

1/ Function

<pre> *Hàm trả về 1 giá trị: create function Tên_Hàm returns Kiểu_dữ_liệu_trả_về as begin return Giá_trị_trả_về end -> PRINT [dbo].Tên_Hàm(Tham số) </pre>	<pre> *Hàm đọc bảng: create function Tên_Hàm(Khai_báo_các_tham_số) returns table as return câu_Select_trả_về_1_bảng_dữ_liệu -> SELECT * FROM [dbo].Tên_Hàm(Tham số) </pre>
<pre> *Hàm tạo bảng: create function Tên_Hàm (Khai_báo_các_tham_số) returns Tên_bảng table (Các_cột_dữ_liệu_bảng_trả_về) as begin --- Có thêm lệnh Insert để thêm data vào bảng tạm -- end </pre>	

2/ Store Procedure

<pre> *Truyền tham số create proc Tên_Hàm(TS vào, TS ra OUTPUT) as begin return Giá_trị_trả_về end -> EXEC [dbo].Tên_hàm – Các tham số -- </pre>	<pre> *Không tham số create proc Tên_Hàm as --- Câu lệnh thực thi --- -> EXEC [dbo].Tên_hàm </pre>
---	---

3/ Cursor

```

DECLARE tên_Cursor CURSOR
FOR SELECT ... (dùng cho bảng nào ?)
OPEN tên_Cursor
DECLARE – Tên biến chứa data từ Cursor –
FETCH NEXT FROM tên_Cursor INTO –tên biến—
WHILE (@@FETCH_STATUS = 0)
    BEGIN
        -- Câu lệnh thực thi --
        FETCH NEXT FROM tên_Cursor INTO –tên biến—
    END
CLOSE tên_Cursor
DEALLOCATE tên_Cursor

```

4/ Trigger

```

CREATE TRIGGER tên_Trigger ON tên_Bảng
INSTEAD OF | FOR (INSERT, UPDATE, DELETE)
AS

```

-- Khởi lệnh –

** Lưu ý:

Data tạm sẽ lưu trong table Inserted hoặc Deleted

Dùng các biến để chứa data lấy từ 1/2 table trên

INSTEAD OF: chạy trước ràng buộc (ex: Thông báo lỗi, ...)

FOR:chạy sau (thực thi ràng buộc xong mới thực hiện)

RAISERROR: Xuất thông báo lỗi

COMMIT TRANSACTION: Báo hiệu thành công

ROLLBACK TRANSACTION: Thất bại, quay lại ban đầu