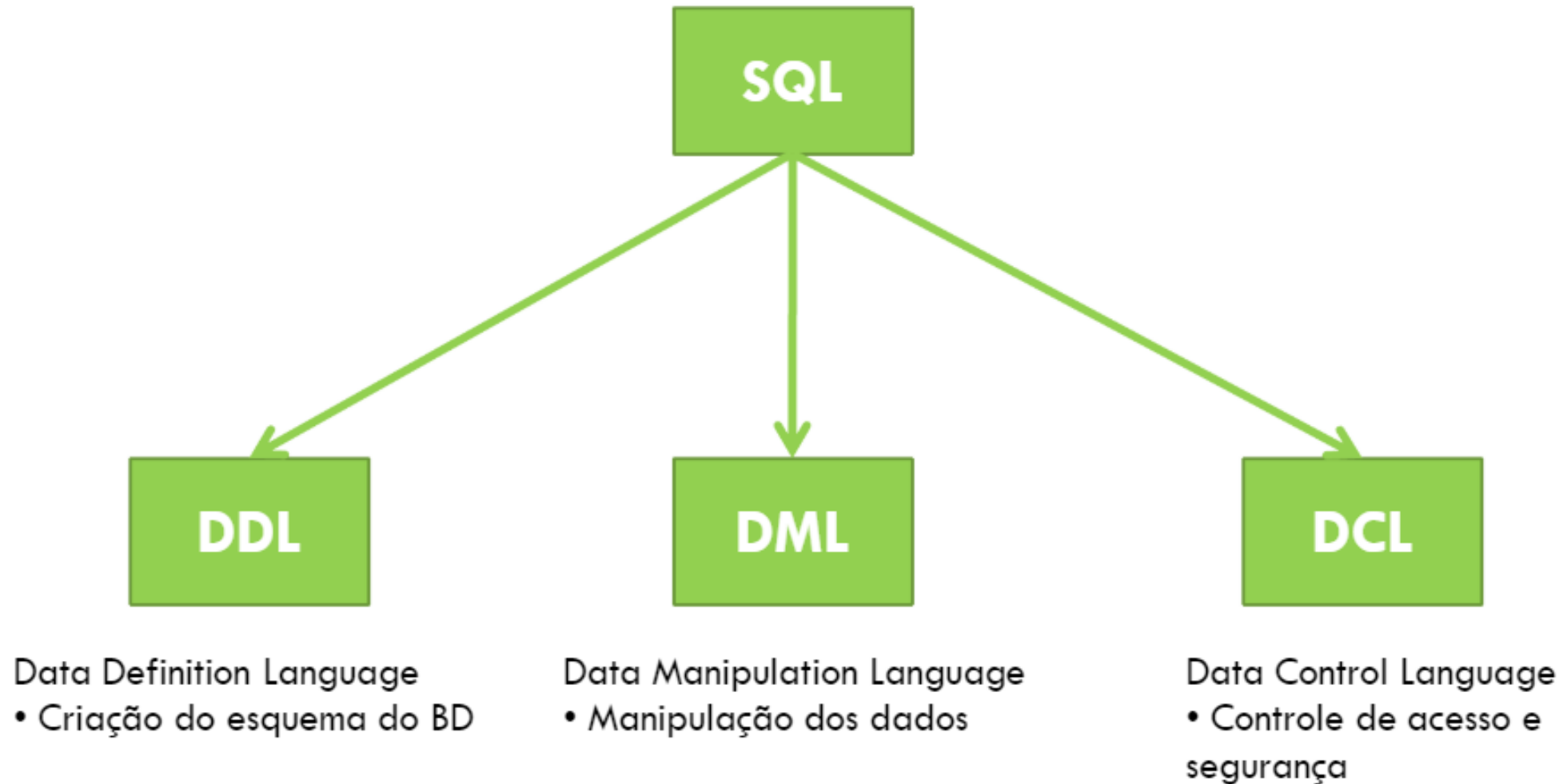


# Linguagem SQL - Structured Query Language

Linguagem de Consulta Estruturada



# Linguagem SQL - Structured Query Language

**BANCO DE DADOS**

# COMANDOS GERAIS DA LINGUAGEM

## SUBGRUPOS

Tipo de Comando	Exemplos de Comandos
Definição de Dados (DDL)	create table/alter table/drop table create index/ drop index
Manipulação de Dados (DML)	insert update delete
Consulta aos Dados (DQL)	select
Transação (DTL)	commit rollback savepoint
Controle (DCL)	grant revoke

# DDL – *Data Definition Language*

## COMANDOS

- ❑ Data Definition Language é utilizada para a criação do esquema do banco de dados.
- ❑ As principais ações desta linguagem são:
  - ❑ Criação de tabelas (CREATE TABLE)
  - ❑ Alteração das tabelas (ALTER TABLE)
  - ❑ Remoção de tabelas (DROP TABLE)
- ❑ Existem outras operações para manipulação dos índices:
  - ❑ CREATE INDEX, ALTER INDEX, DROP INDEX.

- DDL – Criar um banco(CREATE DATABASE)
- Eliminar um banco (DROP DATABASE)
  - Usar um banco (USE DATABASE)

**CREATE DATABASE teste;**

**CREATE DATABASE aula2;**

Cria os bancos de dados teste e aula2.

**DROP DATABASE teste;**

Elimina o banco de dados teste.

**USE aula2;**

Especifica o banco de dados que pretende utilizar.

# DDL – Criar Tabelas (CREATE TABLE)

## – Alterar Tabelas (ALTER TABLE)

```
CREATE TABLE table_name  
(  
    column_name1 data_type(size),  
    column_name2 data_type(size),  
    column_name3 data_type(size),  
    ....  
);
```

```
CREATE table Cliente  
( codigo int PRIMARY KEY,  
  nome varchar(40),  
  data_nascimento varchar(10),  
  endereco varchar(40),  
  telefone varchar(9)  
);
```

```
ALTER TABLE table_name  
ADD column_name datatype;
```

```
ALTER TABLE table_name  
DROP COLUMN column_name;
```

```
ALTER TABLE table_name  
MODIFY (ou ALTER) COLUMN column_name datatype;
```

```
ALTER table Cliente  
ADD email varchar(40);
```

```
ALTER table Cliente  
DROP telefone;
```

```
ALTER table Cliente  
MODIFY COLUMN data_nascimento date;
```

# DDL – Criar Tabelas (CREATE TABLE)

## – Eliminar Tabelas (DROP TABLE)

Os índices são usados para recuperar dados muito rapidamente, são usados apenas para acelerar pesquisas/consultas.

Atualizar uma tabela com índices leva mais tempo do que atualizar uma tabela sem (porque os índices também precisam de uma atualização).

Portanto, crie apenas índices em colunas que serão pesquisadas com frequência.

```
CREATE INDEX index_name ON table_name (column1, column2, ...);
```

```
CREATE INDEX idxnome ON Cliente (nome);
```

```
DROP TABLE table_name;
```

```
DROP TABLE Cliente;
```

# DDL – Criar Tabelas (CREATE TABLE)

## PRINCIPAIS CLÁUSULAS

- **CONSTRAINTS: São Restrições**

- São regras aplicadas nas colunas de uma tabela.
- Usadas para limitar os tipos de dados a serem aceitos.
- Podem ser especificadas na criação da tabela (CREATE) ou após como alteração (ALTER)
- As seguintes restrições são comumente usadas no SQL: NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY, CHECK, INDEX e outras.



# DDL – Criar Tabelas (CREATE TABLE)

## PRINCIPAIS CLÁUSULAS

- **PRIMARY KEY:** chave primária.
- **NOT NULL:** campo obrigatório.
- **UNIQUE:** não pode haver dois registros na tabela/coluna com o mesmo valor.
- **DEFAULT:** valor padrão é adicionado a todos os registros SE outro valor não for especificado.
- **CHECK:** a restrição CHECK em uma única coluna, permitirá apenas determinados valores para essa coluna.
- **AUTO INCREMENT:** padrão valor inicial é 1 e incrementa de 1 em 1 a cada novo registro.
  - Para alterar a sequência inicialize com outro valor. Exemplo: ALTER TABLE Cliente AUTO\_INCREMENT=1000;

# DDL – Criar Tabelas (CREATE TABLE)

## PRINCIPAIS CLÁUSULAS

- **Exemplo de CONSTRAINTS (Restrições)**

```
CREATE table aluno  
( codigo int not null,  
  nome varchar(40) not null,  
  sobrenome varchar(20) not null,  
  PRIMARY KEY (codigo),  
  CONSTRAINT Cpessoa UNIQUE(codigo,sobrenome)  
);
```

# DDL – Criar Tabelas (CREATE TABLE)

## EXEMPLO

```
CREATE table departamento  
( codigo int AUTO_INCREMENT,  
  nome varchar(40) not null,  
  localizacao varchar(20) DEFAULT 'Administrativo',  
  limite_lotacao int,  
  PRIMARY KEY (codigo)  
);
```

```
SHOW COLUMNS FROM departamento;
```

# DDL – Criar Tabelas (CREATE TABLE)

## PRINCIPAIS CLÁUSULAS

**FOREIGN KEY (campo\_estrangeiro)**

**REFERENCES nome-tabela-ref (chave estrangeira)**

A tabela que contém a chave estrangeira é chamada de tabela filha e a tabela que contém a chave primária é chamada de tabela pai ou referenciada.

A restrição FOREIGN KEY impede inserir na chave estrangeira dados que não existam na chave primária.

**ON DELETE <ação>**

**ON UPDATE <ação>**

Na exclusão ou alteração de um registro na tabela pai (PK) especifica o que faz com o registro nas tabelas filhas (FK), de acordo com as ações a seguir:

- **NO ACTION:** impede a ação na tabela pai (que tem a chave primária)
- **CASCADE** – propaga a ação da tabela pai (PK) nas tabelas filhas (FK).
- **SET NULL** – valores de referências alterados para nulo.
- **DEFAULT** – valores de referências alterados para *default*.

# DDL – Criar Tabelas (CREATE TABLE)

## EXEMPLO

```
CREATE table funcionario  
( matricula int,  
  nome varchar(50) not null,  
  cpf int unique,  
  cargo varchar(30) not null,  
  cod_departamento int,  
  PRIMARY KEY (matricula),  
  FOREIGN KEY (cod_departamento) REFERENCES Departamento(codigo)  
);
```

# DDL – *Data Definition Language*

## EXEMPLO

```
CREATE table dependente  
( codigodep int PRIMARY KEY,  
  codigofunc int not null,  
  nomedep varchar(30) not null,  
  data_nasc date,  
  FOREIGN KEY(codigofunc) REFERENCES funcionario(matricula)  
  ON DELETE CASCADE  
  ON UPDATE CASCADE  
);
```

# DML – *Data Manipulation Language*

## COMANDOS

- Linguagem para manipulação dos dados.
- Existem 4 operações principais:
  - Insert – Inclusão de dados
  - Update – Alteração dos dados
  - Delete – Exclusão de dados
  - Select – Seleção de dados

# DML – Inserir Dados (INSERT)

## **Possui duas formas de inserir dados:**

### **1. Informando as colunas que deseja inserir valores:**

```
INSERT INTO departamento (codigo, nome, localizacao, limite_lotacao)  
VALUES ('1', 'TI', 'Adm04', '20');
```

### **2. Não informamos as colunas e valores para todas as colunas:**

```
INSERT INTO departamento VALUES ('1', 'TI', 'Adm04', '20');
```

**Como o código foi declarado como AUTO\_INCREMENT E localização como DEFAULT não é necessário inserir os valores destas colunas.**

```
INSERT INTO departamento (nome, limite_lotacao) VALUES ('Logística', '30');
```



# DML – Atualizar Dados (UPDATE)

**Comando utilizado para alterar registros existentes em uma tabela**

```
UPDATE table_name
```

```
SET column1 = value1, column2 = value2, ...
```

```
WHERE condition;
```

```
UPDATE departamento
```

```
SET nome='Vendas'
```

```
WHERE codigo=1;
```

```
UPDATE departamento
```

```
SET nome='Vendas', localizacao='ADM02'
```

```
WHERE codigo=2;
```

A cláusula WHERE especifica quais registros devem ser atualizados, quando omitida, todos os registros da tabela serão atualizados!

# DML – Deletar Dados (DELETE)

**Comando utilizado para é usada para excluir registros existentes em uma tabela**

```
DELETE FROM table_name  
WHERE condition;
```

```
DELETE FROM departamento  
WHERE codigo=2;
```

```
DELETE FROM departamento  
WHERE localizacao='adm05';
```

A cláusula WHERE especifica quais registros devem ser excluídos, quando omitida, todos os registros da tabela serão atualizados!

# Prática

## Como exportar um BD

The screenshot shows the phpMyAdmin web interface in a browser. The address bar indicates the URL is `localhost/phpmyadmin/db_export.php?db=exportacao`. The left sidebar shows a tree view of databases, with `exportacao` highlighted. The main content area displays the export configuration for the `exportacao` database. The `Exportar` button in the top navigation bar is highlighted with a red box. The `exportacao` database in the sidebar is also highlighted with a red box. The `Executar` button at the bottom is highlighted with a red box.

phpMyAdmin

Recente Favoritos

Base de Dados: exportacao

Estrutura SQL Pesquisar Pesquisa por formulário Exportar Importar Operações Privilegios Rotinas Mais

### Exportando as tabelas da base de dados "exportacao"

**Export templates:**

New template: Existing templates:

Template name:  Criar Template: -- Select a template -- Update Apagar

**Export method:**

☒ Rápido - exibe o mínimo de opções

☐ Personalizado - exibe todas as opções possíveis

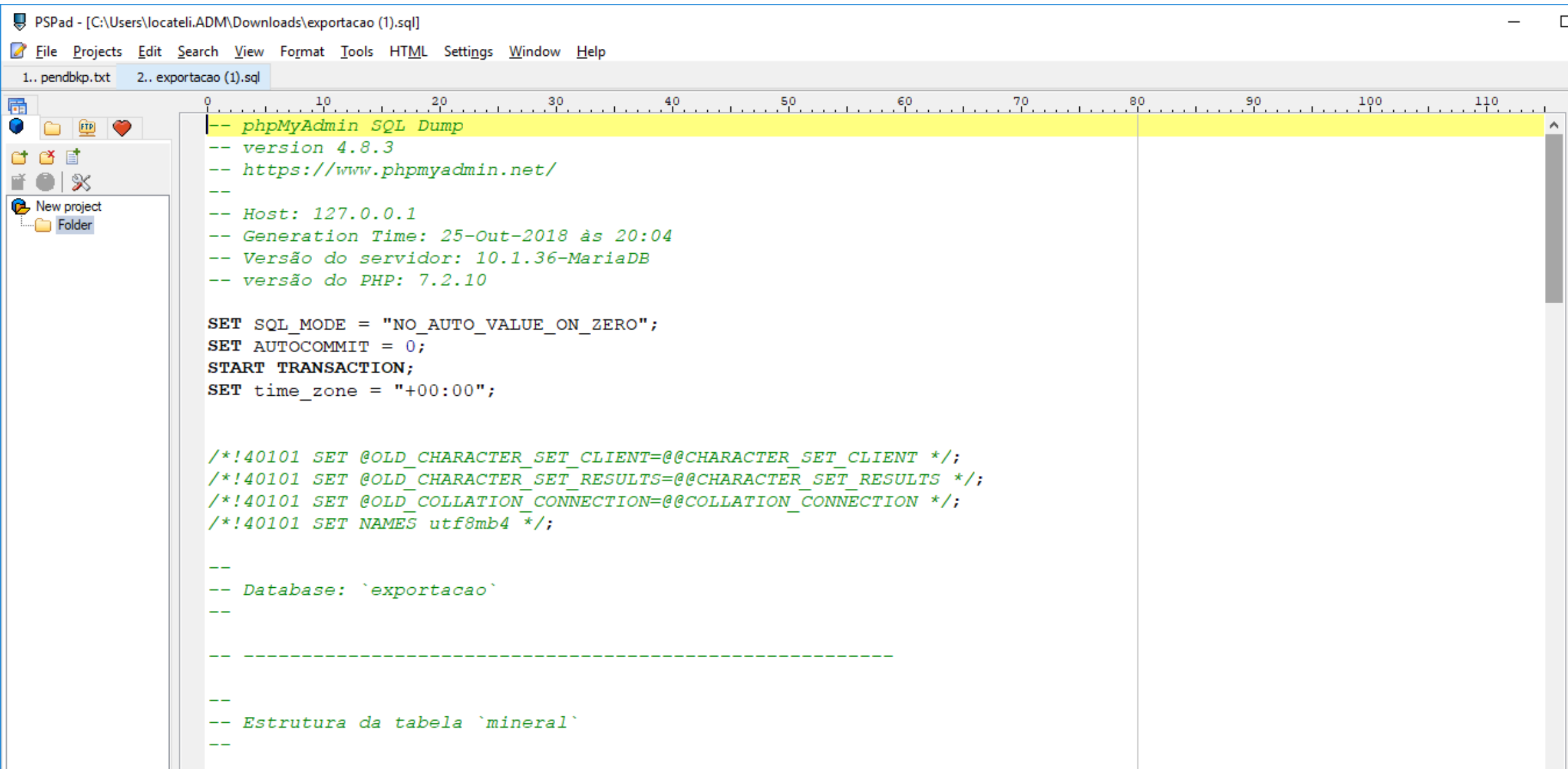
**Formato:**

SQL

Executar

# Prática

## Como exportar um BD



The image shows a screenshot of the PSPad text editor. The title bar indicates the file path: C:\Users\locateli.ADM\Downloads\exportacao (1).sql. The menu bar includes File, Projects, Edit, Search, View, Format, Tools, HTML, Settings, Window, and Help. The tab bar shows two tabs: 1.. pendbkp.txt and 2.. exportacao (1).sql. The left sidebar contains icons for file operations and a 'New project' button. The main editor area displays an SQL dump file with the following content:

```
-- phpMyAdmin SQL Dump
-- version 4.8.3
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Generation Time: 25-Out-2018 às 20:04
-- Versão do servidor: 10.1.36-MariaDB
-- versão do PHP: 7.2.10

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET AUTOCOMMIT = 0;
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `exportacao`
--

--
-- Estrutura da tabela `mineral`
--
```