

Bài 7: Các biến, toán tử, và các cấu trúc điều khiển

Nội dung:

- Biến cục bộ
- Biến hệ thống
- Các toán tử
- Biểu thức Case
- Các cấu trúc điều khiển
- Sử dụng biến kiểu dữ liệu Cursor

2/8/2018

Microsoft SQL Server 2005

1

Biến cục bộ

- Lưu trữ các giá trị tạm thời trong quá trình tính toán
- Phải được khai báo trước khi sử dụng

■ Khai báo biến cục bộ

Cú pháp:

```
Declare @ Tên_biến Kiểu_dữ_liệu
```

Ví dụ 1:

```
Declare @TongSIdat int, @MaxSIdat int,  
        @HotenNhaCC char(50), @Ngayxh datetime
```

2/8/2018

Microsoft SQL Server 2005

2

Biến cục bộ

Lưu ý:

- Tên biến là duy nhất trong phạm vi hoạt động của nó
- Kiểu dữ liệu: là các kiểu dữ liệu sẵn có, ngoại trừ các kiểu text, ntext, image
- Giá trị mặc định khởi tạo của tất cả các biến sau khi khai báo là Null

2/8/2018

Microsoft SQL Server 2005

3

Biến cục bộ

■ Gán giá trị cụ thể cho biến bằng phát biểu Set

Cú pháp:

```
Set @ Tên_biến = Giá_trị
```

Ví dụ 2:

```
Set @Ngayxh = '09-27-2008'
```

```
Set @HotenNhaCC='Nguyễn Văn Nam'
```

2/8/2018

Microsoft SQL Server 2005

4

Biến cục bộ

- Gán các biểu thức tính toán hoặc các giá trị được lấy ra các cột trong các bảng dữ liệu:

Cú pháp:

```
Select @Tên_biến = Tên_cột/Biểu_thức(tên_cột)
From Tên_bảng
```

Ví dụ 3:

```
Select @TongSIDat = sum(SIDat) From CTDONDH
```

Lưu ý: Câu lệnh trên cũng có thể sửa thành:

```
Set @TongSIDat =
(select sum(SIDat) From CTDONDH)
```

2/8/2018

Microsoft SQL Server 2005

5

Biến cục bộ

- Xem giá trị hiện hành của biến:

Cú pháp:

```
Print @Tên_biến/Biểu_thức_chuỗi
```

Ví dụ 4:

```
Print 'Tong so luong dat la:'
Print @TongSIDat
```

* Lưu ý:

- Độ dài tối đa của chuỗi: 8000 ký tự
- Kết hợp 2 hàm chuyển đổi dữ liệu của biến sang kiểu chuỗi (hàm Cast và hàm Convert) để trình bày kết quả

Ví dụ 5:

```
Print 'Tong so luong dat la: ' + cast(@TongSIDat as char(10))
Print 'Tong so luong dat la: ' + convert(char(10),@TongSIDat)
```

2/8/2018

Microsoft SQL Server 2005

6

Biến cục bộ

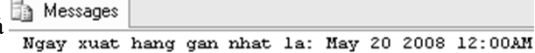
- Phạm vi hoạt động của biến: chỉ hoạt động trong một thủ tục nội tại hoặc một lô mà nó được khai báo

- Thủ tục nội tại
- Lô (batch): tập các lệnh được kết thúc bởi từ khóa Go

Ví dụ 6:

```
Declare @Ngayxhgn datetime
Select @Ngayxhgn = max(Ngayxuat) from PXUAT
Print 'Ngày xuất hàng gần nhất là: '+
Cast(@Ngayxhgn as char(20))
```

Go

→ Kết quả  Messages
Ngày xuất hàng gần nhất là: May 20 2008 12:00AM

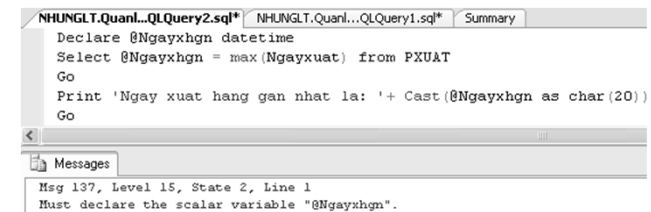
2/8/2018

Microsoft SQL Server 2005

7

Biến cục bộ

- Ví dụ 7: Tương tự ví dụ trước, song đặt lệnh in kết quả trong một lô khác không chứa lệnh khai báo biến → lỗi



```
NHUNGLT.Quanl...QLQuery2.sql* NHUNGLT.Quanl...QLQuery1.sql* Summary
Declare @Ngayxhgn datetime
Select @Ngayxhgn = max(Ngayxuat) from PXUAT
Go
Print 'Ngày xuất hàng gần nhất là: '+ Cast(@Ngayxhgn as char(20))
Go
```

Messages
Msg 137, Level 15, State 2, Line 1
Must declare the scalar variable "@Ngayxhgn".

Hệ thống thông báo lỗi
chưa khai báo biến

2/8/2018

Microsoft SQL Server 2005

8

Biến cục bộ

- **Phạm vi hoạt động của biến (tiếp):** Lưu ý
 - Khi một lô có chứa một câu lệnh bị lỗi:
 - + Nếu lỗi đó là lỗi cú pháp → tất cả các câu lệnh trong lô đó không được thực hiện
 - + Nếu đó là lỗi về ràng buộc toàn vẹn dữ liệu → chỉ riêng lệnh có lỗi không được thực hiện, các lệnh khác được thực hiện bình thường
 - Các câu lệnh Create như: Create default, create rule, create trigger, create view, create procedure không được phép kết hợp với các lệnh khác trong cùng một lô

2/8/2018

Microsoft SQL Server 2005

9

Biến cục bộ

- **Biên kiểu bảng dữ liệu:**
- * **Cú pháp khai báo biên kiểu bảng:**

```
Declare @Tên_biên_kiểu_bảng table
(Tên_cột Kiểu_dữ_liệu [Not Null],
...
)
```

Ví dụ 8:

```
Declare @CTPNHAP_Thang table
(SoPn char(4)not null,
MaVTu char(4)not null,
SINHap int,
DgNhap money)
```

2/8/2018

Microsoft SQL Server 2005

10

Biến cục bộ

- **Biên kiểu bảng dữ liệu (tiếp):**
- * **Chèn bản ghi cụ thể vào biên kiểu bảng:** sử dụng cú pháp Insert Into Values
- Ví dụ 9:**

```
Insert Into @CTPNHAP_Thang
values ('N001','VD02',10,2000000)
```
- * **Chèn dữ liệu lấy từ một bảng cùng kiểu vào biên kiểu:** sử dụng cú pháp Insert Into kết hợp với câu lệnh Select

Ví dụ 10:

```
Insert into @CTPNHAP_Thang
Select * from CTPNHAP where SoPn = 'N001'
```

- * **Lựa chọn hiển thị dữ liệu có trong biên kiểu:** sử dụng cú pháp Select

Ví dụ 11: Select *from @CTPNHAP_Thang

2/8/2018

Microsoft SQL Server 2005

11

Biên hệ thống

→ Là một hàm được Microsoft SQL Server xây dựng sẵn, người dùng có thể tra cứu các giá trị hiện thời của nó nhưng không thể can thiệp trực tiếp để gán lại giá trị

→ Tên biến luôn bắt đầu bằng @@

Ví dụ 12: Xem thông tin phiên bản hệ quản trị

```
Print @@version
```

2/8/2018

Microsoft SQL Server 2005

12

Biến hệ thống

■ Các biến hệ thống thường dùng:

Tên biến	Kiểu dữ liệu	Giá trị
Connections	Số nguyên	Tổng số các kết nối vào Microsoft SQL Server từ khi nó được khởi động
Error	Số nguyên	Mã lỗi của lệnh thực hiện gần nhất (=0 nếu thực hiện thành công)
Fetch_Status	Số nguyên	Trạng thái của việc đọc dữ liệu trong bảng theo cơ chế từng mẫu tin (cursor) (=0 nếu đọc thành công)
Cursor_rows	Số nguyên	Số dòng hiện thời khi đọc dữ liệu theo cơ chế từng mẫu tin

2/8/2018

Microsoft SQL Server 2005

13

Biến hệ thống

■ Các biến hệ thống thường dùng (tiếp):

Tên biến	Kiểu dữ liệu	Giá trị
Language	Chuỗi	Tên ngôn ngữ mà hệ thống Microsoft SQL Server đang sử dụng. Mặc định là US_English
Rowcount	Số nguyên	Tổng số mẫu tin được tác động trong câu lệnh truy vấn gần nhất
ServerName	Chuỗi	Tên của máy tính cục bộ được cài đặt Microsoft SQL Server
ServiceName	Chuỗi	Tên dịch vụ
Version	Chuỗi	Thông tin về phiên bản và máy chủ cài hệ quản trị

2/8/2018

Microsoft SQL Server 2005

14

Các toán tử

■ Toán tử số học:

Áp dụng cho các kiểu dữ liệu số: int, smallint, tinyint, decimal, numeric, float, real, money, smallmoney

Các toán tử:

Ký hiệu	Ý nghĩa
+	Thực hiện phép cộng 2 số
-	Thực hiện phép trừ 2 số
*	Thực hiện phép nhân 2 số
/	Thực hiện phép chia 2 số
%	Thực hiện phép chia lấy phần dư

2/8/2018

Microsoft SQL Server 2005

15

Các toán tử

■ Toán tử nối chuỗi

- Ghép nối các chuỗi lại thành 1 chuỗi
- Ký hiệu: +

Ví dụ 13:

```
Select 'Hi' + ' every body!'
Select 'Ngày đặt hàng là: ' + convert(char(10),Ngaydh)
From DONDH
Where SoDh='D003'
```

2/8/2018

Microsoft SQL Server 2005

16

Các toán tử

■ Toán tử so sánh:

- Áp dụng cho các kiểu dữ liệu số, chuỗi, ngày
- Các toán tử: <, <=, =, >, >=, <>, !=(khác), !=>(không lớn hơn), !=<(không nhỏ hơn)

Ví dụ 14:

```
Select * From VATTU
Where PhanTram<>40
```

2/8/2018

Microsoft SQL Server 2005

17

Các toán tử

■ Toán tử logic: 3 toán tử

- Not
- And
- Or

Ví dụ 15:

```
Select * From VATTU
Where ((DvTinh!='Cái') and (PhanTram=30))
or ((DvTinh='Cái') and (PhanTram>=20))
```

2/8/2018

Microsoft SQL Server 2005

18

Biểu thức Case

■ Cú pháp Case dạng đơn giản:

```
Case Biểu_thức
  When Giá_trị_1 then Biểu_thức_kết_quả_1
  [When Giá_trị_2 then Biểu_thức_kết_quả_2
  ... ]
  [Else Biểu_thức_kết_quả_N+1]
End
```

Ví dụ 16:

Hiện thị danh sách các vật tư có trong bảng vật tư theo từng loại hàng

2/8/2018

Microsoft SQL Server 2005

19

Biểu thức Case

Giải ví dụ 16:

```
Select Loai=case left(MaVTu,2)
  when 'DD' then 'Đầu DVD'
  when 'VD' then 'Đầu VCD'
  when 'TV' then 'Tivi'
  when 'TL' then 'Tủ lạnh'
  when 'BI' then 'Bia lon'
  when 'LO' then 'Loa thùng'
  else 'Chưa phân loại'
End, MaVTu,TenVTu,DvTinh
From VATTU order by(left(MaVTu,2))
Compute count(MaVTu) by (Left(MaVTu,2))
```

2/8/2018

Microsoft SQL Server 2005

20

Biểu thức Case (tiếp)

■ Cú pháp Case dạng tìm kiếm

```
Case
  When BT_logic_1 then Biểu_thức_kết_quả_1
  [When BT_logic_2 then Biểu_thức_kết_quả_2
  ... ]
  [Else Biểu_thức_kết_quả_N+1]
End
```

Ví dụ 17: Hiện thị danh sách các vật tư có trong bảng vật tư, thông tin bổ sung thêm chuỗi **Ghi chú** tùy thuộc vào phần trăm giá bán:

```
PhanTram<20: Lờì ít
20<PhanTram<40: Lờì nhiều
PhanTram>40: Rấì lờì
```

2/8/2018

Microsoft SQL Server 2005

21

Biểu thức Case (tiếp)

Giải ví dụ 17:

```
Select TenVTu,DvTinh,PhanTram,
Ghichu=case
  when Phantram<20 then 'Lờì ít'
  when Phantram between 20 and 40 then 'Lờì nhiều'
  when Phantram>40 then 'Rấì lờì'
End
From VATTU order by PhanTram
```

2/8/2018

Microsoft SQL Server 2005

22

Biểu thức Case

Ví dụ 18: Sử dụng lệnh Update set để cập nhật lại giá bán hàng (cập nhật vào bảng CTPXUAT) trong tháng 2/2002 theo quy tắc giảm giá:

```
Nếu số lượng hàng <=2 thì không giảm giá
Nếu 3 <= số lượng hàng <=10 thì giảm 10%
Nếu số lượng hàng >10 thì giảm giá 20%
```

Giải ví dụ 18?

2/8/2018

Microsoft SQL Server 2005

23

Cấu trúc điều khiển

■ Cấu trúc rẽ nhánh if else

Cú pháp:

```
If Biểu_thức_logic
  Câu_lệnh1/Khỏi_lệnh1
Else
  Câu_lệnh2/Khỏi_lệnh2
```

Ví dụ 19: Declare @a int; set @a=-5

```
if @a>0
  print convert(char(5),@a)+' là số dương'
else if @a<0
  print convert(char(5),@a)+' là số âm'
else print 'Số 0'
```

→ Kết quả: -5 là số âm

2/8/2018

Microsoft SQL Server 2005

24

Cấu trúc điều khiển

■ Kiểm tra sự tồn tại của các dòng dữ liệu trong bảng:

→ Sử dụng If kết hợp với từ khóa exists:

```
If Exists(Câu_lệnh Select)
    Câu_lệnh1/Khối_lệnh1
Else
    Câu_lệnh2/Khối_lệnh2
```

2/8/2018

Microsoft SQL Server 2005

25

Cấu trúc điều khiển

Ví dụ 20: Kiểm tra xem có vật tư nào đã được xuất với số lượng >4 không, đưa ra các thông báo tương ứng trong từng trường hợp.

```
→ If exists(Select * From CTPXUAT where SIXuat>4)
    Begin
        Print 'Danh sách các vật tư đã xuất với số lượng > 4'
        Select CTPX.MaVTu,TenVTu,SIXuat
        From CTPXUAT CTPX inner join VATTU VT
        on CTPX.MaVTu=VT.MaVTu where SIXuat>4
    End
Else Print 'Chưa bán hàng nào với số lượng >4'
```

2/8/2018

Microsoft SQL Server 2005

26

Cấu trúc điều khiển

■ Cấu trúc lặp While

Cú pháp:

```
While Biểu_thức_logic
    Begin
        Các_lệnh
    End
```

Ví dụ 21: Câu lệnh in ra 10 số nguyên dương từ 1 đến 10

```
Declare @a int set @a=1
Print'Các số nguyên từ 1 đến 10:'
While @a<=10
    Begin
        Print convert(char(3),@a)
        Set @a=@a+1
    End
```

2/8/2018

Microsoft SQL Server 2005

27

Cấu trúc điều khiển

* Kết hợp Break (kết thúc câu lệnh lặp) và Continue (quay về đầu câu lệnh lặp) trong cấu trúc lặp While

Sơ đồ hoạt động:

```
While Biểu_thức_logic
    Begin
        Nhóm_lệnh_lặp1
        [If Biểu_thức_lặp_tiếp
            Continue]
        [If Biểu_thức_thoát
            Break]
        Nhóm_lệnh_lặp2
    End
    Các_lệnh_khác
```

2/8/2018

Microsoft SQL Server 2005

28

Cấu trúc điều khiển

■ Ví dụ 22:

```
Declare @a int
Set @a=0
While @a<=20
Begin
    Set @a=@a+1
    If @a=5 continue
    If @a>10 break
    Print convert(char(3),@a)
End
```

	Messages
	1
	2
Kết quả	3
	4
	6
	7
	8
	9
	10

2/8/2018

Microsoft SQL Server 2005

29

Thoát khỏi lô với Return

■ Ví dụ 23:

```
Declare @a int
Set @a=0
While @a<=20
Begin
    Set @a=@a+1
    If @a=5 return
    Print convert(char(3),@a)
End
```

	Messages
	1
Kết quả	2
	3
	4

2/8/2018

Microsoft SQL Server 2005

30

Cấu trúc Try ... Catch

■ Tương tự cấu trúc kiểm soát lỗi trong ngôn ngữ lập trình C#:

```
Begin Try
    Các_lệnh_khỏi_Try
End Try
Begin Catch
    Các_lệnh_khi_khỏi_Try_có_lỗi
End Catch
```

2/8/2018

Microsoft SQL Server 2005

31

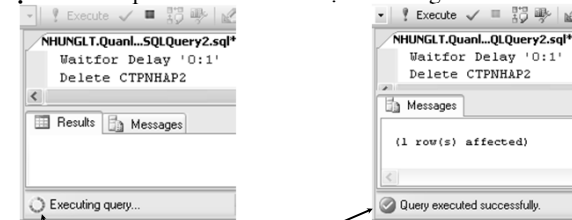
Cấu trúc Waitfor

→ Chặn việc thực thi các lệnh truy vấn SQL, các lô, các thủ tục trong thời gian chỉ định

■ Waitfor chặn thực thi lệnh trong khoảng thời gian xác định

Waitfor Delay Khoảng_thời_gian

Ví dụ 24: Chờ 1 phút sau đó xóa dữ liệu của bảng CTPNHAP2



Chờ thực thi lệnh

Thông báo lệnh đã được thực thi

2/8/2018

Microsoft SQL Server 2005

32

Cấu trúc Waitfor

- **Waitfor chặn thực thi lệnh đến một thời điểm xác định**
Waitfor Time Thời_điểm

Ví dụ 25:

Chặn việc thực thi lệnh cho đến thời điểm 10h10:
Waitfor Time '10:10'

2/8/2018

Microsoft SQL Server 2005

33

Từ khóa Goto

- Cho phép điều khiển chuyển đến vị trí của nhãn: Ví dụ 26

```
Declare @a int
Set @a=0
Khai báo nhãn → Lap:
If @a<5
  Begin
    Set @a=@a+1
    Print @a
  End
Else return
Lệnh goto → Goto Lap
Go
```

Kết quả →

Messages
1
2
3
4
5

2/8/2018

Microsoft SQL Server 2005

34

Kiểu dữ liệu Cursor

■ Khái niệm

- Cho phép làm việc với cơ sở dữ liệu theo cơ chế duyệt từng mẫu tin
- Sử dụng con trỏ có khả năng dịch chuyển truy cập tới các bản ghi trong bảng dữ liệu tham chiếu

■ Các thao tác với biến kiểu con trỏ

- Định nghĩa biến kiểu Cursor
- Mở Cursor đã định nghĩa
- Đọc và xử lý trên từng dòng dữ liệu bên trong Cursor
- Đóng Cursor

2/8/2018

Microsoft SQL Server 2005

35

Kiểu dữ liệu Cursor

■ Định nghĩa biến kiểu cursor:

Cú pháp:

```
Declare Tên_cursor Cursor
  [Local/Global]
  [Forward_Only/Scroll]
  [Static/Dynamic/Keyset]
  [Read_Only/Scroll_Lock]
For Câu_lệnh_Select
  [For update[of Danh_sách_cột_cập_nhật]]
```

2/8/2018

Microsoft SQL Server 2005

36

Kiểu dữ liệu Cursor

Trong đó:

- Tên_Cursor: Tên biến kiểu Cursor
 - Local/Global: Chỉ định phạm vi hoạt động của biến Cursor (cục bộ/toàn cục)
 - Forward_Only/Scroll: Chỉ định hướng đọc dữ liệu (chỉ theo chiều tiến/cả tiến-lùi)
 - Static/Dynamic/Keyset: Dạng dữ liệu mà con trỏ tham chiếu
 - Read_Only/Scroll_Lock: Chỉ định thao tác với dữ liệu
 - Câu_lệnh_Select: Xác định dữ liệu mà con trỏ tham chiếu (không chứa Into, compute [by])
- Danh_sách_cột_cập_nhật: Các cột được phép thay đổi giá trị trong Cursor (lưu ý mặc định cập nhật)

2/8/2018

Microsoft SQL Server 2005

37

Kiểu dữ liệu Cursor

Ví dụ 27:

```
Declare cur_Vattu Cursor  
Dynamic  
For Select * from VATTU
```

Ví dụ 28:

```
Declare cur_Nhacc Cursor  
Forward_Only  
Static  
For Select * from NHACC
```

2/8/2018

Microsoft SQL Server 2005

38

Kiểu dữ liệu Cursor

■ Mở cursor:

Cú pháp: Open Tên_cursor

Ví dụ 29: Open Cur_Vattu

Open Cur_Nhacc

2/8/2018

Microsoft SQL Server 2005

39

Kiểu dữ liệu Cursor

■ Đọc và xử lý dữ liệu trong Cursor:

Đọc dữ liệu:

```
Fetch [Next/Prior/First/Last/Absolute n/Relative n]  
From Tên_cursor  
[Into Danh_sách_biến]
```

Trong đó:

Next/Prior/First/Last: đọc dòng dữ liệu tiếp theo/trước đó/đầu tiên/cuối cùng

Absolute n: đọc dòng thứ n trong cursor

Relative n: đọc dòng dữ liệu cách dòng hiện hành là n

Danh_sách_biến: Dùng để lưu dữ liệu đọc được

2/8/2018

Microsoft SQL Server 2005

40

Kiểu dữ liệu Cursor

Lưu ý:

- Để kiểm tra việc đọc dữ liệu thành công/thất bại, cần sử dụng biến hệ thống @@Fetch_status
- Để xác định dòng dữ liệu hiện thời mà biến kiểu cursor đang tham chiếu, sử dụng cú pháp:

Current of tên_Cursor

Ví dụ 30: đọc dữ liệu từ cur_Vattu

```
Fetch next from cur_Vattu --Đọc dữ liệu
While @@Fetch_status = 0
Fetch next from cur_Vattu --Đọc các dòng kế tiếp
```

2/8/2018

Microsoft SQL Server 2005

41

Kiểu dữ liệu Cursor

■ Đóng Cursor:

Cú pháp:

Close Tên_cursor

Deallocate Tên_cursor

Trong đó:

- Close: chỉ giải phóng các dòng dữ liệu tham chiếu bên trong biến cursor
- Deallocate: xóa biến cursor khỏi bộ nhớ

Ví dụ 31:

Close cur_Vattu

Close cur_Nhacc

2/8/2018

Microsoft SQL Server 2005

42

Kiểu dữ liệu Cursor

Ví dụ 32: Tạo cursor tham chiếu tới các bản ghi lưu dữ liệu vật tư thuộc loại Ti vi trong bảng VATTU, sau đó đọc từng dòng dữ liệu trong bảng tham chiếu

2/8/2018

Microsoft SQL Server 2005

43

Kiểu dữ liệu Cursor

Giải ví dụ 32:

```
Declare cur_Vattu cursor --Khai báo biến cursor
For Select * From Vattu
where MaVTu like 'TV%'
order by MaVTu
Open cur_Vattu --Mở cursor
Fetch next from cur_Vattu --Đọc dữ liệu
While @@Fetch_status=0
Fetch next from cur_Vattu --Đọc các dòng kế tiếp
Close cur_Vattu --Đóng cursor
Deallocate cur_Vattu
```

2/8/2018

Microsoft SQL Server 2005

44

Kiểu dữ liệu Cursor

■ **Ví dụ 33:** Viết đoạn chương trình cho phép thực hiện các thao tác sau:

- Thêm cột TGNhap vào bảng PNHAP
- Sử dụng biến kiểu cursor đọc từng dòng dữ liệu trong bảng PNHAP, khi đọc đến dòng nào, cập nhật giá trị cho cột TGNhap ở dòng đó (với mỗi phiếu nhập, TGNhap được tính bằng tổng Số lượng nhập * đơn giá nhập của các chi tiết phiếu nhập tương ứng)

Giải ví dụ 33?

Kiểu dữ liệu Cursor

■ **Bài tập:** Viết đoạn chương trình cho phép thực hiện các thao tác sau:

- Tạo bảng VATTU_Temp gồm 2 cột MaVTu, TenVTu với dữ liệu lấy từ bảng VATTU
- Xóa từng dòng dữ liệu trong bảng VATTU_Temp vừa tạo, khi xóa đến dòng nào thì hiển thị thông báo Tên vật tư và Mã vật tư ở dòng đang xóa ra màn hình