SENAI SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL

Microsoft SQL Server

Aula 19 – Funções no T-SQL

O que iremos aprender?

- Funções definidas pelo usuário;
- Funções dentro de um loop;
- Usando funções como tabelas;
- Alterando uma função;
- Excluindo uma função.

Funções definidas pelo usuário

Já temos várias funções disponíveis nos SQL Server para serem usadas, inclusive já vimos algumas como:

- Strings – **SUBSTRING, LEFT, LTRIM,** etc ...
- Datas – **DATEADD, DATEDIFF**, etc ...
- Números – **CELLING, FLOOR, RAND,** etc ...
- Conversão de dados – **CAST** e **CONVERT**

Agora vamos poder criar as nossas próprias funções e usá-las nas consultas.

Funções definidas pelo usuário

As funções são rotinas que efetuam um determinado processamento e retornam uma valor (escalar ou uma tabela)

Podemos desenvolver funções em T-SQL ou em .NET e usá-las no SQL Server

Chamamos de funções UDF (*User-Defined Functions*) as funções definidas pelo usuário desenvolvedor no SQL Server.

Daremos foco apenas na criação de funções usando o Transact SQL.

Funções definidas pelo usuário

Uma função UDF escalar é aquela que retorna um valor. Possui as seguintes características:

- O seu corpo deve ser delimitado por um **BEGIN** e **END**;
- Deve sempre retornar um valor através do comando **RETURN;**
- Após o nome da função devemos declarar as variáveis de entrada.

Funções definidas pelo usuário

```
CREATE FUNCTION <NOME DA FUNÇÃO> (<PARÂMETROS DE ENTRADA>)
RETURNS <TIPO DE RETORNO>
AS
BEGIN

      <CORPO DA FUNÇÃO>

RETURN <VARIÁVEL DE RETORNO>
END
```


DESAFIO

Desafio 01

Em exercícios anteriores, construímos um script para obter o número de notas fiscais de uma determinada data. Veja-o abaixo:

```
DECLARE @NUMNOTAS INT
SELECT @NUMNOTAS = COUNT(*) FROM [TABELA DE NOTAS FISCAIS]
    WHERE DATA = '20170101'
PRINT @NUMNOTAS
```

Transforme este script em uma função onde passamos a data como parâmetro e retornamos o número de notas. Chame esta função de NumeroNotas. Após a sua criação, teste seu uso com um SELECT.

Funções dentro de um loop

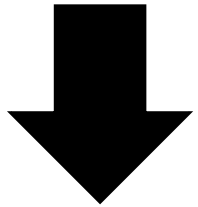
Mais um exemplo


DESAFIO

Desafio 02

Em exercícios anteriores construímos um script para criar uma tabela com o número de notas fiscais para um período de datas. Veja o script abaixo:

```
CREATE TABLE TABELANOTAS (DATA DATE, NUMNOTAS INT)
DECLARE @DATAINICIAL DATE
DECLARE @DATAFINAL DATE
DECLARE @NUMNOTAS INT
SET @DATAINICIAL = '20170101'
SET @DATAFINAL = '20170110'
WHILE @DATAINICIAL <= @DATAFINAL
BEGIN
   SELECT @NUMNOTAS = COUNT(*) FROM [TABELA DE NOTAS
FISCAIS]
    WHERE DATA = @DATAINICIAL
   INSERT INTO TABELANOTAS (DATA, NUMNOTAS)
    VALUES (@DATAINICIAL, @NUMNOTAS)
   SELECT @DATAINICIAL = DATEADD(DAY, 1, @DATAINICIAL)
END
SELECT * FROM TABELANOTAS
```



Reescreva este script usando a função NumeroNotas no momento de inserir dados na tabela. Execute o SELECT para exibir os dados.

Usando funções como tabela

Mais exemplo!!!


DESAFIO

Desafio 03

Veja a consulta abaixo:

```
SELECT DISTINCT DATA, [dbo].[NUMERONOTAS](DATA) AS 'NUMERO'
 FROM [TABELA DE NOTAS FISCAIS]
WHERE DATA >= '20170101' AND DATA <= '20170110'
```

Ela irá retornar o número de notas fiscais entre duas datas. Transforme isto em uma função chamada FuncTabelaNotas, onde o resultado é a consulta acima. Lembrando que a data inicial e final serão parâmetros desta função. Depois, teste a função através de um SELECT.

Alterando Função

Mais um exemplo:

- Criar uma outra função
- Retornar sempre um endereço completo do cliente.
- Alteração dessa função


DESAFIO

Desafio 04

Veja a consulta abaixo:

```
SELECT DATA, COUNT(*) AS NUMERO FROM [TABELA DE NOTAS FISCAIS]
WHERE DATA >= '20170101' AND DATA <= '20170110'
GROUP BY DATA
```

Ela também retorna o número de notas entre duas datas. Modifique a função FuncTabelaNotas para que utilize esta consulta acima.

Excluindo uma função

```
DROP FUNCTION <NOME DA FUNCAO>
```

O que aprendemos nesta aula

- Como ter uma tabela como resultado de uma função;
- Como usar funções dentro de um loop;
- Como funciona as funções no SQL Server;

- Como alterar e excluir funções.