

BANCO DE DADOS

Profa. Andréa Sakai

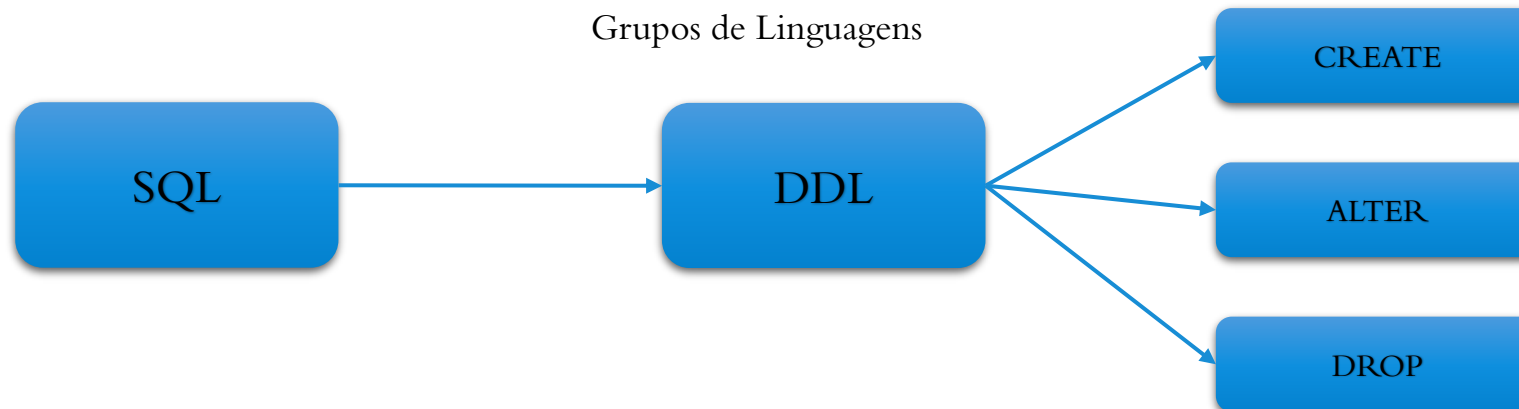


INTRODUÇÃO A LINGUAGEM SQL

- SQL – Linguagem de Consulta Estruturada
- Linguagem padrão dos Bancos de dados relacionais
- PostgreSQL
- 5 grupos de linguagem fazem parte do SQL
 - DQL – Linguagem de Consulta de Dados
 - DML – Linguagem de Manipulação de Dados
 - DDL – Linguagem de Definição de Dados
 - DCL – Linguagem de Controle de Dados
 - DTL – Linguagem de Transação de Dados

DDL – LINGUAGEM DE DEFINIÇÃO DE DADOS

- Comandos em SQL que definem a estrutura de um banco de dados, podendo criar, alterar e deletar banco de dados e tabelas, índices e visões.
- A DDL permite descrever os domínios e as restrições de integridade.



COMANDOS DDL

- CREATE – comando para criar banco de dados, tabelas e outros objetos em um BD.
- ALTER – comando usado para alterar a estrutura de tabelas ou outro objeto em um BD.
- DROP – comando usado para apagar banco de dados, tabelas e outros objetos.

CREATE TABLE – CRIAÇÃO DE TABELAS

- Este comando cria uma nova tabela, inicialmente vazia, no banco de dados.

Sintaxe:

```
CREATE TABLE <nome_da_tabela>
(
    <nome_do_campo>    <tipo_de_dados>    < aceitação>,
    <nome_do_campo>    <tipo_de_dados>    < aceitação>,
    <nome_do_campo>    <tipo_de_dados>    < aceitação>,

    CONSTRAINT <nome_constraint> PRIMARY KEY(<nome_do_campo>),
    CONSTRAINT <nome_constraint> FOREIGN KEY(<nome_do_campo>)
        REFERENCES <nome_da_tabela> (<nome_do_campo>)
);
```

EXEMPLOS

```
CREATE TABLE produto
(
    codigo      INTEGER,
    descricao   VARCHAR(50)
);
```

- Tabela produto com dois campos, sendo o código do tipo numérico e inteiro e descrição do tipo literal e suportando 50 caracteres apenas.

Tabela produto

codigo	descricao

EXEMPLOS

```
CREATE TABLE produto  
(  
    codigo    INTEGER PRIMARY KEY,  
    descricao VARCHAR(50)  
);
```

- Tabela produto com dois campos, sendo o código a **chave primária** do tipo numérico e inteiro e o campo descrição do tipo literal e suportando 50 caracteres apenas.
- Chave primária – os valores deste atributos são únicos, ou seja, não pode haver valores repetidos.

CRIAR UMA TABELA PARA ARMAZENAR OS DADOS DOS ALUNOS CONFORME O DESENHO:

Entidade: Alunos

rgm	nome	endereço	telefone
1256	Ana Luíza	Rua X, 100	11 9999-0000
5689	Caio Gomes	Rua Y, 200	11 9999-1111

A saber:

- Não será permitido que os alunos tenha o mesmo RGM;
- O RGM é o código que será referenciado por outras tabelas;
- Os atributos nome, endereço e telefone são palavras;
- Os atributos nome e telefone são obrigatórios para todos os alunos.

OBSERVAÇÕES

<Aceitação>

Tipo de aceitação do dado, ou seja, se ele pode ou não receber valores nulos ou não.

- NOT NULL – valores não nulos são permitidos na coluna.
- NULL – valores nulos são permitidos na coluna. Este é o padrão.

Primary key

A restrição de chave primária específica que a coluna, da tabela pode conter apenas valores únicos (não duplicados) e não nulos.

SOLUÇÃO

```
CREATE TABLE alunos
(  
    rgm            INTEGER PRIMARY KEY,  
    nome           VARCHAR(50) NOT NULL,  
    endereco       VARCHAR(70),  
    telefone       VARCHAR(11) NOT NULL  
);
```

OU

```
CREATE TABLE alunos
(  
    rgm            INTEGER,  
    nome           VARCHAR(50) NOT NULL,  
    endereco       VARCHAR(70),  
    telefone       VARCHAR(11) NOT NULL,  
    CONSTRAINT fkrgm PRIMARY KEY(rgm)  
);
```

ATIVIDADE 1

Modelo: Conceitual

Entidade ou tabela: cliente

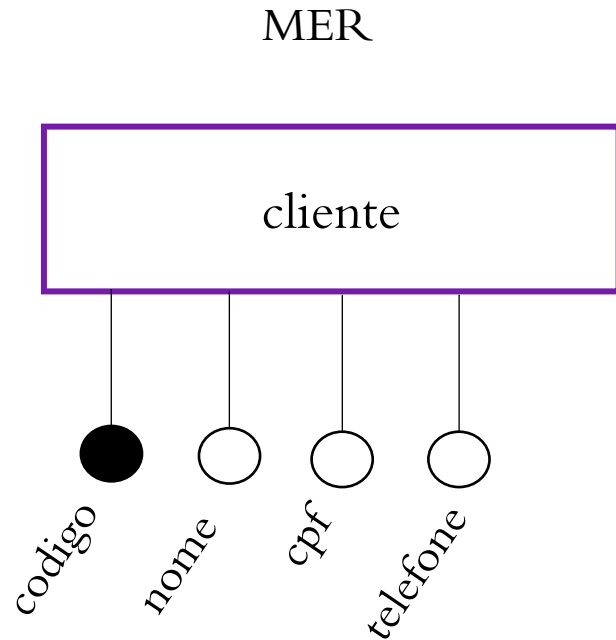
Atributos ou campos: código, nome, cpf, telefone

Tipos de atributos:

Simples ou atômico: nome, cpf

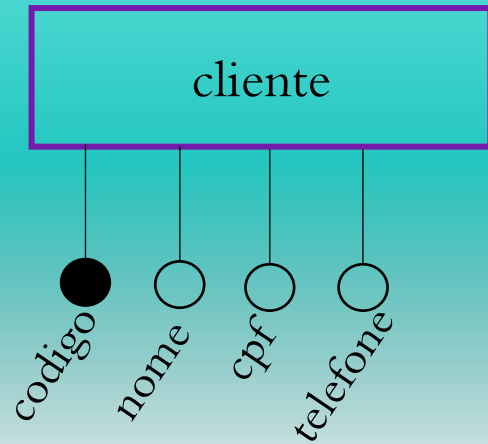
Multivalorado: telefone

Chave: código



ATIVIDADE 1

Modelo: Conceitual



Entidade ou tabela: cliente

Atributos ou campos: código, nome, cpf, telefone

Tipos de atributos:

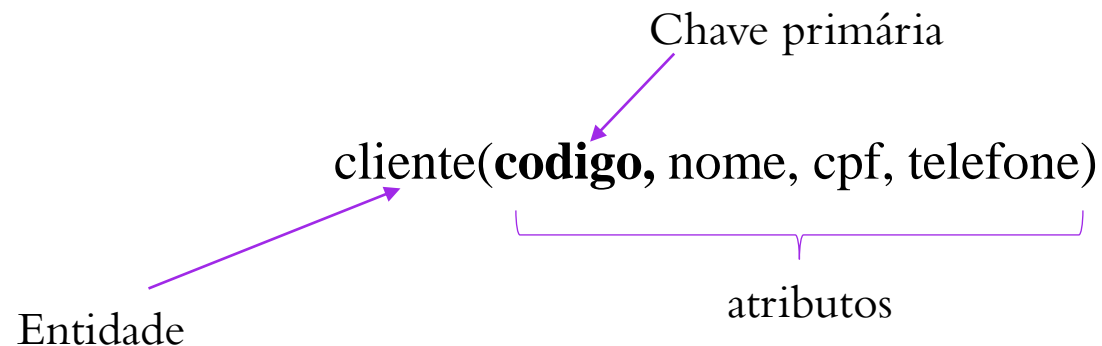
Simple ou atômico: nome, cpf

Multivalorado: telefone

Chave: codigo

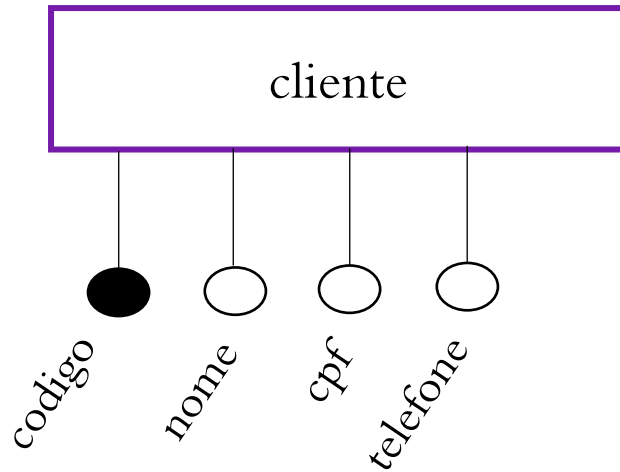
Modelo: Lógico

Modelo Relacional (MR)



Modelo: Conceitual

MER



Modelo: Físico

Modelo: Lógico

MR

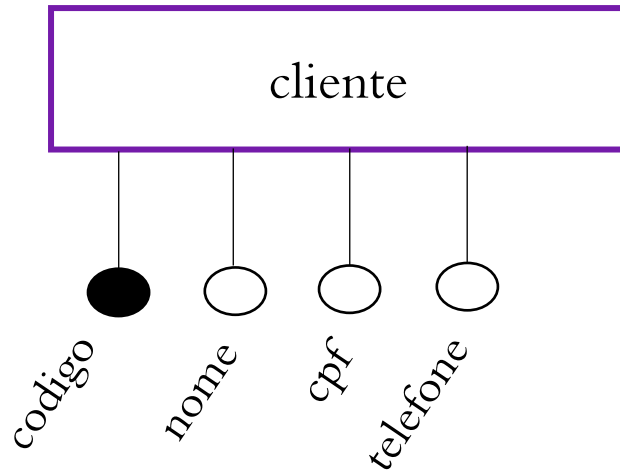
cliente(**codigo**, nome, cpf, telefone)

SCRIPT em SQL

```
CREATE TABLE cliente(  
    codigo        integer,  
    nome          varchar(50),  
    cpf           varchar(11),  
    telefone      varchar(11)  
);
```

Modelo: Conceitual

MER



Modelo: Físico

Modelo: Lógico

MR

cliente(**codigo**, nome, cpf, telefone)

SCRIPT em SQL

```
CREATE TABLE cliente(  
    codigo        integer,  
    nome          varchar(50),  
    cpf           varchar(11),  
    telefone      varchar(11)  
);
```

Modelo: Físico

SCRIPT em SQL

```
CREATE TABLE cliente(  
    codigo        integer,  
    nome          varchar(50),  
    cpf           varchar(11),  
    telefone      varchar(11)  
);
```

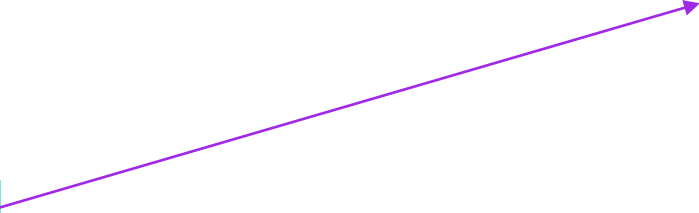
CREATE TABLE

- Comando em SQL usado para criar uma tabela.
- Neste comando são definidos:
 - Atributos da tabela;
 - Tipos de dados de cada atributo;
 - E as restrições;

Modelo: Físico

SCRIPT em SQL

CREATE TABLE Comando em SQL usado para criar uma tabela.



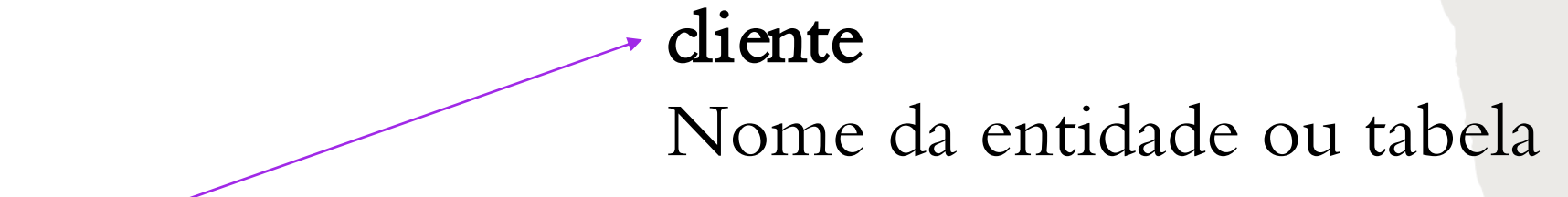
```
CREATE TABLE cliente(  
    codigo        integer,  
    nome          varchar(50),  
    cpf           varchar(11),  
    telefone      varchar(11)  
);
```


Modelo: Físico

SCRIPT em SQL

```
CREATE TABLE cliente(  
    codigo        integer,  
    nome          varchar(50),  
    cpf           varchar(11),  
    telefone      varchar(11)  
);
```

cliente
Nome da entidade ou tabela



Modelo: Físico

SCRIPT em SQL

```
CREATE TABLE cliente(  
  codigo  
  nome  
  cpf  
  telefone  
);
```

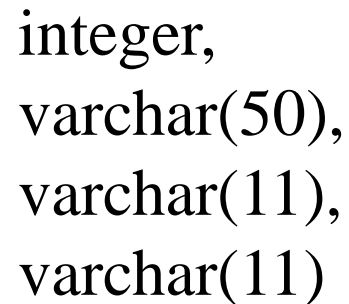
ATRIBUTOS ou CAMPOS

Escrevemos entre os parênteses os atributos da tabela, separando-os por vírgula.

Modelo: Físico

SCRIPT em SQL

```
CREATE TABLE cliente(  
    codigo  
    nome  
    cpf  
    telefone  
);
```



integer,
varchar(50),
varchar(11),
varchar(11)

TIPOS DE DADOS

Para cada atributo deve-se definir um tipo de dado:

Ex: só será possível armazenar dados do tipo literal (caracteres alfanuméricos) no campo nome.

Modelo: Físico

SCRIPT em SQL

```
CREATE TABLE cliente(  
    codigo integer,  
    nome varchar(50),  
    cpf varchar(11),  
    telefone varchar(11)  
);
```

TIPOS DE DADOS

Integer

Tipo de dado numérico
inteiro

Ex: 10, -500, 25, 0

Modelo: Físico

SCRIPT em SQL

```
CREATE TABLE cliente(  
    codigo        integer,  
    nome          varchar(50),  
    cpf           varchar(11),  
    telefone      varchar(11)  
);
```

TIPOS DE DADOS

VARCHAR(<qtde de caracteres>)

Tipo de dado alfanumérico

Ex: Andréa, PLG1025, ...

Modelo: Físico

SCRIPT em SQL

```
CREATE TABLE cliente (  
    codigo        integer,  
    nome          varchar(50),  
    cpf           varchar(11),  
    telefone      varchar(11)  
);
```

PARÊNTESES

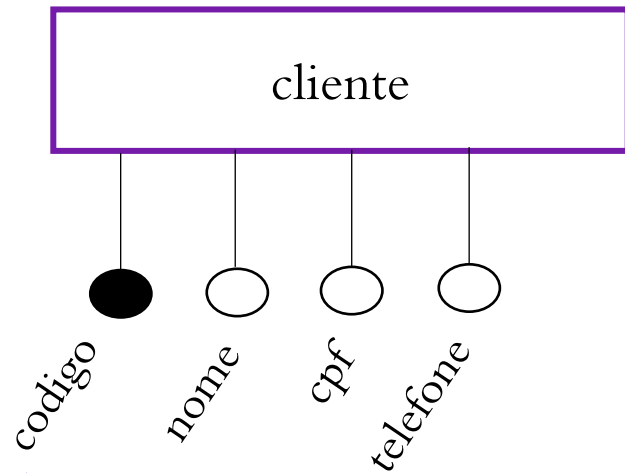
Delimita onde serão colocados os atributos da tabela.

Ponto e vírgula

Usado para indicar o fim de um comando

Modelo: Conceitual

MER



Chave primária

Modelo: Lógico

MR

Chave primária



Modelo: Físico

SCRIPT em SQL

```
CREATE TABLE cliente(  
    codigo        integer PRIMARY KEY,  
    nome          varchar(50),  
    cpf           varchar(11),  
    telefone      varchar(11)  
);
```

PRIMARY KEY

Comando em SQL aplicado em um atributo da tabela para que não seja possível armazenar dados repetidos neste campo. Sendo assim, a coluna de “codigo” da tabela cliente não permitirá valores de códigos repetidos.

Tabela cliente

Codigo (pk)	Nome	cpf	Telefone
1	Ana Maria	1111111	1199991111
2	Carlos Gomes	2222222	1199992222
2	Marcia Souza	3333333	1199993333

Tabela cliente

Codigo (pk)	Nome	cpf	Telefone
1	Ana Maria	1111111	1199991111
2	Carlos Gomes	2222222	1199992222
3	Marcia Souza	3333333	1199993333

ATIVIDADES

1. Criar um MER conforme problema a seguir:
 - A secretária de um consultório dentário recebeu a atividade de catalogar os fornecedores da clínica. Sendo que, os dados mais importantes são o cnpj, nome, telefone e email da empresa;
2. Responda as seguintes questões:
 - a) Qual o nome das entidade:
 - b) Quais são os atributos e seus tipos:
 - c) Qual a chave primária?
3. Criar um banco de dados chamando aula05_ativ01
4. Crie a tabela em SQL definindo a chave primária.