

TRANSACT-SQL Programming

```
DROP {PROC|PROCEDURE} [SchemaName.]ProcedureName [,...n ]

CREATE {PROC|PROCEDURE} [SchemaName.]ProcedureName
    [{@parameter type} [VARYING] [= defaultValue] [OUT|OUTPUT] ] [,...n]
    AS { <sql_block> [ ...n] }{;}
<sql_block> ::= {instruction|BEGIN instructions END }

RETURN [integer_expression]

[{EXEC|EXECUTE}][@variable =] nb_proc  [{@par=value [OUTPUT] }[,...n ]];

EXEC('sql_string');

OBJECT_ID(object_name,type)                @@ROWCOUNT
SUSER_SNAME()

SET IDENTITY_INSERT table_name [ON|OFF];

DECLARE @var type;

PRINT expression;

[SET | SELECT] @variableName = value;

IF condition
    sql_block
[ELSE
    sql_block ]

WHILE condition
    sql_block | BREAK | CONTINUE

WAITFOR {DELAY 'time_to_wait' |TIME 'date_of_execution'}

GOTO LabelName

BEGIN TRY
    sql_block
END TRY
BEGIN CATCH
    [sql_block]
END CATCH [ ; ]

ERROR_NUMBER()          ERROR_SEVERITY()      ERROR_STATE()
ERROR_PROCEDURE()       ERROR_LINE()          ERROR_MESSAGE()

CREATE TRIGGER [SchemaName.]TriggerName ON {table | view } [,...n] ]
    {FOR|AFTER|INSTEAD OF} {[INSERT][,][UPDATE][,][DELETE]}
    AS sql_block [;] [,...n ]

COLUMN(columnName)                COLUMNS_UPDATED()

ALTER TRIGGER [SchemaName.]TriggerName ON {table|view}
{FOR|AFTER|INSTEAD OF} {[INSERT][,][UPDATE][,][DELETE]} [WITH APPEND]
AS sql_block

DISABLE TRIGGER [SchemaName.]TriggerName [,...n] | ALL }
ON {NombreTableView | DATABASE | ALL SERVER} [;]

ENABLE TRIGGER {[SchemaName.]TriggerName [,...n] | ALL }
ON {NombreTableView | DATABASE | ALL SERVER} [ ; ]

DROP TRIGGER SchemaName.TriggerName [,...n] [;]
```

```
DECLARE cursor_name [ INSENSITIVE ] [ SCROLL ] CURSOR
  FOR select_statement
  [ FOR { READ ONLY | UPDATE [ OF column_name [ ,...n ] ] } ][;]
```

```
OPEN { cursor_name | cursor_variable_name }
```

```
CLOSE { cursor_name | cursor_variable_name }
```

```
DEALLOCATE { cursor_name | @cursor_variable_name }
```

```
FETCH [ [NEXT |PRIOR |FIRST |LAST |ABSOLUTE { n | @nvar } |RELATIVE { n | @nvar}]
  FROM ]
  { cursor_name | cursor_variable_name }
  [ INTO @variable_name [ ,...n ] ] [;]
```