

# Quản lý dữ liệu bằng cách sử dụng **Microsoft SQL Server**

**Phiên: 5**

**Transact-SQL**

Chỉ dành cho Trung tâm Aptech



# Objectives

- Giải thích Transact-SQL
- Liệt kê các danh mục khác nhau của câu lệnh Transact-SQL
- Giải thích các kiểu dữ liệu khác nhau được hỗ trợ bởi Transact-SQL
- Giải thích các yếu tố ngôn ngữ Transact-SQL
- Giải thích các tập hợp và logic vị từ
- Mô tả thứ tự logic của các toán tử trong câu lệnh SELECT

Chỉ dành cho Trung tâm Aptech



# Introduction

- SQL là ngôn ngữ phổ biến được sử dụng trong thế giới cơ sở dữ liệu.
- Hầu hết các sản phẩm RDBMS hiện đại sử dụng một số loại phương ngữ SQL làm ngôn ngữ truy vấn chính của chúng.
- SQL có thể được sử dụng để tạo hoặc hủy các đối tượng, chẳng hạn như bảng, trên máy chủ cơ sở dữ liệu và để thực hiện những việc với các đối tượng đó, chẳng hạn như đưa dữ liệu vào chúng hoặc truy vấn dữ liệu.
- Transact-SQL là cách triển khai SQL tiêu chuẩn của Microsoft.
- Thường được gọi là T-SQL, ngôn ngữ này thực hiện một cách chuẩn hóa để giao tiếp với cơ sở dữ liệu.
- Ngôn ngữ Transact-SQL là sự cải tiến của SQL, ngôn ngữ cơ sở dữ liệu quan hệ tiêu chuẩn của Viện Tiêu chuẩn Quốc gia Hoa Kỳ (ANSI).
- Nó cung cấp một ngôn ngữ toàn diện hỗ trợ xác định bảng, chèn, xóa, cập nhật và truy cập dữ liệu trong bảng.



# Transact-SQL 1-2

Transact-SQL là một ngôn ngữ mạnh mẽ cung cấp các tính năng như kiểu dữ liệu, đối tượng tạm thời và các thủ tục được lưu trữ mở rộng.

Ngôn ngữ Transact-SQL trong SQL Server 2012 cung cấp hiệu suất được cải thiện, tăng chức năng và các tính năng nâng cao.



## Transact-SQL 2-2

Đoạn mã sau hiển thị câu lệnh Transact-SQL, LỰA CHỌN, được sử dụng để truy xuất tất cả hồ sơ của nhân viên với 'Kỹ sư thiết kế' như JobTitle từ Nhân viênbàn.

```
CHỌN ID đăng nhập  
TỪ nhân viên  
WHERE JobTitle = 'Kỹ sư thiết kế'
```

- Hình sau cho thấy kết quả của LỰA CHỌN tuyên bố:

	LoginID
1	josi
2	josef
3	sharon

- Transact-SQL bao gồm nhiều phần tử cú pháp được sử dụng bởi hoặc ảnh hưởng đến hầu hết các câu lệnh.
- Các phần tử này bao gồm kiểu dữ liệu, vị từ, hàm, biến, biểu thức, điều khiển luồng, nhận xét và dấu phân tách hàng loạt.



# Data Definition Language (DDL)

DDL được sử dụng để xác định và quản lý tất cả các thuộc tính và thuộc tính của cơ sở dữ liệu, bao gồm bố cục hàng, định nghĩa cột, cột chính, vị trí tệp và chiến lược lưu trữ.

Đối với mỗi đối tượng, thường có CREATE, ALTER, và LÀM RƠI câu lệnh (chẳng hạn như, TẠO BẢNG, BẢNG ALTER, và DROP BẢNG).

**TẠO RA**  
tên của môn học

**ALTER**  
tên của môn học

**LÀM RƠI**  
tên của môn học

Trong các câu lệnh DDL, tên của môn học có thể là một bảng, dạng xem, bộ kích hoạt, thủ tục được lưu trữ, v.v.



# Data Manipulation Language (DML)

DML được sử dụng để chọn, chèn, cập nhật hoặc xóa dữ liệu trong các đối tượng được xác định bằng DDL.

Các câu lệnh DML khác nhau như sau:

**LỰA CHỌN**  
tuyên bố

**CHÈN**  
tuyên bố

**CẬP NHẬT**  
tuyên bố

**XÓA BỎ**  
tuyên bố



# Data Control Language (DCL)

Dữ liệu là một phần quan trọng của cơ sở dữ liệu, vì vậy cần thực hiện các bước thích hợp để kiểm tra xem không có người dùng không hợp lệ nào truy cập vào dữ liệu.

Quyền được kiểm soát bằng cách sử dụng CẤP, THU HỒI, và PHỦ NHẬN các câu lệnh.

KHOẢN TRÍ DÁP  
tuyên bố

HOÀN LẠI  
tuyên bố

PHỦ NHẬN  
tuyên bố





# Data Types 1-6

- Kiểu dữ liệu là một thuộc tính xác định kiểu dữ liệu mà một đối tượng có thể chứa.
- Các kiểu dữ liệu phải được cung cấp cho các cột, tham số biến và hàm trả về giá trị dữ liệu và các thủ tục được lưu trữ có mã trả về.
- Transact-SQL bao gồm một số kiểu dữ liệu cơ sở, chẳng hạn như varchar, văn bản, và int.
- Tất cả dữ liệu được lưu trữ trong SQL Server phải tương thích với một trong các kiểu dữ liệu cơ sở.
- Các đối tượng sau có kiểu dữ liệu:

Các cột hiển thị trong bảng và dạng xem

Các tham số trong các thủ tục được lưu trữ

Biến

Các hàm Transact-SQL trả về một hoặc nhiều giá trị dữ liệu của một kiểu dữ liệu cụ thể

Các thủ tục được lưu trữ có mã trả về thuộc kiểu dữ liệu số nguyên

- Các mục khác nhau trong SQL Server 2012 như cột, biến và biểu thức được gán kiểu dữ liệu.



# Data Types 2-6

- SQL Server 2012 hỗ trợ ba loại kiểu dữ liệu:

Các kiểu dữ liệu do hệ thống xác định

- Các kiểu dữ liệu này được cung cấp bởi SQL Server 2012.
- Bảng sau đây cho thấy các kiểu dữ liệu do hệ thống xác định thường được sử dụng của SQL Server 2012.

Category	Data Type	Description
Exact Numerics	int	A column of this type occupies 4 bytes of memory space. Is typically used to hold integer values. Can hold integer data from $-2^{31}$ (-2,147,483,648) to $2^{31}-1$ (2,147,483,647).
	smallint	A column of this type occupies 2 bytes of memory space. Can hold integer data from -32,768 to 32,767.
	tinyint	A column of this type occupies 1 byte of memory space. Can hold integer data from 0 to 255.
	bigint	A column of this type occupies 8 bytes of memory space. Can hold data in the range $-2^{63}$ (-9,223,372,036,854,775,808) to $2^{63}-1$ (9,223,372,036,854,775,807).
	numeric	A column of this type has fixed precision and scale.
	money	A column of this type occupies 8 bytes of memory space. Represents monetary data values ranging from $-2^{63}/10000$ (-922,337,203,685,477.5808) to $2^{63}-1$ (922,337,203,685,477.5807).



## Data Types 3-6

Category	Data Type	Description
Approximate Numerics	float	A column of this type occupies 8 bytes of memory space. Represents floating point number ranging from $-1.79E+308$ through $1.79E+308$ .
	real	A column of this type occupies 4 bytes of memory space. Represents floating precision number ranging from $-3.40E+38$ through $3.40E+38$ .
Date and Time	datetime	Represents date and time. Stored as two 4-byte integers.
	smalldatetime	Represents date and time.
Character String	char	Stores character data that is fixed-length and non-Unicode.
	varchar	Stores character data that is variable-length and non-Unicode with a maximum of 8,000 characters.
	text	Stores character data that is variable-length and non-Unicode with a maximum length of $2^{31} - 1$ (2,147,483,647) characters.
Unicode Types	nchar	Stores Unicode character data of fixed-length.
	nvarchar	Stores variable-length Unicode character data.



# Data Types 4-6

Category	Data Type	Description
Other Data Types	timestamp	A column of this type occupies 8 bytes of memory space. Can hold automatically generated, unique binary numbers that are generated for a database.
	binary(n)	Stores fixed-length binary data with a maximum length of 8000 bytes.
	varbinary(n)	Stores variable-length binary data with a maximum length of 8000 bytes.
	image	Stores variable-length binary data with a maximum length of $2^{30}-1$ (1,073,741,823) bytes.
	uniqueidentifier	A column of this type occupies 16 bytes of memory space. Also, stores a globally unique identifier (GUID).

## Loại dữ liệu bí danh

- Dựa trên các kiểu dữ liệu do hệ thống cung cấp.
- Được sử dụng khi có nhiều bảng lưu trữ cùng một kiểu dữ liệu trong một cột và có các đặc điểm tương tự như độ dài, tính nullability và kiểu.
- Có thể được tạo khi nó có thể được sử dụng phổ biến bởi tất cả các bảng này.



## Data Types 5-6

- Các kiểu dữ liệu bí danh có thể được tạo bằng cách sử dụng TẠO LOẠI tuyên bố.
- Cú pháp cho TẠO LOẠI như sau:

### Cú pháp:

```
TẠO LOẠI [schema_name. ] type_name {FROM base_type [(precision [, scale])]} [NULL | CÓ GIÁ TRỊ] } [ ; ]
```

ở đây,

schema\_name: xác định tên của lược đồ mà kiểu dữ liệu bí danh đang được tạo.  
Lược đồ là một tập hợp các đối tượng như bảng, dạng xem, v.v. trong cơ sở dữ liệu.

tên loại: xác định tên của loại bí danh đang được tạo.

base\_type: xác định tên của kiểu dữ liệu do hệ thống xác định dựa trên đó kiểu dữ liệu bí danh đang được tạo.

độ chính xác và tỷ lệ: chỉ định độ chính xác và tỷ lệ cho dữ liệu số. NULL | CÓ

GIÁ TRỊ: chỉ định kiểu dữ liệu có thể giữ giá trị null hay không.



## Data Types 6-6

- Đoạn mã sau đây cho biết cách tạo loại dữ liệu bí danh bằng cách sử dụng TẠO LOẠItuyên bố.

TẠO kiểu người dùng TYPE TỪ varchar (20) KHÔNG ĐỦ

- Trong mã, kiểu dữ liệu tích hợpvarchar được lưu trữ dưới dạng một kiểu dữ liệu mới có tên kiểu người dùngbằng cách sử dụngTẠO LOẠItuyên bố.

Các loại do người dùng xác định

- Được tạo bằng các ngôn ngữ lập trình được .NET Framework hỗ trợ.



# Transact-SQL Language Elements

- Được sử dụng trong SQL Server 2012 để làm việc trên dữ liệu được nhập vào cơ sở dữ liệu SQL Server.
- Bao gồm các yếu tố sau:

**Dự đoán**

Các nhà khai thác

**Chức năng**

**Biến**

**Biểu thức**

Kiểm soát dòng chảy

**Lỗi**

**Giao dịch**

**Bình luận**

**Bộ phân tách lô**



# Predicates and Operators 1-3

- Vị ngữ được sử dụng để đánh giá xem một biểu thức có ĐÚNG SAI, hoặc KHÔNG XÁC ĐỊNH. Một số vị từ có sẵn trong Transact-SQL như sau:

TRONG	• Xác định xem một giá trị đã chỉ định có khớp với bất kỳ giá trị nào trong một truy vấn con hoặc một danh sách hay không.
Ở GIỮA	• Chỉ định một loạt các giá trị để kiểm tra.
NHƯ	• Được sử dụng để khớp các ký tự với một mẫu được chỉ định.
CHỨA	• Tìm kiếm các kết quả phù hợp chính xác hoặc ít chính xác hơn với các từ và cụm từ đơn lẻ, các từ trong một khoảng cách nhất định với nhau hoặc các kết quả phù hợp có trọng số.

- Bảng sau đây cho thấy một số ví dụ về các vị từ:

Thuộc tính	Ví dụ
TRONG	CHỌN UserID, FirstName, LastName, Mức lương TỪ Nhân viên Ở ĐẦU Lương TRONG (5000,20000);
Ở GIỮA	Chọn UserID, FirstName, LastName, Mức lương TỪ nhân viên NƠI Mức lương GIỮA 5000 và 20000;
NHƯ	Chọn UserID, FirstName, LastName, Mức lương TỪ nhân viên NƠI FirstName LIKE '%h%'
CHỨA	CHỌN UserID, FirstName, LastName, Mức lương TỪ Nhân viên NƠI Mức lương CHỨA (5000);





# Predicates and Operators 2-3

- Các toán tử được sử dụng để thực hiện phép tính số học, so sánh, nối hoặc gán các giá trị.
- SQL Server có bảy loại toán tử. Bảng sau mô tả các toán tử khác nhau được hỗ trợ trong SQL Server 2012.

Nhà điều hành	Sự miêu tả	Ví dụ
So sánh	So sánh một giá trị với một giá trị khác hoặc một biểu thức	=, <, >, >=, <=, !=, !=>
Hợp lý	Kiểm tra sự thật của một điều kiện	VÀ, HOẶC, KHÔNG
Môn số học	Thực hiện các phép toán số học như cộng, trừ, nhân và chia	+, -, *, /, %
Sự kết hợp	Kết hợp hai chuỗi thành một chuỗi	+
Phân công	Gán giá trị cho một biến	=

- Bảng sau đây cho thấy mức độ ưu tiên của các vị từ và toán tử:

Order	Operators	Order	Operators
1	() Parentheses	5	NOT
2	*, /, %	6	AND
3	+, -	7	BETWEEN, IN, CONTAINS, LIKE, OR
4	=, <, >, >=, <=, !=, !=>	8	=

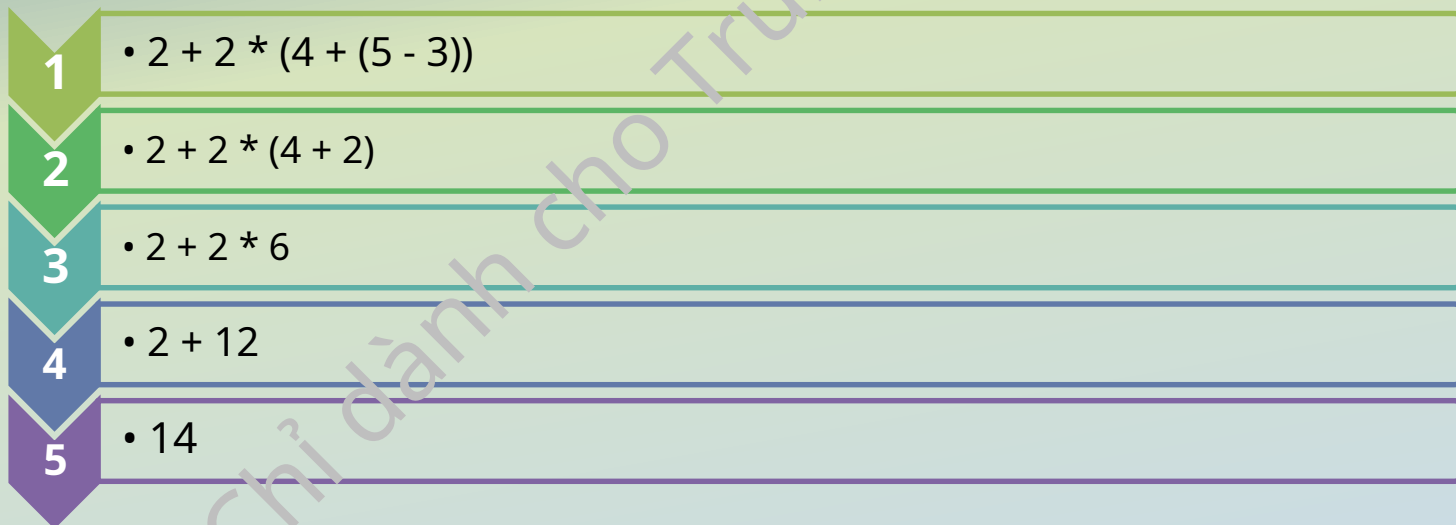


## Predicates and Operators 3-3

- Đoạn mã sau hiển thị việc thực thi các toán tử theo thứ tự ưu tiên:

```
DECLARE @Number int;  
SET @Number = 2 + 2 * (4 + (5 - 3)) CHỌN  
@Number
```

- Ở đây, các bước để đi đến kết quả như sau:



- Do đó, mã sẽ hiển thị 14.



# Functions 1-2

- Là một tập hợp các câu lệnh Transact-SQL được sử dụng để thực hiện một số tác vụ.
- Bốn loại hàm trong SQL Server 2012 như sau:

## Chức năng Rowset

- Được sử dụng để trả về một đối tượng có thể được sử dụng thay cho tham chiếu bảng.
- Ví dụ, OPENDATASOURCE, OPENQUERY, OPENROWSET, và OPENXML.

## Chức năng tổng hợp

- Cung cấp các chức năng tổng hợp để hỗ trợ việc tóm tắt khối lượng lớn dữ liệu.
- Ví dụ, SUM, MIN, MAX, AVG, COUNT, COUNTBIG, và như thế.

## Chức năng Xếp hạng

- Có thể triển khai nhiều tác vụ, như tạo mảng, tạo số tuần tự, tìm thứ hạng, v.v. một cách dễ dàng hơn và nhanh hơn bằng cách sử dụng các chức năng xếp hạng.
- Ví dụ, RANK, DENSE\_RANK, NTILE, và SỐ LƯỢNG HÀNG.

## Hàm vô hướng

- Đầu vào là một giá trị duy nhất và đầu ra nhận được cũng là một giá trị duy nhất.



## Functions 2-2

- Bảng sau đây cho thấy các hàm vô hướng thường được sử dụng trong SQL:

Hàm số	Sự miêu tả	Ví dụ
Chuyển đổi hàm số	Hàm chuyển đổi được sử dụng để chuyển đổi giá trị của một kiểu dữ liệu này sang một kiểu dữ liệu khác. Ngoài ra, nó có thể được sử dụng để lấy nhiều định dạng ngày tháng đặc biệt.	ĐỔI
Ngày và giờ hàm số	Các hàm ngày và giờ được sử dụng để thao tác các giá trị ngày và giờ. Chúng rất hữu ích để thực hiện các phép tính dựa trên thời gian và ngày tháng.	GETDATE, SYSDATETIME, GETUTCDATE, DATEADD, HIỆU QUẢ, NĂM, THÁNG, NGÀY
Toán học hàm số	Các hàm toán học thực hiện các phép toán đại số trên các giá trị số.	RAND, ROUND, POWER, ABS, TRẦN, SÀN
Hệ thống hàm số	SQL Server cung cấp các chức năng hệ thống để trả về siêu dữ liệu hoặc cài đặt cấu hình.	HOST_ID, HOST_NAME, ISNULL
Hàm chuỗi	Các hàm chuỗi được sử dụng cho các đầu vào chuỗi như char và varchar. Đầu ra có thể là một chuỗi hoặc một giá trị số.	SUBSTRING, LEFT, RIGHT, LEN, DATALENGTH, THAY THẾ, THAY THẾ, LÊN, LOWER, RTRIM, LTRIM

- Ngoài ra còn có các hàm vô hướng khác như hàm con trỏ, hàm logic, hàm siêu dữ liệu, hàm bảo mật, v.v. có sẵn trong SQL Server 2012.



# Variables

Một biến là một đối tượng có thể chứa một giá trị dữ liệu. Trong Transact-SQL, các biến có thể được phân loại thành biến cục bộ và biến toàn cục.

Dữ liệu có thể được chuyển đến các câu lệnh SQL bằng cách sử dụng các biến cục bộ. Tên của một biến cục bộ phải được bắt đầu bằng dấu '@'.

Giá trị của bất kỳ biến nào trong số này có thể được truy xuất bằng một LỰA CHỌN truy vấn.



# Expressions 1-2

Một biểu thức là sự kết hợp của các số nhận dạng, giá trị và toán tử mà SQL Server có thể đánh giá để thu được kết quả.

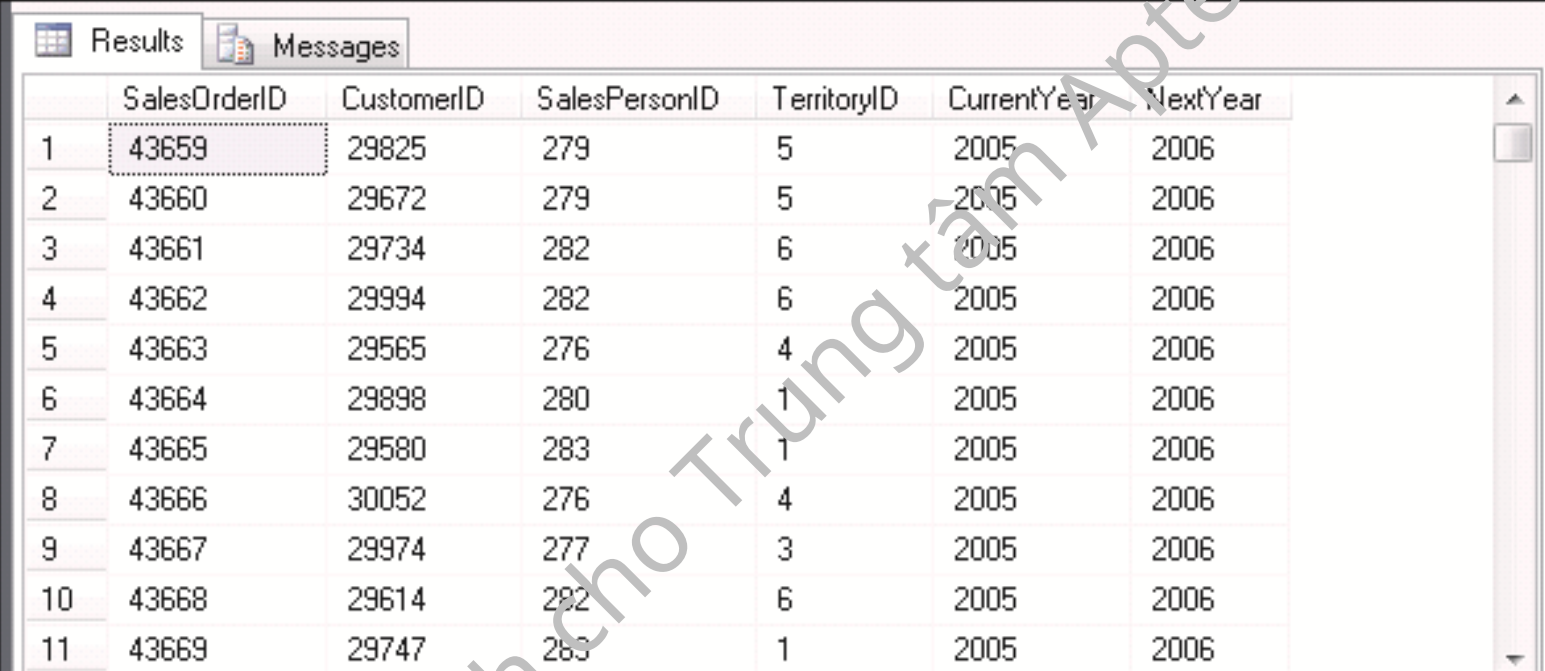
Đoạn mã sau hiển thị một biểu thức hoạt động trên một cột để thêm một số nguyên vào kết quả của `NĂM` chức năng trên một ngày giờ cột:

```
CHON ID đơn đặt hàng, ID khách hàng, ID người bán hàng, ID  
lãnh thổ, YEAR (Ngày đặt hàng)  
NHƯ Hiện tại Năm, NĂM (Ngày đặt hàng) + 1 NHƯ NĂM tới TỪ  
Bán hàng.SalesOrderHeader
```



## Expressions 2-2

- Hình sau cho thấy kết quả của biểu thức:



	SalesOrderID	CustomerID	SalesPersonID	TerritoryID	CurrentYear	NextYear
1	43659	29825	279	5	2005	2006
2	43660	29672	279	5	2005	2006
3	43661	29734	282	6	2005	2006
4	43662	29994	282	6	2005	2006
5	43663	29565	276	4	2005	2006
6	43664	29898	280	1	2005	2006
7	43665	29580	283	1	2005	2006
8	43666	30052	276	4	2005	2006
9	43667	29974	277	3	2005	2006
10	43668	29614	282	6	2005	2006
11	43669	29747	283	1	2005	2006



# Control-of-Flow, Errors, and Transactions

- Mặc dù Transact-SQL chủ yếu là một ngôn ngữ truy xuất dữ liệu, nhưng nó hỗ trợ các câu lệnh control-of-flow để thực thi và tìm lỗi.
- Ngôn ngữ điều khiển luồng xác định luồng thực thi của câu lệnh Transact-SQL, khối câu lệnh, hàm do người dùng định nghĩa và thủ tục được lưu trữ.
- Bảng sau đây cho thấy một số câu lệnh điều khiển luồng thường được sử dụng trong Transact-SQL:

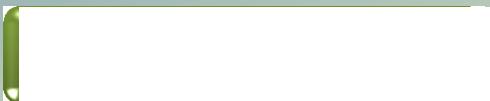
Kiểm soát luồng Tuyên bố	Sự miêu tả
NẾU. . . KHÁC	Cung cấp điều khiển phân nhánh dựa trên một bài kiểm tra logic.
TRONG KHI	Lặp lại một câu lệnh hoặc một khối câu lệnh miễn là điều kiện đúng.
BẮT ĐẦU. . . CHẤM DỨT	Xác định phạm vi của một khối câu lệnh Transact-SQL.
THỬ. . . NẮM LẤY	Xác định cấu trúc cho việc xử lý ngoại lệ và lỗi.
BẮT ĐẦU GIAO DỊCH	Đánh dấu một khối câu lệnh là một phần của giao dịch rõ ràng.





## Comments 1-2

- Chú thích là chuỗi văn bản mô tả, còn được gọi là chú thích, trong mã chương trình sẽ bị trình biên dịch bỏ qua.
- Chú thích có thể được bao gồm bên trong mã nguồn của một câu lệnh, một lô hoặc một thủ tục được lưu trữ.
- Nhận xét giải thích mục đích của chương trình, các điều kiện thực hiện đặc biệt và cung cấp thông tin lịch sử sửa đổi.
- Microsoft SQL Server hỗ trợ hai loại kiểu bình luận:



- Một dòng mã hoàn chỉnh hoặc một phần của mã có thể được đánh dấu là một nhận xét, nếu hai dấu gạch ngang (- -) được đặt ở đầu.
- Phần còn lại của dòng trở thành nhận xét.
- Đoạn mã sau hiển thị việc sử dụng kiểu nhận xét này:

SỬ DỤNG AdventureWorks2012

- - Bảng HumanResources.EmpLOYEE chứa thông tin chi tiết của một nhân viên.
- - Câu lệnh này truy xuất tất cả các hàng của bảng
- - Nguồn nhân lực. Nhân viên. CHỌN \* TỪ
- Nguồn nhân lực. Người lao động



## Comments 2-2

- Các ký tự chú thích này có thể được sử dụng trên cùng một dòng với mã được thực thi, trên các dòng của chính chúng, hoặc thậm chí trong mã thực thi.
- Mọi thứ trong dòng bắt đầu từ cặp chú thích mở (/ \*) đến cặp chú thích đóng (\* /) đều được coi là một phần của chú thích.
- Đối với một nhận xét nhiều dòng, cặp ký tự nhận xét mở (/ \*) phải bắt đầu nhận xét và cặp ký tự nhận xét đóng (\* /) phải kết thúc nhận xét.
- Đoạn mã sau hiển thị việc sử dụng kiểu nhận xét này:

SỬ DỤNG AdventureWorks2012

/ \* Bảng HumanResources.E Employee chứa thông tin chi tiết của một nhân viên. Câu lệnh này truy xuất tất cả các hàng của bảng  
Nguồn nhân lực. Nhân viên \* / CHỌN \* TỪ  
Nguồn nhân lực. Người lao động



# Batch Separators

Lô là tập hợp một hoặc nhiều câu lệnh Transact-SQL được gửi cùng một lúc từ một ứng dụng đến SQL Server để thực thi.

Các câu lệnh trong kế hoạch thực thi sau đó được thực hiện lần lượt.

Một bộ phân tách hàng loạt được xử lý bởi các công cụ máy khách SQL Server như SSMS để thực thi các lệnh. Ví dụ: bạn cần chỉ định GO làm bộ phân tách hàng loạt trong SSMS.

- Một ví dụ về câu lệnh lô được đưa ra trong đoạn mã sau:

```
SỬ DỤNG AdventureWorks2012
CHỌN * TỪ Nguồn nhân lực. Nhân viên ĐI
```

- Ở đây, hai câu lệnh sẽ được nhóm lại thành một kế hoạch thực thi nhưng được thực thi một câu lệnh tại một thời điểm.
- Các từ khóa báo hiệu sự kết thúc của một lô.



# Set Theory

Lý thuyết tập hợp là một nền tảng toán học được sử dụng trong mô hình cơ sở dữ liệu quan hệ.

Ví dụ: tất cả các nhân viên dưới quyền Nhân viên bảng có thể được coi là một bộ.

- Bảng sau đây cho thấy các ứng dụng khác nhau trong lý thuyết tập hợp và ứng dụng tương ứng của chúng trong các truy vấn SQL Server.

Đặt ứng dụng lý thuyết	Ứng dụng trong truy vấn SQL Server
Hành động trên toàn bộ tập hợp cùng một lúc.	Truy vấn toàn bộ bảng cùng một lúc.
Sử dụng xử lý dựa trên khai báo, dựa trên tập hợp.	Sử dụng các thuộc tính trong SQL Server để truy xuất dữ liệu cụ thể.
Các phần tử trong tập hợp phải là duy nhất.	Xác định các khóa duy nhất trong bảng.
Không có hướng dẫn sắp xếp.	Kết quả của truy vấn không được truy xuất theo bất kỳ thứ tự nào.



# Predicate Logic

- Logic vị từ là một khung toán học bao gồm các bài kiểm tra logic đưa ra một kết quả. Kết quả luôn được hiển thị là true hoặc false.
- Trong Transact-SQL, các biểu thức chẳng hạn như **Ở ĐÂU** và **TRƯỜNG HỢP** biểu thức dựa trên logic vị từ.
- Logic vị từ cũng được sử dụng trong các tình huống khác trong Transact-SQL. Một số ứng dụng của logic vị từ trong Transact-SQL như sau:

Thực thi tính toán vẹn của dữ liệu bằng cách sử dụng **Kiểm tra** hạn chế

Kiểm soát luồng bằng cách sử dụng **Nếu** tuyên bố

Nối các bảng bằng cách sử dụng **TRÊN** lọc

Lọc dữ liệu trong các truy vấn bằng cách sử dụng **Ở ĐÂU** và **ĐANG CÓ** mệnh đề

Cung cấp logic có điều kiện để **TRƯỜNG HỢP** biểu thức

Xác định truy vấn con



# Logical Order of Operators in SELECT Statement 1-5

- Cùng với cú pháp của các phần tử SQL Server khác nhau, người dùng SQL Server cũng phải biết quá trình thực thi toàn bộ truy vấn như thế nào.
- Quá trình này là một quá trình logic phá vỡ truy vấn và thực hiện truy vấn theo một trình tự được xác định trước trong SQL Server 2012.
- Các LỰA CHỌN câu lệnh là một truy vấn sẽ được sử dụng để giải thích quy trình logic của việc thực thi truy vấn.
- Cú pháp của LỰA CHỌN như sau:

## Cú pháp:

```
CHỌN <chọn danh sách>  
TỪ <nguồn bảng>  
WHERE <điều kiện tìm kiếm> NHÓM  
THEO <nhóm theo danh sách> CÓ  
<điều kiện tìm kiếm> ĐẶT HÀNG  
THEO <thứ tự theo danh sách>
```



# Logical Order of Operators in SELECT Statement 2-5

- Bảng sau giải thích các yếu tố của LỰA CHỌN tuyên bố:

Thành phần	Sự miêu tả
CHỌN <chọn danh sách>	Xác định các cột được trả về
TỪ <nguồn bảng>	Xác định bảng được truy vấn
WHERE <điều kiện tìm kiếm>	Lọc các hàng bằng cách sử dụng các vị từ
GROUP BY <nhóm theo danh sách>	Sắp xếp các hàng theo nhóm
CÓ <điều kiện tìm kiếm>	Lọc các nhóm bằng cách sử dụng các vị từ
ĐẶT HÀNG THEO <thứ tự theo danh sách>	Sắp xếp đầu ra

- Đoạn mã sau đây cho thấy một LỰA CHỌN tuyên bố:

```
CHỌN ID người bán hàng, NĂM (Ngày đặt hàng) NHƯ Đơn hàng
Năm từ Bán hàng.SalesOrderHeader
WHERE CustomerID = 30084
NHÓM THEO ID người bán hàng, NĂM (Ngày đặt
hàng) CÓ ĐẾM (*) > 1
ĐẶT HÀNG THEO SalesPersonID, OrderYear;
```



# Logical Order of Operators in SELECT Statement 3-5

- Trong ví dụ, thứ tự mà SQL Server sẽ thực thi LỰA CHỌN như sau:

1

- Đầu tiên TỪ mệnh đề được đánh giá để xác định bảng nguồn sẽ được truy vấn.

2

- Tiếp theo, Ở ĐẦU mệnh đề được đánh giá để lọc các hàng trong bảng nguồn.
- Bộ lọc này được xác định bởi vị từ được đề cập trong Ở ĐẦU mệnh đề.

3

- Sau đó, NHÓM THEO mệnh đề được đánh giá.
- Mệnh đề này sắp xếp các giá trị đã lọc nhận được từ Ở ĐẦU mệnh đề.

4

- Tiếp theo, ĐANG CÓ mệnh đề được đánh giá dựa trên vị ngữ được cung cấp.

5

- Tiếp theo, LỰA CHỌN mệnh đề được thực thi để xác định các cột sẽ xuất hiện trong kết quả truy vấn.

6

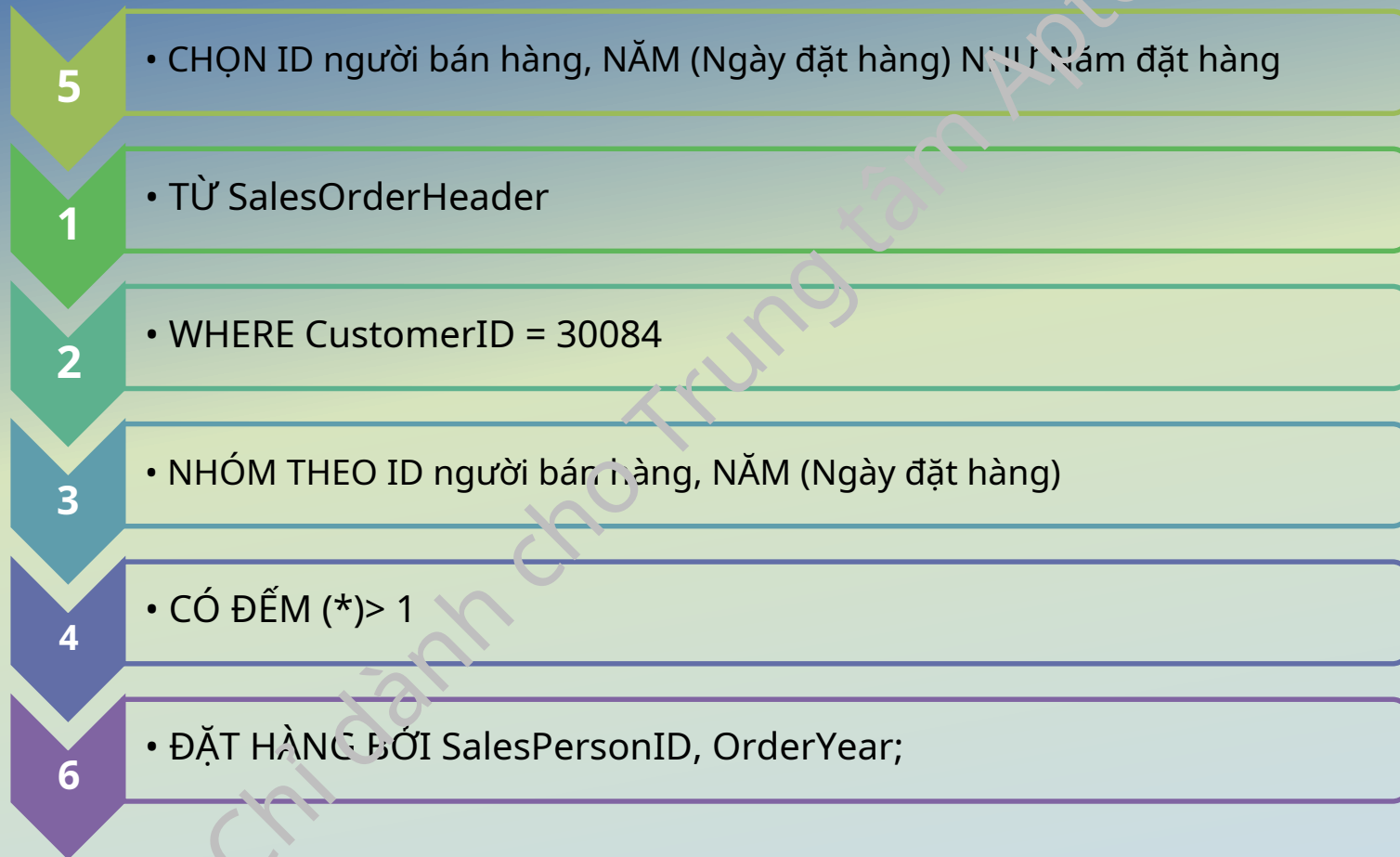
- Cuối cùng ĐẶT BỞI câu lệnh được đánh giá để hiển thị kết quả đầu ra.





# Logical Order of Operators in SELECT Statement 4-5

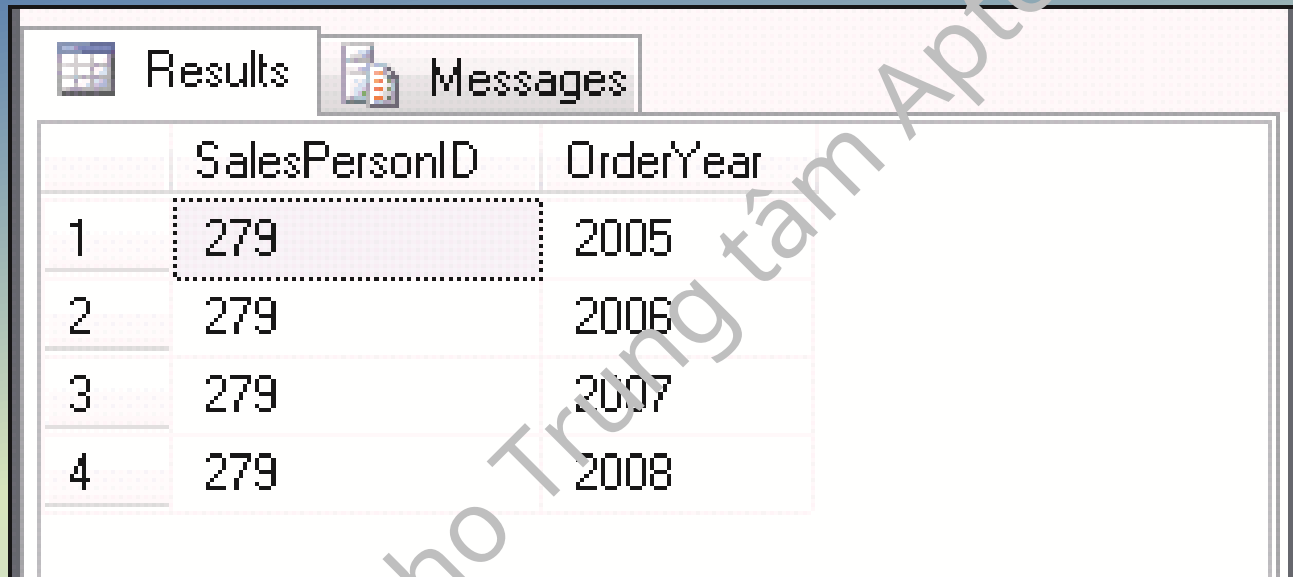
- Thứ tự thực hiện cho LỰA CHỌN câu lệnh trong đoạn mã sẽ như sau:





# Logical Order of Operators in SELECT Statement 5-5

- Hình sau cho thấy kết quả của LỰA CHỌN tuyên bố:

A screenshot of a SQL Server query results window. The window has two tabs: 'Results' (active) and 'Messages'. The 'Results' tab displays a table with two columns: 'SalesPersonID' and 'OrderYear'. The table contains four rows of data. The first row is highlighted with a dotted border. A large, diagonal watermark text 'Chỉ dành cho Trung tâm Aptech' is overlaid on the table.

	SalesPersonID	OrderYear
1	279	2005
2	279	2006
3	279	2007
4	279	2008



# Summary

- Transact-SQL là một ngôn ngữ mạnh mẽ cung cấp các tính năng như kiểu dữ liệu, đối tượng tạm thời và các thủ tục được lưu trữ mở rộng.
- SQL Server hỗ trợ ba loại câu lệnh Transact-SQL, đó là DDL, DML và DCL.
- Kiểu dữ liệu là một thuộc tính xác định kiểu dữ liệu mà một đối tượng có thể chứa.
- Các phần tử ngôn ngữ Transact-SQL bao gồm các vị từ, toán tử, hàm, biến, biểu thức, điều khiển luồng, lỗi và giao dịch, nhận xét và dấu phân tách hàng loạt.
- Bộ và Logic vị từ là hai nguyên tắc toán học cơ bản được sử dụng trong SQL Server 2012.
- Lý thuyết tập hợp là một nền tảng toán học được sử dụng trong mô hình cơ sở dữ liệu quan hệ, trong đó tập hợp là một tập hợp các đối tượng riêng biệt được coi là một tổng thể.
- Logic vị từ là một khung toán học bao gồm các bài kiểm tra logic đưa ra một kết quả.