

# Microsoft SQL Server 2016 (T-SQL)

## Aula 04

Professor Anderson Henrique

## Assuntos tratados nessa aula:

01 – INSERT INTO (Populando Tabelas)

02 – TRUNCATE TABLE

03 – SELECT FROM (Consultas simples)

Professor Anderson Henrique



## 01 – INSERT INTO (Inserir registros nas tabelas)

Para criar inserir, utilizamos o comando DML com a seguinte sintaxe:

**INSERT INTO tabela(campo1, campo2,...) VALUES(valor1, valor2,...);**

Inserir dados – Tabela Autor (Populando):

```
INSERT INTO tbl_Autor(ID_Autor, Nome_Autor, Sobrenome_Autor)  
VALUES(1, 'Daniel', 'Barret');
```

```
INSERT INTO tbl_Autor(ID_Autor, Nome_Autor, Sobrenome_Autor)  
VALUES(2, 'Gerald', 'Carter');
```

```
INSERT INTO tbl_Autor(ID_Autor, Nome_Autor, Sobrenome_Autor)  
VALUES(3, 'Mark', 'Sobell');
```

```
INSERT INTO tbl_Autor(ID_Autor, Nome_Autor, Sobrenome_Autor)  
VALUES(4, 'William', 'Stanek');
```

```
INSERT INTO tbl_Autor(ID_Autor, Nome_Autor, Sobrenome_Autor)  
VALUES(5, 'Richard', 'Blum');
```

Inserir dados – Tabela Editora (Populando):

```
INSERT INTO tbl_Editora(Nome_Editora)VALUES('Prentice Hall');  
INSERT INTO tbl_Editora(Nome_Editora)VALUES('O Reilly');  
INSERT INTO tbl_Editora(Nome_Editora)VALUES('Microsoft Press');  
INSERT INTO tbl_Editora(Nome_Editora)VALUES('Wiley');
```

Inserir dados – Tabela Livro (Populando):

```
INSERT INTO tbl_Livro(Nome_Livro, ISBN_Livro, Data_Pub, Preco_Livro,  
ID_Autor, ID_Editora)  
VALUES('Linux Command Line and Shell Scripting', 143586, '20091221',  
68.35, 5, 4);
```

```
INSERT INTO tbl_Livro(Nome_Livro, ISBN_Livro, Data_Pub, Preco_Livro,  
ID_Autor, ID_Editora)  
VALUES('SSH, The Secure Shell', 127658, '20091221', 58.30, 1, 2);
```

```
INSERT INTO tbl_Livro(Nome_Livro, ISBN_Livro, Data_Pub, Preco_Livro,  
ID_Autor, ID_Editora)  
VALUES('Using Samba', 123856, '20001221', 61.55, 2, 2);
```

```
INSERT INTO tbl_Livro(Nome_Livro, ISBN_Livro, Data_Pub, Preco_Livro,  
ID_Autor, ID_Editora)  
VALUES('Fedora and Red Hat Linux', 123456, '20101101', 62.24, 3, 1);
```

```
INSERT INTO tbl_Livro(Nome_Livro, ISBN_Livro, Data_Pub, Preco_Livro,  
ID_Autor, ID_Editora)  
VALUES('Windows Server 2012 Inside Out', 123356, '20040517', 66.80, 4, 3);
```



```
INSERT INTO tbl_Livro(Nome_Livro, ISBN_Livro, Data_Pub, Preco_Livro,  
ID_Autor, ID_Editora)  
VALUES('Microsoft Exchange Server 2010', 123366, '20001221', 45.30, 4, 3);
```

## 02 – TRUNCATE TABLE

O comando remove todas as linhas de uma tabela sem registrar as exclusões de linhas individuais

TRUNCATE TABLE é como a instrução DELETE sem usar a cláusula WHERE

Entretanto, TRUNCATE TABLE é mais rápida e utiliza menos recursos de sistema e log de transações

Vamos utilizar esse comando após esse pequeno exercício:

CRIE UM BANCO DE DADOS (aulas)  
CRIE UMA TABELA (tbl\_Cliente)  
CAMPOS (ID\_Cliente, Nome, Telefone)  
INSIRA TRÊS REGISTROS NESSA TABELA

Utilizando o TRUNCATE, com a seguinte sintaxe:

**TRUNCATE TABLE cliente;**

**Vamos lá, pratiquem!**

```
CREATE DATABASE aulas;
```

```
USE aulas;
```

```
CREATE TABLE tbl_Cliente(  
    ID_Cliente SMALLINT PRIMARY KEY IDENTITY(1,1),  
    Nome VARCHAR(100),  
    Telefone VARCHAR(15)  
);
```

```
INSERT INTO tbl_Cliente(Nome, Telefone)VALUES('Ana Maria','9999-9999');  
INSERT INTO tbl_Cliente(Nome, Telefone)VALUES('Carlos','8888-8888');  
INSERT INTO tbl_Cliente(Nome, Telefone)VALUES('Aline','7777-7777');
```

Veremos no próximo tópico, adiantando, podemos realizar consultas na tabela retornando todos os registros:

```
SELECT COUNT(*) AS Tabela_Antes FROM tbl_Cliente;  
TRUNCATE TABLE tbl_Cliente;  
SELECT COUNT(*) AS Tabela_Depois FROM tbl_Cliente;
```

## 03 – SELECT FROM (Consultas Simples)

É possível consultar os registros em uma tabela, utilizando o comando DML com a seguinte sintaxe:

**SELECT campo, campo FROM tabela;**

Vamos ao exercício:

CRIAR UMA TABELA tbl\_Produto no BANCO DE DADOS aulas  
CAMPOS (ID\_Produto, Descricao, Quantidade, Preco)  
INSERIR TRÊS REGISTROS NESSA TABELA

INSERIR NOVAMENTE TRÊS REGISTROS NA TABELA tbl\_Cliente

**Vamos lá, pratiquem!**

```
CREATE TABLE tbl_Produto(  
    ID_Produto SMALLINT PRIMARY KEY IDENTITY,  
    Descricao VARCHAR(60),  
    Quantidade INT,  
    Preço decimal(10,2)  
);
```

```
INSERT INTO tbl_Produto(Descricao, Quantidade, Preço)VALUES('Teclado  
USB', 10, 19.90);
```

```
INSERT INTO tbl_Produto(Descricao, Quantidade, Preço)VALUES('MouseUSB',  
15, 12.90);
```

```
INSERT INTO tbl_Produto(Descricao, Quantidade, Preço)VALUES('Monitor LCD  
19', 8, 249.90);
```



Utilizando as consultas nas tabelas tbl\_Cliente e tbl\_Produto:

**SELECT Nome FROM tbl\_Cliente;**

**SELECT \* FROM tbl\_Cliente;**

**SELECT Nome, Telefone FROM tbl\_Cliente;**

**SELECT Descricao, Preco FROM tbl\_Produto;**

**SELECT \* FROM tbl\_Produto;**

**SELECT ID\_Produto, Descricao, Preco FROM tbl\_Produto;**

**Dúvidas?**

**Professor Anderson Henrique**



## Para a próxima aula

01 – Cláusula ORDER BY (Consultas com Ordenação)

02 – SELECT DISTINCT (Consultas exibindo valores distintos)

03 – Cláusula WHERE (Filtrando registros em consultas)

Professor Anderson Henrique

