

PARA
SER,
UTP.

utp.edu.br

TRIGGERS

TRIGGERS

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

Triggers

- Triggers são “gatilhos” disparados mediante alguma ação.

Schema Browser

- [CAIXA](#) (TABLE)
 - DATA datetime(23)
 - SALDO_INICIAL decimal(10)
 - SALDO_FINAL decimal(10)
- [VENDAS](#) (TABLE)
 - DATA datetime(23)
 - CODIGO int(10)
 - VALOR decimal(10)

```
CREATE TABLE Caixa (  
    DATA          DATETIME,  
    SALDO_INICIAL  DECIMAL(10,2),  
    SALDO_FINAL    DECIMAL(10,2)  
)  
  
INSERT INTO Caixa  
VALUES (CONVERT(DATETIME, CONVERT(VARCHAR, GETDATE()), 101), 100,  
100)  
  
CREATE TABLE Vendas (  
    DATA          DATETIME,  
    CODIGO         INT,  
    VALOR          DECIMAL(10,2)  
)
```

Triggers

```
CREATE TRIGGER TGR_VENDAS_AI
ON VENDAS
FOR INSERT
AS
BEGIN
    DECLARE
        @VALOR    DECIMAL(10,2),
        @DATA     DATETIME

    SELECT @DATA = DATA, @VALOR = VALOR FROM
    INSERTED

    UPDATE CAIXA SET SALDO_FINAL = SALDO_FINAL +
    @VALOR
    WHERE DATA = @DATA
END
```



Instrução Join

Inserindo dados em vendas

```
INSERT INTO Vendas  
VALUES (CONVERT(DATETIME, CONVERT(VARCHAR, GETDATE(),  
101)), 1, 10);
```

Selecionando os dados

```
SELECT * FROM Caixa
```

DATA	SALDO_INICIAL	SALDO_FINAL
2018-02-14T00:00:00Z	100	110

```
SELECT * FROM Vendas
```

DATA	CODIGO	VALOR
2018-02-14T00:00:00Z	1	10



Triggers

```
CREATE TRIGGER TGR_VENDAS_AD
ON VENDAS
FOR DELETE
AS
BEGIN
    DECLARE
        @VALOR  DECIMAL(10,2),
        @DATA   DATETIME

    SELECT @DATA = DATA, @VALOR = VALOR FROM DELETED

    UPDATE CAIXA SET SALDO_FINAL = SALDO_FINAL - @VALOR
    WHERE DATA = @DATA
END
```

```
DELETE FROM VENDAS WHERE CODIGO = 1
```

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

PARA
SER,
UTP.

utp.edu.br

STORED PROCEDURES

Procedures

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

Stored Procedures

- É uma coleção de comandos SQL executados no banco de dados que encapsula tarefas repetitivas, aceita parâmetros de entrada e retorna um valor de status para indicar se houve ou não falha na execução.

```
CREATE PROCEDURE ListaAvioes  
AS  
    SELECT * FROM Planes  
    RETURN  
  
EXEC ListaAvioes
```

ID	Model	Built_date	Number_of_seats
1	Cesna	2012-06-18	10
2	Foker 100	2012-06-18	100

Stored Procedures

- É possível fornecer parâmetros de entrada para tornar o procedimento mais flexíveis.

```
CREATE PROCEDURE ListaAvioesLugares @Lugares
int
AS
    SELECT * FROM Planes WHERE Number_of_seats
    >= @Lugares
    RETURN

EXEC ListaAvioesLugares 50
```

ID	Model	Built_date	Number_of_seats
2	Foker 100	2012-06-18	100

PARA
SER,
UTP.

utp.edu.br

STORED FUNCTIONS

Functions

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

Stored Functions

- É uma coleção de comandos SQL executados no banco de dados que encapsula tarefas repetitivas, aceita parâmetros de entrada e retorna um valor.

```
CREATE FUNCTION dbo.Ola()  
RETURNS varchar(100)  
AS  
BEGIN  
    RETURN "Ola!!"  
END;
```

```
SELECT dbo.ola() AS Funcao_Ola
```

Resultados		Mensagens	
Funcao_Ola			
1	Ola!		

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

Stored Functions

- É uma coleção de comandos SQL executados no banco de dados que encapsula tarefas repetitivas, aceita parâmetros de entrada e retorna um valor.

```
CREATE FUNCTION dbo.CELSIUS_TO_FAHRENHEIT (@celsius real)
RETURNS real
AS
BEGIN
    RETURN @celsius*1.8+32
END;
```

```
SELECT dbo.CELSIUS_TO_FAHRENHEIT(0)
```

Resultados		Mensagens	
(Nenhum nome de coluna)			
1	32		

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

Procedure/Function IMC

- Escreva uma Função/Procedimento para cálculo do IMC.
 - <16 Magreza grave
 - 16 a 16,9 Muito abaixo do peso
 - 17 a 18,4 Abaixo do peso
 - 18,5 a 24,9 **Peso normal**
 - **25 a 29,9** Acima do peso
 - 30 a 34,9 Obesidade Grau I
 - 35 a 40 Obesidade Grau II
 - >40 Obesidade Grau III
-
- $IMC = Massa / (Altura * Altura)$

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

Stored Functions

- É uma coleção de comandos SQL executados no banco de dados que encapsula tarefas repetitivas, aceita parâmetros de entrada e retorna um valor.

```
CREATE FUNCTION
dbo.CELSIUS_TO_FAHRENHEIT(@celsiusreal)
RETURNS real
AS
BEGIN
    RETURN @celsius*1.8+32
END;
```

```
SELECT dbo.CELSIUS_TO_FAHRENHEIT(0)
```

Resultados		Mensagens	
(Nenhum nome de coluna)			
1	32		

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

```
CREATE FUNCTION dbo.IMC2(@peso REAL, @altura REAL)
RETURNS VARCHAR(100)
AS
BEGIN
    DECLARE @imc REAL
    DECLARE @retorno VARCHAR(100)

    SET @imc = @peso / (@altura * @altura)

    SET @retorno = 'Nao determinado!'
    IF @imc < 16
        SET @retorno = 'Magreza grave'
    ELSE
        IF (@imc >= 16) AND (@imc < 17)
            SET @retorno = 'Muito abaixo do peso'
        ELSE
            IF (@imc >= 17) AND (@imc < 18.5)
                SET @retorno = 'Abaixo do peso'
            ELSE
                IF (@imc >= 18.5) AND (@imc < 25)
                    SET @retorno = 'Normal'
                ELSE
                    IF (@imc > 25) AND (@imc < 30)
                        SET @retorno = 'Obesidade Grau I'
                    ELSE
                        IF (@imc > 35) AND (@imc < 40)
                            SET @retorno = 'Obesidade Grau II'
                        ELSE
                            IF (@imc > 40)
                                SET @retorno = 'Obesidade Grau III'

    RETURN @retorno
END;
```

Stored Functions

- É uma coleção de comandos SQL executados no banco de dados que encapsula tarefas repetitivas, aceita parâmetros de entrada e retorna um valor.

```
SELECT dbo.IMC2(140, 1.80)
```

Resultados		Mensagens
(Nenhum nome de coluna)		
1	Obesidade Grau III	

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

Stored Functions

- Carregar variável a partir do Select

```
DECLARE @total INT  
  
SELECT @total = SUM(Qtd) FROM ITENS
```

PARA SER, UTP.



**Universidade
Tuiuti do
Paraná**

Funções com valor de tabela

```
CREATE FUNCTION dbo.FUNC_PEDIDOS (@ped REAL)
RETURNS TABLE
AS
RETURN
(
SELECT pd.PedidoID, p.Descricao, i.Qtd, p.Preco
FROM PRODUTOS as p
INNER JOIN ITENS as i
ON p.ProdutoID = i.ProdutoID
INNER JOIN PEDIDOS as pd
ON pd.PedidoID = i.PedidoID
WHERE pd.PedidoID = @Ped
);
```

SELECT * FROM **dbo.FUNC_PEDIDOS(1)**

Resultados		Mensagens		
	PedidoID	Descricao	Qtd	Preco
1	1	Sabonete	2	10
2	1	Pao	4	4,5

PARA SER, UTP.



**Universidade
Tuiuti do
Paraná**

Funções com valor de tabela

```
CREATE FUNCTION dbo.FUNC_PEDIDOS2 (@ped REAL)
RETURNS @retFinalTable TABLE
(
    PedidoID INT,
    Descricao VARCHAR(100),
    Total REAL
)
AS
BEGIN
    INSERT @retFinalTable
    SELECT pd.PedidoID, p.Descricao, i.Qtd * p.Preco AS TOT
        FROM PRODUTOS as p
        INNER JOIN ITENS as i
        ON p.ProdutoID = i.ProdutoID
        INNER JOIN PEDIDOS as pd
        ON pd.PedidoID = i.PedidoID
        WHERE pd.PedidoID = @Ped;
    RETURN
END;
```

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

Funções com valor de tabela

```
SELECT * FROM dbo.FUNC_PEDIDOS2(1)
```

Resultados				Mensagens			
	PedidoID	Descricao	Total				
1	1	Sabonete	20				
2	1	Pao	18				

PARA SER, UTP.



Universidade
Tuiuti do
Paraná