



FPT POLYTECHNIC



THỰC HỌC – THỰC NGHIỆP



Conceive Design Implement Operate

QUẢN TRỊ CƠ SỞ DỮ LIỆU VỚI SQL SERVER

BÀI 2: TẠO CSDL QUAN HỆ & NGÔN NGỮ TRUY VẤN T-SQL

- ① **Tạo cơ sở dữ liệu quan hệ**
- ① **Ngôn ngữ truy vấn T-SQL**





Tạo cơ sở dữ liệu quan hệ

- ❖ Kiểu dữ liệu
- ❖ Tạo cơ sở dữ liệu quan hệ
- ❖ Thiết lập các ràng buộc toàn vẹn



Ngôn ngữ truy vấn T-SQL

- ❖ Khái niệm biến
- ❖ Các lệnh T-SQL





PHẦN 1



Tạo cơ sở dữ liệu quan hệ

- ❖ Kiểu dữ liệu
- ❖ Tạo cơ sở dữ liệu quan hệ
- ❖ Thiết lập các ràng buộc toàn vẹn



Ngôn ngữ truy vấn T-SQL

- ❖ Khái niệm biến
- ❖ Các lệnh T-SQL



❑ Kiểu dữ liệu SQL là một thuộc tính xác định loại dữ liệu cho các đối tượng: biến, cột...

Nhóm	Mô Tả
Chuỗi	Các chuỗi dữ liệu ký tự
Số	Các số nguyên, số có dấu phẩy động, tiền tệ và các kiểu dữ liệu số khác.
Thời gian (ngày/giờ)	Ngày, giờ, hoặc cả ngày và giờ
Kiểu khác	Các giá trị chuỗi và nhị phân lớn, XML, dữ liệu hình học, dữ liệu địa lý, dữ liệu phân cấp

□ Kiểu dữ liệu chuỗi

Kiểu	Byte	Mô tả
char[(n)]	n	<ul style="list-style-type: none"> Chuỗi độ dài cố định. n là số ký tự, nằm giữa 1 và 8,000. Mặc định là 1.
varchar[(n)]	n	<ul style="list-style-type: none"> Chuỗi độ dài biến đổi. Số lượng byte được sử dụng để lưu trữ chuỗi phụ thuộc vào độ dài thực sự của chuỗi. n là số lượng ký tự lớn nhất giữa 1 và 8,000. Mặc định là 1.
nchar[(n)]	2n	<ul style="list-style-type: none"> Chuỗi độ dài cố định chứa dữ liệu ký tự Unicode. n là số ký tự, nằm giữa 1 và 4,000. Mặc định là 1.
nvarchar[(n)]	2n	<ul style="list-style-type: none"> Chuỗi độ dài biến đổi chứa dữ liệu ký tự Unicode. Số lượng byte được sử dụng để lưu trữ chuỗi phụ thuộc vào độ dài thực sự của chuỗi. n là số lượng ký tự lớn nhất giữa 1 và 4,000. Mặc định là 1.

❑ Kiểu dữ liệu chuỗi (tt)

- ❖ Kiểu **char** và **nchar** dùng để lưu trữ chuỗi có độ dài cố định.
- ❖ Kiểu **varchar**, **nvarchar** dùng để lưu trữ chuỗi có độ dài thay đổi.
- ❖ Kiểu **nchar**, **nvarchar** dùng để lưu trữ kí tự Unicode. Hệ thống hỗ trợ đa ngôn ngữ nên sử dụng kiểu này.

□ Kiểu dữ liệu số

Kiểu	Byte	Mô tả
bigint	8	Số nguyên lớn từ -9,223,372,036,854,775,808 tới 9,223,372,036,854,775,807.
int	4	Số nguyên từ -2,147,483,648 tới 2,147,483,647.
smallint	2	Số nguyên nhỏ từ -32,768 tới 32,767
tinyint	1	Số nguyên rất nhỏ từ 0 tới 255
bit	1	Số nguyên với các giá trị 1 hoặc 0.
decimal[(p[,s])]	5-17	Số thập phân với độ chính xác (p) và tỉ lệ (s) cố định từ $-10^{38} + 1$ tới $10^{38} - 1$
numeric[(p[,s])]	5-17	Đồng nghĩa với decimal

❑ Kiểu dữ liệu số (tt)

Kiểu	Byte	Mô tả
money	8	Giá trị tiền tệ với bốn vị trí thập phân từ -922,337,203,685,477.5808 tới 922,337,203,685,477.5807. Đồng nghĩa với decimal(19,4).
smallmoney	4	Giá trị tiền tệ với bốn vị trí thập phân từ -214,748.3648 tới 214,748.3647. Đồng nghĩa với decimal(10,4).
float[(n)]	4 - 8	Số dấu phẩy động độ chính xác kép từ -1.79E + 308 tới 1.79E + 308.
real	4	Số dấu phẩy động độ chính xác đơn từ -3.40E + 38 tới 3.40E + 38.

❑ Kiểu dữ liệu số

- ❖ Dùng lưu trữ các dữ liệu số, sử dụng trong tính toán
- ❖ Trong một số trường hợp thường sử dụng kiểu chuỗi làm kiểu dữ liệu thay thế khi lưu trữ dữ liệu số như:
số điện thoại, số chứng minh, mã ZipCode

❑ Kiểu dữ liệu thời gian

Kiểu	Byte	Mô tả
date	3	Chỉ có ngày (không chứa phần giờ). Định dạng YYYY-MM-DD có giá trị từ '0001-01-01' tới '9999-12-31'.
time(n)	3-5	Chỉ có giờ từ 00:00:00,0000000 tới 23:59:59,9999999, với độ chính xác 0,0000001 giây; n là số từ 0 tới 7 được sử dụng cho độ chính xác phân đoạn giây.
datetime		Kết hợp ngày và giờ. Định dạng YYYY-MM-DD hh:mm:ss từ '1000-01-01 00:00:00' tới '9999-12-31 23:59:59'
Datetime2(n)	6-8	Từ 1/1/0001 đến 31/12/9999 với độ chính xác 100 nano giây
datetimeoffset(n)	8-10	Mở rộng của kiểu datetime2, thêm múi giờ (time zone) được biểu diễn bởi giá trị từ -14 tới +14.

Các kiểu dữ liệu trong csdl QLDA

Tên thuộc tính	Diễn giải	Kiểu dữ liệu
HONV	Họ của nhân viên	Chuỗi (15)
TENLOT	Tên lót của nhân viên	Chuỗi (15)
TENNV	Tên của nhân viên	Chuỗi (15)
MANV, MA_NVIEN	Mã số nhân viên	Chuỗi (9)
NGSINH	Ngày tháng năm sinh nhân viên	Ngày (mm/dd/yy)
DCHI	Địa chỉ cư ngụ của nhân viên	Chuỗi (30)
PHAI	Giới tính	Chuỗi (3) { Nam, Nữ }
LUONG	Lương nhân viên	Số thực
MA_NQL	Mã người quản lý	Chuỗi (9)
TENPHG	Tên phòng ban	Chuỗi (15)
PHG, MAPHG	Mã phòng ban	Số nguyên
TRPHG	Mã người trưởng phòng	Chuỗi (9)
NG_NHANCHUC	Ngày nhận chức trưởng phòng	Ngày (mm/dd/yy)
DIADIEM	Địa điểm của phòng ban	Chuỗi (15)
TENTN	Tên thân nhân của nhân viên	Chuỗi (15)
QUANHE	Mối quan hệ với nhân viên	Chuỗi (15)
MADA, SODA	Mã số đề án Nguyên	Số nguyên
TENDA	Tên đề án	Chuỗi (15)
DDIEM_DA	Địa điểm diễn ra đề án	Chuỗi (15)
PHONG	Mã số phòng ban phụ trách đề án	Số nguyên
THOIGIAN	Thời gian làm việc 1 đề án của 1 nhân viên trong 1 tuần	Số thực
STT	Số thứ tự công việc trong một đề án	Số nguyên
TEN_CONG_VIEC	Tên công việc	Chuỗi (50)

SQL Server bao gồm 4 system databases và một hay nhiều user database. Các system databases bao gồm:

- ❑ **Cơ sở dữ liệu Master:** Chứa tất cả những thông tin cấp hệ thống (system-level information) bao gồm thông tin về các database khác trong hệ thống như vị trí của các data files, các login account và các thiết đặt cấu hình hệ thống của SQL Server (system configuration settings).
- ❑ **Cơ sở dữ liệu Model:** Database này đóng vai trò như một bảng mẫu (template) cho các database khác. Nghĩa là khi một user database được tạo ra thì SQL Server sẽ copy toàn bộ các system objects (tables, stored procedures...) từ Model database sang database mới vừa tạo.

- ❑ **Cơ sở dữ liệu Tempdb:** Chứa tất cả những table hay stored procedure được tạm thời tạo ra trong quá trình làm việc bởi user hay do bản thân SQL Server engine. Các table hay stored procedure này sẽ biến mất khi khởi động lại SQL Server hay khi ta disconnect.
- ❑ **Cơ sở dữ liệu Msdb:** được sử dụng cho SQL Server Agent để lập lịch các công việc và các cảnh báo (schedule alerts and jobs).

❑ Một database bao gồm tối thiểu hai file

- ❖ mdf: lưu trữ các đối tượng trong database như table, view, ...
 - Có thể bổ sung thêm các tập tin lưu trữ khác
 - Tổ chức tốt các tập tin lưu trữ giúp tăng tốc độ xử lý
- ❖ ldf: lưu trữ quá trình cập nhật/thay đổi dữ liệu
 - Hỗ trợ phục hồi dữ liệu
 - Hỗ trợ backup/restore dữ liệu

❑ Các thông số về kích thước file

- ❖ Initial size: kích thước ban đầu
- ❖ File growth: Kích thước tăng trưởng
- ❖ Maximum file size: kích thước tối đa.

❑ Tạo database

❖ Sử dụng công cụ Sql Server Management Studio

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'Object Explorer' pane displays the server hierarchy for 'DESKTOP-86LVI3P (SQL Server 14.0.1000.169 - DESK)'. The 'Databases' folder is selected, and a right-click context menu is open, with 'New Database...' highlighted. The 'New Database Wizard' dialog box is open in the foreground. It has a 'Script' button and a 'Help' icon. The 'Database name' field contains 'QLDA', the 'Owner' is '<default>', and the 'Use full-text indexing' checkbox is checked. Below, the 'Database files' section contains a table with the following data:

Logical Name	File Type	Filegroup	Initial Size (MB)	Autogrowth / Maxsize
QLDA	ROWS...	PRIMARY	8	By 64 MB, Unlimited
QLDA_log	LOG	Not Applicable	8	By 64 MB, Unlimited

❑ Tạo database

❖ Sử dụng câu query

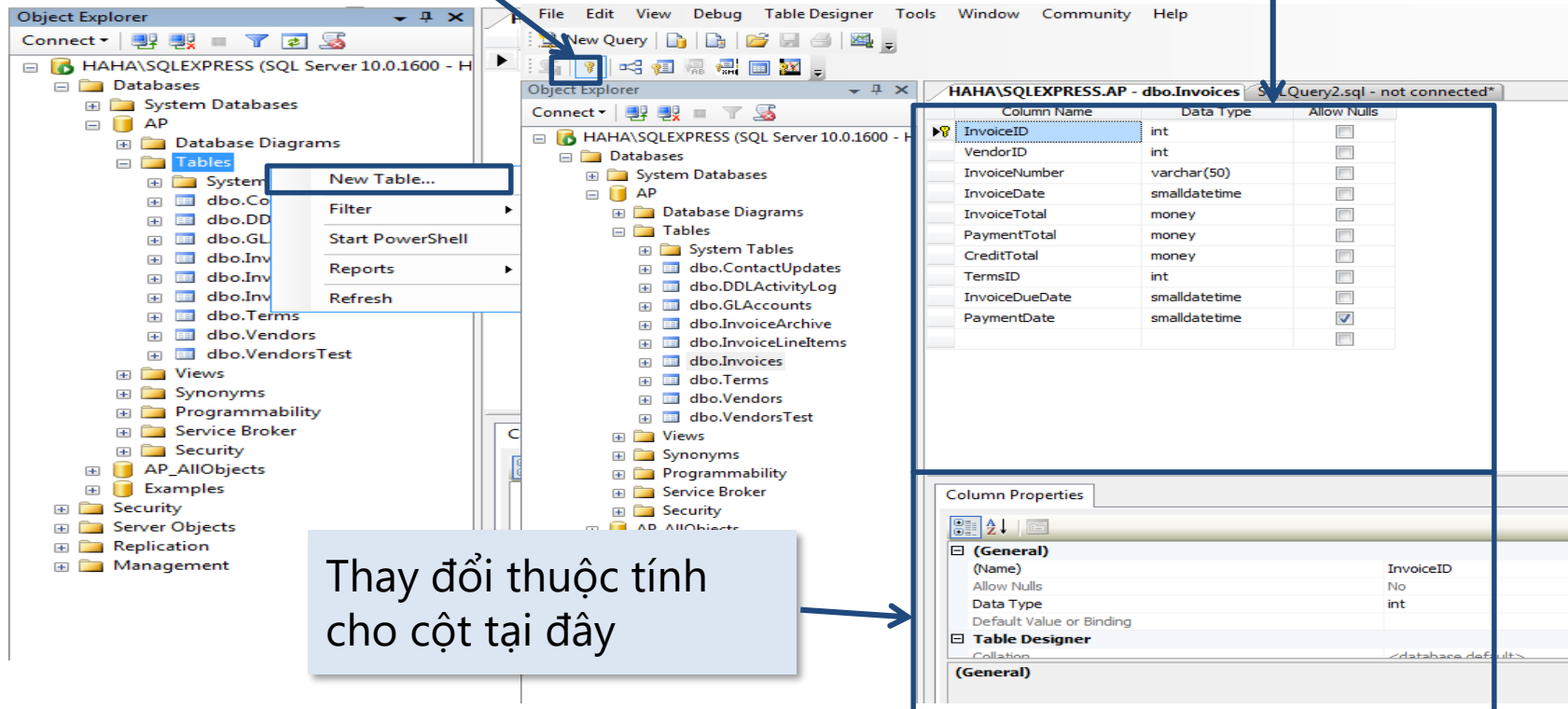
```
USE master ;
GO
CREATE DATABASE Music
ON
( NAME = Music_dat,
  FILENAME = 'C:\Program Files\Microsoft SQL
Server\MSSQL13.MSSQLSERVER\MSSQL\DATA\Music.mdf',
  SIZE = 10,
  MAXSIZE = 50,
  FILEGROWTH = 5 )
LOG ON
( NAME = Music_log,
  FILENAME = 'C:\Program Files\Microsoft SQL
Server\MSSQL13.MSSQLSERVER\MSSQL\DATA\Music_log.ldf',
  SIZE = 5MB,
  MAXSIZE = 25MB,
  FILEGROWTH = 5MB ) ;
GO
```

❑ Tạo bảng

❖ Sử dụng công cụ Sql Server Management Studio

Thiết lập khóa chính cho cột

Nhập tên các cột, kiểu dữ liệu, chấp nhận thuộc tính NULL tại đây



Object Explorer

Connect | File Edit View Debug Table Designer Tools Window Community Help

New Query

Object Explorer

HAHA\SQLEXPRESS (SQL Server 10.0.1600 - H)

Databases

System Databases

AP

Database Diagrams

Tables

System Tables

dbo.ContactUpdates

dbo.DDLActivityLog

dbo.GLAAccounts

dbo.InvoiceArchive

dbo.InvoiceLineItems

dbo.Invoices

dbo.Terms

dbo.Vendors

dbo.VendorsTest

Views

Synonyms

Programmability

Service Broker

Security

AP_AllObjects

Examples

Security

Server Objects

Replication

Management

New Table...

Filter

Start PowerShell

Reports

Refresh

HAHA\SQLEXPRESS.AP - dbo.Invoices

Column Name	Data Type	Allow Nulls
InvoiceID	int	<input checked="" type="checkbox"/>
VendorID	int	<input type="checkbox"/>
InvoiceNumber	varchar(50)	<input type="checkbox"/>
InvoiceDate	smalldatetime	<input type="checkbox"/>
InvoiceTotal	money	<input type="checkbox"/>
PaymentTotal	money	<input type="checkbox"/>
CreditTotal	money	<input type="checkbox"/>
TermsID	int	<input type="checkbox"/>
InvoiceDueDate	smalldatetime	<input type="checkbox"/>
PaymentDate	smalldatetime	<input checked="" type="checkbox"/>

Column Properties

(General)

(Name) InvoiceID

Allow Nulls No

Data Type int

Default Value or Binding

Table Designer

Collation <database defaults>

(General)

Thay đổi thuộc tính cho cột tại đây

❑ Tạo bảng

❖ Sử dụng câu query

```
CREATE TABLE table_name (  
    column1 datatype,  
    column2 datatype,  
    column3 datatype,  
    ....  
);
```

❖ Ví dụ:

```
CREATE TABLE PhongBan(  
    MaPhong int IDENTITY(1,1) NOT NULL PRIMARY KEY,  
    TenPhong nvarchar(50) NOT NULL  
)
```

❑ Chỉnh sửa bảng

```
ALTER TABLE table_name  
ADD column_name datatype;
```

```
ALTER TABLE table_name  
DROP COLUMN column_name;
```

```
ALTER TABLE table_name  
ALTER COLUMN column_name datatype;
```

```
ALTER TABLE Customers  
ADD Email varchar(255);
```

```
ALTER TABLE Customers  
DROP COLUMN Email;
```

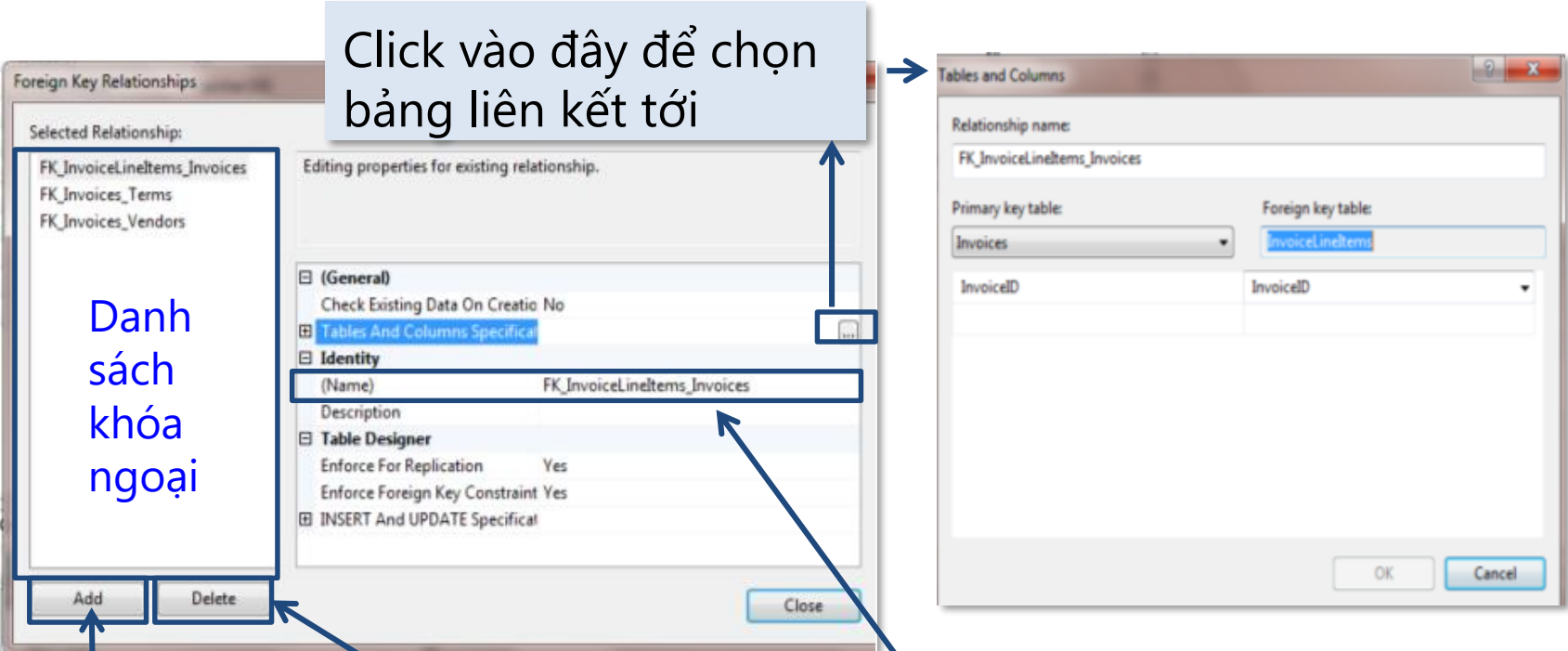
```
ALTER TABLE Persons  
ALTER COLUMN DateOfBirth year;
```

❑ Xóa bảng

```
DROP TABLE table_name;
```

❑ Thiết lập khóa ngoại

Click vào button **Relationships** trên thanh công cụ



Click vào đây để chọn bảng liên kết tới

Danh sách khóa ngoại

Thêm khóa ngoại mới

Xóa khóa ngoại được chọn

Chỉ định tên cho khóa ngoại tại đây

Foreign Key Relationships

Selected Relationship:

- FK_InvoiceLineItems_Invoices
- FK_Invoices_Terms
- FK_Invoices_Vendors

Editing properties for existing relationship.

(General)

Check Existing Data On Creation: No

Tables And Columns Specific

Identity

(Name) FK_InvoiceLineItems_Invoices

Description

Table Designer

Enforce For Replication: Yes

Enforce Foreign Key Constraint: Yes

INSERT And UPDATE Specific

Close

Tables and Columns

Relationship name: FK_InvoiceLineItems_Invoices

Primary key table: Invoices

Foreign key table: InvoiceLineItems

InvoiceID

InvoiceID

OK

Cancel

❑ Thiết lập khóa ngoại (tt)

```
CREATE TABLE NhanVien(  
    MaNV int NOT NULL PRIMARY KEY,  
    MaPhg int FOREIGN KEY REFERENCES PhongBan(MaPhong)  
)
```

❑ Sinh viên tham khảo các lệnh đầy đủ tại:

https://www.w3schools.com/sql/sql_foreignkey.asp



- ❖ Tạo cơ sở dữ liệu QLDA có 2 table là PhongBan và NhanVien với khóa chính, khóa ngoại theo sơ đồ (dùng tool và lệnh):





PHẦN 2



Tạo cơ sở dữ liệu quan hệ

- ❖ Kiểu dữ liệu
- ❖ Tạo cơ sở dữ liệu quan hệ
- ❖ Thiết lập các ràng buộc toàn vẹn



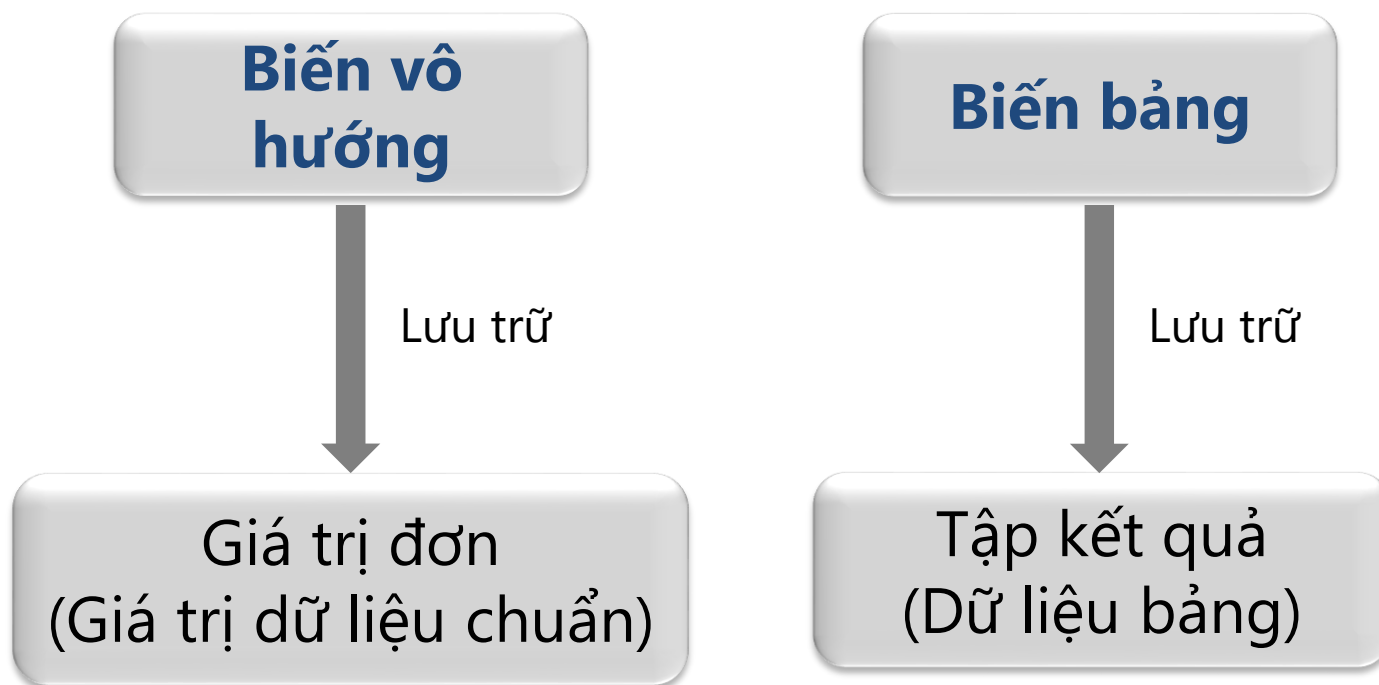
Ngôn ngữ truy vấn T-SQL

- ❖ Khái niệm biến
- ❖ Các lệnh T-SQL



❑ Khái niệm biến

- ❖ Biến (Variable) dùng để lưu trữ các giá trị tạm thời trong quá trình thực thi chương trình.
- ❖ Phân loại biến



□ Khái niệm biến (tt)

❖ Biến vô hướng

→ Khai báo

Cú pháp

***DECLARE @<tên biến 1> <kiểu dữ liệu>
[, @<tên biến 2> <kiểu dữ liệu>]...***

→ Gán giá trị

Cú pháp của câu lệnh SET

SET @<tên biến> = <biểu thức>

Sử dụng để gán giá trị cho một biến

Cú pháp câu lệnh SELECT

SELECT @<tên biến> = <biểu thức cột>

SELECT có thể dùng để gán giá trị cho 2 hoặc nhiều biến.

→ Truy xuất giá trị biến:

- **SELECT @<tên biến>**

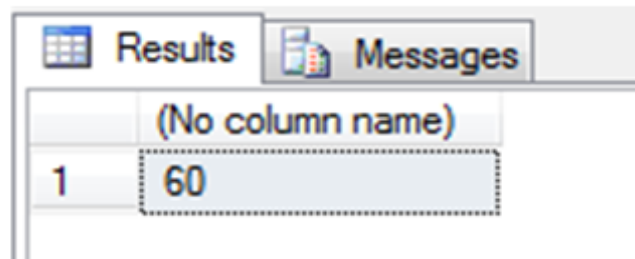
❑ Khái niệm biến (tt)

❖ Biến vô hướng (tt)

Ví dụ: Tính diện tích hình chữ nhật

```
DECLARE @ChieuDai int, @ChieuRong int, @DienTich int
SET @ChieuDai=10  -- Gán cho biến ChieuDai là 10
SET @ChieuRong=6  -- Gán cho biến ChieuRong là 6
SET @DienTich=@ChieuDai*@ChieuRong  -- Tính giá trị của
                                      DienTich
```

```
SELECT @DienTich  -- Hiển thị kết quả
```



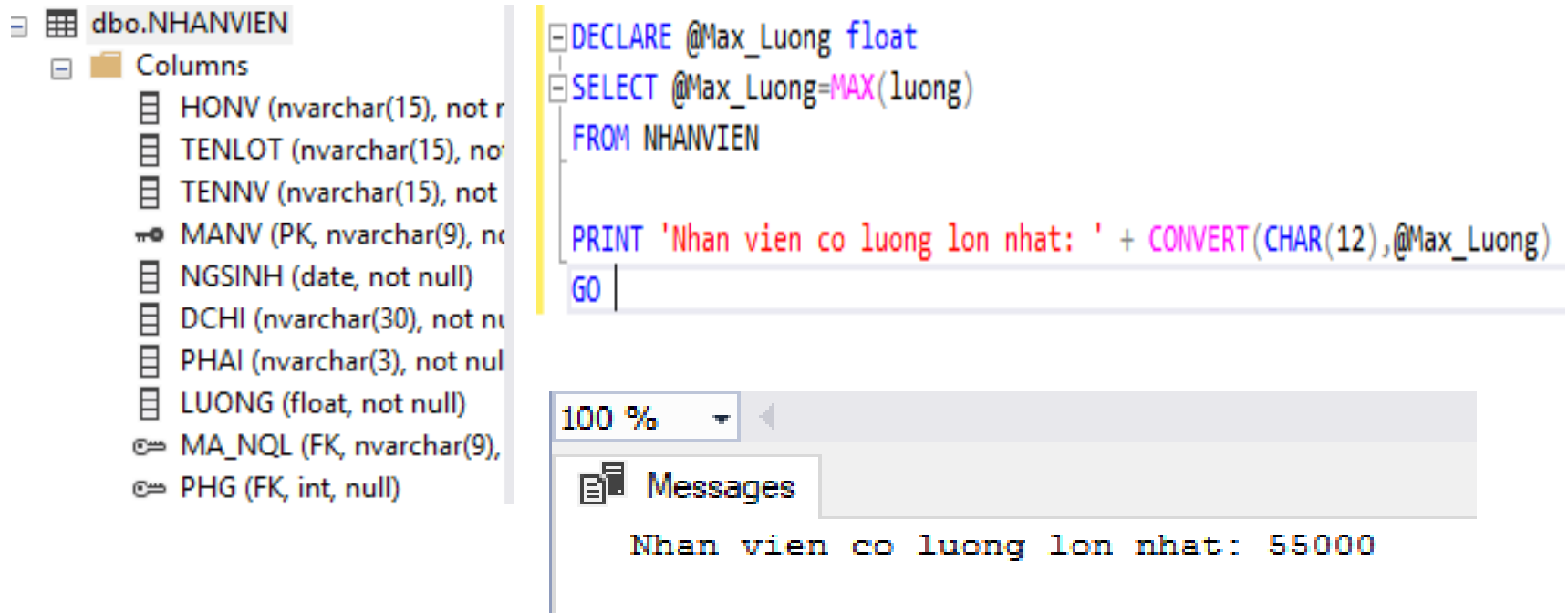
The screenshot shows a SQL Server interface with two tabs: 'Results' and 'Messages'. The 'Results' tab is active, displaying a table with one column labeled '(No column name)' and one row containing the value '60'.

	(No column name)
1	60

❑ Khái niệm biến(tt)

❖ Biến vô hướng (tt)

Ví dụ: Cho bảng NhanVien. Tìm lương lớn nhất của nhân viên



The screenshot displays the SQL Server Enterprise Manager interface on the left, showing the database structure for 'dbo.NHANVIEN'. The 'Columns' folder is expanded, listing various attributes including 'HONV', 'TENLOT', 'TENNV', 'MANV' (primary key), 'NGSINH', 'DCHI', 'PHAI', 'LUONG' (float, not null), 'MA_NQL' (foreign key), and 'PHG' (foreign key).

The central Query Editor window contains the following T-SQL code:

```
DECLARE @Max_Luong float
SELECT @Max_Luong=MAX(luong)
FROM NHANVIEN

PRINT 'Nhan vien co luong lon nhat: ' + CONVERT(CHAR(12),@Max_Luong)
GO
```

The bottom status bar shows the query execution results in the 'Messages' pane:

```
Nhan vien co luong lon nhat: 55000
```

❑ Khái niệm biến (tt)


- ❖ Biến bảng: lưu trữ dữ liệu dạng bảng


```
DECLARE @<tên biến bảng> TABLE  
(<tên cột 1> <kiểu dữ liệu> [<Các thuộc tính cột>]  
[, <tên cột 2> <kiểu dữ liệu> [<Các thuộc tính cột>]]...  
[, <Các thuộc tính bảng>])
```











- ❖ Có thể tham chiếu đến biến bảng trong câu lệnh **SELECT, INSERT, UPDATE, DELETE**
- ❖ **Ngoại lệ:** Không thể sử dụng câu lệnh **SELECT INTO** với biến bảng

❑ Khái niệm biến (tt)

❖ Biến bảng: Tạo biến chứa các nhân viên ở HCM

☰  dbo.NHANVIEN

☰  Columns

-  HONV (nvarchar(15), not null)
-  TENLOT (nvarchar(15), not null)
-  TENNV (nvarchar(15), not null)
-  MANV (PK, nvarchar(9), not null)
-  NGSINH (date, not null)
-  DCHI (nvarchar(30), not null)
-  PHAI (nvarchar(3), not null)
-  LUONG (float, not null)
-  MA_NQL (FK, nvarchar(9), null)
-  PHG (FK, int, null)

--Khai báo biến bảng chứa 2 thuộc tính

```
DECLARE @NhanVien_HCM Table ( MaNV nvarchar(9), TenNV nvarchar(15))
```

-- Chèn dữ liệu vào biến bảng

```
INSERT INTO @NhanVien_HCM
SELECT MaNV, TenNV
From NHANVIEN
Where DCHI like '%HCM'
```

❖ Truy cập Biến bảng

```
SELECT * FROM @NhanVien_HCM
-- Hoặc truy cập đến từng trường
SELECT MaNV, HoTen
From @NhanVien_HCM
```

Results			Messages	
	MaNV	TenNV		
1	005	Tùng		
2	007	Hành		
3	008	Quang		
4	009	Tiền		

❑ Khái niệm biến (tt)

- ❖ Biến bảng: Chèn dữ liệu vào biến bảng

```
DECLARE @NhanVien Table
( MaNV int, HoTen char (50) )
-- Chèn dữ liệu vào biến bảng
INSERT INTO @NhanVien VALUES (01, 'Phan Viet The')
-- Truy cập dữ liệu từ biến bảng
SELECT * FROM @NhanVien
```

- ❖ Sửa bản ghi biến bảng

```
DECLARE @NhanVien Table
( MaNV int, HoTen char (50) )
-- Chèn dữ liệu vào biến bảng
UPDATE @NhanVien
SET HoTen='Linda1'
WHERE MaNV=1
```

❑ Transact-SQL (còn gọi là T-SQL)

- ❖ T-SQL là một ngôn ngữ lập trình Database hướng thủ tục độc quyền của Microsoft sử dụng trong SQL Server.
- ❖ T-SQL tổ chức theo từng khối lệnh
- ❖ Một khối lệnh có thể lồng bên trong một khối lệnh khác
- ❖ Một khối lệnh bắt đầu bởi BEGIN và kết thúc bởi END, bên trong khối lệnh có nhiều lệnh, và các lệnh ngăn cách nhau bởi dấu chấm phẩy.

```
1 BEGIN
2     -- Khai báo biến
3     -- Các câu lệnh T-SQL
4 END;
```

□ Transact-SQL

Begin

```

Declare @v_Result Int;
Declare @v_a Int = 50;
Declare @v_b Int = 100;

Print 'v_a= ' + Cast(@v_a as varchar(15));
Print 'v_b= ' + Cast(@v_b as varchar(15));

Set @v_Result = @v_a + @v_b;

Print 'v_Result= ' + Cast(@v_Result as varchar(15));

End;

```

100 %

Messages

```

v_a= 50
v_b= 100
v_Result= 150

```



- ❖ Sử dụng T-SQL áp dụng cách khai báo biến để thực hiện các truy vấn:
 - Cho biết lương cao nhất nhân viên có thể nhận
 - Cho biết họ tên nhân viên (HONV, TENLOT, TENNV) có mức lương trên mức lương trung bình của phòng "Nghiên cứu"
 - Với các phòng ban có mức lương trung bình trên 30,000, liệt kê tên phòng ban và số lượng nhân viên của phòng ban đó

☑ **Tạo cơ sở dữ liệu quan hệ**

- ❖ Kiểu dữ liệu
- ❖ Tạo cơ sở dữ liệu quan hệ
- ❖ Thiết lập các ràng buộc toàn vẹn

☑ **Ngôn ngữ truy vấn T-SQL**

- ❖ Khái niệm biến
- ❖ Các lệnh T-SQL



thank
you!