

Banco de Dados : Functions

Prof. Márcio Funes



Plano de aula

Functions

Parâmetros

Variáveis

Estruturas de controle

Exercícios

Introdução

O contexto

Uma **Function** é um rotina armazenada no banco de dados. Elas utilizadas para retornar valores quando chamadas.

Podem conter comandos SQL, além de algoritmos com variáveis, loops e estruturas de decisão.

Um **Function** é semelhante a um método com retorno na linguagem Java.

Podem ainda ser usadas para restringir acesso a grupos de usuários, aumentando com isso a segurança do banco de dados e diminuindo o tráfego na rede.

Functions

Functions

Veja abaixo uma sintaxe simples a para criação de uma Function:

SQL Server

```
CREATE FUNCTION nome_da_function  
RETURNS tipo_dado AS  
BEGIN  
    --código sql  
    RETURN valor  
END
```

Functions

Veja abaixo uma sintaxe simples a para criação de uma Function:

MySQL

```
create function nome_da_function  
    returns tipo_retorno  
    returns valor;
```

Functions

Comparativo entre SQL Server e MySQL:

MySQL

```
create function nome_da_function  
    returns tipo_retorno  
    returns valor;
```

SQL Server

```
CREATE FUNCTION nome_da_function  
RETURNS tipo_dado AS  
BEGIN  
    --código sql  
    RETURN valor  
END
```

Functions

Para criar uma function que simplesmente mostra uma mensagem:

```
CREATE FUNCTION hello()  
RETURNS varchar(30) AS  
BEGIN  
    RETURN('Hello World')  
END
```

```
SELECT dbo.hello()
```


Functions

Para criar uma function que simplesmente mostra uma mensagem:

```
create function hello()  
  returns varchar(30) deterministic  
  return "Hello World" $
```

Functions

Para criar uma function que simplesmente mostra uma mensagem:

```
CREATE FUNCTION hello2(@nome varchar(30))  
RETURNS varchar(30) AS  
BEGIN  
    RETURN concat('Hello', @nome)  
END  
  
SELECT dbo.hello2('Marcio')
```

Functions

Para criar uma function que simplesmente mostra uma mensagem:

```
create function hello(str varchar(20))  
  returns varchar(30) deterministic  
  return concat("Hello ", str) $
```

Functions

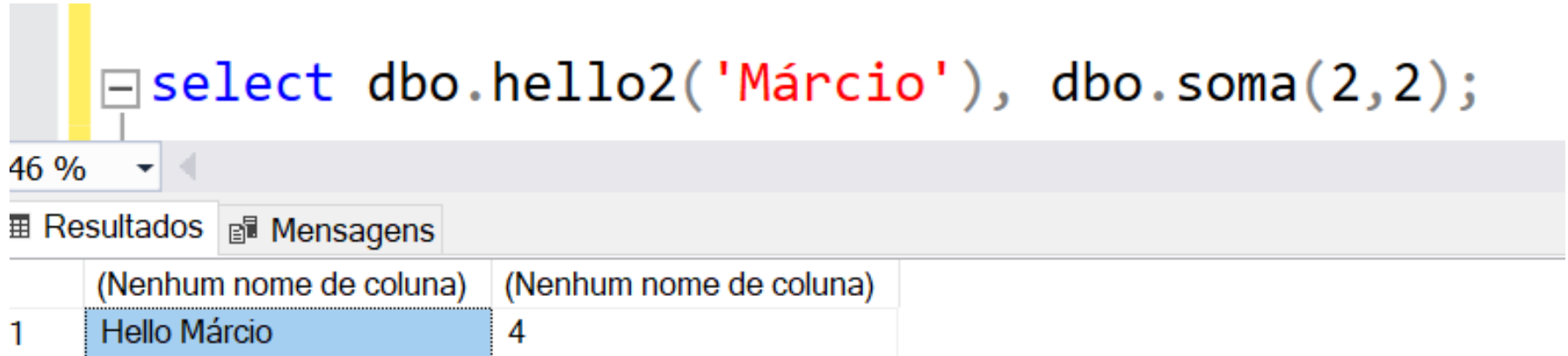
Para criar uma function que simplesmente soma dois números:

```
CREATE FUNCTION soma(@a int, @b int)
RETURNS int AS
BEGIN
    RETURN @a + @b;
END

SELECT dbo.soma(2,2)
```

Functions

Uma function pode ser chamada em qualquer contexto, bastando inserir seu nome e lista de parâmetros.



The screenshot shows a SQL query editor with the following text:

```
select dbo.hello2('Márcio'), dbo.soma(2,2);
```

Below the query, there is a progress bar at 46 % and two tabs: "Resultados" (selected) and "Mensagens". The "Resultados" tab displays the following table:

| | (Nenhum nome de coluna) | (Nenhum nome de coluna) |
|---|-------------------------|-------------------------|
| 1 | Hello Márcio | 4 |

Functions

Exercício 1 - Faça uma função que retorne o lucro de um produto tendo como parâmetro seu ID

```
CREATE TABLE produtos(  
    id int primary key identity,  
    nome varchar(30),  
    preco_custo decimal(9,2),  
    preco_venda decimal(9,2),  
)
```

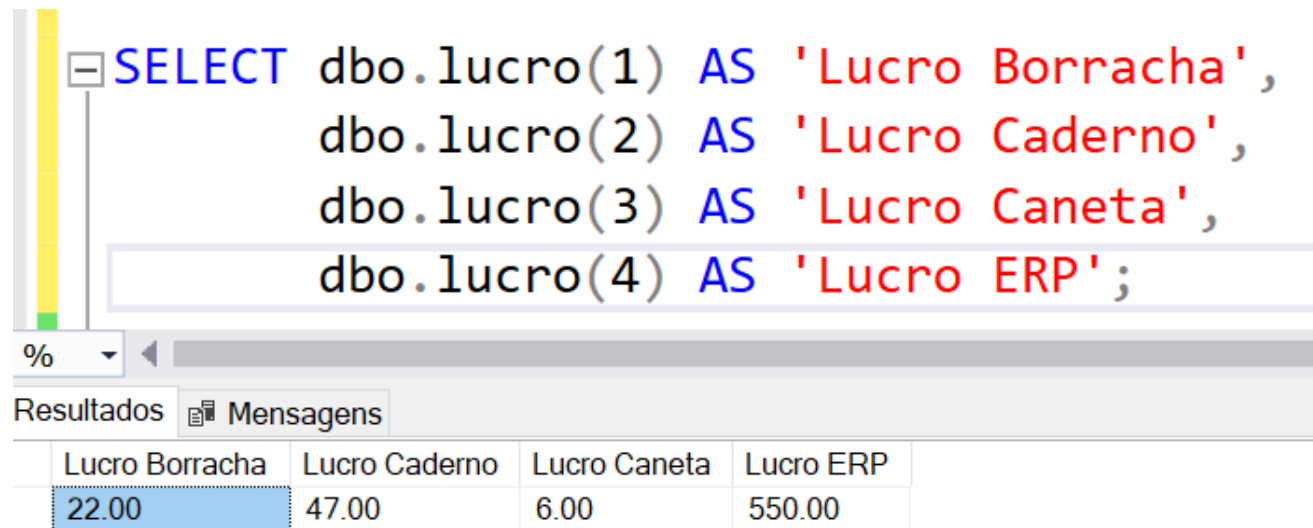
```
insert into produtos values ('borracha', 10, 12);  
insert into produtos values ('caderno', 20, 27);  
insert into produtos values ('caneta', 1, 5);  
insert into produtos values ('ERP', 200, 350);
```

Functions

Uma function pode fazer consultas a tabelas, criar variáveis a fim de retornar o resultado esperado.

```
CREATE FUNCTION lucro(@produto_id int)
RETURNS decimal(9,2) AS
BEGIN
    DECLARE @custo AS decimal(9,2), @venda AS decimal(9,2), @lucro AS decimal(9,2);
    SET @custo = (SELECT preco_custo FROM produtos WHERE id = @produto_id);
    SET @venda = (SELECT preco_venda FROM produtos WHERE id = @produto_id);
    SET @lucro = @custo + @venda;

    RETURN @lucro;
END
GO;
```



The screenshot shows a SQL query window with the following query:

```
SELECT dbo.lucro(1) AS 'Lucro Borracha',
       dbo.lucro(2) AS 'Lucro Caderno',
       dbo.lucro(3) AS 'Lucro Caneta',
       dbo.lucro(4) AS 'Lucro ERP';
```

Below the query window, the 'Resultados' (Results) tab is active, displaying the following table:

| Lucro Borracha | Lucro Caderno | Lucro Caneta | Lucro ERP |
|----------------|---------------|--------------|-----------|
| 22.00 | 47.00 | 6.00 | 550.00 |

Functions prédefinidas

Há muitas funções já implementadas e disponíveis. Veja alguns dos principais exemplos:

```
SELECT UPPER('Fatec Franca SP');
SELECT LOWER('Fatec Franca SP');
SELECT LEN('Fatec Franca SP');
SELECT DATALENGTH(1234);
SELECT TRIM('  Fatec Franca SP  ');
SELECT REVERSE('Fatec Franca SP');
SELECT ROUND(5.1234,3);
SELECT DIFFERENCE('Teste', 'Teste');
SELECT LEFT('Fatec Franca', 3);
SELECT RIGHT('Fatec Franca', 3);
SELECT REPLACE('Fatec Franca SP', 'Franca', 'São Paulo');
SELECT REPLICATE('Fatec Franca SP', 5);
SELECT SPACE(10); --????
SELECT STUFF('Fatec Franca', 1, 5, 'Unifran');
SELECT SUBSTRING('Fatec Franca', 1, 3);
SELECT UNICODE('Atlanta');
```


Functions prédefinidas

Há muitas funções já implementadas e disponíveis. Veja alguns dos principais exemplos:

https://www.w3schools.com/sql/func_sqlserver_unicode.asp

Stored Procedures x Functions

Procedimentos e funções podem ser utilizadas para automatizar e parametrizar tarefas. Contudo,

- Functions são pensadas para **retornar valores** para consultas e outros funções.
- Procedures por sua vez, são utilizadas em aplicações que se comunicam com o banco.

Functions prédefinidas

Há muitas funções já implementadas e disponíveis. Veja alguns dos principais exemplos:

https://www.w3schools.com/sql/sql_ref_sqlserver.asp

<https://learn.microsoft.com/en-us/sql/t-sql/functions/functions?view=sql-server-ver16>

Exercícios

Exercícios Extra

Utilize o script Universidade.sql onde cada curso pode pertencer a somente uma área.

Utilize Stored Procedures e Functions para automatizar a inserção e seleção dos cursos.

1 – Crie uma função que ao inserir um aluno deve ter seu endereço gerado automaticamente no seguinte formato: nome.sobrenome@dominio.com

3- Crie uma função que recebe o nome de um curso e sua área, em seguida retorna o id do curso.

3 - Crie uma procedure que recebe o os dados do aluno e de um curso e faz sua matrícula. Caso o aluno já esteja matriculado em um curso, essa matrícula não pode ser realizada.