

SSMS: SQL Server Management Studio

O SQL Server Management Studio (ambiente de gerenciamento SQL Server) é um software que foi lançado pela primeira vez com o Microsoft SQL Server 2005 e é usado para configurar, gerenciar e administrar todos os componentes do Microsoft SQL Server. É o sucessor do Enterprise Manager no SQL 2000 ou anteriores.

É através do SSMS que criaremos a "Query" onde digitaremos os comandos "SQL" necessários para a criação e abertura da "base de dados", os comandos para manutenção da estrutura desta base de dados e os comandos para manipulação dos registros.

Acesso ao SSMS:

Abra o SSMS, em poucos segundos será aberta a janela para efetuar o "login":

-- clicar no campo "nome do servidor", abrirá uma janela, ampliar a opção "Mecanismo de banco de dados" (clicar no sinal de +)

-- escolher a "máquina" sugerida. Exemplo: LAB203-10\SQLEXPRESS (será sempre o nome da máquina em que está)

-- Clicar no botão ok, clicar no botão conectar.

Pronto, agora é só criar a "consulta" (nova query) e digitar os comandos.

Comando SQL para criação do banco de dados:

Vamos iniciar criando um banco de dados, então, com o Management Studio aberto, clique no **botão New Query**, como mostra a **Figura 1**.

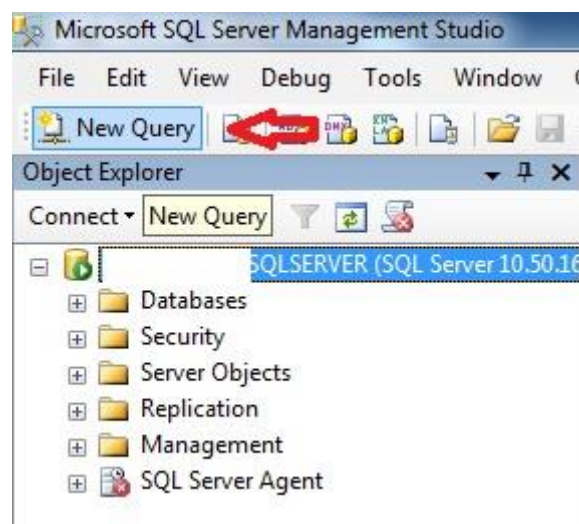


Figura 1. Criando nova Query no Management Studio

Em seguida, na parte central da interface aparecerá uma tela em branco, na qual você poderá digitar os códigos para criar o banco de dados, tabelas, efetuar consultas, etc.

CREATE DATABASE NomeBanco

Nesse momento, vamos executar o código a seguir e pressionar F5 ou clicar em **Execute**, como mostra a **Figura 2**:



Figura 2. Executando o script

Com o banco criado, precisamos entrar no seu contexto (abri-lo) para poder executar novos scripts dentro dele. No caso, precisamos acessá-lo para criar nossa tabela. Para isso, devemos executar o seguinte comando:

USE NomeBanco

Para confirmar que a operação foi executada como esperado, veja na parte superior esquerda do Management Studio se o banco criado está em uso, como mostra a **Figura 3**.

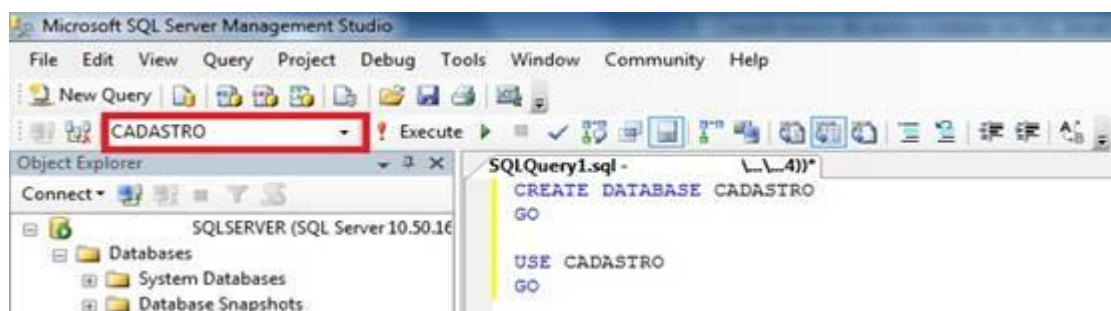


Figura 3. Verificando o banco em uso no Management Studio

O próximo passo será criar uma tabela, nesse caso contendo apenas três campos. Isso pode ser feito executando o código abaixo.

```
CREATE TABLE CLIENTE (  
    COD_CLI INT PRIMARY KEY NOT NULL,  
    NOME VARCHAR (50),  
    SOBRENOME VARCHAR (50)  
)
```

Obs 1: “not null” significa que neste campo a digitação de algum dado é obrigatória.

Obs 2: Ao colocar “Primary key” após o tipo de dado, este campo será definido como “chave primária”.

Em seguida, para verificar se a tabela foi criada, acesse a árvore de elementos do banco de dados no lado esquerdo, como vemos na **Figura 4**.

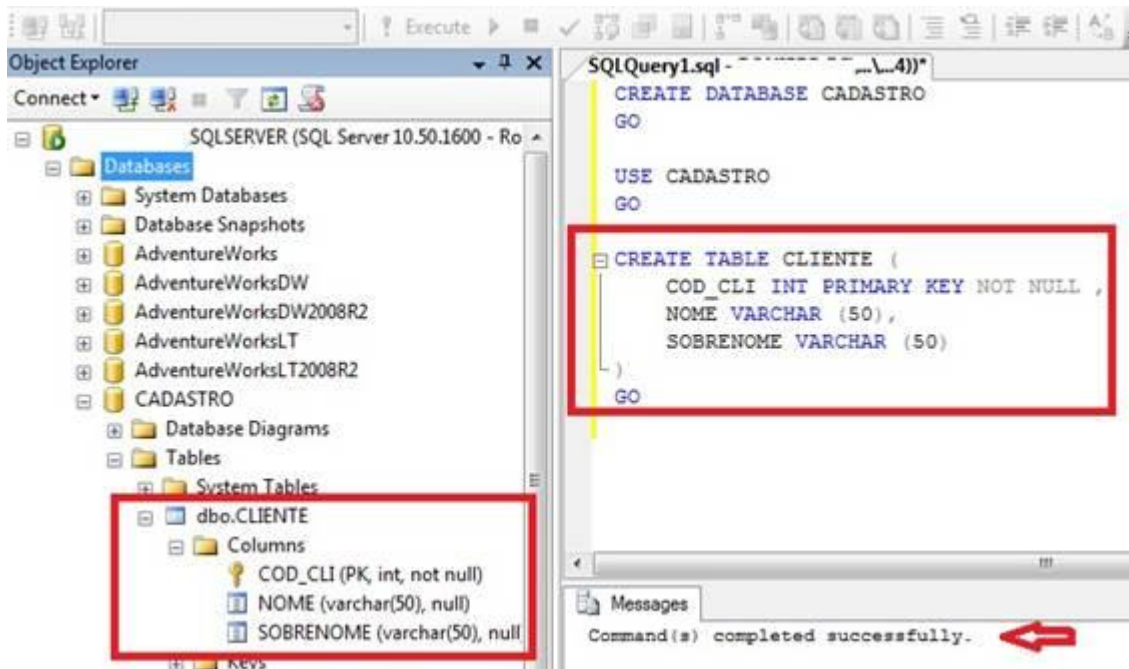


Figura 4. Estrutura do banco com tabela criada

Tipos de dados suportado pelo SQL

No SQL Server, cada coluna (campo da tabela) tem um tipo de dados relacionado. O tipo de dados é um atributo que especifica o tipo de dados que o objeto pode manter: dados inteiros, dados de caractere, dados monetários, data e hora, cadeias de caracteres binárias etc.

Os tipos de dados no SQL Server são organizados nas seguintes categorias:

Numéricos exatos == **int**

Numéricos aproximados == **float**

Data e hora == **date, time , datetime**

Cadeias de caracteres == **varchar**

Obs 1: O tipo de dado “varchar” necessita da propriedade “tamanho”.

Exemplo: nome varchar(35);

Obs 2: Para tornar um campo inteiro em “auto numeração” basta incluir o “identity”.

Exemplo: num1 int identity;

Alterando a estrutura de uma tabela

É possível modificar toda a estrutura de uma tabela após sua criação, para essas modificações, utilizamos o comando **alter tab**. É possível inserir um novo campo, alterar o nome e o tipo de um campo ao mesmo tempo, como podemos observar a seguir.

Para adicionar um novo campo em uma tabela, usamos a cláusula <add>:

```
alter table <nome tabela> add <nome campo novo> <tipo>;
```

Para alterar o tipo de um campo, usamos cláusula <modify>:

```
alter table <nome tabela> modify <nome campo novo> <tipo>;
```

Para alterar o nome de um campo de uma tabela, usamos a cláusula <change>:

```
alter table <nome tabela> change <nome velho> <nome novo> <tipo>;
```

Para excluir um campo de uma tabela, usamos a cláusula <drop>:

```
alter table <nome tabela> drop <nome campo>;
```

Para adicionar uma chave primaria a uma tabela existente:

```
alter table <nome tabela> add primary key (<campo chave>;
```

Para remover uma chave primaria a uma tabela existente:

```
alter table <nome tabela> drop primary key;
```

Apagando uma tabela ou um banco de dados

Para os casos que é necessário apagar uma tabela inteira, utilizamos o comando que também é utilizado para apagar a base de dados, importante ressaltar que este comando

pode causar danos irreparáveis, sendo necessário a restauração de um backup (cópia de segurança), assim as sintaxes para excluir uma tabela e um banco é:

drop table <nome da tabela>

drop database <nome do banco de dados>