Aonde você quer chegar?

Vai com a

↓

UniCesumar

EDUCAÇÃO PRESENCIAL E A DISTÂNCIA

Disciplina: Banco de Dados

Prof. Maurício P. de Freitas MSc.

Aula 06 – 22/08/2024
Data Definition Language
Data Manipulation Language

UniCesumar
EDUCAÇÃO PRESENCIAL E A DISTÂNCIA

DDL – Data Definition Language

Esquema lógico 02

Tutor(id, cpf, nome, email, fone)

Tutor_endereço(id, id_resp, cep, rua, numero, complemento, cidade, uf)
id_resp referencia Responsavel(id)

Animal(id, id_resp, peso, raca, especie, cor, sexo, data_nasc)
id_resp referencia Responsável(id)

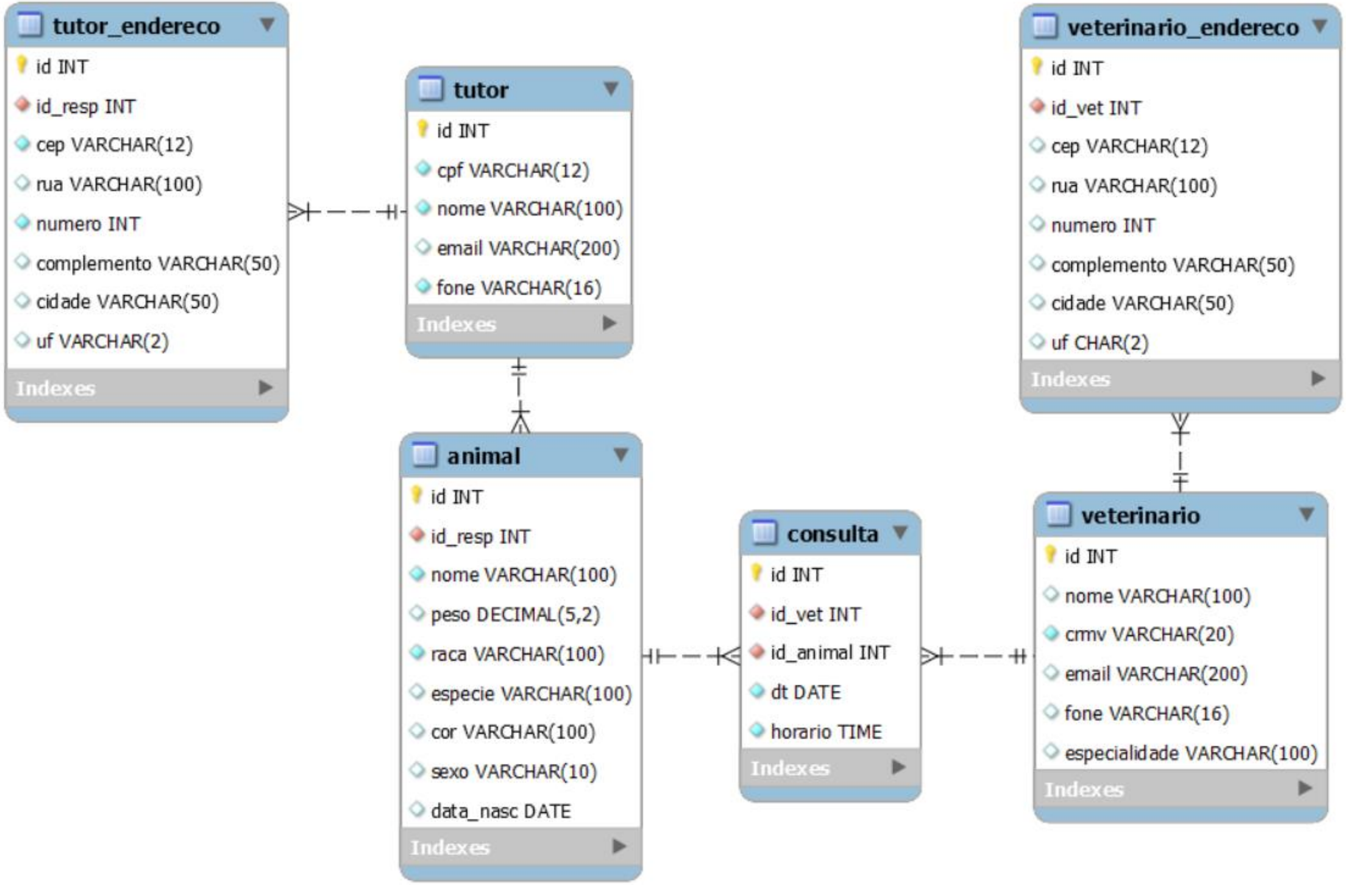
Veterinário(id, nome, crmv, email, fone, especialidade)

Dados_bancarios(id, id_vet, banco, agencia, conta, tipo)
id_pessoa referencia Pessoa(id)

Veterinario_endereço(id, id_vet, cep, rua, numero, complemento, cidade, uf)
id_vet referencia Veterinario(id)

Consulta(id, id_vet, id_animal, dt, horario)
id_vet referencia Veterinario(id)
id_animal referencia Veterinario(id)

MER → Esquema lógico 02



tutor_endereco
- id INT
- id_resp INT
- cep VARCHAR(12)
- rua VARCHAR(100)
- numero INT
- complemento VARCHAR(50)
- cidade VARCHAR(50)
- uf VARCHAR(2)

Indexes

tutor
- id INT
- cpf VARCHAR(12)
- nome VARCHAR(100)
- email VARCHAR(200)
- fone VARCHAR(16)

Indexes

veterinario_endereco
- id INT
- id_vet INT
- cep VARCHAR(12)
- rua VARCHAR(100)
- numero INT
- complemento VARCHAR(50)
- cidade VARCHAR(50)
- uf CHAR(2)

Indexes

animal
- id INT
- id_resp INT
- nome VARCHAR(100)
- peso DECIMAL(5,2)
- raca VARCHAR(100)
- especie VARCHAR(100)
- cor VARCHAR(100)
- sexo VARCHAR(10)
- data_nasc DATE

Indexes

consulta
- id INT
- id_vet INT
- id_animal INT
- dt DATE
- horario TIME

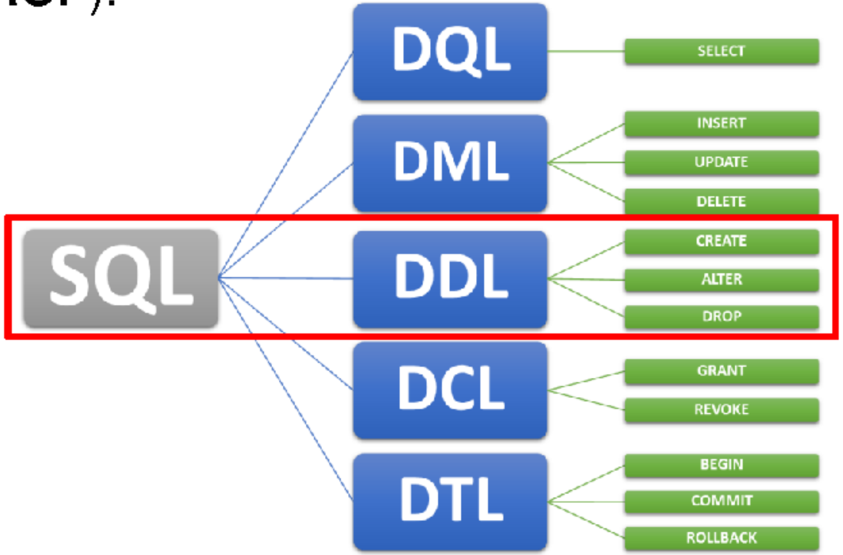
Indexes

veterinario
- id INT
- nome VARCHAR(100)
- crmv VARCHAR(20)
- email VARCHAR(200)
- fone VARCHAR(16)
- especialidade VARCHAR(100)

Indexes

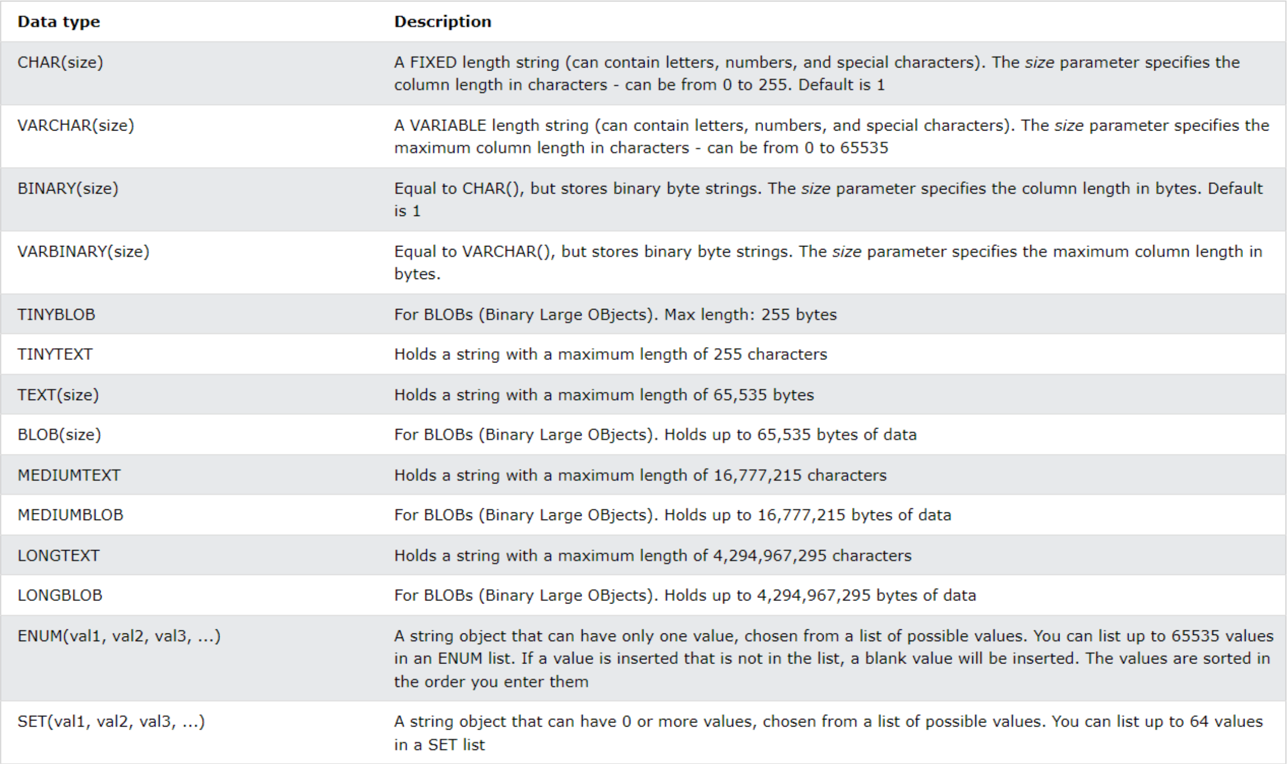
DDL - Linguagem de Definição de Dados:

- Comandos utilizados para:
  - Criação (**CREATE**) de esquemas, tabelas e visualizações;
  - Atualização (**ALTER**) dessas estruturas;
  - Assim como a remoção (**DROP**).



MYSQL – Tipos de dados (Domínio)

Dado do tipo Texto

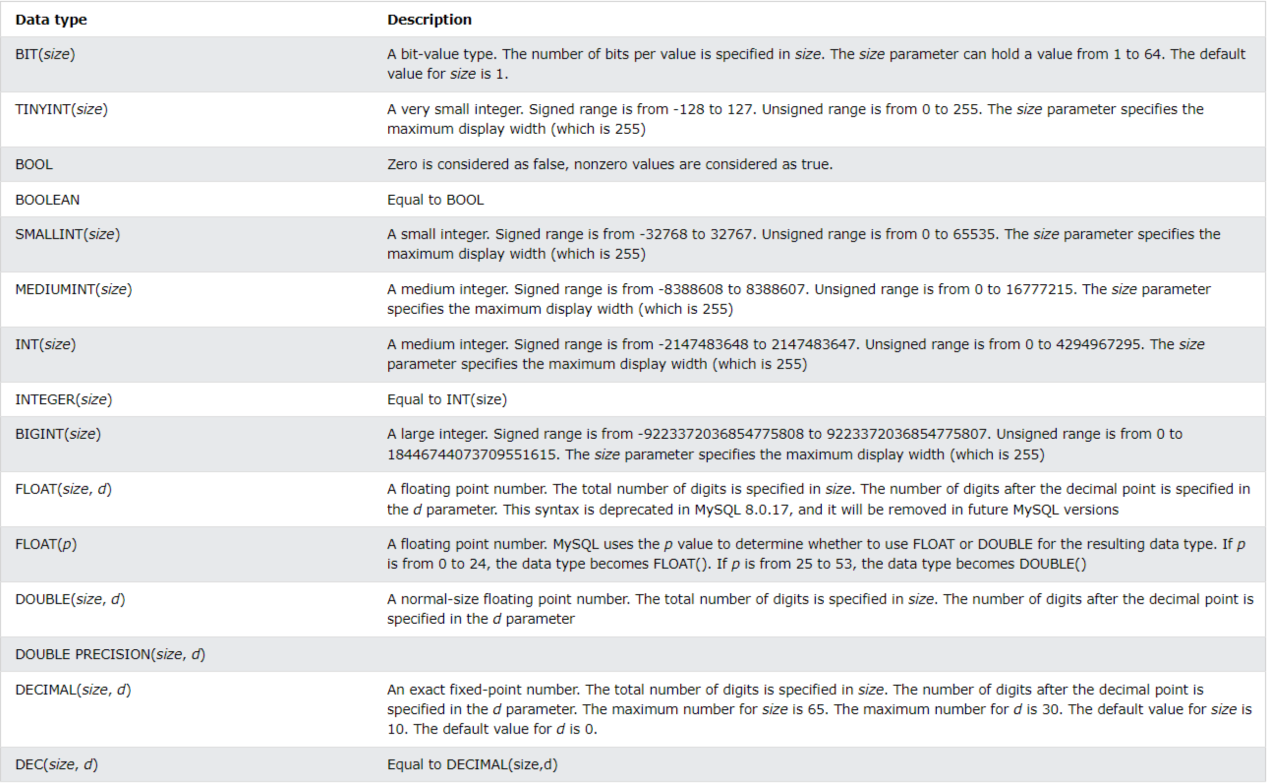


Data type	Description
CHAR(size)	A FIXED length string (can contain letters, numbers, and special characters). The *size* parameter specifies the column length in characters - can be from 0 to 255. Default is 1
VARCHAR(size)	A VARIABLE length string (can contain letters, numbers, and special characters). The *size* parameter specifies the maximum column length in characters - can be from 0 to 65535
BINARY(size)	Equal to CHAR(), but stores binary byte strings. The *size* parameter specifies the column length in bytes. Default is 1
VARBINARY(size)	Equal to VARCHAR(), but stores binary byte strings. The *size* parameter specifies the maximum column length in bytes.
TINYBLOB	For BLOBs (Binary Large OBjects). Max length: 255 bytes
TINYTEXT	Holds a string with a maximum length of 255 characters
TEXT(size)	Holds a string with a maximum length of 65,535 bytes
BLOB(size)	For BLOBs (Binary Large OBjects). Holds up to 65,535 bytes of data
MEDIUMTEXT	Holds a string with a maximum length of 16,777,215 characters
MEDIUMBLOB	For BLOBs (Binary Large OBjects). Holds up to 16,777,215 bytes of data
LONGTEXT	Holds a string with a maximum length of 4,294,967,295 characters
LONGBLOB	For BLOBs (Binary Large OBjects). Holds up to 4,294,967,295 bytes of data
ENUM(val1, val2, val3, ...)	A string object that can have only one value, chosen from a list of possible values. You can list up to 65535 values in an ENUM list. If a value is inserted that is not in the list, a blank value will be inserted. The values are sorted in the order you enter them
SET(val1, val2, val3, ...)	A string object that can have 0 or more values, chosen from a list of possible values. You can list up to 64 values in a SET list

Data types:
https://dev.mysql.com/doc/refman/8.0/en/data-types.html
https://www.w3schools.com/mysql/mysql_datatypes.asp

MYSQL – Tipos de dados (Domínio)

Dado do tipo Numéricos

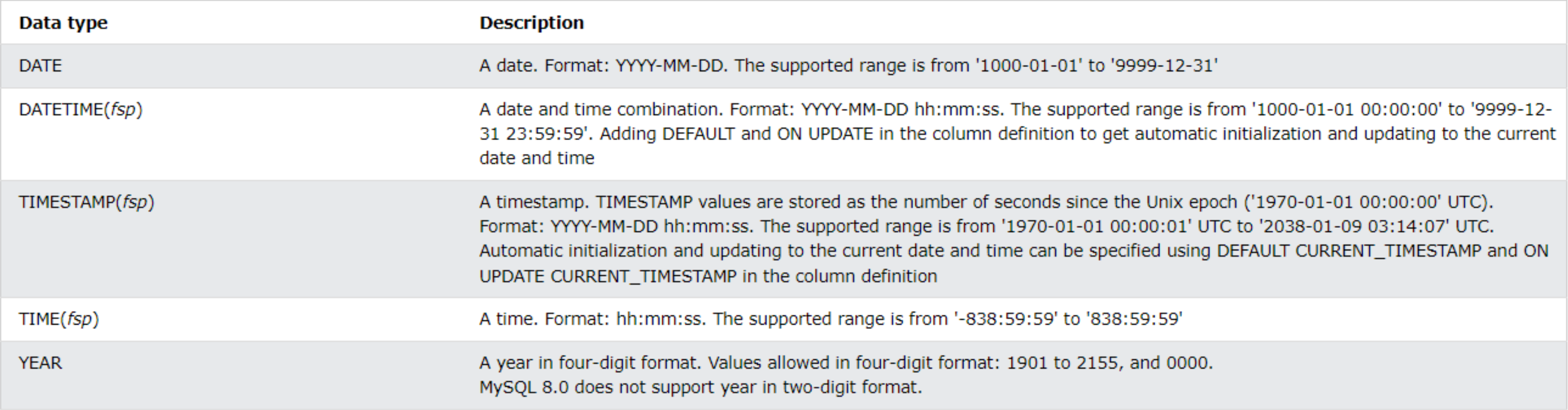


Data type	Description
BIT(*size*)	A bit-value type. The number of bits per value is specified in *size*. The *size* parameter can hold a value from 1 to 64. The default value for *size* is 1.
TINYINT(*size*)	A very small integer. Signed range is from -128 to 127. Unsigned range is from 0 to 255. The *size* parameter specifies the maximum display width (which is 255)
BOOL	Zero is considered as false, nonzero values are considered as true.
BOOLEAN	Equal to BOOL
SMALLINT(*size*)	A small integer. Signed range is from -32768 to 32767. Unsigned range is from 0 to 65535. The *size* parameter specifies the maximum display width (which is 255)
MEDIUMINT(*size*)	A medium integer. Signed range is from -8388608 to 8388607. Unsigned range is from 0 to 16777215. The *size* parameter specifies the maximum display width (which is 255)
INT(*size*)	A medium integer. Signed range is from -2147483648 to 2147483647. Unsigned range is from 0 to 4294967295. The *size* parameter specifies the maximum display width (which is 255)
INTEGER(*size*)	Equal to INT(size)
BIGINT(*size*)	A large integer. Signed range is from -9223372036854775808 to 9223372036854775807. Unsigned range is from 0 to 18446744073709551615. The *size* parameter specifies the maximum display width (which is 255)
FLOAT(*size*, *d*)	A floating point number. The total number of digits is specified in *size*. The number of digits after the decimal point is specified in the *d* parameter. This syntax is deprecated in MySQL 8.0.17, and it will be removed in future MySQL versions
FLOAT(*p*)	A floating point number. MySQL uses the *p* value to determine whether to use FLOAT or DOUBLE for the resulting data type. If *p* is from 0 to 24, the data type becomes FLOAT(). If *p* is from 25 to 53, the data type becomes DOUBLE()
DOUBLE(*size*, *d*)	A normal-size floating point number. The total number of digits is specified in *size*. The number of digits after the decimal point is specified in the *d* parameter
DOUBLE PRECISION(*size*, *d*)	
DECIMAL(*size*, *d*)	An exact fixed-point number. The total number of digits is specified in *size*. The number of digits after the decimal point is specified in the *d* parameter. The maximum number for *size* is 65. The maximum number for *d* is 30. The default value for *size* is 10. The default value for *d* is 0.
DEC(*size*, *d*)	Equal to DECIMAL(size,d)

Data types:
https://dev.mysql.com/doc/refman/8.0/en/data-types.html
https://www.w3schools.com/mysql/mysql_datatypes.asp

MYSQL – Tipos de dados (Domínio)

Dado do tipo
Date e Time



Data type	Description
DATE	A date. Format: YYYY-MM-DD. The supported range is from '1000-01-01' to '9999-12-31'
DATETIME(*fsp*)	A date and time combination. Format: YYYY-MM-DD hh:mm:ss. The supported range is from '1000-01-01 00:00:00' to '9999-12-31 23:59:59'. Adding DEFAULT and ON UPDATE in the column definition to get automatic initialization and updating to the current date and time
TIMESTAMP(*fsp*)	A timestamp. TIMESTAMP values are stored as the number of seconds since the Unix epoch ('1970-01-01 00:00:00' UTC). Format: YYYY-MM-DD hh:mm:ss. The supported range is from '1970-01-01 00:00:01' UTC to '2038-01-09 03:14:07' UTC. Automatic initialization and updating to the current date and time can be specified using DEFAULT CURRENT_TIMESTAMP and ON UPDATE CURRENT_TIMESTAMP in the column definition
TIME(*fsp*)	A time. Format: hh:mm:ss. The supported range is from '-838:59:59' to '838:59:59'
YEAR	A year in four-digit format. Values allowed in four-digit format: 1901 to 2155, and 0000. MySQL 8.0 does not support year in two-digit format.

Data types:
https://dev.mysql.com/doc/refman/8.0/en/data-types.html
https://www.w3schools.com/mysql/mysql_datatypes.asp

Schema

- Schema = banco de dados.

- Conjunto de objetos de banco de dados inter-relacionados:
  - Tabelas, colunas, tipos de dados, índices, chaves estrangeiras, entre outros.

Schema – Criar

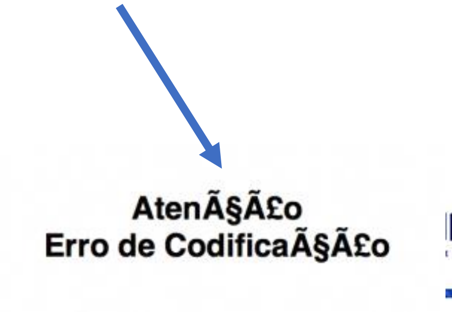
- CREATE {DATABASE | SCHEMA} [IF NOT EXISTS] nome_bd [**opções**];
  - **opções** : {[DEFAULT] CHARACTER SET [=] nome_charset | [DEFAULT] COLLATE [=] nome_agrupamento | [DEFAULT] ENCRYPTION [=] {'Y' | 'N’}}

UniCesumar

EDUCAÇÃO PRESENCIAL E A DISTÂNCIA

Schema – Criar – Exemplo

- CREATE SCHEMA **clinicavet** DEFAULT CHARACTER SET utf8;
- CREATE DATABASE IF NOT EXISTS **clinicavet** DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8_general_ci;



AtenÃ§Ã£o
Erro de CodificaÃ§Ã£o

Schema – Alterar

- ALTER {DATABASE | SCHEMA} nome_bd [**opções**];
  - **opções**: {[DEFAULT] CHARACTER SET [=] nome_charset | [DEFAULT] COLLATE [=] nome_agrupamento | [DEFAULT] ENCRYPTION [=] {'Y' | 'N’} | READ ONLY [=] {DEFAULT | 0 | 1}}

- Exemplo:
  - ALTER DATABASE **clinicavet** READ ONLY = 1 DEFAULT COLLATE utf8mb4_bin;

Schema – Excluir

- DROP {DATABASE | SCHEMA} [IF EXISTS] nome_bd;

- Exemplo:
  - DROP SCHEMA **clinicavet**;

Schema – Executar Comandos

- Executar os comandos SQL em um Banco de Dados:
  - USE nome_db;

- Por exemplo:
  - USE **clinicavet**;

Tabela – Criar

- CREATE [TEMPORARY] TABLE [IF NOT EXISTS] nome_tabela (definicao_colunas) [opções_tabela];

Tabela – Criar – Exemplo

```
CREATE TABLE Veterinario(
                    id integer PRIMARY KEY auto_increment,
                    nome varchar(100),
                    crmv numeric(10),
                    email varchar(200),
                    fone varchar(16),
                    especialidade varchar(100));
```

```
CREATE TABLE Veterinario_endereco (
                    id integer PRIMARY KEY auto_increment,
                    cep varchar(12),
                    rua varchar(100),
                    numero integer,
                    complemento varchar(50),
                    cidade varchar(50),
                    uf varchar(2),
                    id_vet integer NOT NULL,
                    CONSTRAINT fk_veterinário_endereço FOREIGN KEY (id_vet) REFERENCES Veterinario(id));
```

Tabela – Excluir

- DROP TABLE [IF EXISTS] nome_tabela;

- Exemplos:
    - DROP TABLE Veterinario_endereco;
    - DROP TABLE IF EXISTS Veterinario_endereco;

REFERÊNCIAS

RAMAKRISHNAN, Raghu; GEHRKE, Johannes. Sistemas de gerenciamento de banco de dados. São Paulo, SP: McGraw-Hill Interamericana do Brasil, 2008. xxvii, 884 p.

NAVATHE, Shamkant B.; ELMASRI, Ramez. Sistemas de banco de dados. Sham, Addison. Ribeirão Preto SP, 2005.

“Sucesso é o acúmulo de pequenos esforços, repetidos dia e noite.”

Robert Collier

OBRIGADO E BONS ESTUDOS!

UniCesumar
EDUCAÇÃO PRESENCIAL E A DISTÂNCIA

UniCesumar

EDUCAÇÃO PRESENCIAL E A DISTÂNCIA