www.PragimTech.com

Pragim@PragimTech.com

Part 30 – User defined functions

Venkat
PRAGIM Technologies
kudvenkat@gmail.com
http://csharp-video-tutorials.blogspot.com

In this session we will learn

- User Defined Functions
- Types of User Defined Functions
- Creating a Scalar User Defined Function
- Calling a Scalar User Defined Function
- Places where we can use Scalar User Defined Function
- Altering and Dropping User Defined Function

SCALAR UDF

From Parts 22 to 29, we have learnt how to use many of the system functions that are available in SQL Server. In this session, we will turn our attention, to creating user defined functions. In short UDF.

In SQL Server there are 3 types of User Defined functions
1. Scalar functions
2. Inline table-valued functions
3. Multi-statement table-valued functions

Scalar functions may or may not have parameters, but always return a single (scalar) value. The returned value can be of any data type, except text, ntext, image, cursor, and timestamp

```
--To create a function, we use the following syntax:
CREATE FUNCTION Function_Name(@Parameter1 DataType, @Parameter2 DataType,..@ParameterN Datatype)
RETURNS Return_Datatype
AS
BEGIN
   --Function Body
    Return Return_Datatype
END
```

SCALAR UDF

```
CREATE FUNCTION Age(@DOB Date)
RETURNS INT
AS
BEGIN
    DECLARE @Age INT
    SET @Age = DATEDIFF(YEAR, @DOB, GETDATE()) -
                    CASE
                      WHEN (MONTH(@DOB) > MONTH(GETDATE())) OR
                           (MONTH(@DOB) = MONTH(GETDATE()) AND DAY(@DOB) > DAY(GETDATE()))
                      THEN 1
                      ELSE 0
                     END
    RETURN @Age
END
```

When calling a scalar user-defined function, you must supply a two-part name, OwnerName.FunctionName. dbo stands for database owner.

```
Select dbo.Age('10/08/1982')
```

You can also invoke it using the complete 3 part name, DatabaseName.OwnerName.FunctionName

```
Select SampleDB.dbo.Age( dbo.Age('10/08/1982')
```

SCALAR UDF

```
--Scalar user defined functions can be used in the Select clause
Select Name, DateOfBirth, dbo.Age(DateOfBirth) as Age from tblEmployees
```

Id	Name	DateOfBirth
1	Sam	1980-12-30 00:00:00.000
2	Pam	1982-09-01 12:02:36.260
3	John	1985-08-22 12:03:30.370
4	Sara	1979-11-29 12:59:30.670

Name	DateOfBirth	Age
Sam	1980-12-30 00:00:00.000	31
Pam	1982-09-01 12:02:36.260	30
John	1985-08-22 12:03:30.370	27
Sara	1979-11-29 12:59:30.670	32

```
--Scalar user defined functions can be used in the Select Where clauses
Select Name, DateOfBirth, dbo.Age(DateOfBirth) as Age from tblEmployees
Where dbo.Age(DateOfBirth) > 30
```

Id	Name	DateOfBirth
1	Sam	1980-12-30 00:00:00.000
2	Pam	1982-09-01 12:02:36.260
3	John	1985-08-22 12:03:30.370
4	Sara	1979-11-29 12:59:30.670

Name	DateOfBirth	Age
Sam	1980-12-30 00:00:00.000	31
Sara	1979-11-29 12:59:30.670	32

A stored procedure also can accept DateOfBirth and return Age, but you cannot use stored procedures in a select or where clause. This is just one difference between a function and a stored procedure. There are several other differences, which we will talk about in a later session.

To alter a function we use ALTER FUNCTION FuncationName statement and to delete it, we use DROP FUNCTION FuncationName.

Additional Resources

- PRAGIM Home Page:
  - http://www.PragimTech.com
- Resources:
  - ASP.NET Interview Questions
  - http://www.VenkatASPInterview.Blogspot.com
  - C# Interview Questions
  - http://www.VenkatCSharpInterview.Blogspot.com

PRAGIM Technologies | www.pragimtech.com | 9900113931

http://csharp-video-tutorials.blogspot.com