang'`

h) Xóa cơ sở dữ liệu: Khi 1 CSDL bị xóa thì tất cả các file vật lý của nó sẽ bị xóa

Cú pháp:

DROP DATABASE *database_name*

Ví dụ:

Drop database Banhang

8. Kiểu dữ liệu (System Data Type)

Có 2 nhóm:

- **System-Supplied datatype:** Các kiểu dữ liệu cơ bản được hỗ trợ bởi SQL Server.
- **User-defined datatype:** Các kiểu dữ liệu của người dùng tự định nghĩa dựa trên các kiểu dữ liệu cơ bản.

a) Tạo một User-Defined Data Type

□ Dùng thủ tục hệ thống *sp_addtype* để tạo một user-defined data type.

Cú pháp: **sp_addtype** *type, system_data_type* ['NULL' | 'NOT NULL']

Ví dụ: Tạo kiểu dữ liệu tên là **isbn** với kiểu dữ liệu cơ bản là **smallint** và không chấp nhận giá trị Null

EXEC sp_addtype isbn, 'smallint', 'NOT NULL'

b) Xem các user-defined data types trong CSDL hiện hành:

Dùng thủ tục **sp_help** hoặc truy vấn trong **information_schema.domains**

Ví dụ: **Use** **QLBHDB**

Sp_help

hoặc

```
SELECT domain_name, data_type, character_maximum_length
FROM information_schema.domains
ORDER BY domain_name
```

c) Xóa một User-Defined Data Type: dùng thủ tục hệ thống *sp_droptype* để xóa một user-defined data type từ bảng systypes. Một user-defined data type không thể xóa được nếu nó được tham chiếu bởi các bảng và những đối tượng khác.

Cú pháp: **Sp_droptype** *type*

Ví dụ:

EXEC sp_droptype isbn

PHẦN THỰC HÀNH

1. Tạo CSDL QLSach bằng SSMS có tham số như sau:

Tham số	Giá trị
Database name	QLSach
Tên logic của data file chính	QLSach_data
Tên tập tin và đường dẫn của data file chính	T:\QLTV\QLSach_Data.mdf
Kích cỡ khởi tạo của CSDL	20 MB
Kích cỡ tối đa của CSDL	40 MB

Gia số gia tăng tập tin CSDL	1 MB
Tên logic của transaction log	QLSach_Log
Tên tập tin và đường dẫn của transaction log	T:\QLTV\QLSach_Log.ldf
Kích cỡ khởi tạo của transaction log	6 MB
Kích cỡ tối đa của transaction log	8 MB
Gia số gia tăng tập tin transaction log	1 MB

- a. Xem lại thuộc tính (properties) của CSDL QLSach. (HD: Nhấp phải chuột tại tên CSDL, chọn properties). Quan sát và cho biết các trang thể hiện thông tin gì?.
- b. Tại cửa sổ properties của CSDL, khai báo thêm
 - Một Group File mới có tên là DuLieuSach
 - Một tập tin dữ liệu (data file) thứ hai nằm trong Group file vừa tạo ở trên và có thông số như sau Tên login của data file là QLSach_Data2; Tên tập tin và đường dẫn vật lý của data file là T:\QLTV\QLSach_Data2.ndf.
 - Chọn thuộc tính ReadOnly, sau đó đóng cửa sổ properties. Quan sát màu sắc của CSDL. Bỏ thuộc tính ReadOnly.
 - Thay đổi Owner: tên server đang kết nối.
2. Tạo cơ sở dữ liệu bằng lệnh T-SQL tại cửa sổ Query Analyzer (lưu ý: sau mỗi lần có sự thay đổi thì phải dùng các lệnh để kiểm tra sự thay đổi đó)
 - a. Dùng lệnh Create DataBase, tạo một CSDL với các tham số được liệt kê như trong bảng dưới. Lưu ý rằng CSDL này gồm một data file và nó được nằm trong primary filegroup

THAM SỐ	GIÁ TRỊ
Database name	QLBH
Tên logic của data file chính	QLBH_data1
Tên tập tin và đường dẫn của data file chính	T:\QLBH_data1.mdf
Kích cỡ khởi tạo của CSDL	10 MB
Kích cỡ tối đa của CSDL	40 MB
Gia số gia tăng tập tin CSDL	1 MB
Tên logic của transaction log	QLBH_Log
Tên tập tin và đường dẫn của transaction log	T:\QLBH.ldf
Kích cỡ khởi tạo của transaction log	6 MB
Kích cỡ tối đa của transaction log	8 MB
Gia số gia tăng tập tin transaction log	1 MB

- b. Xem lại thuộc tính của CSDL QLBH bằng cách Click phải vào tên CSDL chọn Property và bằng thủ tục hệ thống sp_helpDb, sp_spaceused, sp_helpfile.
- c. Thêm một filegroup có tên là DuLieuQLBH (HD: dùng lệnh Alter DataBase <Tên Database> ADD FILEGROUP <Tên filegroup>)

- d. Khai báo một secondary file có tên logic là QLBH_data2, tên vật lý QLBH_data2.ndf nằm ở T:\, các thông số khác tùy bạn chọn, data file này nằm trong file group là DuLieuQLBH. (HD: Dùng lệnh Alter Database ADD FILE TO FILEGROUP ...)
 - e. Cho biết thủ tục hệ thống sp_helpfilegroup dùng để làm gì?
 - f. Dùng lệnh Alter Database ... Set ... để cấu hình cho CSDL QLBH có thuộc tính là Read_Only. Dùng sp_helpDB để xem lại thuộc tính của CSDL. Hủy bỏ thuộc tính Read_Only.
 - g. Dùng lệnh Alter DataBase ... MODIFY FILE ... để tăng SIZE của QLBH_data1 thành 50 MB. Tương tự tăng SIZE của tập tin QLBH_log thành 10 MB. Để thay đổi SIZE của các tập tin bằng công cụ Design bạn làm như thế nào? Bạn hãy thực hiện thay đổi kích thước của tập tin QLBH_log với kích thước là 15MB. Nếu thay đổi kích cỡ nhỏ hơn ban đầu có được không? Nếu thay đổi kích cỡ MAXSIZE nhỏ hơn kích cỡ SIZE thì có được không? Giải thích.
3. Tạo CSDL QLSV, các thông số tùy chọn. Dùng công cụ design tạo cấu trúc của các bảng sau trong CSDL QLSV:
 LOP (MaLop char(5) , TenLop Nvarchar(20), SiSoDuKien Int, NgayKhaiGiang DateTime)
 SINHVIEN (MaSV char(5), TenHo Nvarchar(40), NgaySinh DateTime, MALOP char(5))
 MONHOC(MaMh char(5), Tenmh Nvarchar(30), SoTC int)
 KETQUA(MaSV char(5), MAMH char(5), Diem real)
 Lưu ý: cột in đậm gạch chân là khóa chính và không chấp nhận giá trị Null, cột in đậm không chấp nhận giá trị Null
 - a. Tạo Diagram giữa hai bảng vừa tạo.
 - b. Nhập dữ liệu tùy ý vào hai các bảng bằng SSMS, mỗi bảng khoảng 3 mẫu tin. Giả sử bạn nhập dữ liệu cho bảng KETQUA trước, sau đó mới nhập dữ liệu cho các bảng còn lại thì bạn có nhập được không? Vì sao? Theo bạn nên nhập dữ liệu theo thứ tự nào?
 - c. Dùng tác vụ General Script, để tạo đoạn Script cho CSDL QLSV và tất cả các đối tượng của CSDL thành một tập tin Script có tên là QLSV.SQL
 - d. Vào Query Analyzer, mở tập tin Script vừa tạo và khảo sát công dụng và cú pháp của các lệnh có trong tập tin script.
 - e. Đổi tên CSDL QLSV thành QLHS
 - f. Dùng thao tác xóa để xóa toàn bộ CSDL QLHS

PHẦN 3: KIỂU DỮ LIỆU (DATA TYPE)

1. Tìm hiểu về kiểu dữ liệu (datatype):

- a. Tìm hiểu và trả lời các câu hỏi sau:
 - Có mấy loại datatype, hãy liệt kê.

- Các system datatype được SQL Server lưu trữ trong Table nào ở trong CSDL nào.
- Các User-defined datatype được SQL Server lưu trữ trong Table nào, ở trong CSDL nào?

b. Vào Query Analyzer, chọn QL BH là CSDL hiện hành, định nghĩa các datatype:

Kiểu dữ liệu (Data type)	Mô tả dữ liệu (Description of data)
Mavung	10 ký tự
STT	STT không vượt quá 30,000
SoDienThoai	13 ký tự , chấp nhận NULL
Shortstring	Số ký tự thay đổi đến 15 ký tự

HD: Dùng thủ tục sp_addtype để định nghĩa

Ví dụ: EXEC sp_addtype SODienThoai, 'char(13)', NULL

- c. Các User-defined datatype vừa định nghĩa được lưu trữ ở đâu và phạm vi sử dụng của nó ở đâu (trong toàn bộ một instance hay chỉ ở trong CSDL hiện hành).
- d. Có bao nhiêu cách liệt kê danh sách các User-Defined datatype vừa định nghĩa.

**SELECT domain_name, data_type, character_maximum_length
FROM information_schema.domains
ORDER BY domain_name**

Hoặc **SELECT * From Systype**

- e. Tạo 1 bảng có tên là ThôngTinKH(MaKH (khóa chính) kiểu dữ liệu STT , Vung kiểu là Mavung , Diachi kiểu là Shortstring, DienThoai kiểu là SoDienThoai) trong CSDL QL BH và sử dụng User-defined data type vừa định nghĩa ở trên. Bạn có tạo được không? Nếu được bạn nhập thử dữ liệu 2 record bằng design.
- f. Muốn User-Defined datatype được dùng trong tất cả các CSDL thì bạn định nghĩa nó ở đâu?
- g. Xóa kiểu dữ liệu SoDienThoai.
- h. Thực hiện việc Backup và Retore CSDL QL BH

BÀI TẬP TUẦN 2 & 3**Số tiết: 6****Mục tiêu:**

- ✗ Tạo CSDL cùng các bảng trong CSDL bằng T-SQL
- ✗ Tạo các ràng buộc (constraint) cho các bảng bằng T-SQL
- ✗ Phát sinh tập tin script
- ✗ Thực hiện chức năng attach và detach CSDL
- ✗ Thực hiện chức năng import/export

PHẦN LÝ THUYẾT**1) Bảng dữ liệu – Table**

Bảng là một đối tượng của CSDL được dùng để lưu trữ dữ liệu. Dữ liệu trong bảng được tổ chức thành các hàng (rows) và cột (columns). Mỗi hàng trong bảng biểu diễn một bản ghi (record) duy nhất. Mỗi cột biểu diễn một thuộc tính (attribute). Tên cột trong 1 bảng không được trùng nhau nhưng cho phép tên cột có thể trùng nhau trong những bảng khác nhau của cùng 1 CSDL.

SQL Server cho phép:

Tối đa 2 triệu bảng trong 1 CSDL.

Tối đa 1024 cột trong 1 bảng

Tối đa 8060 bytes trong 1 hàng

2) Tạo Table

Cú pháp:

CREATE TABLE

```
[ database_name.[ owner ] .|owner.] table_name
({ < column_definition >
|column_name AS computed_column_expression
| < table_constraint > ::= [ CONSTRAINT
constraint_name ] }
[ { PRIMARY KEY | UNIQUE } [ ,...n ]
)
[ ON { filegroup | DEFAULT } ]
[ TEXTIMAGE_ON { filegroup | DEFAULT } ]
```

Ví dụ:

a) Tạo bảng có khóa chính

```
CREATE TABLE Sanpham
( Masp CHAR(5) primary key, Tensp VARCHAR(15), Dvt
VARCHAR(10), Dongia SMALLMONEY, SlTon INT )
```

b) Tạo bảng có giá trị phát sinh tự động

```
CREATE TABLE NhaCungCap
```

```
(MaNCC int Identity NOT NULL Primary key, TenNCC
VarChar(25))
```

- c) Tạo bảng có cột tính toán

```
CREATE TABLE cthoadon
(   sohd int NOT NULL,
    MaHang char(5) NOT NULL,
    SoLuong int NOT NULL,
    DonGia money,
    ThanhTien AS SoLuong*DonGia
)
```

- d) Khai báo Filegroup chứa Table

```
CREATE TABLE KH
(   MaKh int Identity(1000,1) NOT NULL, TenKH Varchar(40))
ON FGROUPl
```

3) Sửa cấu trúc của bảng

Cú pháp:

```
ALTER TABLE <table_name>
{ALTER COLUMN <column_name> <new_data_type>}
| {ADD [<column_name> <data_type>]}
| {DROP COLUMN <column_name>}
```

Ví dụ:

- a) Thêm cột

```
ALTER TABLE SanPham ADD NgayNhap SmallDateTime
```

- b) Sửa kiểu dữ liệu cho cột

```
ALTER TABLE SanPham ALTER COLUMN NgayNhap DateTime NOT
NULL
```

- c) Xóa cột

```
ALTER TABLE Sanpham DROP COLUMN NgayNhap
```

4) Xóa bảng

Cú pháp: DROP TABLE <Table_Name>

Ví dụ: DROP TABLE SanPham

5) Xem thông tin Table

Cú pháp: sp_help <table_name>

Ví dụ: Sp_help cthoadon

Quy tắc nghiệp vụ (Toàn vẹn dữ liệu)

- TVDL là đề cập đến trạng thái của tất cả các giá trị dữ liệu lưu trữ trong CSDL là đúng. Nếu dữ liệu không đúng mà đã được lưu trữ trong CSDL thì gọi là vi phạm TVDL.
- Các loại ràng buộc toàn vẹn: Not Null, Default, Identity, Constraints, Rule, Triggers, Indexs.

6) Các loại ràng buộc toàn vẹn

Cú pháp:

Cú pháp: < column_constraint > ::= [CONSTRAINT constraint_name]


```
{ [ NULL | NOT NULL ]
  | [ { PRIMARY KEY | UNIQUE } ]
  | [ [ FOREIGN KEY ]
    REFERENCES ref_table [ ( ref_column ) ]
    [ ON DELETE { CASCADE | NO ACTION } ]
    [ ON UPDATE { CASCADE | NO ACTION } ]
  ]
  | CHECK ( logical_expression )
}
```

a) Định nghĩa NULL/NOT NULL

- Giá trị NULL dùng để chỉ các giá trị chưa biết, hay sẽ được bổ sung sau. Nó khác với giá trị rỗng (empty) hay zero. Hai giá trị null không được xem là bằng nhau. Khi so sánh hai giá trị null, hay 1 giá trị null với 1 giá trị khác thì kết quả trả về sẽ là unknown.

• **Ví dụ:**

```
USE QLBH
CREATE TABLE SanPham
(
    Masp          smallint NOT NULL,
    Tensp         char(20) NOT NULL,
    Mota          char(30) NULL,
    Gia           smallmoney NOT NULL
)
```

- b) Ràng buộc Default:** dùng để xác định giá trị “có sẵn” được gán cho 1 cột khi thêm 1 bản ghi mới vào bảng. DEFAULT có thể áp dụng cho bất kỳ cột nào trong bảng ngoại trừ cột có kiểu timestamp hay có thuộc tính IDENTITY.

Cách tạo ràng buộc Default:

Cách 1: Tạo cùng với lúc tạo Table

Cú pháp: CREATE TABLE <TableName>

(<Column_Name> <DataType> DEFAULT (<expresion>))

Ví dụ:

```
CREATE TABLE HoaDon (MaHD int, LoaiHD Char(1) DEFAULT 'X',
    NgayLap DateTime NOT NULL)
```

Cách 2: Tạo khi đã có Table

Cú pháp: ALTER TABLE tablename

ADD [CONSTRAINT *constraintname*]

DEFAULT *expression* FOR *columnname*

Ví dụ:

```
ALTER TABLE HoaDon
    ADD CONSTRAINT Ngay_DF DEFAULT Getdate() FOR NgayLap
```

Cách 3:

Cú pháp: CREATE DEFAULT fieldname AS 'value'

Sau khi tạo được DEFAULT, nó cần được gắn kết vào 1 cột hay kiểu dữ liệu người dùng.

```
sp_bindefault default_name, object_name [,
FUTUREONLY]
```

Xóa gắn kết default làm cho nó không còn áp dụng được vào cột của bảng hay kiểu dữ liệu người dùng.

```
sp_unbindefault object_name
[, FUTUREONLY]
```

Ví dụ:

```
CREATE DEFAULT dienthoai AS 'unknown'
GO
sp_bindefault dienthoai, 'KH.DT'
GO
sp_unbindefault 'KH.DT'
```

c) Xóa ràng buộc Default - mặc định

Cú pháp: DROP DEFAULT { *default* } [,...n]

Hay

```
ALTER TABLE <TenTable>
DROP CONSTRAINT <TenDefault>
```

Lệnh drop có thể xóa cùng lúc nhiều default

Ví dụ:

```
DROP DEFAULT phonedflt
DROP DEFAULT Ngay_DF
```

Hay

```
ALTER TABLE Hoadon
DROP CONSTRAINT Ngay_DF
```

d) Ràng buộc Check: Qui định nhập dữ liệu phải thỏa mãn điều kiện của biểu thức check_logic.

Cách tạo:

Cách 1: Tạo cùng với tạo Table

Cú pháp: CREATE TABLE <Table_Name>

```
(<Column_Name> <Data_Type>[,...] CONSTRAINT
ConstraintName] CHECK (NOT FOR REPLICATION) <Logical
expresion>),....)
```

Ví dụ:

```
CREATE TABLE NhanVien
(MaNV char(4) CHECK (Manv LIKE '[0-9][0-9][0-9][0-9]',
Hoten Varchar(40), LCB int CHECK (LCB BETWEEN 0 AND 50000,
HSPC real,
Thanhpho varchar(10) CONSTRAINT chkCity CHECK(Thanhpho IN
('Berkeley', 'Boston', 'Chicago', 'Dallas'))
```

Cách 2: Tạo sau khi đã tạo bảng

Cú pháp: ALTER TABLE <Table_Name>
 [WITH CHECK | WITH NOCHECK] ADD
 [CONSTRAINT ConstraintName]
 CHECK (NOT FOR REPLICATION) <Logical expression>,...)

Ví dụ: ALTER TABLE Nhanvien
 ADD CONSTRAINT NV_HSPC
 CHECK (HSPC>=0.1 AND HSPC<0.5)

Rule: Định nghĩa các qui tắc hợp lệ mà có thể kết buộc vào các cột của bảng hay các kiểu dữ liệu do người dùng định nghĩa. Rule được tạo nên chính nó trước khi kết buộc vào đối tượng khác

Định nghĩa Rule:

CREATE RULE rulename AS condition_expression

Kết buộc rule vào một cột

sp_bindrule rulename, tablename.columnname

Kết buộc Rule vào user-defined datatype

sp_binrule rulename, datatype_name[, futureonly]

Ví dụ:

CREATE RULE ActiveDate AS
 @Date Between '01/01/70' AND Getdate()
 sp_bindrule ActiveDate, 'Orders.OrderDate'

e) Xoá ràng buộc CHECK

ALTER TABLE <TenTable>
 DROP CONSTRAINT <TenCheck>

Ví dụ:

ALTER TABLE NhanVien
 DROP CONSTRAINT NV_HSPC

7) Các ràng buộc khóa chính, khóa ngoại, unique– Constraints

Cú pháp: < column_constraint > ::= [CONSTRAINT constraint_name]

{ [NULL | NOT NULL]
 | [{ PRIMARY KEY | UNIQUE }]
 | [[FOREIGN KEY]
 REFERENCES ref_table [(ref_column)]
 [ON DELETE { CASCADE | NO ACTION }]
 [ON UPDATE { CASCADE | NO ACTION }]
]
 | CHECK (logical_expression)
 }

a) **Ràng buộc Primary Key:** Ràng buộc Primary key gồm một hay nhiều cột dùng để nhận diện các record, giá trị của primary key không được phép trùng nhau và không chứa giá trị Null. Chỉ mục sẽ được tự động tạo ra khi có khai

báo 1 ràng buộc primary key. Mỗi bảng chỉ có một khóa chính, chỉ mục do primary key tạo ra mặc định thường là clustered

Cú pháp:

Cách 1: Tạo cùng với lệnh tạo bảng

```
CREATE TABLE TableName
    (columnname datatype [...],[CONSTRAINT constraint_name]
    PRIMARY KEY [CLUSTERED|NONCLUSTERED]
    {(column [ASC |DESC][,...,n])}
    [WITH FILLFACTOR = fillfactor]
    [ON {filegroup|DEFAULT}])
```

```
CREATE TABLE DEAN
(
    MADA smallint PRIMARY KEY
    CLUSTERED NOT NULL,
    TENDA varchar(50) NOT NULL
    DEFAULT 'Chua '
)
```

Cách 2: Tạo sau khi có Table

Cú pháp: ALTER TABLE TableName

```
ADD [CONSTRAINT constraint_name]
PRIMARY KEY {(column [ASC |DESC][,...,n])}
[ON {filegroup|DEFAULT}]
```

Ví dụ: ALTER TABLE Sanpham
ADD CONSTRAINT Masp_PK
PRIMARY KEY (Cmasp)
EXEC Sp_helpconstraint Sanpham

- b) **Ràng buộc Unique:** Dùng để đảm bảo không có giá trị trùng ở các cột. Một cột hay sự kết hợp giữa các cột vốn không phải là khóa chính. Chấp nhận một hàng chứa giá trị Null. Một bảng có thể có nhiều Unique constraint.

Cách tạo:

Cách 1: Tạo cùng với lệnh tạo bảng

Cú pháp: CREATE TABLE TableName

```
(columnname datatype [...],[CONSTRAINT constraint_name]
UNIQUE [CLUSTERED|NONCLUSTERED]
{(column [ASC |DESC][,...,n])}
[WITH FILLFACTOR = fillfactor]
[ON {filegroup|DEFAULT}])
```

Ví dụ: CREATE TABLE jobs

```
(job_id smallint UNIQUE
CLUSTERED NOT NULL,
job_desc varchar(50) NOT NULL
DEFAULT 'New Position - title not
formalized yet')
```

Cách 2: Tạo sau khi đã có bảng

Cú pháp: ALTER TABLE TableName

```
ADD [CONSTRAINT constraint_name]
UNIQUE {(column [ASC | DESC][,...n])}
[ON {filegroup|DEFAULT}]
```

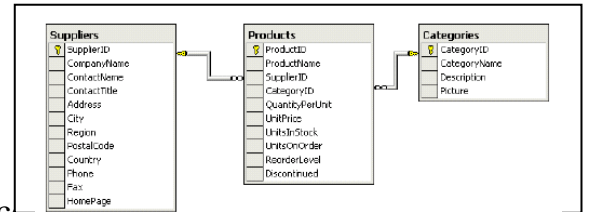
Ví dụ: ALTER TABLE Table3Unique

```
ADD col3 char(5) CONSTRAINT Table3_Unique UNIQUE
EXEC Sp_helpconstraint Table3
```

- c) **Ràng buộc Foreign key:** Khoá ngoại chỉ có thể tham chiếu đến một cột sau trong bảng chính:

- Là 1 cột hay 1 phần của khoá chính
- Là cột có ràng buộc unique
- Là cột có chỉ mục unique

Một bảng có thể có tối đa 253 khoá ngoại và có thể khác nhau.



Định nghĩa FOREIGN KEY CONSTRAINT khi tạo bảng

CREATE TABLE TableName

```
(columnName datatype [...],
[CONSTRAINT constraintName]
FOREIGN KEY[(column[,...n])]
REFERENCES ref_table [ ( ref_column [...n])] )
[ ON DELETE { CASCADE | NO ACTION } ]
[ ON UPDATE { CASCADE | NO ACTION } ]
[ NOT FOR REPLICATION]
```

- ON UPDATE|DELETE {CASCADE | NO ACTION}
- Xác định hành động cần phải thực hiện cho 1 hàng trong bảng đang tạo nếu hàng đó có quan hệ tham chiếu và hàng tham chiếu bị xoá khỏi bảng chính. Mặc định là NO ACTION.
- CASCADE: dùng để xác định là hàng sẽ bị cập nhật/xoá khỏi bảng tham chiếu nếu hàng đó bị cập nhật/xoá khỏi bảng chính
- NO ACTION: SQL Server sẽ đưa ra thông báo lỗi và việc xoá hàng trên bảng chính sẽ bị từ chối.

• Ví dụ 1

```
CREATE TABLE VITRI
(MaVt int Primary key, DiaChi varchar(40))
CREATE TABLE PhongBan
(
    Mapb int primary key,
    TenPb varchar(30),
    MaVT int REFERENCES VITRI(MaVt)
)
```

Định nghĩa FOREIGN KEY CONSTRAINT khi bảng đã tồn tại

```
ALTER TABLE TableName
    [WITH CHECK | WITH NOCHECK] ADD
    [CONSTRAINT constraintName]
    FOREIGN KEY[(column[,...n])]
    REFERENCES ref_table [ ( ref_column [,...n])] )
    [ ON DELETE { CASCADE | NO ACTION } ]
    [ ON UPDATE { CASCADE | NO ACTION } ]
    [ NOT FOR REPLICATION]
```

- **WITH CHECK:** trước khi tạo ràng buộc, SQL Server sẽ kiểm tra dữ liệu hiện có vi phạm ràng buộc hay không, nếu có sẽ không tạo constraint.
- **WITH NOCHECK:** tạo constraint mà không cần kiểm tra dữ liệu hiện có vi phạm ràng buộc hay không.

Ví dụ:

```
CREATE TABLE ChucVu
(Macv int primary key, tench varchar(30))
ALTER TABLE NhanVien1
    ADD CV int
ALTER TABLE Nhanvien1
    ADD CONSTRAINT Cv_FK Foreign key (Macv) REFERENCES
    Chucvu(Macv))
```

d) Xem trợ giúp ràng buộc

Cú pháp: Sp_helpConstraint ConstraintName

Ví dụ: Sp_helpConstraint NhanVien

e) Xóa ràng buộc constraint

Cú pháp:

ALTER TABLE <TenTable> DROP CONSTRAINT <ConstraintName>

Ví dụ: ALTER TABLE NhanVien1 DROP CONSTRAINT Cv_FK

PHẦN THỰC HÀNH

BÀI TẬP 1:

Cho mô tả nghiệp vụ của hệ thống quản lý bán hàng của một siêu thị như sau:

- Siêu thị bán nhiều sản phẩm khác nhau. Các sản phẩm được phân loại theo từng nhóm sản phẩm, mỗi nhóm sản phẩm có một mã nhóm (MANHOM) duy nhất, mỗi mã nhóm hàng xác định tên nhóm hàng (TENNHOM), tất nhiên một nhóm hàng có thể có nhiều sản phẩm. Mỗi sản phẩm được đánh một mã số (MASP) duy nhất, mỗi mã số sản phẩm xác định các thông tin về sản phẩm đó như : tên sản phẩm (TENS P), mô tả sản phẩm (MoTa), đơn vị tính (Đơn vị tính), đơn giá mua (ĐONGIA), số lượng tồn (SLTON).
- Siêu thị lấy hàng về từ nhiều nhà cung cấp khác nhau. Mỗi sản phẩm được lấy từ một nhà cung cấp. Hệ thống phải lưu trữ các thông tin về các nhà cung cấp hàng cho siêu thị. Mỗi nhà cung cấp có một mã số (MaNCC) duy nhất, mỗi mã

nhà cung cấp sẽ xác định tên nhà cung cấp (TenNCC), địa chỉ (Diachi), số điện thoại (Phone), số fax (Sofax) và địa chỉ mail (DCMail).

- Siêu thị bán hàng cho nhiều loại khách hàng khác nhau. Mỗi khách hàng có một mã khách hàng (MAKH) duy nhất, mỗi MAKH xác định được các thông tin về khách hàng như : họ tên khách hàng (HOTEN), địa chỉ (ĐIACHI), số điện thoại (ĐIENTHOAI), Ngày đăng ký thẻ thành viên (NgayDKThe), địa chỉ mail (DCMail), điểm tích lũy (DiemTL). Siêu thị chia khách hàng thành 3 loại khách hàng: VIP, TV, VL. Khách hàng VIP là những khách hàng đã là thành viên trên 5 năm và có tổng số hóa đơn mua hàng trên 100, khách hàng TV(thành viên là các khách hàng đã làm thẻ thành viên nhưng không đủ điều kiện của khách hàng VIP). Khách hàng vắng lai (VL) là khách hàng chưa có thẻ thành viên. Đối với khách hàng vắng lai thì MaKH sẽ được hệ thống tự cấp phát MaKH cho mỗi lần mua hàng do đó hệ thống không cần lưu các thông tin còn lại của khách hàng vắng lai.
 - Mỗi lần mua hàng, khách hàng có một hóa đơn. Mỗi hóa đơn bán hàng có một số hóa đơn (SOHD) duy nhất, mỗi số hóa đơn xác định được khách hàng và ngày lập hóa đơn (NGAYLAPHD), ngày giao hàng (NGAYGIAO) và nơi chuyển hàng (NoiChuyen). Ứng với mỗi hóa đơn siêu thị qui định như sau: Nếu khách hàng VIP sẽ được tặng 20% tổng tiền vào điểm tích lũy của khách hàng, nếu là thành viên là 10% tổng tiền, vắng lai thì không được tặng điểm tích lũy. Dựa vào điểm tích lũy siêu thị sẽ tặng phiếu quà tặng vào cuối năm cho các khách hàng. Mỗi đơn hàng có thể mua nhiều sản phẩm, Với mỗi sản phẩm trong một hóa đơn cho biết số lượng bán (SLBAN) của mặt hàng đó.
1. Dựa vào nghiệp vụ trên, xây dựng mô hình thực thể kết hợp ERD, sau đó chuyển qua lược đồ cơ sở dữ liệu cho hệ thống trên và xác định các ràng buộc khóa chính và khóa ngoại cho lược đồ CSDL.
 2. Xác định các qui tắc nghiệp vụ của hệ thống trên.
 3. Tại cửa sổ Query analyzer, thực hiện:
 - a. Dùng lệnh Create Table ... để tạo cấu trúc của các bảng sau trong CSDL

QLBH:

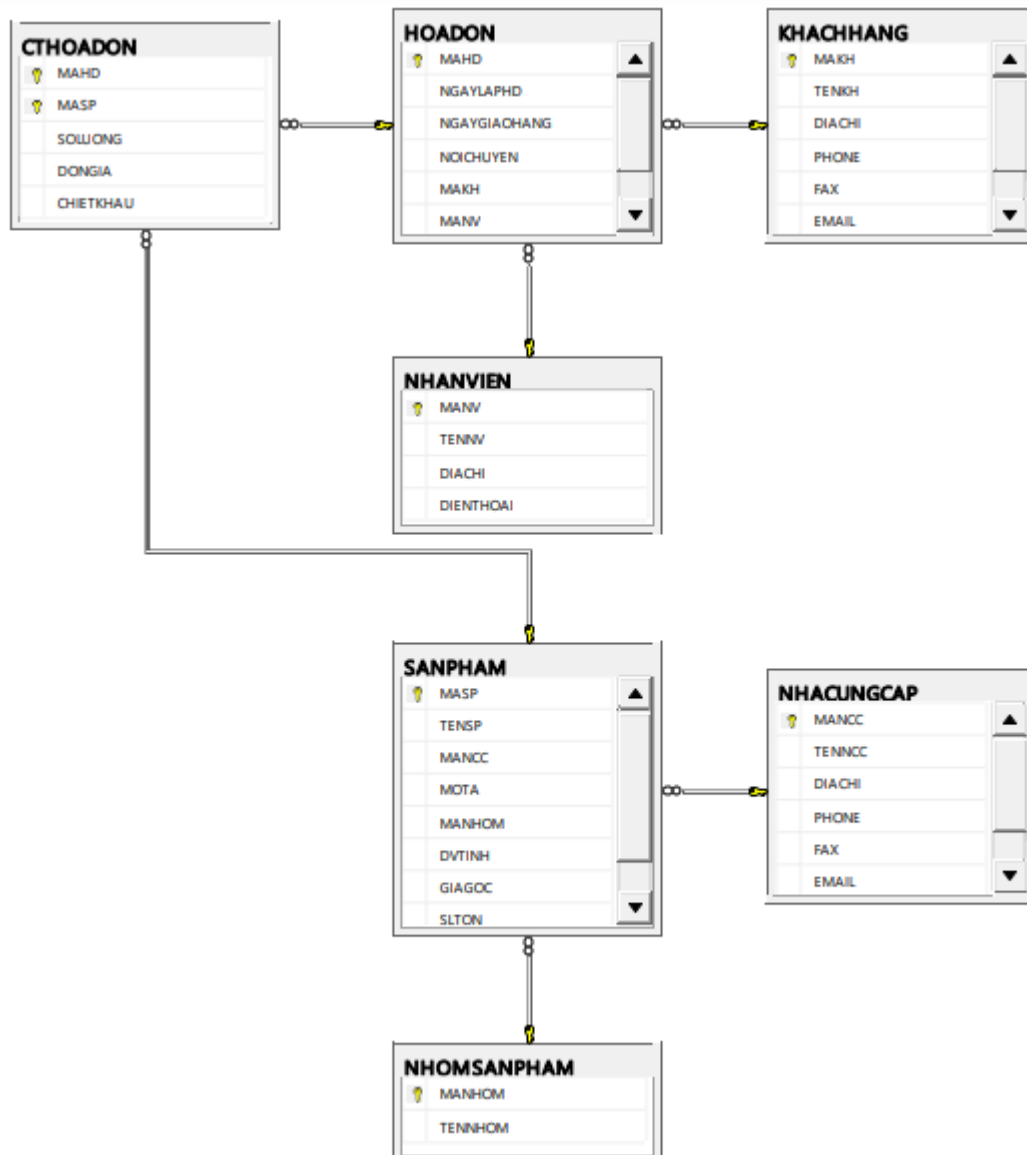
Lưu ý: Các bảng KHACHHANG, HOADON, CT_HOADON khai báo cột có Null hoặc Not Null, không cần khai báo khóa chính, khóa ngoại. Bảng NhómSanPham, SanPham, NhaCungCap yêu cầu tạo khóa chính và khóa ngoại trong lệnh Create Table luôn, các bảng còn lại thì dùng lệnh Alter Table để tạo khóa chính và khóa ngoại.

NhómSanPham		
MaNhóm	Int	Not null
TenNhóm	Nvarchar(15)	
SanPham		

MaSp	int	Not null
TenSp	nvarchar(40)	Not null
MaNCC	Int	
MoTa	nvarchar(50)	
MaNhom	Int	
Donvitinh	nvarchar(20)	
GiaGoc	Money	>0
SLTON	Int	>0
HoaDon		
MaHD	Int	Not null
NgayLapHD	DateTime	>=Ngày hiện hành Giá trị mặc định là ngày hiện hành
NgayGiao	DateTime	
Noichuyen	NVarchar(60)	Not Null
MaKh	Nchar(5)	
CT_HoaDon		
MaHD	Int	Not null
MaSp	int	Not null
Soluong	SmallInt	>0
Dongia	Money	
ChietKhau	Money	>=0
NhaCungCap		
MaNCC	Int	Not null
TenNcc	Nvarchar(40)	Not Null
Diachi	Nvarchar(60)	
Phone	NVarchar(24)	
SoFax	NVarchar(24)	
DCMail	NVarchar(50)	
KhachHang		
MaKh	NChar(5)	Not null
TenKh	Nvarchar(40)	Not null
LoaiKh	Nvarchar(3)	Chỉ nhập VIP, TV, VL
DiaChi	Nvarchar(60)	
Phone	NVarchar(24)	
SoFax	NVarchar(24)	
DCMail	NVarchar(50)	

DiemTL	Int	>=0
--------	-----	-----

- Dùng lệnh Alter Table ... khai báo các ràng khóa chính (Primary Key Constraint) ở các bảng còn lại KHACHHANG, HOADON, CT_HOADON.
- Dùng lệnh Alter Table ... khai báo các ràng khóa ngoại (Foreign Key Constraint) ở các bảng còn lại KHACHHANG, HOADON, CT_HOADON. .
- Dùng lệnh Alter Table ... khai báo các ràng buộc miền giá trị (Check Constraint) và ràng buộc giá trị mặc định cho các bảng trên
- Thêm cột LoaiHD vào bảng HOADON, LoaiHD có kiểu dữ liệu char(1), Chỉ nhập N(Nhập), X(Xuất), C(Chuyển từ cửa hàng này sang cửa hàng khác), T (Trả), giá trị mặc định là 'N'.
- Tạo ràng buộc cho bảng HoaDon với yêu cầu NgayGiao>=NgayLapHD
- Xây dựng sơ đồ quan hệ cho các bảng của cơ sở dữ liệu QLBH theo mẫu sau:



BÀI TẬP 2 (làm thêm tại nhà và nộp lại cho GV):

- Dùng T-SQL tạo CSDL Movies với các tham số sau:

- Tập tin Datafile có: Name: Movies_data; pathname: C:\Movies\Movies_data.mdf; Size: 25 MB; Maxsize: 40 MB; FileGrowth: 1 MB.
 - Tập tin Log file có: Name: Movies_log; pathname: C:\Movies\Movies_log.ldf; Size: 6 MB; Maxsize: 8 MB; FileGrowth: 1 MB.
2. Thực hiện, kiểm tra kết quả sau mỗi lần thực hiện:
- Thêm một Datafile thứ 2 có Name: Movies_data2; pathname: C:\Movies\Movies_data2.ndf; Size: 10 MB; thông số khác không cần chỉ định.
 - Lần lược cấu hình CSDL Movies với chế độ single_user, restricted user, multi user. Sau đó cài đặt lại là multi_user. Nhớ dùng lệnh để kiểm tra lại.
 - Tăng kích cỡ của data file thứ 2 từ 10 MB lên 15 MB. Kiểm tra lại.
 - Cấu hình CSDL về chế độ tự động SHRINK
 - Phát sinh tập tin Script từ CSDL Movies, tạo script cho tất cả các đối tượng kể cả database. Lưu vào đĩa với tên là Movies.SQL.
 - Xóa CSDL Movies
3. Mở tập tin Movies.SQL. Thực hiện:
- Bổ sung thêm câu lệnh tạo một filegroup tên là Data.
 - Hiệu chỉnh maxsize của tập tin transaction log thành 10 MB
 - Size của tập tin datafile thứ 2 thành 10 MB.
 - Cho datafile thứ 2 nằm trong filegroup có tên là Data.
 - Lưu tập tin script
 - Cho thực thi toàn bộ tập tin script.
 - Dùng sp_helpDB để kiểm tra sự tồn tại của Movies và các thông số của nó.
4. Các bảng có trong CSDL Movies là

Tên bảng	Nội dung lưu chứa
Movie	Danh sách các phim có trong cửa hàng
Customer	Thông tin khách hàng
Category	Danh sách các loại phim
Rental	Thông tin thuê phim
Rental_detail	Chi tiết thuê phim

Bạn hãy suy nghĩ xem mỗi bảng trên cần lưu những thông tin cụ thể nào (tức là các cột nào), kiểu dữ liệu ra sao? Khóa chính của từng bảng, mối quan hệ giữa các bảng, có những ràng buộc toàn vẹn nào?

5. Thực hiện định nghĩa các user-defined datatype sau vào trong CSDL Movies. Kiểm tra sau khi tạo.

Kiểu dữ liệu (Data type)	Mô tả dữ liệu (Description of data)
--------------------------	-------------------------------------

Movie_num	Int, không chấp nhận Null
Category_num	Int, không chấp nhận Null
Cust_num	Int, không chấp nhận Null
Invoice_num	Int, không chấp nhận Null

6. Thực hiện tạo các bảng vào CSDL Movies, nhớ kiểm tra lại cấu trúc bằng sp_help

Customer

Tên cột	kiểu dữ liệu	cho phép Null
Cust_num	cust_num IDENTITY(300,1)	No
Lname	varchar(20)	No
Fname	varchar(20)	No
Address1	varchar(30)	Yes
Address2	varchar(20)	Yes
City	varchar(20)	Yes
State	Char(2)	Yes
Zip	Char(10)	Yes
Phone	Varchar(10)	No
Join_date	Smalldatetime	No

Category

Tên cột	kiểu dữ liệu	cho phép Null
Category_num	category_num IDENTITY(1,1)	No
Description	Varchar(20)	No

Movie

Tên cột	kiểu dữ liệu	cho phép Null
Movie_num	Movie_num	No
Title	Cust_num	No
Category_Num	category_num	No
Date_purch	Smalldatetime	Yes
Rental_price	Int	Yes
Rating	Char(5)	Yes

Rental:

Tên cột	Kiểu dữ liệu	cho phép Null
Invoice_num	Invoice_num	No
Cust_num	Cust_num	No
Rental_date	Smalldatetime	No
Due_date	Smalldatetime	No

Rental:_Detail

Tên cột	Kiểu dữ liệu	cho phép Null
---------	--------------	---------------

Invoice_num	Invoice_num	No
Line_num	Int	No
Movie_num	Movie_num	No
Rental_price	Smallmoney	No

7. Thực hiện phát sinh tập tin script cho CSDL Movies với các lựa chọn sau, lưu với tên Table.sql:
 - All Tables, All user-defined data types
 - Generate the CREATE <object> command for each object
 - Generate the DROP <object> command for each object
8. Thực hiện tạo Diagram cho các bảng trong Movies. Bạn có tạo được không? Tại sao? Tạm thời lưu diagram với tên là Movies.
9. Thực hiện định nghĩa các khoá chính (Primary Key Constraint) cho các bảng như sau, nhớ kiểm tra các Constraint bằng lệnh sp_helpconstraint

Tên bảng	cột làm khóa	Tên của Primary Constraint
Movie	Movie_num	PK_movie
Customer	Cust_num	PK_customer
Category	Category_num	PK_category
Rental	Invoice_num	PK_rental

10. Thực hiện định nghĩa các khoá ngoại (Foreign Key Constraint) cho các bảng như sau, nhớ kiểm tra các Constraint bằng lệnh sp_helpconstraint

Tên bảng	cột làm khóa	Tên bảng được tham chiếu đến	Cột được tham chiếu	tên của Primary Constraint	Cascade
Movie	Category_num	Category	Category_num	FK_movie	
Rental	Cust_num	Customer	Cust_num	FK_rental	
Rental_detail	Invoice_num	Rental	Invoice_num	FK_detail_invoice	delete
Rental_detail	Movie_num	Movie	Movie_num	PK_detail_movie	

11. Mở lại Diagram có tên Movie, xem khóa chính, mối quan hệ giữa các bảng.
12. Thực hiện định nghĩa các giá trị mặc định (Default Constraint) cho các cột ở các bảng như sau, nhớ kiểm tra các Constraint bằng lệnh sp_helpconstraint

Tên bảng	Cột có giá trị Default	Giá trị Default	tên của Primary Constraint
Movie	Date_purch	Ngày hiện hành	DK_movie_date_purch
Customer	join_date	Ngày hiện hành	DK_customer_join_date
Rental	Rental_date	Ngày hiện hành	DK_rental_rental_date

Rental	Due_date	Ngày hiện hành + 2	DK_rental_due_date
--------	----------	-----------------------	--------------------

13. Thực hiện định nghĩa các miền giá trị (Check Constraint) cho các cột ở các bảng như sau, nhớ kiểm tra các Constraint bằng lệnh sp_helpconstraint

Tên bảng	cột có giá trị Default	Miền giá trị	tên của Primary Constraint
Movie	Rating	'G', 'PG', 'R', 'NC17', 'NR'	CK_movie
Rental	Due_date	>= Rental_date	CK_Due_date

14. Thực hiện phát sinh tập tin script cho các đối tượng trong CSDL Movie. Tên của tập tin là Constraint.sql. Với lựa chọn Script Primary Keys, Foreign Keys, Default, and Check Constraints.

BÀI TẬP TUẦN 4, 5, 6, 7, 8

Số tiết: 15

Mục tiêu:

- ✎ Thực hiện được thêm, xóa, sửa đơn giản bằng lệnh INSERT, UPDATE, DELETE
- ✎ Biết cách truy vấn dữ liệu bằng lệnh SELECT

TUẦN 4 (3 tiết)

PHẦN I: Lệnh INSERT – UPDATE – DELETE đơn giản

PHẦN LÝ THUYẾT

1) Cập nhật nội dung Table

a) Thêm dữ liệu vào table

Cú pháp: INSERT [INTO] <table_name> VALUES <values>

Ví dụ: `Insert into cthoadon values(2, 'b', 10,100)`

b) Sửa dữ liệu cho table

Cú pháp: UPDATE <table_name> SET <column_name = value>
WHERE <condition>

Ví dụ: `Update cthoadon set dongia = dongia+10*1.0/100*dongia`

c) Xóa dữ liệu cho table

Cú pháp: DELETE FROM <table_name> WHERE <condition>

Ví dụ: `Delete from cthoadon where sohd =2`

d) Câu lệnh TRUNCATE TABLE

Dùng để xóa các dòng của table

Nhanh hơn lệnh DELETE

Không dùng với Trigger

Cú pháp: **TRUNCATE TABLE** *table_name*

Ví dụ: `TRUNCATE TABLE NewProducts`

2) Truy vấn dữ liệu

Cú pháp:

```
SELECT [ALL | DISTINCT] [TOP n [WITH TIES]] select_list  
[ INTO new_table ]  
FROM table_source  
[ WHERE search_condition ]  
[ GROUP BY group_by_expression ]  
[ HAVING search_condition ]  
[ ORDER BY order_expression [ ASC | DESC ] ]
```

- *ORDER BY* : Sắp xếp
- *WHERE*: Điều kiện

- *GROUP BY*: Nhóm
- *HAVING*: Điều kiện nhóm

a) Truy vấn đơn giản

Hiển thị tất cả các cột có trong một Table

Cú pháp: `SELECT * FROM <tablename>`

Ví dụ: `SELECT * FROM Customers`

Chọn một vài cột trong một Table

`SELECT OrderID, OrderDate, CustomerID FROM Orders`

Truy vấn loại bỏ các dòng bị trùng:

Cú pháp: `Select Distinct`

Ví dụ:

`SELECT DISTINCT Order_Date as 'Date of Order' FROM Orders`

Truy vấn dùng các toán tử chuẩn trong biểu thức điều kiện:

Ví dụ 1:

`SELECT ProductID, UnitPrice
FROM Product_T
WHERE UnitPrice < 275`

Ví dụ 2:

`SELECT ProductID, ProductName, UnitPrice
FROM Products
WHERE ProductName like 'N%'
AND UnitPrice > 300`

b) Truy vấn trên nhiều bảng

Ví dụ:

`SELECT c.CustomerID, CompanyName, OrderID, OrderDate
FROM Customers C INNER JOIN Orders O
ON C.CustomerID = O.CustomerID`

c) Truy vấn seft join: Truy vấn trên một bảng liên kết với chính nó

Ví dụ:

`SELECT e.Firstname+' '+e.Lastname AS 'Employee' ,
m.Firstname+' '+m.Lastname AS 'Manager'
FROM Employees e JOIN Employees m
ON e.ManagerID = m.EmployeeID`

d) Lệnh Union: Union nối kết quả từ nhiều câu lệnh select

Ví dụ:

`SELECT Firstname+' '+Lastname AS name, Homephone
FROM Employees
UNION
SELECT Companyname, Phone
FROM Customers`

e) Truy vấn kết nhóm:

Các hàm tổng hợp - Aggregate Functions: tổng hợp thông tin từ nhiều bộ thành một bộ.

Chức năng grouping được sử dụng để tạo nhóm trước khi thực hiện tổng hợp dữ liệu.

Các hàm tổng hợp: COUNT, SUM, MAX, MIN, AVG.

Ví dụ:

```
SELECT SUM(quantity)
      AS 'Sum of Quantity' FROM [Order Details]
```

Mệnh đề GROUP BY: chỉ định các thuộc tính kết nhóm xuất hiện trong mệnh đề **select**, kết quả của hàm thống kê được áp dụng cho các bộ trong cùng một nhóm.

```
SELECTorderid, SUM(quantity+unitprice)
      AS 'Total' FROM [Order Details]
      GROUP BYorderid
```

Mệnh đề Having: xác định điều kiện lọc sau khi nhóm dữ liệu

```
SELECTorderid, SUM(quantity+unitprice)
      AS 'Total' FROM [Order Details]
      GROUP BYorderid
      HAVING SUM(quantity+unitprice)>10000000
```

f) Lệnh select into

Có thể tạo một bảng mới dựa vào kết quả của câu lệnh select.

Ví dụ:

```
SELECT C.CustomerID AS NameId, OrderID, OrderDate
      INTO Customer_Order
      FROM Customers C INNER JOIN Orders O
      ON C.CustomerID = O.CustomerID
      WHERE month(OrderDate) = 7
```

g) Truy vấn con - Nested Queries

Nested query là một query chứa một query khác, query được chứa bên trong gọi là *subquery*. Subquery thường xuất hiện trong mệnh đề WHERE của query. Ngoài ra Subquery cũng có thể xuất hiện trong mệnh đề FROM hoặc HAVING.

Các phép toán dùng trong nested query

- **IN:** so sánh một giá trị v với một tập giá trị V, kết quả là TRUE nếu v tồn tại trong V.

Ví dụ:

```
SELECT Orderid, orderdate FROM [Orders] WHERE Orderid IN (SELECT
      Orderid FROM [Customers] WHERE city = 'Lon don')
```

- **NOT IN:** so sánh một giá trị v với một tập giá trị V, kết quả là TRUE nếu v không tồn tại trong V

Ví dụ:

```
SELECT Orderid, orderdate FROM [Orders] WHERE Orderid NOT
      IN (SELECT Orderid FROM [Customers] WHERE city = 'Lon
      don')
```


- **ANY:** kết hợp với các phép toán **op** (>, >=, <, <=, and <>), kết quả là TRUE nếu và chỉ nếu các giá trị trong tập **v** thỏa mãn phép toán **op** với ít nhất là một giá trị trong **V**.

Ví dụ 1:

```
Select Productid, Productname, Unitprice from [Products]
Where Unitprice>ANY (Select Unitprice from [Products]
where ProductName like 'B%')
```

Ví dụ 2:

```
Select Productid, Productname, Unitprice from [Products]
Where Unitprice=ANY (Select Unitprice from [Products]
where ProductName like 'B%')
```

- **ALL:** kết hợp với các phép toán **op** (>, >=, <, <=, and <>), kết quả là TRUE nếu và chỉ nếu các giá trị trong tập **v** thỏa mãn phép toán **op** với tất cả giá trị trong **V**

Ví dụ:

```
Select Productid, Productname, Unitprice from [Products]
Where Unitprice>ALL (Select Unitprice from [Products]
where ProductName like 'B%')
```

- **EXISTS:** kiểm tra kết quả của subquery có rỗng hay không, exists trả về giá trị là TRUE nếu kết quả của subquery chứa ít nhất là một bộ giá trị.

Ví dụ:

```
Select * from [Products] s
where EXISTS ( SELECT * from [Order Details] c
where s.Productid=c.Productid)
```

- **NOT EXISTS:** trả về giá trị là TRUE nếu kết quả của subquery không chứa bộ giá trị nào.

Ví dụ:

```
Select * from [Products] s
where NOT EXISTS ( SELECT * from [Order Details] c
where s.Productid=c.Productid)
```

PHẦN THỰC HÀNH

Mở CSDL QL BH, thực hiện các lệnh sau:

1. Thêm vào mỗi Table 2 dòng dữ liệu thông qua cửa sổ Design. Dữ liệu sinh viên tự nghĩ.

Chú ý: Các ràng buộc của các Table

2. Dùng lệnh Insert thêm dữ liệu vào các bảng sau:

Chú ý: Nếu bảng nào có dữ liệu bạn hãy xóa hết các dữ liệu đó trước rồi mới nhập vào

Table NhomSanPham

IBM-35E18473... - dbo.NhomHang	
SQLQuery2.sql	
Manhom	TenNhom
1	Điện Tử
2	Gia Dụng
3	Dụng Cụ Gia Đình
4	Các Mặt Hàng Khác
NULL	NULL

Table NhaCungCap

IBM-35E18473...bo.NhaCungCap						
SQLQuery1.sql - ...inistrator (54))						
	Mancc	TenNcc	Diachi	Phone	SoFax	DCMail
	1	Công ty TNHH Nam Phương ...	1 Lê Lợi Phường 4 Quận Gò Vấp	083843456...	32343434 ...	NamPhuong@yahoo.com
	2	Công Ty Lan Ngọc	12 Cao bá Quát Quận 1 Tp. Hồ Chí Minh	086234567...	83434355 ...	LanNgoc@gmail.com
►*	NULL	NULL	NULL	NULL	NULL	NULL

Table SanPham

IBM-35E184733... - dbo.Sanpham								
SQLQuery1.sql - ...inistrator (54))								
	Masp	Tensp	DonVitinh	GiaGoc	SiTon	Manhom	MaNCC	MoTa
	1	Máy Tính	Cái	7000.0000	100	1	1	Máy Sony Ram 2GB
	2	Bàn phím	Cái	1000.0000	50	1	1	Bàn phím 101 phím
	3	Chuột	Cái	800.0000	150	1	1	Chuột không dây
	4	CPU	Cái	3000.0000	200	1	1	CPU
	5	USB	Cái	500.0000	100	1	1	8GB
	6	Lò Vi Sóng	Cái	1000000.0000	20	3	2	NULL
►*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Table KhachHang

IBM-35E18473...bo.KhachHang								
SQLQuery1.sql - ...inistrator (54))								
	Makh	Tenkh	DiaChi	DienThoai	LoaiKH	SoFax	DCMail	DiemTL
	KH1 ...	Nguyễn Thu Hằng	12 Nguyễn Du		VL	NULL	NULL	NULL
	KH2 ...	Lê Minh	34 Điện Biên Phủ	0123943455	TV	NULL	LeMinh@yahoo.com	100
►	KH3 ...	Nguyễn Minh Trung	3 Lê Lợi Quận Gò Vấp	098343434	VP	NULL	Trung@gmail.com	800
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Table HoaDon

IBM-35E18473... - dbo.HoaDon					
SQLQuery1.sql - ...inistrator (54))					
	Mahd	NgayLap	Makh	NgayGiao	Noichuyen
	1	2015-09-30 00:00:00.000	KH1	2015-10-05 00:00:00.000	Cửa Hàng ABC 3 Lý Chính Thắng Quận 3
	2	2015-07-29 00:00:00.000	KH2	2015-08-10 00:00:00.000	23 Lê Lợi Quận Gò Vấp
	3	2015-10-01 00:00:00.000	KH3	2015-10-01 00:00:00.000	2 Nguyễn Du Quận Gò Vấp
►*	NULL	NULL	NULL	NULL	NULL

Table CT_HoaDon

IBM-35E184733...ChiTietHoaDon		SQLQuery1.sql - ...inistrator (54))		
	Mahd	Masp	Dongia	Soluong
	1	1	8000.0000	5
	1	2	1200.0000	4
	1	3	1000.0000	15
	2	2	1200.0000	9
	2	4	800.0000	5
	3	2	3500.0000	20
	3	3	1000.0000	15
►*	NULL	NULL	NULL	NULL

3. Dùng lệnh Update sửa chữa dữ liệu theo yêu cầu sau

- Tăng đơn giá bán lên 5% cho các sản phẩm có mã là 2
- Tăng số lượng tồn lên 100 cho các sản phẩm có nhóm mặt hàng là 3 của nhà cung cấp có mã là 2
- Tăng điểm lũy lên 50 cho những khách hàng không phải là khách hàng vắng lai
- Cập nhật cột mô tả cho sản phẩm có tên là Lò vi sóng (dữ liệu cột mô tả sinh viên tự thêm)
- Tăng đơn giá gốc lên 2% cho những sản phẩm mà phần tên có chứa chữ u

4. Dùng lệnh Delete thực hiện các yêu cầu sau:

- Xóa các sản phẩm có SLTon <2
- Xóa các hóa đơn của khách hàng vắng lai
- Xóa khách hàng thuộc loại VIP mà có điểm tích lũy bằng 0

TUẦN 5 (3 TIẾT)

PHẦN 2: Lệnh SELECT

BÀI TẬP 1: LỆNH SELECT – TRUY VẤN ĐƠN GIẢN

Chọn CSDL hiện hành là NorthWind, tìm hiểu cấu trúc và dữ liệu của các bảng. Xây dựng sơ đồ quan hệ cho các bảng dưới. Sau đó dùng câu lệnh Select ... From ... Where ... để truy vấn dữ liệu như sau, với cấu trúc của các bảng như sau:

Products(ProductID, ProductName, SupplierID, UnitPrice, UnitInStock, ...)

Customers(CustomerID, CompanyName, Address, City, Region, Country, ...)

Employees(EmployeeID, LastName, FirstName, BirthDate, City, ...)

Orders(OderID, CustomerID, EmployeeID, OrderDate,...)

Order Details(OrderID, ProductID, UnitPrice, Quantity, Discount)

Suppliers(SupplierID, SupplierName,...)

Chú ý: Nếu câu nào mà bạn đánh đúng lệnh mà kết quả không có, bạn cho biết tại sao?

1. Liệt kê thông tin của tất cả các sản phẩm (Products)
2. Liệt kê danh sách các khách hàng (Customers). Thông tin bao gồm CustomerID, CompanyName, City, Phone.
3. Liệt kê danh sách các sản phẩm (Products). Thông tin bao gồm ProductId, ProductName, UnitPrice.
4. Liệt kê danh sách các nhân viên (Employees). Thông tin bao gồm EmployeeId, EmployeeName, Phone, Age. Trong đó EmployeeName được ghép từ LastName và FirstName; Age là tuổi được tính từ năm hiện hành (GetDate()) và năm của Birthdate.
5. Liệt kê danh sách các khách hàng (Customers) có ContactTitle bắt đầu bằng chữ O
6. Liệt kê danh sách khách hàng (Customers) ở thành phố LonDon, Boise và Paris
7. Liệt kê danh sách khách hàng (Customers) có tên bắt đầu bằng chữ V mà ở thành phố Lyon
8. Liệt kê danh sách các khách hàng (Customers) không có số fax
9. Liệt kê danh sách các khách hàng (Customers) có số Fax
10. Liệt kê danh sách nhân viên (Employees) có năm sinh ≤ 1960 .
11. Liệt kê danh sách các sản phẩm (Products) có từ 'Boxes' trong cột QuantityPerUnit.
12. Liệt kê danh sách các sản phẩm (Products) có Unitprice lớn hơn 10 và nhỏ hơn 15.
13. Liệt kê danh sách các hóa đơn (orders) có OrderDate được lập trong tháng 9 năm 1996. Được sắp xếp theo mã khách hàng, cùng mã khách hàng sắp xếp theo ngày lập hóa đơn giảm dần.
14. Liệt kê danh sách các hóa đơn (Orders) được lập trong quý 4 năm 1997. Thông tin gồm OrderID, OrderDate, Customerid, EmployeeID. Được sắp xếp theo tháng của ngày lập hóa đơn.
15. Liệt kê danh sách các hóa đơn (Orders) được lập trong trong ngày thứ 7 và chủ nhật của tháng 12 năm 1997. Thông tin gồm OrderID, OrderDate, Customerid, EmployeeID, WeekDayOfOrdinate (Ngày thứ mấy trong tuần).
16. Liệt kê danh sách các sản phẩm (Products) ứng với tiền tồn vốn. Thông tin bao gồm ProductId, ProductName, Unitprice, UnitsInStock, TotalAccount. Trong đó $TotalAccount = UnitsInStock * Unitprice$. Được sắp xếp theo TotalAccount giảm dần.
17. Liệt kê danh sách 5 customers có city bắt đầu 'M'.
18. Liệt kê danh sách 2 employees có tuổi lớn nhất. Thông tin bao gồm EmployeeID, EmployeeName, Age. Trong đó, EmployeeName được ghép từ LastName và FirstName; Age là năm hiện hành trừ năm sinh.
19. Liệt kê danh sách các Products có số lượng tồn nhỏ hơn 5.

20. Liệt kê danh sách các Orders gồm OrderId, Productid, Quantity, Unitprice, Discount, ToTal = Quantity * unitPrice – 20%*Discount.

TUẦN 6 (3 tiết)

BÀI TẬP 2: LỆNH SELECT – TRUY VẤN CÓ KẾT NỐI

1. Liệt kê các customer lập hóa đơn trong tháng 7/1997. Thông tin gồm CustomerID, CompanyName, Address, OrderID, Orderdate. Được sắp xếp theo Customerid, cùng CustomerID thì sắp xếp theo OrderDate giảm dần.
2. Liệt kê các customer có lập hóa đơn trong 15 ngày đầu tiên của 1/1997
3. Liệt kê danh sách các sản phẩm được giao vào ngày 16/7/1996
4. Liệt kê danh sách các hóa đơn của các Customers mua hàng trong tháng 4, 9 của năm 1997. Thông tin gồm Orderid, CompanyName, OrderDate, RequiredDate, được sắp xếp theo CompanyName, cùng Companyname thì theo OrderDate giảm dần.
5. Liệt kê danh sách các hóa đơn do nhân viên có Lastname là Fuller lập.
6. Liệt kê danh sách các Products do nhà cung cấp (supplier) có mã 1,3,6 bán được trong tháng quý 2 của năm 1997, được sắp xếp theo mã nhà cung cấp (SupplierID), cùng mã nhà cung cấp thì sắp xếp theo ProductID.
7. Liệt kê danh sách các Products có đơn giá bán bằng đơn giá mua.
8. Liệt kê danh sách các Products mà hóa đơn có OrderID là 10248 đã mua.
9. Liệt kê danh sách các Employers đã lập các hóa đơn trong tháng 7 của năm 1996.
10. Danh sách các sản phẩm bán trong ngày thứ 7 và chủ nhật của tháng 12 năm 1996, thông tin gồm [ProductID], [ProductName], OrderID, OrderDate, CustomerID, Unitprice, Quantity, ToTal= Quantity*UnitPrice. Được sắp xếp theo ProductID, cùng ProductID thì sắp xếp theo Quantity giảm dần.
11. Liệt kê bảng doanh thu của mỗi nhân viên theo từng hóa đơn trong năm 1996 gồm EmployeeID, EmployName, OrderID, Orderdate, Productid, quantity, unitprice, ToTal=quantity*unitprice.
12. Danh sách các đơn hàng sẽ được giao trong các thứ 7 của tháng 12 năm 1996.
13. Liệt kê danh sách các nhân viên chưa lập hóa đơn (dùng LEFT JOIN/RIGHT JOIN).
14. Liệt kê danh sách các sản phẩm chưa bán được (dùng LEFT JOIN/RIGHT JOIN).
15. Liệt kê danh sách các khách hàng chưa mua hàng lần nào (dùng LEFT JOIN/RIGHT JOIN).

TUẦN 7 (3 TIẾT)

BÀI TẬP 3: LỆNH SELECT – TRUY VẤN GOM NHÓM

1. Danh sách các orders ứng với tổng tiền của từng hóa đơn. Thông tin bao gồm OrdersId, OrderDate, TotalAccount. Trong đó TotalAccount là Sum của Quantity * Unitprice, kết nhóm theo OrderId.
2. Danh sách các orders ứng với tổng tiền của từng hóa đơn được lập ở thành phố (Shipcity) là 'Madrid'. Thông tin bao gồm OrdersId, OrderDate, TotalAccount. Trong đó TotalAccount là Sum của Quantity * Unitprice, kết nhóm theo OrderId.
3. Danh sách các Products có tổng số lượng lập hóa đơn lớn nhất. Thông tin gồm ProductID, ProductName, CountOfOrders.
4. Cho biết mỗi customers đã lập bao nhiêu hóa đơn. Thông tin gồm CustomerID, CompanyName, CountOfOrder. Trong đó CountOfOrder (tổng số hóa đơn) được đếm (Count) theo từng Customers.
5. Cho biết mỗi Employee đã lập được bao nhiêu hóa đơn, ứng với tổng tiền.
6. Liệt kê bảng lương của mỗi nhân viên theo từng tháng trong năm 1996 gồm EmployeeID, EmployName, Month_Salary, Salary = sum(quantity*unitprice)*10%. Được sắp xếp theo Month_Salary, cùng Month_Salary thì sắp xếp theo Salary giảm dần.
7. Danh sách các customer ứng với tổng tiền các hoá đơn, mà các hóa đơn được lập từ 31/12/1996 đến 1/1/1998 và tổng tiền các hóa đơn >20000.
8. Danh sách các customer ứng với tổng số hoá đơn, tổng tiền các hoá đơn, mà các hóa đơn được lập từ 31/12/1996 đến 1/1/1998 và tổng tiền các hóa đơn >20000. Thông tin được sắp xếp theo CustomerID, cùng mã thì sắp xếp theo tổng tiền giảm dần.
9. Danh sách các Category có tổng số lượng tồn (UnitsInStock) lớn hơn 300, đơn giá trung bình nhỏ hơn 25. Thông tin kết quả bao gồm CategoryID, CategoryName, Total_UnitsInStock, Average_Unitprice.
10. Danh sách các loại sản phẩm (Category) có tổng số sản phẩm (product) nhỏ hơn 10. Thông tin kết quả bao gồm CategoryID, CategoryName, TotalOfProducts. Được sắp xếp theo CategoryName, cùng tên loại theo TotalOfProducts giảm dần.
11. Liệt kê danh sách các sản phẩm bán trong quý 1 năm 1998 có tổng số lượng >200, thông tin gồm [ProductID], [ProductName], SumofQuatity
12. Danh sách các Customer ứng với tổng tiền của các hóa đơn ở từng tháng. Thông tin bao gồm CustomerID, CompanyName, Month_Year, Total. Trong đó Month_year là tháng và năm lập hóa đơn, Total là tổng của Unitprice* Quantity.
13. Cho biết Employees nào bán được nhiều tiền nhất trong 7 của năm 1997
14. Danh sách 3 khách có nhiều đơn hàng nhất của năm 1996.

15. Tính tổng số hóa đơn và tổng tiền của mỗi nhân viên đã bán trong tháng 3/1997, có tổng tiền >4000, thông tin gồm [EmployeeID],[LastName],[FirstName], countofOrderid, sumoftotal.

TUẦN 8 (3 TIẾT)

BÀI TẬP 4: LỆNH SELECT – TRUY VẤN CON SUBQUERY

1. Các product có đơn giá bán lớn hơn đơn giá bán trung bình của các product. Thông tin gồm ProductID, ProductName, Unitprice (Bên bảng Order Details).
2. Các product có đơn giá bán lớn hơn đơn giá bán trung bình của các product có ProductName bắt đầu là 'N'
3. Cho biết những sản phẩm có tên bắt đầu bằng chữ N và đơn giá bán > đơn giá bán của sản phẩm khác
4. Danh sách các products đã có khách hàng đặt hàng (tức là ProductId có trong Order Details). Thông tin bao gồm ProductId, ProductName, Unitprice
5. Danh sách các products có đơn giá nhập lớn hơn đơn giá bán nhỏ nhất của tất cả các Products
6. Danh sách các hóa đơn của những Customers mà Customers ở thành phố London và Madrid.
7. Danh sách các products có đơn vị tính có chữ Box và có đơn giá mua nhỏ hơn đơn giá bán trung bình của tất cả các Products.
8. Danh sách các Products có số lượng (Quantity) bán được lớn nhất.
9. Danh sách các Customer chưa từng lập hóa đơn (viết bằng ba cách: dùng NOT EXISTS, dùng LEFT JOIN, dùng NOT IN)
10. Cho biết các sản phẩm có đơn vị tính có chứa chữ box và có đơn giá bán cao nhất.
11. Danh sách các products có đơn giá bán lớn hơn đơn giá bán trung bình của các Products có ProductId<=5.
12. Cho biết những sản phẩm nào có tổng số lượng bán được lớn hơn số lượng trung bình bán ra.
13. Liệt kê danh sách các khách hàng mua các hóa đơn mà các hóa đơn này chỉ mua những sản phẩm có mã >=3
14. Liệt kê các sản phẩm có trên 20 đơn hàng trong quý 3 năm 1998, thông tin gồm [ProductID], [ProductName]
15. Liệt kê danh sách các sản phẩm Producers chưa bán được trong tháng 6 năm 1996
16. Liệt kê danh sách các Employees không lập hóa đơn vào ngày hôm nay
17. Liệt kê danh sách các Customers chưa mua hàng trong năm 1997
18. Tìm tất cả các Customers mua các sản phẩm có tên bắt đầu bằng chữ T trong tháng 7.
19. Danh sách các City có nhiều hơn 3 customer.

20. Bạn hãy đưa ra câu hỏi cho 3 câu truy vấn sau và cho biết sự khác nhau của 3 câu truy vấn này:

```
Select ProductId, ProductName, UnitPrice from [Products]
Where Unitprice>ALL (Select Unitprice from [Products]      where
ProductName like 'B%')
```

```
Select ProductId, ProductName, UnitPrice from [Products]
Where Unitprice>ANY (Select Unitprice from [Products]      where
ProductName like 'B%')
```

```
Select ProductId, ProductName, UnitPrice from [Products]
Where Unitprice=ANY (Select Unitprice from [Products]      where
ProductName like 'B%')
```

BÀI TẬP 5: LỆNH SELECT – CÁC LOẠI TRUY VẤN KHÁC

1. Kết danh sách các Customer và Employee lại với nhau. Thông tin gồm CodeID, Name, Address, Phone. Trong đó CodeID là CustomerID/EmployeeID, Name là Companyname/LastName + FirstName, Phone là Homephone.
2. Dùng lệnh SELECT...INTO tạo bảng HDKH_71997 cho biết tổng tiền khách hàng đã mua trong tháng 7 năm 1997 gồm CustomerID, CompanyName, Address, ToTal =sum(quantity*Unitprice)
3. Dùng lệnh SELECT...INTO tạo bảng LuongNV cho biết tổng lương của nhân viên trong tháng 12 năm 1996 gồm EmployeeID, Name là LastName + FirstName, Address, ToTal =sum(quantity*Unitprice)
4. Liệt kê các khách hàng có đơn hàng chuyển đến các quốc gia ([ShipCountry]) là 'Germany' và 'USA' trong quý 1 năm 1998, do công ty vận chuyển (CompanyName) Speedy Express thực hiện, thông tin gồm [CustomerID], [CompanyName] (tên khách hàng), tổng tiền.
5. Pivot Query

Tạo bảng dbo.HoaDonBanHang có cấu trúc sau

```
CREATE TABLE dbo.HoaDonBanHang
(
    orderid INT NOT NULL,
    orderdate DATE NOT NULL,
    empid INT NOT NULL,
    custid VARCHAR(5) NOT NULL,
    qty INT NOT NULL,
    CONSTRAINT PK_Orders PRIMARY KEY(orderid)
)
```


Chèn dữ liệu vào bảng

(30001, '20070802', 3, 'A', 10),
 (10001, '20071224', 2, 'A', 12),
 (10005, '20071224', 1, 'B', 20),
 (40001, '20080109', 2, 'A', 40),
 (10006, '20080118', 1, 'C', 14),
 (20001, '20080212', 2, 'B', 12),
 (40005, '20090212', 3, 'A', 10),
 (20002, '20090216', 1, 'C', 20),
 (30003, '20090418', 2, 'B', 15),
 (30004, '20070418', 3, 'C', 22),
 (30007, '20090907', 3, 'D', 30)


- Tính tổng Qty cho mỗi nhân viên. Thông tin gồm empid, custid
- Tạo bảng Pivot có dạng sau

empid	A	B	C	D
1	NULL	20	34	NULL
2	52	27	NULL	NULL
3	20	NULL	22	30

Gợi ý:

```
SELECT empid, A, B, C, D
FROM (SELECT empid, custid, qty
FROM dbo.Orders) AS D
PIVOT(SUM(qty) FOR custid IN(A, B, C, D)) AS P;
```

- Tạo 1 query lấy dữ liệu từ bảng dbo.HoaDonBanHang trả về số hóa đơn đã lập của nhân viên employee trong mỗi năm.
- Tạo bảng pivot hiển thị số đơn đặt hàng được thực hiện bởi nhân viên có mã 164, 198, 223, 231, and 233.

BÀI TẬP TUẦN 9**Số tiết: 3****Mục tiêu:** **Thêm, sửa, xóa dữ liệu nâng cao bằng các lệnh Insert, Update, Delete****Phần bài tập này dùng CSDL QLBH đã làm ở Tuần 2****BÀI TẬP 1: LỆNH INSERT****Cách 1: Dùng chức năng Import/Export (kiểm tra kết quả sau mỗi lần thực hiện)**

- 1) Tất cả các thông tin nhân viên có trong bảng Employees trong NorthWind ra thành tập tin NhanVien.txt.
- 2) Dữ liệu của các bảng Products, Orders, Order Details trong bảng NorthWind vào tập tin QLHH.MDB. Lưu ý: Tập tin QLHH.MDB phải tồn tại trên đĩa trước khi thực hiện Export.
- 3) Dữ liệu các bảng Products, Suppliers trong NorthWind ra thành tập tin SP_NCC.XLS
- 4) Các khách hàng có City là LonDon từ bảng Customers trong NorthWind ra thành tập tin KH.TXT.
- 5) Danh sách các sản phẩm ở Products trong NorthWind thành tập tin SanPham.TXT, thông tin cần lấy bao gồm ProductID, ProductName, QuantityPerUnit, Unitprice.
- 6) Các sản phẩm có SupplierID là **1** hoặc **2** hoặc **3** ở bảng Products trong NorthWind vào bảng SanPham trong QLBH. Lưu ý chỉ chọn những cột mà trong bảng sản phẩm cần.
- 7) Các nhà cung cấp có Country là **USA** ở bảng Suppliers trong NorthWin vào bảng NhaCungCap trong QLBH. Lưu ý: chỉ chọn những cột mà trong bản Nhacungcap cần.
- 8) Danh sách các nhân viên có trong tập tin Nhanvien.TXT vào bảng NhanVien

Cách 2: Dùng lệnh Insert...Select...

- 1) Xóa hết các dữ liệu đang có trong các Table của cơ sở dữ liệu QLBH bằng lệnh Delete.
- 2) Trong trường hợp nào thì không xóa được dữ liệu bảng SanPham khi chưa xóa dữ liệu bảng con của SanPham?
- 3) Nếu bạn muốn xóa bất kỳ Bảng cha thì xóa luôn các bảng quan hệ thì bạn phải làm gì? Bạn thực hiện một ví dụ minh họa
- 4) Dùng lệnh Insert thêm vào mỗi bảng của CSDL QLBH 5 record với nội dung do sinh viên tự nghĩ.
- 5) Dùng câu lệnh INSERT ... SELECT với các cột chọn cần thiết để thêm dữ liệu vào (nhớ kiểm tra kết quả sau mỗi lần thực hiện):

- a. Các khách hàng có trong bảng Customers trong NorthWind vào bảng KháchHang trong QL BH.
- b. Các sản phẩm có SupplierID từ 4 đến 29 ở bảng Products trong CSDL NorthWind vào bảng Sanpham trong QL BH.
- c. Danh sách tất cả các hoá đơn có OrderID nằm trong khoảng 10248 đến 10350 trong bảng Orders trong Northwind vào bảng HoaDon, các hoá đơn này được xem là hoá đơn xuất - tức LoaiHD là 'X'
- d. Danh sách tất cả các hoá đơn có OrderID nằm trong khoảng 10351 đến 10446 trong bảng Orders trong Northwind vào bảng HoaDon, các hoá đơn này được xem là hoá đơn nhập - tức LoaiHD là 'N'
- e. Danh sách tất cả các chi tiết hoá đơn có OderID nằm trong khoảng 10248 đến 10270 trong bảng Order Detail trong NorthWind vào bảng CT_HoaDon.

Chú ý: các ràng buộc khóa chính, khóa ngoại và các ràng buộc khác. Chỉ lấy các cột tương ứng với các bảng trong CSDL QL BH

BÀI TẬP 2: LỆNH UPDATE

1. Cập nhật đơn giá bán 100000 cho mã sản phẩm có tên bắt đầu bằng chữ T
2. Cập nhật số lượng tồn = 50% số lượng tồn hiện có cho những sản phẩm có đơn vị tính có chữ box
3. Cập nhật mã nhà cung cấp là 1 trong bảng NHACUNGCAP thành 100? Bạn có cập nhật được hay không?. Vì sao?.
4. Tăng điểm tích lũy lên 100 cho những khách hàng mua hàng trong tháng 7 năm 1997
5. Giảm 10% đơn giá bán cho những sản phẩm có số lượng tồn <10.
6. Cập nhật giá bán trong bảng CT_HoaDon bằng với đơn giá mua trong bảng SanPham của các sản phẩm do nhà cung cấp có mã là 4 hay 7.

BÀI TẬP 3: LỆNH DELETE

Lưu ý, việc xóa dữ liệu là công việc cần thận trọng, nên chúng ta ít thao tác trên CSDL với lệnh DELETE, trừ khi loại bỏ dữ liệu tạm. Nên **phần này yêu cầu chúng ta phải sao chép dữ liệu trước khi thực hiện các công việc sau:**

1. Xóa các hóa đơn được lập trong tháng 7 năm 1996. Bạn có thực hiện được không? Vì sao?
2. Xóa các hóa đơn của các khách hàng có loại là VL mua hàng trong năm 1996.
3. Xóa các sản phẩm chưa bán được trong năm 1996.
4. Xóa các khách hàng vắng lai. Lưu ý khi xóa xong thì phải xóa luôn các hóa đơn và các chi tiết của các hóa đơn này trong bảng HoaDon và bảng CTHoaDon
5. Tạo bảng HoaDon797 chứa các hóa đơn được lập trong tháng 7 năm 1997. Sau đó xóa toàn bộ dữ liệu của bảng này bằng lệnh Truncate

BÀI TẬP TUẦN 10**Số tiết: 3****Mục tiêu:** **View** **Index****1) View**

1. Tạo view vw_Products_Info hiển thị danh sách các sản phẩm từ bảng Products và bảng Categories. Thông tin bao gồm CategoryName, Description, ProductName, QuantityPerUnit, UnitPrice, UnitsInStock.
2. Tạo view List_Product_view chứa danh sách các sản phẩm dạng hộp (box) có đơn giá > 16, thông tin gồm ProductID, ProductName, UnitPrice, QuantityPerUnit, COUNT of OrderID
3. Tạo view vw_CustomerTotals hiển thị tổng tiền bán được từ mỗi khách hàng theo tháng và theo năm. Thông tin gồm CustomerID, YEAR(OrderDate) AS OrderYear, MONTH(OrderDate) AS OrderMonth, SUM(UnitPrice*Quantity). Xem lại cú pháp lệnh tạo view này.
4. Tạo view trả về tổng số lượng sản phẩm bán được của mỗi nhân viên (Employee) theo từng năm. Thông tin gồm EmployeeID, OrderYear, sumOfOrderQuantity. Yêu cầu sau khi tạo view, người dùng không xem được cú pháp lệnh đã tạo view này.
5. Tạo view ListCustomer_view chứa danh sách các khách hàng có trên 5 hóa đơn đặt hàng từ năm 1997 đến 1998, thông tin gồm mã khách (CustomerID), họ tên (CompanyName), Số hóa đơn (CountOfOrders).
6. Tạo view ListProduct_view chứa danh sách những sản phẩm nhóm Beverages và Seafood có tổng số lượng bán trong mỗi năm trên 30 sản phẩm, thông tin gồm CategoryName, ProductName, Year, SumOfOrderQuantity.
7. Tạo view vw_OrderSummary với từ khóa WITH ENCRYPTION gồm OrderYear (năm của ngày lập hóa đơn), OrderMonth (tháng của ngày lập hóa đơn), OrderTotal (tổng tiền, =UnitPrice*Quantity). Sau đó xem thông tin và trợ giúp về mã lệnh của view này
8. Tạo view vwProducts với từ khóa WITH SCHEMABINDING gồm ProductID, ProductName, Discount. Xem thông tin của View. Xóa cột Discount. Có xóa được không? Vì sao?
9. Tạo view vw_Customer với từ khóa WITH CHECK OPTION chỉ chứa các khách hàng ở thành phố London và Madrid, thông tin gồm: CustomerID, CompanyName, City.
 - a. Chèn thêm một khách hàng mới không ở thành phố London và Madrid thông qua view vừa tạo. Có chèn được không? Giải thích.

- b. Chèn thêm một khách hàng mới ở thành phố London và một khách hàng mới ở thành phố Madrid. Dùng câu lệnh select trên bảng Customers để xem kết quả .
10. Tạo 3 bảng lần lượt có tên là KhangHang_Bac, KhachHang_Trung, KhachHang_Nam, dùng để lưu danh sách các khách hàng ở ba miền, có cấu trúc như sau: MaKh, TenKH, DiaChi, KhuVuc. Trong đó,
- KhachHang_Bac có một Check Constraint là KhuVuc là 'Bac Bo'*
- KhachHang_Nam có một Check Constraint là KhuVuc là 'Nam Bo'*
- KhachHang_Trung có một Check Constraint là KhuVuc là 'Trung Bo'*
- Khoá chính là MaKH và KhuVuc .
- Tạo một partition view từ ba bảng trên, sau đó chèn mẫu tin tùy ý thông qua view. Kiểm tra xem mẫu tin được lưu vào bảng nào khi thêm/sửa/xóa dữ liệu vào view?
11. Lần lượt tạo các view sau, đặt tên tùy ý, sau khi tạo kiểm tra sự tồn tại và kết quả truy vấn từ view.
- Danh sách các sản phẩm có chữ 'Boxes' trong DonViTinh.
 - Danh sách các sản phẩm có đơn giá <10.
 - Các sản phẩm có đơn giá gốc lớn hơn hay bằng đơn giá gốc trung bình.
 - Danh sách các khách hàng ứng với các hóa đơn được lập. Thông tin gồm MaKH, TenKH, và tất cả các cột trong bảng HoaDon và CT_HoaDon.
- Trong các view ở câu trên view nào có thể INSERT, UPDATE, DELETE dữ liệu thông qua view được? Hãy Insert/Update/Delete thử dữ liệu tùy ý.

2) Index

1. Tạo chỉ mục dạng CLUSTERED cho bảng Orders với cột làm chỉ mục là Customerid. Xem trợ giúp về chỉ mục vừa tạo. Dùng lệnh select xem thông tin bảng orders.
 2. Tạo chỉ mục dạng NONCLUSTERED cho bảng Orders với cột làm chỉ mục là Employeeid. Xem trợ giúp về chỉ mục vừa tạo. Dùng lệnh select xem thông tin bảng orders.
- Nhận xét sự khác nhau giữa hai loại chỉ mục vừa tạo.
3. Thêm vào bảng Orders cột DiemTL. Tạo chỉ mục dạng unique cho cột DiemTL. Sau khi tạo chỉ mục này, nếu nhập dữ liệu cho 2 hóa đơn có cùng điểm tích lũy có được không? Giải thích
 4. Giả sử bạn có nhu cầu truy vấn thường xuyên câu lệnh sau:


```
SELECT *
FROM Orders
WHERE orderdate= getdate();
```

Bạn hãy thực hiện việc tạo chỉ mục thích hợp để việc truy vấn câu trên thực hiện nhanh hơn?
 5. Giả sử có nhu cầu truy vấn thường xuyên câu sau:

```
SELECT *  
FROM Products  
WHERE ProductID = 57
```

Bạn hãy thực hiện việc tạo chỉ mục thích hợp để việc truy vấn câu trên thực hiện nhanh hơn?