



FPT POLYTECHNIC



THỰC HỌC – THỰC NGHIỆP



Conceive Design Implement Operate

QUẢN TRỊ CƠ SỞ DỮ LIỆU VỚI SQL SERVER

BÀI 3: CÁC HÀM HỆ THỐNG & XỬ LÝ CHUỖI

- ⦿ Các hàm hệ thống Sql
- ⦿ Xử lý chuỗi





Các hàm hệ thống Sql

- ❖ Các hàm chuyển đổi kiểu dữ liệu
- ❖ Các hàm toán học
- ❖ Các hàm xử lý chuỗi
- ❖ Các hàm ngày tháng năm





PHẦN 1

- ❑ Khi làm việc với các biểu thức chứa nhiều kiểu dữ liệu khác nhau, phải thực hiện chuyển đổi giữa các kiểu dữ liệu.
- ❑ Hai loại chuyển đổi kiểu dữ liệu
 - ❖ Chuyển đổi ngầm (do SQL server tự thực hiện)
 - ❖ Chuyển đổi tường minh (sử dụng các hàm thư viện)

❑ Chuyển đổi ngầm

- ❖ Gán giá trị cho một cột có kiểu dữ liệu khác với giá trị được gán.
- ❖ Biểu thức tính toán có sự tham gia của nhiều loại dữ liệu khác nhau (SQL chuyển kiểu có độ ưu tiên thấp sang kiểu có độ ưu tiên cao hơn).
- ❖ Lập trình viên cần nắm chuyển đổi ngầm tránh lỗi.
- ❖ Việc chuyển đổi xảy ra tự động.

$\text{TotalCost} = \text{StandardCost} * \text{Quantity}$

	Name	Quantity	StandardCost	TotalCost
1	LL Mountain Seat Assembly	267	98.77	26371.59
2	LL Mountain Seat Assembly	318	98.77	31408.86
3	LL Mountain Seat Assembly	212	98.77	20939.24
4	ML Mountain Seat Assembly	268	108.99	29209.32
5	ML Mountain Seat Assembly	320	108.99	34876.80
6	ML Mountain Seat Assembly	214	108.99	23305.59
7	ML Mountain Seat Assembly	270	145.87	39385.90
8	HL Mountain Seat Assembly	32	145.87	4688.24
9	HL Mountain Seat Assembly	21	145.87	31507.92
10	LL Road Seat Assembly	272	98.77	26865.44
11	LL Road Seat Assembly	323	98.77	31902.71
12	LL Road Seat Assembly	217	98.77	21433.09
13	ML Road Seat Assembly	273	108.99	29754.27
14	ML Road Seat Assembly	324	108.99	35312.76

SMALLINT

MONEY

Implicit Conversion:
Quantity converted to Money

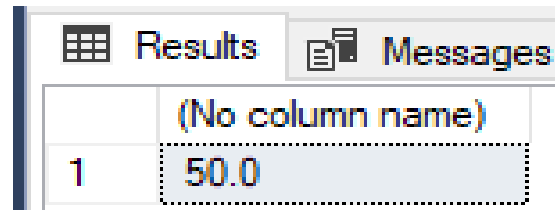
❑ Chuyển đổi ngầm

- ❖ Thứ tự ưu tiên của các kiểu dữ liệu phổ biến trong SQL Server

<u>Độ ưu tiên</u>	<u>Nhóm</u>	<u>Kiểu dữ liệu</u>
Cao nhất	Ngày/Giờ	datetime smalldatetime
	Số	float real decimal money smallmoney int smallint tinyint bit
Thấp nhất	Chuỗi	nvarchar nchar varchar char

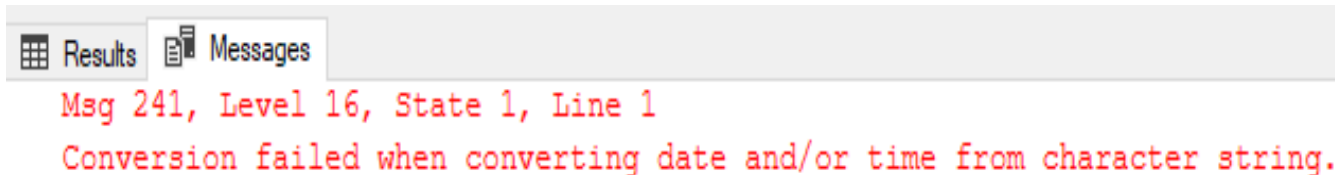
❑ Chuyển đổi ngầm

- ❖ Xét ví dụ: `SELECT 100 * .5` → Kết quả có kiểu dữ liệu gì?



	(No column name)
1	50.0

- ❖ Kết là 50.0 vì .5 có độ ưu tiên cao hơn
- ❖ Xét ví dụ: `SELECT 'Today is ' + GETDATE()` => Kết quả kiểu dữ liệu là gì?



- ❖ Không thể chuyển "Today is" thành kiểu DateTime

❑ Chuyển đổi tường minh

- ❖ Chuyển dữ liệu có kiểu dữ liệu với độ ưu tiên cao hơn về kiểu dữ liệu có độ ưu tiên thấp hơn.
- ❖ Sử dụng hàm **CAST** hoặc **CONVERT** để thực hiện phép chuyển đổi tường minh.
- ❖ Ví dụ chuyển kiểu có độ ưu tiên cao DateTime về kiểu có độ ưu tiên thấp hơn varchar:

```
SELECT 'Today is ' + cast(GETDATE() as varchar)
```

Results		Messages	
	(No column name)		
1	Today is Dec 16 2018 3:04PM		

❑ Chuyển đổi tường minh

❖ Sử dụng hàm chuyển đổi dữ liệu Hàm CAST

➔ Cú pháp

CAST(<Biểu thức> AS <Kiểu dữ liệu> [(length)])

➔ Sử dụng

Hàm CAST dùng để chuyển đổi tường minh, hay ép kiểu, một biểu thức từ kiểu dữ liệu này sang kiểu dữ liệu khác

❑ Chuyển đổi tường minh

❖ Một số ví dụ về hàm CAST

Hàm	Kết quả
CAST('123.4' AS Decimal)	123
CAST(123 as decimal(6,2))	123.00
CAST(2.78128 AS integer)	2
CAST(2.78128 AS money)	2.7813
CAST('11/11/72' as smalldatetime)	1972-11-11 00:00:00
CAST('123' AS Int)	123
CAST('1994-02-01 14:16:26' AS varchar)	1994-02-01 14:16:26
SELECT Name, CAST(ID AS VarChar(5))	

❑ Chuyển đổi tường minh

❖ Một số ví dụ về hàm CAST

- Chuyển đổi dữ liệu khi thực hiện phép chia số nguyên

Thao tác	Kết quả
50/100	0 (Kiểu int)
50/CAST(100 AS decimal(3))	.500000 (Kiểu decimal)

- Câu lệnh SELECT sử dụng hàm CAST

```
SELECT LUONG,
       CAST(NHANVIEN.LUONG AS int) AS integerLuong,
       CAST(NHANVIEN.LUONG AS varchar) AS varcharLuong
FROM NHANVIEN
```

	LUONG	integerLuong	varcharLuong
1	43000.5	43000	43000.5
2	25000	25000	25000
3	38000.18	38000	38000.2
4	40000	40000	40000
5	55000.27	55000	55000.3

❑ Chuyển đổi tường minh

- ❖ Sử dụng hàm chuyển đổi dữ liệu Hàm CONVERT

➔ Cú pháp

CONVERT(<Kiểu dữ liệu>[(length)], <Biểu thức>[, <Tham số định dạng>])

➔ Sử dụng

- Sử dụng hàm **CONVERT** để chuyển đổi tường minh một biểu thức từ kiểu dữ liệu này sang kiểu dữ liệu khác
- Khi chuyển đổi sang kiểu dữ liệu kí tự và cần định dạng hiển thị cho dữ liệu
- **Tham số định dạng**: định dạng hiển thị cho các giá trị ngày/giờ, số thực, tiền tệ khi chuyển đổi sang kiểu kí tự

<http://msdn.microsoft.com/en-us/library/ms187928.aspx>

❑ Chuyển đổi tương minh

❖ Sử dụng hàm chuyển đổi dữ liệu Hàm CONVERT

➤ Một số ví dụ về hàm CONVERT

Hàm	Kết quả
CONVERT(VarChar(20), 123.56)	123.56
CONVERT(INT, '123')	123
CONVERT(dec(10,2),BillingTotal + PaymentTotal) CONVERT(varchar, InvoiceDate, 107)	
CONVERT(varchar,@d , 1)	mm/dd/yy
CONVERT(varchar,@d , 105)	dd/mm/yyyy

❑ Chuyển đổi tường minh

❖ Sử dụng hàm chuyển đổi dữ liệu Hàm CONVERT

➤ So sánh giữa Cast và CONVERT:

```
SELECT 'Today's date is ' + CAST(GETDATE() as varchar)
```

Results		Messages	
	SuDungCast		
1	Today's date is Dec 16 2018 3:37PM		

```
SELECT 'Today's date is ' + CONVERT(VARCHAR, GETDATE(), 101)
```

Results		Messages	
		SuDungConvert	
1	Today's date is 12/16/2018		

Tham số định dạng

❑ Chuyển đổi tường minh

❖ Sử dụng hàm chuyển đổi dữ liệu Hàm CONVERT

➤ Ví dụ dùng Convert (chú ý các định dạng khi xuất dữ liệu)

```
SELECT NgSinh,
Cast(NGSINH as Varchar) as VarcharDate,
CONVERT(Varchar, NGSINH, 103) as VarCharVNDate,
CONVERT(Varchar, NGSINH, 101) as VarCharForDate,
CONVERT(Varchar, NGSINH, 105) as VarCharVNDate1,
CONVERT(Varchar, NGSINH, 110) as VarCharForDate1
FROM NHANVIEN
```

101 = mm/dd/yyyy
 103 = dd/mm/yyyy
 105 = dd-mm-yyyy
 107 = Mon dd, yyyy
 110 = mm-dd-yyyy

Results		Messages				
	NgSinh	VarcharDate	VarCharVNDate	VarCharForDate	VarCharVNDate1	VarCharForDate1
1	1967-02-01	1967-02-01	01/02/1967	02/01/1967	01-02-1967	02-01-1967
2	1957-05-04	1957-05-04	04/05/1957	05/04/1957	04-05-1957	05-04-1957
3	1967-03-04	1967-03-04	04/03/1967	03/04/1967	04-03-1967	03-04-1967
4	1962-08-20	1962-08-20	20/08/1962	08/20/1962	20-08-1962	08-20-1962
5	1965-01-01	1965-01-01	01/01/1965	01/01/1965	01-01-1965	01-01-1965

❑ PI: trả về số Pi

```
--Tính chu vi dương tròn có bán kính 10
```

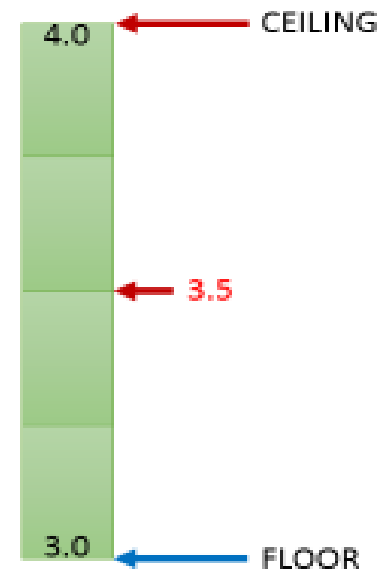
```
SELECT 2 * PI() * 10
```

❑ SQRT: tính căn bậc 2, trả về dữ liệu kiểu float -
SELECT SQRT(25) → 5

❑ SQUARE: bình phương - SELECT SQUARE(3) => ?

❑ CEILING và FLOOR

SELECT CEILING(9/ 4.0) → ?



- ❑ ROUND: làm tròn giá trị theo vị trí thập phân xác định

```
ROUND(value, number of decimal places)
```

```
SELECT ROUND(5.153745,0)==>5.000
```

```
SELECT ROUND(5.153745,1)==>5.2000
```

- ❑ ABS: Lấy giá trị tuyệt đối, ví dụ lương trung bình của nv là 31000, thống kê mức độ chênh lệch lương của các nv so với lương trung bình → không quan tâm số âm hay dương

```
SELECT LUONG ,ABS(LUONG - 31000) MucDoChenhLech_usd  
FROM NHANVIEN  
ORDER BY MucDoChenhLech_usd
```

	LUONG	MucDoChenhLech_usd
1	30000	1000
2	25000.89	5999.11
3	25000	6000



❖ Demo các ví dụ trong các slide bên trên



PHẦN 2

❑ Làm việc với kiểu dữ liệu chuỗi

Hàm	Mô tả
LEN(string)	Trả về số lượng ký tự trong chuỗi, tính cả ký tự trắng đầu chuỗi.
LTRIM(string)	Trả về chuỗi với các ký tự trắng đầu chuỗi bị loại bỏ.
RTRIM(string)	Trả về chuỗi với các ký tự trắng cuối chuỗi bị loại bỏ.
LEFT(string,length)	Trả về chuỗi con có chiều dài length tính từ đầu
RIGHT(string,length)	Trả về chuỗi con có chiều dài length tính từ cuối

□ Ví dụ

Hàm	Kết quả
LEN('SQL Server')	10
LEN(' SQL Server ')	12
LEFT('SQL Server', 3)	'SQL'
LTRIM(' SQL Server ')	'SQL Server '
RTRIM(' SQL Server ')	' SQL Server'
LTRIM(RTRIM(' SQL Server '))	'SQL Server'

□ Hàm CHARINDEX

→ Cú pháp

CHARINDEX (*find*, *search* [, *start*])

CHARINDEX – Position within a string

S	Q	L		S	e	r	v	e	r
1	2	3	4	5	6	7	8	9	10

CHARINDEX('Se', 'SQL Server') = 5

□ Hàm SubString

➔ Cú pháp

SUBSTRING(string,start,length)

SUBSTRING – Return characters within another string

S	Q	L		S	e	r	v	e	r
1	2	3	4	5	6	7	8	9	10

SUBSTRING('SQL Server',5,3)

```
DECLARE @FullName VarChar(25)  
SET @FullName = 'www.poly.edu.vn'  
SELECT SUBSTRING(@FullName, 5, 4)
```

➔ poly

❑ Hàm REPLACE

➔ Cú pháp

REPLACE (*search*, *find*, *replace*)

Trả về chuỗi *search* với tất cả các chuỗi *find* được thay thế bởi chuỗi *replace*.

REPLACE ('0973-456-226', '-', '.')



0973.456.226

CÁC HÀM XỬ LÝ CHUỖI

Hàm	Mô tả
REVERSE(string)	Trả về chuỗi với các ký tự đảo ngược.
LOWER(string)	Trả về chuỗi được chuyển đổi thành các chữ cái thường.
UPPER(string)	Trả về chuỗi được chuyển đổi thành các chữ cái hoa.
SPACE(integer)	Trả về chuỗi với số lượng ký tự trắng được chỉ định qua tham số integer

Hàm	Kết quả
LOWER('SQL Server')	'sql server'
UPPER('ca')	CA
REVERSE('ABCDE')	EDCBA

□ Demo các hàm xử lý chuỗi

```
SELECT DCHI, LEFT(DCHI,2) as Left2, RIGHT(DCHI,3) as Right3,
SUBSTRING(DCHI, LEN(DCHI)-2,3) as Substring3
FROM NHANVIEN
```

Results		Messages		
	DCHI	Left2	Right3	Substring3
1	291 Hồ Văn Huê, TP HCM	29	HCM	HCM
2	34 Mai Thị Lự, Tp Hồ Chí Minh	34	inh	inh
3	95 Bà Rịa, Vũng Tàu	95	Tàu	Tàu
4	222 Nguyễn Văn Cừ, Tp HCM	22	HCM	HCM
5	15 Trưng Vương, Hà Nội	15	Nội	Nội
6	332 Nguyễn Thái Học, Tp HCM	33	HCM	HCM
7	80 Lê Hồng Phong, Tp HCM	80	HCM	HCM
8	119 Cổng Quỳnh, Tp HCM	11	HCM	HCM

❑ Vấn đề thường xảy ra với dữ liệu chuỗi

❖ Sắp thứ tự

- Khi sắp xếp một cột kiểu chuỗi chứa số, bạn có thể nhận được kết quả không mong đợi.
- Để tránh điều đó, bạn chuyển đổi cột kiểu chuỗi thành giá trị số trong mệnh đề ORDER BY
- Bảng NhanVien được sắp xếp bởi cột MaNV khi có dùng Cast

```
SELECT * FROM NHANVIEN
ORDER BY MANV
```

TENNV	MANV	I
Như	001	
Tâm	003	
Hùng	004	
Tùng	005	
Vinh	006	
Hành	007	
Quang	008	
Tiên	009	

```
SELECT * FROM NHANVIEN
ORDER BY CAST(MANV AS int)
```

TENNV	MANV	N
Như	001	1
Tâm	003	1
Hùng	004	1
Tùng	005	1
Vinh	006	1
Hành	007	1
Quang	008	1
Tiên	009	1

❑ Vấn đề thường xảy ra với dữ liệu chuỗi

❖ Phân tách kí tự

- Nếu chuỗi gồm hai hay nhiều thành phần, bạn có thể phân tách chuỗi thành những thành phần độc lập.
- Sử dụng hàm CHARINDEX để định vị những ký tự phân tách. Sau đó, dùng hàm LEFT, RIGHT, SUBSTRING và LEN để trích ra những thành phần độc lập

```
SELECT Name,  
       LEFT(Name, CHARINDEX(' ', Name) - 1) AS First,  
       RIGHT(Name, LEN(Name) - CHARINDEX(' ', Name) ) AS Last  
FROM StringSample
```

Name	First	Last
Lizbeth Darien	Lizbeth	Darien
Damell O'Sullivan	Damell	O'Sullivan
Lance Pinos-Potter	Lance	Pinos-Potter
Jean Paul Renard	Jean	Paul Renard
Alisha von Strump	Alisha	von Strump

- ❑ GETDATE trả về ngày tháng năm hiện tại (bao gồm ngày, tháng năm, giờ, phút, giây)

```
SELECT GETDATE()
```



(No column name)
2018-12-17 19:02:04.353

- ❖ Lược bỏ thông tin giờ phút giây

```
SELECT CONVERT(date, GETDATE())
```



(No column name)
2018-12-17

- ❖ Lược bỏ thông tin ngày tháng năm

```
select CONVERT(time, GETDATE())
```



	thoigian
1	09:45:36.0866667

- ❖ Hiển thị từng phần ngày tháng năm

```
SELECT year(GETDATE()) as year,  
month(GETDATE()) as month,  
day(GETDATE()) as day
```



	year	month	day
1	2018	12	17

❑ DATENAME: truy cập tới các thành phần liên quan ngày tháng

```
SELECT DATENAME(year, GETDATE()) as Year,
       DATENAME(week, GETDATE()) as Week,
       DATENAME(dayofyear, GETDATE()) as DayOfYear,
       DATENAME(month, GETDATE()) as Month,
       DATENAME(day, GETDATE()) as Day,
       DATENAME(weekday, GETDATE()) as WEEKDAY
```

Results		Messages				
	Year	Week	DayOfYear	Month	Day	WEEKDAY
1	2018	51	352	December	18	Tuesday

❑ Chú ý khi lưu và truy vấn dữ liệu ngày/giờ

- ❖ Trong SQL Server dữ liệu ngày/giờ được xử lý dưới định dạng tháng/ngày/năm
- ❖ Để sử dụng định dạng ngày/giờ dưới dạng ngày/tháng/năm. Cần chú ý
 - Khi sử dụng câu lệnh INSERT phải truyền dữ liệu ngày/giờ theo định dạng tháng/ngày/năm
 - Khi truy vấn dữ liệu, để lấy về giá trị có định dạng ngày/tháng/năm có thể sử dụng hàm CONVERT với mã định dạng 3 hoặc 103



❖ Thực hiện các ví dụ bên trên cho phần xử lý chuỗi và ngày tháng.

☑ Các hàm hệ thống Sql

- ❖ Các hàm chuyển đổi kiểu dữ liệu
- ❖ Các hàm toán học
- ❖ Các hàm xử lý chuỗi
- ❖ Các hàm ngày tháng năm



thank
you!