

## THỦ TỤC LƯU TRỮ (Stored Procedures)

### Nội dung

- Khái niệm thủ tục lưu trữ
- Ưu điểm của thủ tục lưu trữ
- Các loại thủ tục lưu trữ
- Danh mục thủ tục lưu trữ hệ thống
- Tạo lập, thực thi thủ tục lưu trữ
- Tham số trong thủ tục lưu trữ
- Biên dịch lại thủ tục lưu trữ

### Thủ tục lưu trữ

- Là một nhóm các câu lệnh được biên dịch thành một kế hoạch thực thi nhằm thực hiện một nhiệm vụ cụ thể.
- Được viết bởi những người phát triển CSDL (database developers) hoặc DBA
- Sử dụng để thực hiện các chức năng quản trị, hoặc áp dụng các quy luật giao dịch phức tạp
- Chứa các câu lệnh DML

### Thuận lợi:

- Tăng tốc độ thực thi
- Truy cập dữ liệu nhanh hơn
- Lập trình theo Mô-đun
- Đồng nhất
- Tăng cường bảo mật

## Hai loại thủ tục lưu trữ

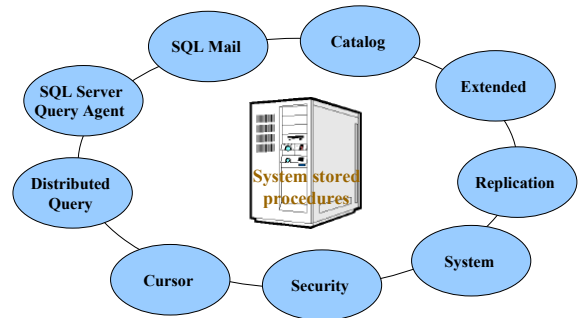
- Các thủ tục lưu trữ hệ thống  
(Chỉ có thể được thực thi)



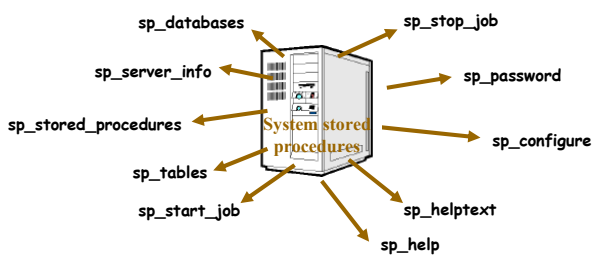
- Các thủ tục người dùng định nghĩa  
(Có thể được tạo ra và thực thi)



## Danh mục thủ tục lưu trữ hệ thống



## Ví dụ:



## Thủ tục lưu trữ do người dùng định nghĩa

- Được tạo bởi người dùng trong CSDL hiện hành
- Chủ sở hữu CSDL có quyền mặc định tạo thủ tục lưu trữ

### Cú pháp:

```
CREATE PROC[EDURE] {procedure_name}  
AS {STATEMENTS}
```

### Ví dụ

USE Nothwind

```
CREATE PROCEDURE PROC_CUSTOMERS AS  
SELECT * FROM CUSTOMERS  
WHERE CITY = 'London'
```

### Cú pháp

```
EXEC[UTE] procedure_name
```

### Ví dụ

```
EXEC PROC_CUSTOMERS
```

### Tham số trong thủ tục lưu trữ

### Cú pháp

```
CREATE PROCEDURE procedure_name  
@Parameter_name data_type  
AS {STATEMENTS}
```

### Ví dụ:

```
USE PUBS  
-----  
CREATE PROCEDURE Proc_Titles  
    @v_pubid char(4)  
AS  
    SELECT * FROM titles WHERE pub_id = @v_pubid  
-----  
EXEC Proc_Titles '1389'
```

## Chỉnh sửa các thủ tục lưu trữ

- Câu lệnh ALTER PROCEDURE được sử dụng để sửa chữa một thủ tục lưu trữ (Cú pháp giống như lệnh CREATE PROCEDURE)
- Sự thay đổi này vẫn giữ lại các quyền người dùng

## Biên dịch lại các thủ tục lưu trữ

Có ba cách để biên dịch lại các thủ tục:

- Sử dụng thủ tục hệ thống sp\_recompile
- WITH RECOMPILE với lệnh CREATE PROCEDURE
- WITH RECOMPILE với lệnh EXECUTE

## Biên dịch lại các thủ tục lưu trữ

Syntax:

```
sp_recompile [ @objname = ] 'object'
```

Example:

```
EXEC sp_recompile titles
```

## Biên dịch lại thủ tục lưu trữ

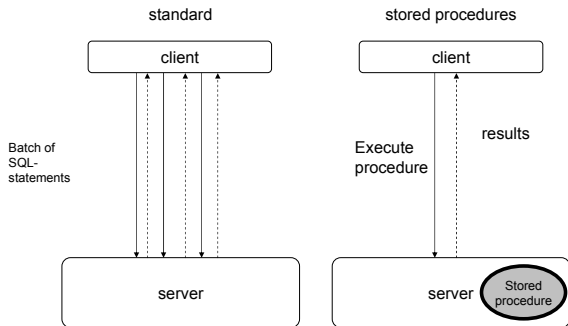
Syntax: WITH RECOMPILE với câu lệnh Create Procedure

```
CREATE PROCEDURE procedure_name  
  @Parameter_name data_type  
  With Recompile  
  AS {STATEMENTS}
```

Syntax: WITH RECOMPILE với câu lệnh Execute

```
EXEC {Procedure Name} With Recompile
```

## Stored Procedures vs SQL



## Hàm (Functions)

### ■ Hàm

- Là một phần của câu lệnh T-SQL
- Được biên dịch và tối ưu trước
- Thực hiện như một đơn vị duy nhất
- Có thể thực hiện trong 1 câu truy vấn

### ■ Hai loại

- Hàm trả về giá trị đơn
- Hàm trả về một bảng

## Hàm trả về giá trị đơn (1)

```
CREATE FUNCTION DayOnly (@date DATETIME)
RETURNS varchar(10)
AS
BEGIN
RETURN CONVERT(VARCHAR(10), @date, 101)
END
```

```
SELECT dbo.DayOnly(GETDATE())
```

Results:

-----  
09/16/2008

## Hàm trả về giá trị đơn (2)

```
CREATE TABLE test (id INT IDENTITY PRIMARY KEY,
testDate DATETIME NOT NULL)

GO

DECLARE @count INT
SET @count = 1
WHILE @count <= 10
BEGIN
INSERT test VALUES (DATEADD(MINUTE, @count, GETDATE()))
SET @count = @count + 1
END

GO

-- No results:
SELECT * FROM test WHERE testDate = GETDATE()

GO

-- This works:
SELECT * FROM test WHERE dbo.DayOnly(testDate) = dbo.DayOnly(GETDATE())
```

## Hàm trả về một bảng (1)

```
Create Function fnProducts(@type varchar(20))
```

```
Returns Table
```

```
As Return
```

```
Select prod_nr,[name]
```

```
From product
```

```
Where [type]=@type
```

```
Select * From dbo.fnProducts('electronics')
```