

Linguagem SQL

Conceitos e Comandos DDL



Relembrando alguns conceitos

Banco de Dados (BD)	Modelo de Dados	Sistema Gerenciador de Banco de Dados (SGBD)
conjunto de dados integrados que tem por objetivo atender a uma comunidade de usuários.	descrição formal das estruturas de dados para representação de um BD.	software que incorpora as funções de definição, recuperação e alteração de dados em um BD.

Relembrando alguns conceitos

Existem modelos para diferentes níveis de abstração de representação de dados

Modelos **DESCRITIVOS**

Modelos **CONCEITUAIS**

Modelos **LÓGICOS**

Modelos **FÍSICOS**

Relembrando alguns conceitos

MODELAGEM DE DADOS

Ação de **representar/abstrair dados** do minimundo com o objetivo de **criar** projetos **conceituais** e **lógicos** de um BD.

Alguns autores incluem os **projetos físicos** como parte da **modelagem de dados**, pelo fato de que as otimizações são oriundas de análises do comportamento dinâmico do BD.

Relembrando alguns conceitos

Projeto Conceitual

ação que produz o **esquema** de dados **abstratos** que descreve a estrutura de um BD de forma **independente** de um **SGBD**.

Projeto Lógico

ação que produz o **esquema lógico** de dados que representa a estrutura de dados de um BD em acordo com o modelo de dados **subjacente** a um **SGBD**

Projeto Físico

ação que produz o **esquema físico** de dados a partir do **esquema lógico** de dados com a **adição** das estratégias de **otimização** para manipulação das estruturas de dados, sendo **dependentes** dos fabricantes dos **SGBDs** e de suas versões

Structured Query Language,
linguagem de consulta estruturada foi
desenvolvida pela IBM no início da
década de **1970**, posteriormente,
adotada como **linguagem padrão** pela
ANSI e pela ISO em **1986 e 1987**,
respectivamente.

Linguagem SQL

Structured Query Language Linguagem de Consulta Estrurada

Linguagem desenvolvida pela **IBM** no início da década de **1970**, posteriormente, adotada como linguagem **padrão** pela **ANSI** e pela **ISO** em **1986** e **1987**, respectivamente.

A T-SQL (**Transact-SQL**) é uma **implementação** da **Microsoft** para a SQL padrão ANSI.

Linguagem SQL

Sub-Linguagens SQL

Data Definition Language (DDL)	Data Manipulation Language (DML)	Data Control Language (DCL)
Utilizada para definir a estrutura de banco de dados.	Utilizada para realizar inclusões, consultas, exclusões e alterações de dados.	Controla os aspectos de autorização de dados e a utilização de licenças por usuários.
CREATE ALTER DROP TRUNCATE RENAME	SELECT* UPDATE INSERT DELETE	GRANT REVOKE DENY

Testando os conhecimentos



<https://www.menti.com/er53axk43f>

Nossa ferramenta de trabalho!

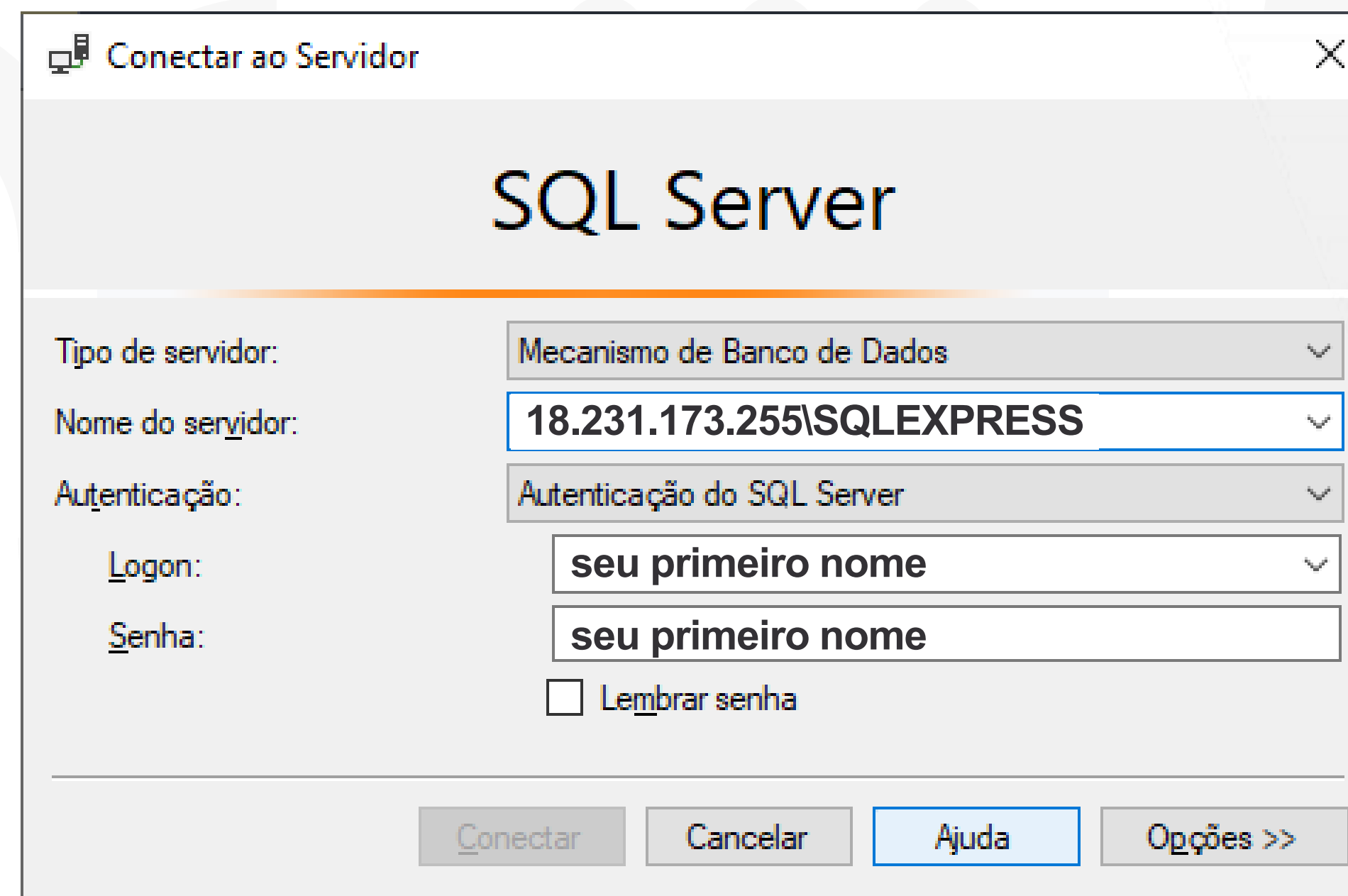
SQL Server Management Studio (SSMS)

Ambiente integrado para **acessar, configurar, gerenciar, administrar e desenvolver** todos os componentes do SQL Server.

SSMS combina um amplo grupo de **ferramentas gráficas** com um número de editores de script ricos para fornecer aos desenvolvedores e administradores de todos os níveis de acesso ao SQL Server.

Nossa ferramenta de trabalho!

Conectando no servidor



Conectar ao Servidor

SQL Server

Tipo de servidor: Mecanismo de Banco de Dados

Nome do servidor: 18.231.173.255\SQLEXPRESS

Autenticação: Autenticação do SQL Server

Logon: seu primeiro nome

Senha: seu primeiro nome

☐ Lembrar senha

Conectar Cancelar Ajuda Opções >>

PRIMEIROS COMANDOS DDL ...

Comandos DDL

TRUNCATE
TABLE
TAMBÉM É
DDL,
VEREMOS EM
BREVE!

CREATE TABLE

Cria uma nova
tabela.

ALTER TABLE

Modifica uma
tabela. Alterando,
adicionando ou
excluindo
colunas e
restrições.

DROP TABLE

Remove uma
tabela e todos os
dados, índices,
gatilhos,
restrições e
especificações
de permissão
dessas tabelas.

Comandos DDL

CREATE TABLE criação de tabela

-- Simple CREATE TABLE Syntax

CREATE TABLE

```
    { database_name.schema_name.table_name | schema_name.table_name | table_name }  
    ( { <column_definition> } [ ,...n ] )  
[ ; ]
```

Comandos DDL

ALTER TABLE alterar a estrutura da tabela

```
ALTER TABLE { database_name.schema_name.table_name |  
schema_name.table_name | table_name }  
{  
    ALTER COLUMN column_name { <column_definition> }  
    |  
    ADD {<column_definition> | CONSTRAINT <table_constraint>}  
    |  
    DROP { constraint_name | COLUMN {column_name}}  
} [ ,...n ]  
[ ; ]
```

Comandos DDL

DROP TABLE excluir tabela

-- Syntax for SQL Server

```
DROP TABLE [ IF EXISTS ]  
{ database_name.schema_name.table_name | schema_name.table_name | table_name }  
[ ,...n ]  
[ ; ]
```


O que é e para que
servem os data
types?

Data types

DATA TYPES

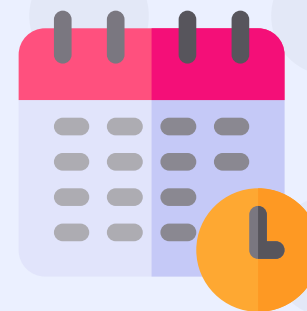
- ✓ Cada elemento, como uma **coluna**, **variável** ou **expressão**, possui um **tipo de dado**.
- ✓ O tipo de dado **especifica o tipo de valor** que o objeto pode armazenar, como **números inteiros**, **texto**, **data** e **hora**, etc.

Data types

O SQL Server organiza os tipos de dados dividindo-os em categorias

12
3

Numéricos exatos
Numéricos;
aproximados



Data e hora

Ab
c

Cadeias de caracteres;
Cadeias de
caracteres Unicode;
Cadeia de caracteres
binária;

Data types

Numéricos Exatos - Inteiros

Tipo de dados	Intervalo	Armazenamento
bigint	-2^{63} (-9.223.372.036.854.775.808) a $2^{63}-1$ (9.223.372.036.854.775.807)	8 bytes
int	-2^{31} (-2.147.483.648) a $2^{31}-1$ (2.147.483.647)	4 bytes
smallint	-2^{15} (-32.768) a $2^{15}-1$ (32.767)	2 bytes
tinyint	0 a 255	1 byte

Data types

Numéricos Exatos - Decimais

Tipo de dados	Intervalo
decimal [(p[,s])] e numeric[(p[,s])]	- 10^38 +1 a 10^38 - 1
precision	Bytes de armazenamento
1 - 9	5
10-19	9
20-28	13
29-38	17

Data types

Numéricos Exatos - Bit

Tipo de dados	Intervalo
bit	1, 0 ou NULL

Colunas bit	Bytes de armazenamento
1 – 8	1
9 – 18	2
19 – 27	3
...	...

Data types

Numéricos Exatos - Moeda

Tipo de dados	Intervalo	Armazenamento
money	- 922.337.203.685.477,5 808 a 922.337.203.685.477,5 807	8 bytes
smallmoney	-214.748,3648 a 214.748,3647	4 bytes

Data types

Numéricos Aproximados

Tipo de dados	Intervalo	Armazenamento
float [(n)]	- 1,79E+308 a -2,23E-308, 0 e 2,23E-308 a 1,79E+308	Depende do valor de <i>n</i>
real	- 3,40E + 38 a -1,18E - 38, 0 e 1,18E - 38 a 3,40E + 38	4 bytes
<i>n</i> value	precisão	Tamanho de armazenamento
1-24	7 dígitos	4 bytes
25-53	15 dígitos	8 bytes

Data types

Data e Hora

Tipo de dados	Intervalo
datetime - 8 bytes	Data e hora compreendidas entre 1/1/ 1753 e 31/12/9999, com a exatidão de 3.33 milissegundos.
Smalldatetime - 4 bytes	Data e hora compreendidas entre 1º de janeiro de 1900 e 6 de junho de 2079, com a exatidão de 1 minuto.
datetime2[(p)] - 8 bytes	Data e hora compreendidas entre 01/01/0001 e 31/12/9999 com precisão de até 100 nanossegundos, dependendo do valor de p.
Date - 3 bytes	Data compreendida entre 01/01/0001 e 31/12/9999, com precisão de 1 dia.
time[(p)] - 5 bytes	Hora no intervalo de 00:00:00.0000000 a 23.59.59.9999999.
Datetimeoffset[(p)]	Data e hora compreendidas entre 1/1/0001 e 31/12/9999 com precisão de até 100 nanossegundos e com indicação do fuso.

Data types

Strings (Cadeias de Caracteres)

Tipo de dados	Intervalo
char (n)	Comprimento fixo de no máximo 8.000 caracteres no padrão ANSI. Cada caractere é armazenado em 1 byte.
varchar (n)	Comprimento variável de no máximo 8.000 caracteres no padrão ANSI. Cada caractere é armazenado em 1 byte.
varchar (max)	Comprimento variável de no máximo $2^{31} - 1$ (2,147,483,647) – 2 GB para caracteres no padrão ANSI. Cada caractere é armazenado em 1 byte.
nchar (n)	Comprimento fixo de no máximo 4.000 caracteres UNICODE. Cada caractere é armazenado em 2 bytes.
nvarchar (n)	Comprimento variável de no máximo 4.000 caracteres UNICODE. Cada caractere é armazenado em 2 bytes.
nvarchar (max)	Comprimento variável de no máximo $2^{31} - 1$ (2,147,483,647) – 2 GB para caracteres no padrão UNICODE. Cada caractere é armazenado em 2 bytes.

Outras características

Além dos Data Types...

Nulabilidade

Permitir ou **não** aceitar **nulos** (ausência de valor) em uma coluna.

Auto numeração

Propriedade utilizada para gerar **valores incrementais**.
Apenas uma coluna de identidade pode ser criada por tabela.



Constraint é de comer ?

Constraint

Constraints são restrições que evitam dados inválidos

PRIMARY KEY

FOREIGN KEY

UNIQUE

CHECK

DEFAULT

PRIMARY KEY

Impõe a exclusividade dos dados para as colunas de chave primária.

```
CONSTRAINT constraint_name  
    PRIMARY KEY (column [ ASC | DESC ] [  
, ... n ])
```

FOREIGN KEY

Criar relacionamento entre duas tabelas, para associar linhas de uma tabela com linhas de outra.

```
CONSTRAINT constraint_name  
  FOREIGN KEY ( column [ ,...n ] )  
    REFERENCES referenced_table_name (ref_column [ ,...n ] )
```

UNIQUE

Assegurar que nenhum valor duplicado seja digitado em colunas específicas que não integram uma chave primária.

```
CONSTRAINT constraint_name  
    UNIQUE (column [ ASC | DESC ] [ , ... n  
])
```

CHECK

Impõe integridade de domínio limitando os valores aceitos por uma ou mais colunas.

```
CONSTRAINT constraint_name  
CHECK ( logical_expression )
```

DEFAULT

Especificar um valor padrão que será inserido na coluna da tabela quando, no momento da inserção, não for informado nenhum valor.

```
-- DEFAULT CREATE TABLE
column_name <data_type> CONSTRAINT constraint_name DEFAULT
constant_expression

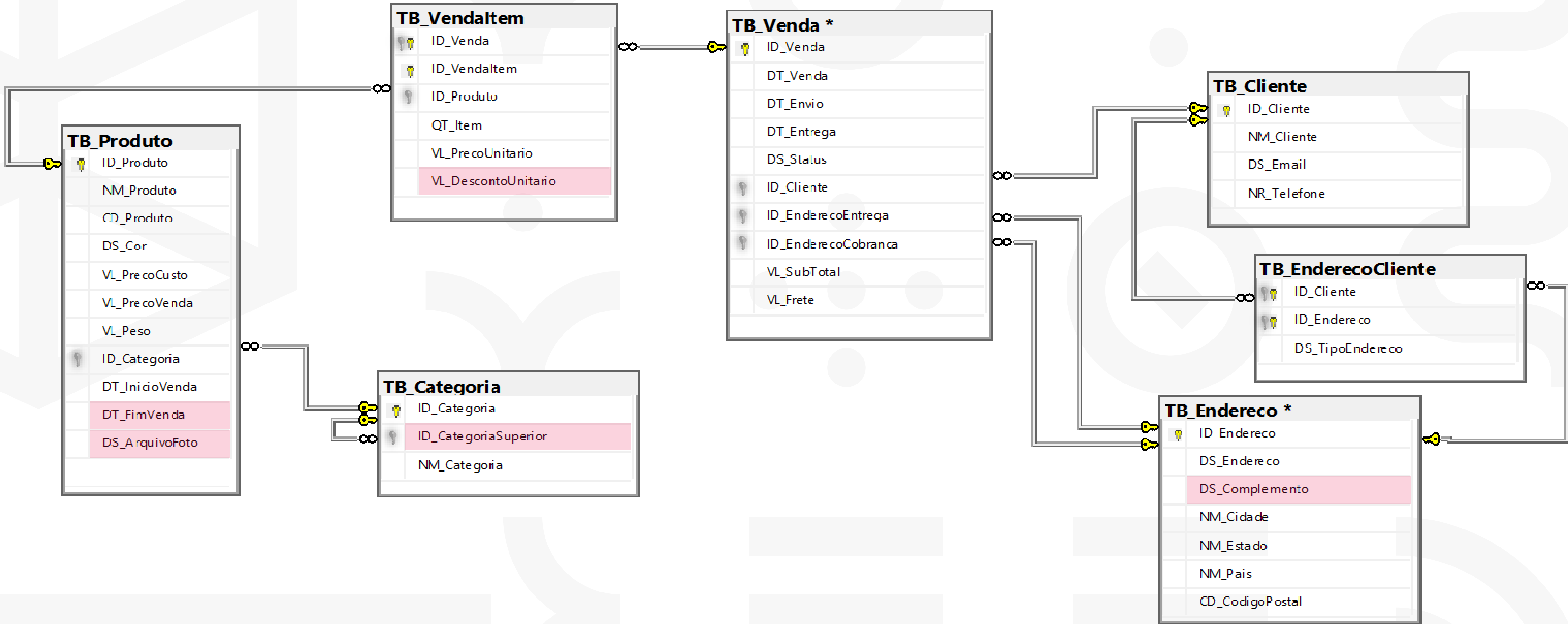
-- DEFAULT ALTER TABLE
CONSTRAINT constraint_name DEFAULT constant_expression FOR
column_name
```


Mão na massa – Em Grupo

Conhecendo os DataTypes, a nulabilidade, auto numeração e constraints. Discutam em grupo qual o melhor datatype para cada coluna do modelo lógico de uma rede fictícia de Varejo, quais podem ou não aceitar Nulos, quais devem ser auto numeráveis e quais constraints devem ser criadas. Neste varejo há mais de 100 mil de produtos diferentes disponíveis em centenas de categorias e eles possuem mais de 1 milhão de clientes.



Mão na massa – Em Grupo



LEGENDA	
	Chave Primária
	Chave Estrangeira
	Coluna permite NULO

TABELA	OBSERVAÇÕES
TB_PRODUTO	O CD_Produto será armazenado o número do EAN, código sempre composto por 13 caracteres e não se repete A data final de venda não é de preenchimento obrigatório, porém quando informada deve ser maior ou igual a data de início da venda
TB_VENDA	Por padrão a data da venda é a data e hora atual do sistema. A data de envio, quando preenchida, deve ser menor que a data da venda, que, quando preenchida, deve ser menor que a data de entrega. Os valores permitidos para status são: Aguardando Pagamento, Em transporte, Entregue ou Cancelado
TB_VENDA_ITEM	ID_VENDAITEM é o identificador do item para determinada venda (Ex: Venda 100, item 1; venda 100 item 2; venda 200 item 1;
TB_ENDERECOCLIENTE	Os valores permitidos para Tipo Endereço são: Comercial ou Residencial

Mão na massa – Individual



Com o resultado do exercício anterior, crie um script SQL (query) para criar as tabelas e suas respectivas colunas. Lembre-se de utilizar a auto numeração, a propriedade da nulabilidade e criar as constraints.

Obrigado.

Colabore com sua opinião!



<https://www.menti.com/353szq8333>

