

BÀI THỰC HÀNH 3

Programming Language (PL)

1. Mục tiêu

Sau khi thực hành bài này, sinh viên có thể biết sử dụng các cú pháp lệnh cơ bản, cũng như viết các chương trình đơn giản trong SQL SERVER 2000. Cụ thể như sau:

- Hiểu cách khai báo biến
- Biết vận dụng các câu lệnh đơn giản để lập trình giải quyết bài toán.
- Biết cách sử dụng một số hàm có sẵn trong SQL SERVER 2000.

2. Yêu cầu

- Sinh viên đọc kỹ các nội dung lý thuyết đã học về PL.
- Các bài tập phải được giải quyết thông qua lập trình câu lệnh.

3. Bài tập

• Bài tập 1: thực hiện các ví dụ của bài học

1. Sử dụng lệnh If ... else.... để viết đoạn mã chương trình kiểm tra nếu số mẫu tin trong bảng ORDERS >10000 thì hiển thị chi tiết bảng shippers ngược lại hiển thị chi tiết bảng customers.

Hướng dẫn:

```
If (select count(orderid)from orders) > 10000
begin
    goto x
end
Else
begin
    select * from customers
end
select * from orders
x:
select * from shippers
```

Nếu không dùng lệnh Goto thì đoạn mã trên được sửa lại như thế nào?

2. Tạo bảng Test như câu lệnh sau:

```
Create table Test (  
    Col1 int primary key,  
    Col2 char (3) not null)
```

Viết đoạn chương trình chèn cột 1 các giá trị từ 0 đến 25, cột 2 chèn các giá trị từ a đến z.

Hướng dẫn:

```
Declare @i int  
Declare @temp int  
Set @i=0  
Set @temp = 0  
While @i < 26  
    Begin  
        Insert into test values (@I, char (@temp+97))  
        Set @i = @i + 1  
        Set @temp = @temp + 1  
        If @temp > 25  
            Set @temp = 0  
    End  
select * from test
```

3. Sử dụng lệnh Case .. when để tính hàng tồn kho của từng đại lý từ cơ sở dữ liệu CSDL3

Hướng dẫn:

```
Use CSDL3  
  
select stt_dl,ma_hang,sum(solg_mua) as tg_mua into m  
from mua  
group by stt_dl,ma_hang
```

```

select stt_dl,ma_hang,sum(solg_ban) as tg_ban into b
from ban
group by stt_dl,ma_hang

```

```

select m.stt_dl,m.ma_hang,m.tg_mua,b.tg_ban,
      ton= case
            when b.tg_ban is null then tg_mua
            else (tg_mua-tg_ban)
      end
from m left join b on (m.stt_dl=b.stt_dl and m.ma_hang=b.ma_hang)
order by m.stt_dl,m.ma_hang

```

4. Từ cơ sở dữ liệu **Pubs**, viết câu lệnh SQL hiển thị nội dung của bảng **authors**, trong đó nội dung hiển thị của cột **state** cần hiển thị tên tiểu bang một cách đầy đủ.

Hướng dẫn:

```

USE PUBS

SELECT au_fname, au_lname,
      CASE state
            WHEN 'CA' THEN 'California'
            WHEN 'KS' THEN 'Kansas'
            WHEN 'TN' THEN 'Tennessee'
            WHEN 'OR' THEN 'Oregon'
            WHEN 'MI' THEN 'Michigan'
            WHEN 'IN' THEN 'Indiana'
            WHEN 'MD' THEN 'Maryland'
            WHEN 'UT' THEN 'Utah'
      END AS StateName
FROM pubs.dbo.authors
ORDER BY au_lname

```

• **Bài tập 2: Sử dụng các hàm/thủ tục có sẵn của hệ thống**

1. Sử dụng biến cục bộ và biến toàn cục

a. Thực hiện đoạn mã sau:

```
DECLARE @TEN varchar(30),@SOLG varchar(30)
SELECT @TEN=ProductName, @SOLG=QuantityPerUnit
FROM Products
WHERE ProductID=13
PRINT @TEN + ':' + @SOLG
```

Hãy cho biết đoạn mã trên làm gì? Liệt kê các biến cục bộ?

b. Thực hiện đoạn mã sau và tìm hiểu kết quả.

```
SELECT @@VERSION AS SQL_SERVER_VERSION
```

Hãy cho biết đoạn mã trên làm gì?

2. Hàm chuyển kiểu

```
SELECT 'EMP ID:' + CONVERT (CHAR(4), EMPLOYEEID)
FROM EMPLOYEES
DECLARE @myval decimal (5, 2)
SET @myval = 193.57
SELECT CAST(@myval as VARCHAR(6)) + ' VND'
```

Hãy cho biết đoạn mã trên làm gì?

3. Tra cứu từ Book-online của SQL Server 2000 để tìm hiểu về các hàm ngày tháng năm và số học.

- GETDATE()
- DATEADD(datepart,number,date)
- DATEDIFF(datepart,date1,date2)
- DATENAME(datepart,date)
- DATEPART(datepart,date)
- ABS(num_expr)
- CEILING(num_expr)
- FLOOR(num_expr)

- POWER(num_expr,y)
- ROUND(num_expr,length)
- Sign(num_expr)
- Sqrt(float_expr)

• **Bài tập 3:**

1. Thực hiện câu lệnh tạo bảng sau:

```
create table product
```

```
(prod_nr int not null primary key,  
  name varchar(30) not null,  
  price money not null,  
  type varchar(30) not null)
```

2. Thực hiện các câu lệnh thêm dữ liệu vào bảng sau:

```
insert into product values (4,'ColorTv',700,'electronic')
```

```
insert into product values (5,'Fan',350,'electronic')
```

```
insert into product values (6,'Heater',200,'electronic')
```

```
insert into product values (7,'Webcam',60,'Computer')
```

3. Viết câu truy vấn bảng product và quan sát dữ liệu.
4. Viết một chương trình để tính trị giá trung bình của mỗi sản phẩm, ghi trị giá này ra một biến và hiển thị trị giá trung bình này thông qua biến vừa ghi.
5. Viết một chương trình để tính trị giá trung bình của mỗi sản phẩm. *Kiểm tra nếu trị giá trung bình của mỗi sản phẩm ≥ 500 thì hiển thị bảng giá chi tiết của mỗi sản phẩm. Ngược lại thực hiện công việc tăng đều giá mỗi sản phẩm lên 5% so với giá hiện tại của nó. Công việc kiểm tra và tăng giá được thực hiện liên tục đến khi nào trị giá trung bình ≥ 500 thì dừng lại và hiển thị trị giá trung bình và liệt kê bảng giá chi tiết của mỗi sản phẩm sau khi tăng.*

• **Bài tập 4:**

Viết đoạn mã chương trình cập nhật cột **hire-date** của bảng **employee** trong cơ sở dữ liệu **Pubs** tăng thêm 1 năm cho mỗi mẫu tin.