

# PHPMyAdmin

---

Prof. Dr. Tiago Eugenio de Melo  
[tmelo@uea.edu.br](mailto:tmelo@uea.edu.br)

# PHPMyAdmin

—

# Apresentação

# Apresentação

- É um software livre escrito em PHP.

# Apresentação

- É um software livre escrito em PHP.
- O objetivo é gerenciar o MySQL através da Web.

# Apresentação

- É um software livre escrito em PHP.
- O objetivo é gerenciar o MySQL através da Web.
- Vasta documentação [[1](#)].

# Apresentação

- É um software livre escrito em PHP.
- O objetivo é gerenciar o MySQL através da Web.
- Vasta documentação [[1](#)].
- Suporte a vários idiomas, inclusive PT-BR.

# Apresentação

- É um software livre escrito em PHP.
- O objetivo é gerenciar o MySQL através da Web.
- Vasta documentação [[1](#)].
- Suporte a vários idiomas, inclusive PT-BR.
- Versão atual: 5.2.1.



# Apresentação

- É um software livre escrito em PHP.
- O objetivo é gerenciar o MySQL através da Web.
- Vasta documentação [[1](#)].
- Suporte a vários idiomas, inclusive PT-BR.
- Versão atual: 5.2.1.
- Site: <https://www.phpmyadmin.net>

# Apresentação

- É um software livre escrito em PHP.
- O objetivo é gerenciar o MySQL através da Web.
- Vasta documentação [[1](#)].
- Suporte a vários idiomas, inclusive PT-BR.
- Versão atual: 5.2.1.
- Site: <https://www.phpmyadmin.net>

# Instalação



# Requisitos

# Requisitos

- Servidor Web

# Requisitos

- Servidor Web
- PHP

# Requisitos

- Servidor Web
- PHP
- Banco de datos

# Requisitos

- Servidor Web
- PHP
- Banco de dados
  - MySQL 5.5 ou adiante.



# Requisitos

- Servidor Web
- PHP
- Banco de dados
  - MySQL 5.5 ou adiante.
  - MariaDB 5.5 ou adiante.

# Linux

# Linux

- Uso de apt-get.

# Linux

- Uso de apt-get.
- Fonte: [https://docs.phpmyadmin.net/pt\\_BR/latest/setup.html](https://docs.phpmyadmin.net/pt_BR/latest/setup.html)

# Linux

- Uso de apt-get.
- Fonte: [https://docs.phpmyadmin.net/pt\\_BR/latest/setup.html](https://docs.phpmyadmin.net/pt_BR/latest/setup.html)

# Windows

# Windows

- A forma mais simples de instalação é através do XAMPP.

# Windows

- A forma mais simples de instalação é através do XAMPP.
- Processo de instalação: <https://www.apachefriends.org/index.html>



# Windows

- A forma mais simples de instalação é através do XAMPP.
- Processo de instalação: <https://www.apachefriends.org/index.html>

# Acesso

---

# Tela de Acesso

- O acesso local é através do localhost.
- Exemplo:
  - <http://localhost/phpmyadmin>



phpMyAdmin

Bem-vindo ao phpMyAdmin

Linguagem - *Language*

Português - Brazilian Portuguese ▾

Entrar ⓘ

Usuário:

Senha:

Escolha de servidor: localhost:3306 ▾

Executar

# Tela Principal

The screenshot displays the phpMyAdmin web interface. At the top, a navigation bar contains tabs for 'Base de Dados', 'SQL', 'Estado', 'Exportar', 'Importar', 'Configurações', 'Variáveis', 'Mapas de Caracteres', 'Motores', and 'Plugins'. The 'Configurações' tab is active.

On the left side, there is a sidebar with the phpMyAdmin logo and a list of recent databases: 'information\_schema' and 'u351623632\_bd1'.

The main content area is divided into two panels:

- Definições gerais:** Contains a dropdown menu for 'Ordenação de caracteres da ligação ao servidor' set to 'utf8mb4\_unicode\_ci'.
- Configurações de aspecto:** Contains settings for 'Língua - Language' (set to 'Português - Portuguese'), 'Tema' (set to 'pmahome'), and 'Tamanho da fonte' (set to '82%'). A link for 'Mais definições' is also present.

On the right side, a 'phpMyAdmin' sidebar provides a list of links:

- Informação da versão: 4.9.5
- Documentação
- Página Oficial do phpMyAdmin
- Contributo
- Obter suporte
- Lista de alterações
- Licença

A 'Consola' tab is visible at the bottom left of the interface.

# Operações

—

# Criação de Tabelas

Servidor: 127.0.0.1:3306 » Base de Dados: u351623632\_bd1

Estrutura SQL Pesquisar Pesquisa por formulário Exportar Importar Operações Rotinas Eventos Acionadores Desenhador

⚠ Nenhuma tabela encontrada na base de dados.

**Criar tabela**

Nome:  Número de colunas:

Executar

# Criação de Tabelas

Servidor: 127.0.0.1:3306 » Base de Dados: u351623632\_bd1 » Tabela: aluno

Procurar Estrutura SQL Pesquisar Inserir Exportar Importar Operações Acionadores

Nome da Tabela:  Adicionar  coluna(s)

**Estrutura**

Nome	Tipo	Tamanho/Valores*	Predefinido	Agrupamento (Collation)	Atributos	Nulo	Índice	A_I	Comentários	Virtualidade	Mover coluna(s)
<input type="text"/>	INT	<input type="text"/>	Nenhum	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	INT	<input type="text"/>	Nenhum	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	INT	<input type="text"/>	Nenhum	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	INT	<input type="text"/>	Nenhum	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Comentários da tabela:

Colação:

Motor de armazenamento:

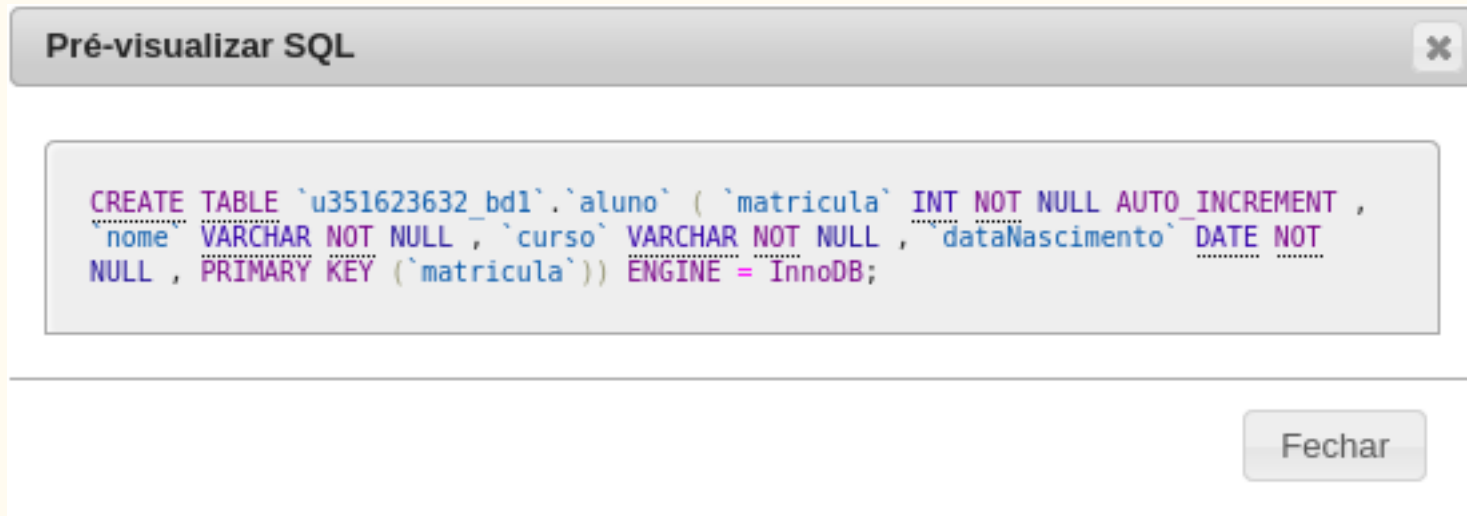
**Definição da PARTIÇÃO:**

Partição por:  ( Expressão ou lista de colunas )

Partições:

# Criação de Tabelas

- Pré-visualização



```
CREATE TABLE `u351623632_bdl`.`aluno` ( `matricula` INT NOT NULL AUTO_INCREMENT ,  
`nome` VARCHAR NOT NULL , `curso` VARCHAR NOT NULL , `dataNascimento` DATE NOT  
NULL , PRIMARY KEY (`matricula`)) ENGINE = InnoDB;
```

Fechar



# Criação de Tabelas

## ● Estrutura das Tabelas

The screenshot displays the 'Estrutura da tabela' (Table Structure) view for a table named 'aluno'. The interface includes a menu bar with options like 'Procurar', 'Estrutura', 'SQL', 'Pesquisar', 'Inserir', 'Exportar', 'Importar', 'Operações', and 'Acionadores'. Below the menu, there are tabs for 'Estrutura da tabela' and 'Visão de relação(ões)'. The main area shows a table with columns: #, Nome, Tipo, Agrupamento (Collation), Atributos, Nulo, Predefinido, Comentários, Extra, and Acções. The columns are: 1. matricula (int(11)), 2. nome (varchar(45)), 3. curso (varchar(45)), and 4. dataNascimento (date). Below the table, there are options to 'Marcar todos', 'Com os seleccionados', and 'Executar'. There is also a section for 'Índices' (Indices) showing a primary key on the 'matricula' column. The 'Partições' (Partitions) section shows 'Nenhum particionamento definido!'. The 'Informação' (Information) section provides details about the table's size and statistics.

#	Nome	Tipo	Agrupamento (Collation)	Atributos	Nulo	Predefinido	Comentários	Extra	Acções
1	matricula	int(11)			Não	Nenhum		AUTO_INCREMENT	Muda Elimina Mais
2	nome	varchar(45)	utf8mb4_unicode_ci		Não	Nenhum			Muda Elimina Mais
3	curso	varchar(45)	utf8mb4_unicode_ci		Não	Nenhum			Muda Elimina Mais
4	dataNascimento	date			Não	Nenhum			Muda Elimina Mais

Acções	Nome da chave	Tipo	Único	Pacote	Coluna	Quantidade	Agrupamento (Collation)	Nulo	Comentário
Edita Elimina	PRIMARY	BTREE	Sim	Não	matricula	0	A	Não	

Espaço ocupado		Estatísticas do registo	
Dados	16.0 KB	Formato	dinâmico
Índice	0 Bytes	Agrupamento (Collation)	utf8mb4_unicode_ci
Suspensão		Próximo auto-índice	1
Em uso	16.0 KB	Criação	23-Nov-2021 às 14:07
Total	16.0 KB	Última actualização	23-Nov-2021 às 14:07
		Última Verificação	23-Nov-2021 às 14:07

# Inserção

Servidor: 127.0.0.1:3306 » Base de Dados: u351623632\_bd1 » Tabela: aluno

Procurar Estrutura SQL Pesquisar Inserir Exportar Importar Operações Acionadores

Coluna	Tipo	Funções	Nulo	Valor
matricula	int(11)			
nome	varchar(45)			
curso	varchar(45)			
dataNascimento	date			

Executar

Ignora

Coluna	Tipo	Funções	Nulo	Valor
matricula	int(11)			
nome	varchar(45)			
curso	varchar(45)			
dataNascimento	date			

Executar

Inserir como novo registo e então Voltar atrás

Executar Pré-visualizar SQL Reiniciar

Continuar a inserção com 2 linhas

# Inserção

Servidor: 127.0.0.1:3306 » Base de Dados: u351623632\_bd1 » Tabela: aluno

Procurar Estrutura SQL Pesquisar Inserir Exportar Importar Operações Acionadores

✓ 1 linha inserida.  
Id da linha inserida: 1

```
INSERT INTO `aluno` (`matricula`, `nome`, `curso`, `dataNascimento`) VALUES (NULL, 'Zé Carioca', 'Engenharia da Computação', '2000-01-01');
```

[Editar em linha] [Edita] [Criar código PHP]

Executar consulta(s) SQL na tabela u351623632\_bd1.aluno:

```
1 INSERT INTO `aluno` (`matricula`, `nome`, `curso`, `dataNascimento`) VALUES (NULL, 'Zé Carioca', 'Engenharia da Computação', '2000-01-01');
```

Colunas

- matricula
- nome
- curso
- dataNascimento

SELECT \* SELECT INSERT UPDATE DELETE Limpar Formato Obter consulta auto-salva

Unir parâmetros

[ Delimitador ; ]  Mostrar de novo aqui este comando  Reter a caixa da consulta (query)  Rollback quando terminado  Ativa a verificação de chaves estrangeiras

Executar

# Inserção

## ● Linha inserida:

Servidor: 127.0.0.1:3306 » Base de Dados: u351623632\_bd1 » Tabela: aluno

Procurar Estrutura SQL Pesquisar Inserir Exportar Importar Operações Acionadores

```
INSERT INTO `aluno` (`matricula`, `nome`, `curso`, `dataNascimento`) VALUES (NULL, 'Zé Carioca', 'Engenharia da Computação', '2000-01-01');
```

[Editar em linha] [ Edita ] [ Criar código PHP ]

Mostrar Caixa do query

✓ 1 linha inserida.  
Id da linha inserida: 2 (A consulta demorou 0.0003 segundos.)

```
INSERT INTO `aluno` (`matricula`, `nome`, `curso`, `dataNascimento`) VALUES (NULL, 'Zé Carioca', 'Engenharia da Computação', '2000-01-01');
```

[Editar em linha] [ Edita ] [ Criar código PHP ]

# Consultas

- É possível realizar consultas SQL.

The screenshot shows a database management tool interface. At the top, the address bar indicates the server IP (127.0.0.1:3306), the database name (u351623632\_bd1), and the table name (aluno). The main menu includes options like Procurar, Estrutura, SQL, Pesquisar, Inserir, Exportar, Importar, Operações, and Acionadores. The central area is titled "Executar consulta(s) SQL na tabela u351623632\_bd1.aluno:" and contains a text editor with the SQL query: `1 SELECT * FROM 'aluno' WHERE 1`. Below the editor are buttons for SQL operations: SELECT \*, SELECT, INSERT, UPDATE, DELETE, Limpar, Formato, and Obter consulta auto-salva. A checkbox for "Unir parâmetros" is also present. On the right, a "Colunas" panel lists the table's columns: matricula, nome, curso, and dataNascimento. At the bottom, there are settings for the query delimiter (set to semicolon), checkboxes for "Mostrar de novo aqui este comando", "Reter a caixa da consulta (query)", "Rollback quando terminado", and "Ativa a verificação de chaves estrangeiras", and an "Executar" button.

# Consultas

## ● Saída da consulta:

Servidor: 127.0.0.1:3306 » Base de Dados: u351623632\_bd1 » Tabela: aluno

Procurar Estrutura SQL Pesquisar Inserir Exportar Importar Operações Acionadores

Mostrar Caixa do query

✔ A mostrar registros de 0 - 1 (2 total, A consulta demorou 0.0002 segundos.)

```
SELECT * FROM `aluno` WHERE 1
```

Perfil [Editar em linha] [Edita] [Explicar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de registros: 25 | Filtrar registros: Pesquisar esta tabela | Ordenar por chave: Nenhum

+ Opções

	matricula	nome	curso	dataNascimento
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Apagar	1	Zé Carioca	Engenharia da Computação	2000-01-01
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Apagar	2	Recruta Zero	Engenharia Química	2000-01-31

Marcar todos | Com os seleccionados:  Editar  Copiar  Apagar  Exportar

Mostrar tudo | Número de registros: 25 | Filtrar registros: Pesquisar esta tabela | Ordenar por chave: Nenhum

**Operações resultantes das consultas**

Imprimir  Copiar para área de transferência  Exportar  Mostrar gráfico  Criar visualização

# Exportar

← Servidor: 127.0.0.1:3306 > Base de Dados: u351623632\_bd1 > Tabela: aluno

Procurar Estrutura SQL Pesquisar Inserir Exportar Importar Operações Acionadores

## Exportando as linhas da tabela "aluno"

**Método de exportação:**

Rápido - exibe o mínimo de opções

Personalizado - exibe todas as opções possíveis

**Formato:**

SQL

**Linhas:**

Eliminar alguma(s) linha(s)

Número de registos:

Começar na linha:

Eliminar todas as linhas

**Executar**

# Importar

Servidor: 127.0.0.1:3306 » Base de Dados: u351623632\_bd1 » tabela: aluno

Procurar Estrutura SQL Pesquisar Inserir Exportar Importar Operações Acionadores

## Fazendo importação para a tabela "aluno"

**Ficheiro a importar:**

O ficheiro pode ser comprimido (gzip, bzip2, zip) ou descomprimido.  
O nome de um ficheiro compactado deve terminar em **.[formato].[compactação]**. Exemplo: **.sql.zip**

Procurar no seu computador:  No file selected. (Tamanho máximo: 256MB)

Você também pode arrastar um ficheiro em qualquer página.

Configurar o Mapa de Caracteres do ficheiro:

**Importação parcial:**

Permite a interrupção da importação caso o script detecte que está perto do tempo limite do PHP. *(Isso pode ser uma boa forma de importar ficheiros grandes, mas pode interromper as transações.)*

Pular esta quantidade de consultas (para SQL), iniciando da primeira:

**Outras opções:**

Ativa a verificação de chaves estrangeiras

**Formato:**

**Opções específicas do formato:**

Modo de compatibilidade SQL:

Não use AUTO\_INCREMENT para valores zerados



# Visualização Gráfica

## ● Desenhador

Servidor: 127.0.0.1:3306 » Base de Dados: u351623632\_bd1

Estrutura SQL Pesquisar Pesquisa por formulário Exportar Importar Operações Rotinas Eventos Acionadores Desenhador

Filtros

Contendo a palavra:

Tabela	Acções	Registos	Tipo	Agrupamento (Collation)	Tamanho	Suspensão
<input type="checkbox"/> aluno	★ Procurar Estrutura Pesquisar Inserir Limpar Eliminar	2	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
<input type="checkbox"/> disciplina	★ Procurar Estrutura Pesquisar Inserir Limpar Eliminar	0	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
<b>2 tabelas</b>	<b>Soma</b>	<b>2</b>	<b>InnoDB</b>	<b>utf8mb4_unicode_ci</b>	<b>48.0 KB</b>	<b>0 Bytes</b>

↑  Marcar todos Com os seleccionados:

Imprimir Dicionário de dados

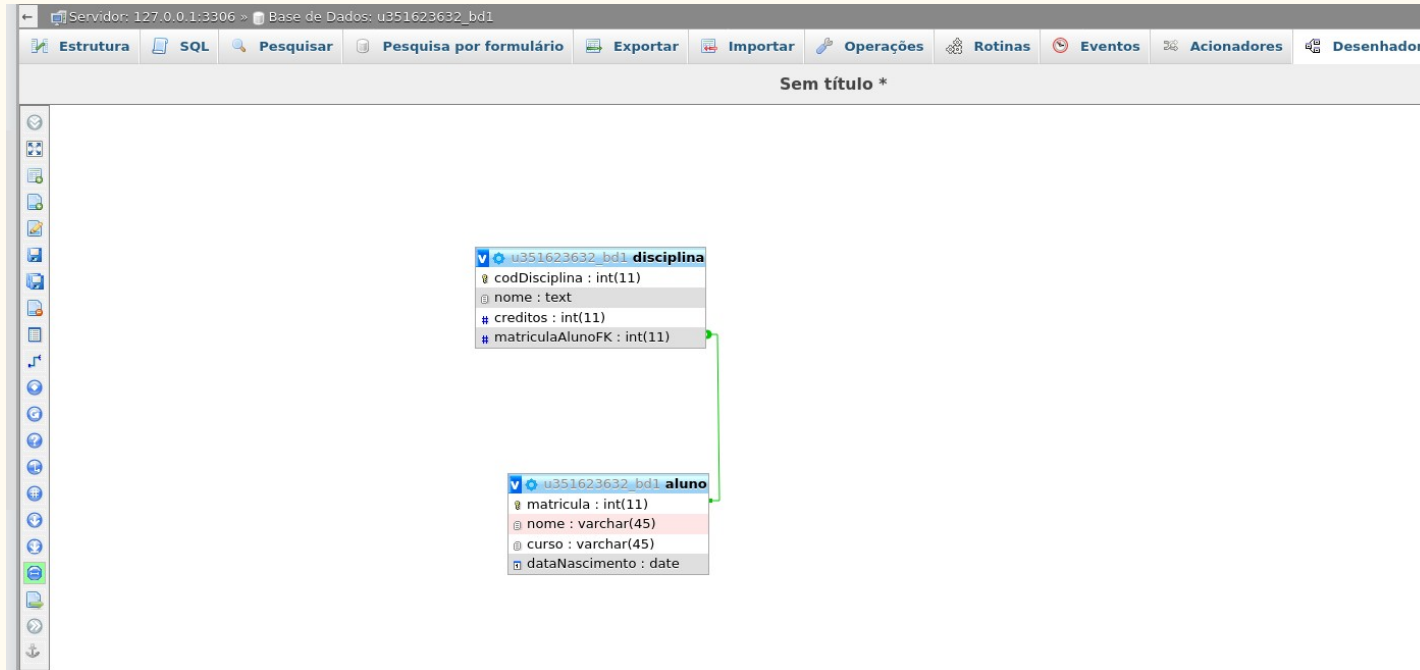
Criar tabela

Nome:  Número de colunas:

Executar

# Visualização Gráfica

## ● Exemplo:





# Mão na massa

---

# Criação BD

---



# Create DataBase

# Create DataBase

- Sintaxe:

# Create DataBase

- Sintaxe:

```
CREATE DATABASE [IF NOT EXISTS] database_name  
[CHARACTER SET charset_name]  
[COLLATE collation_name]
```

# Create DataBase

- Sintaxe:

```
CREATE DATABASE [IF NOT EXISTS] database_name  
[CHARACTER SET charset_name]  
[COLLATE collation_name]
```

- CHARACTER SET



# Create DataBase

- Sintaxe:

```
CREATE DATABASE [IF NOT EXISTS] database_name  
[CHARACTER SET charset_name]  
[COLLATE collation_name]
```

- CHARACTER SET

- É um conjunto de caracteres que são permitidos em uma string.

# Create DataBase

- Sintaxe:

```
CREATE DATABASE [IF NOT EXISTS] database_name  
[CHARACTER SET charset_name]  
[COLLATE collation_name]
```

- CHARACTER SET

- É um conjunto de caracteres que são permitidos em uma string.

- COLLATE

# Create DataBase

- Sintaxe:

```
CREATE DATABASE [IF NOT EXISTS] database_name  
[CHARACTER SET charset_name]  
[COLLATE collation_name]
```

- CHARACTER SET

- É um conjunto de caracteres que são permitidos em uma string.

- COLLATE

- É um conjunto de regras usadas para comparar strings em um particular *character set*.

# Create DataBase

- Exemplo:

```
CREATE DATABASE mydb  
CHARACTER SET utf8  
COLLATE utf8_unicode_ci;
```

# Apagar BD

---



# Drop DataBase

# Drop DataBase

- Sintaxe:

# Drop DataBase

- Sintaxe:

```
DROP DATABASE [IF EXISTS] database_name;
```



# Drop DataBase

- Sintaxe:

```
DROP DATABASE [IF EXISTS] database_name;
```

- Sintaxe [alternativa]:

# Drop DataBase

- Sintaxe:

```
DROP DATABASE [IF EXISTS] database_name;
```

- Sintaxe [alternativa]:

```
DROP SCHEMA [IF EXISTS] database_name;
```

# Drop DataBase

- Sintaxe:

```
DROP DATABASE [IF EXISTS] database_name;
```

- Sintaxe [alternativa]:

```
DROP SCHEMA [IF EXISTS] database_name;
```

- No MySQL, *schema* é sinônimo de *database*.



# Criação Tabela

---

# Create Table

# Create Table

- Sintaxe:

# Create Table

- Sintaxe:

```
CREATE TABLE [IF NOT EXISTS] table_name(  
    column_1_definition,  
    column_2_definition,  
    ...,  
    table_constraints  
) ENGINE=storage_engine;
```

# Create Table

- Sintaxe:

```
CREATE TABLE [IF NOT EXISTS] table_name(  
    column_1_definition,  
    column_2_definition,  
    ...,  
    table_constraints  
) ENGINE=storage_engine;
```

- [IF NOT EXISTS] é opcional.



# Create Table

- Sintaxe:

```
CREATE TABLE [IF NOT EXISTS] table_name(  
    column_1_definition,  
    column_2_definition,  
    ...,  
    table_constraints  
) ENGINE=storage_engine;
```

- [IF NOT EXISTS] é opcional.
- Engine

# Create Table

- Sintaxe:

```
CREATE TABLE [IF NOT EXISTS] table_name(  
    column_1_definition,  
    column_2_definition,  
    ...,  
    table_constraints  
) ENGINE=storage_engine;
```

- [IF NOT EXISTS] é opcional.

- Engine

- InnoDB e MyISAM.

# Create Table

# Create Table

- Engine

# Create Table

- Engine
  - InnoDB e MyISAM.

# Create Table

- Engine
  - InnoDB e MyISAM.
- InnoDB

# Create Table

- Engine
  - InnoDB e MyISAM.
- InnoDB
  - É o padrão do MySQL a partir da versão 5.5.

# Create Table

- Engine
  - InnoDB e MyISAM.
- InnoDB
  - É o padrão do MySQL a partir da versão 5.5.
  - Implementa as regras de ACID.



# Create Table

- Engine
  - InnoDB e MyISAM.
- InnoDB
  - É o padrão do MySQL a partir da versão 5.5.
  - Implementa as regras de ACID.

ACID: *atomicity, consistency, isolation, durability.*

# Create Table

- Engine
  - InnoDB e MyISAM.
- InnoDB
  - É o padrão do MySQL a partir da versão 5.5.
  - Implementa as regras de ACID.
  - Integridade referencial.

ACID: *atomicity, consistency, isolation, durability.*

# Create Table

- Engine
  - InnoDB e MyISAM.
- InnoDB
  - É o padrão do MySQL a partir da versão 5.5.
  - Implementa as regras de ACID.
  - Integridade referencial.
  - Recuperação de falhas (*crash*).

ACID: *atomicity, consistency, isolation, durability*.

# Create Table

- Exemplo:

```
CREATE TABLE IF NOT EXISTS tasks (  
    task_id INT AUTO_INCREMENT PRIMARY KEY,  
    title VARCHAR(255) NOT NULL,  
    start_date DATE,  
    due_date DATE,  
    status TINYINT NOT NULL,  
    priority TINYINT NOT NULL,  
    description TEXT,  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
) ENGINE=INNODB;
```

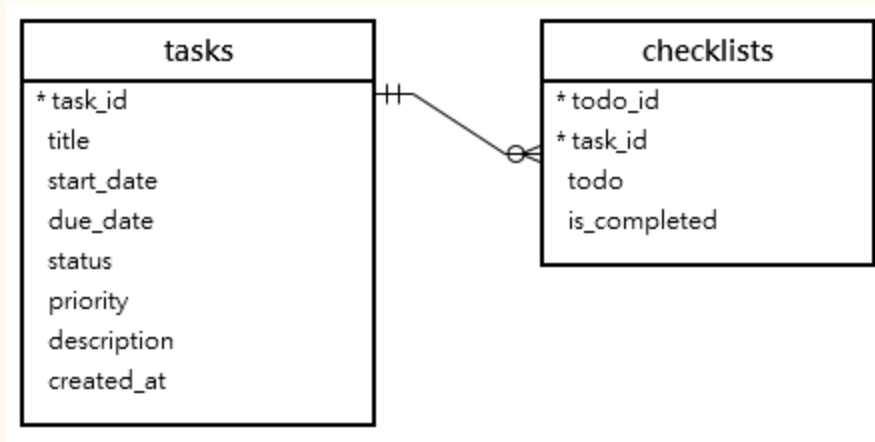
# Create Table

# Create Table

- Exemplo com **chave estrangeira**:

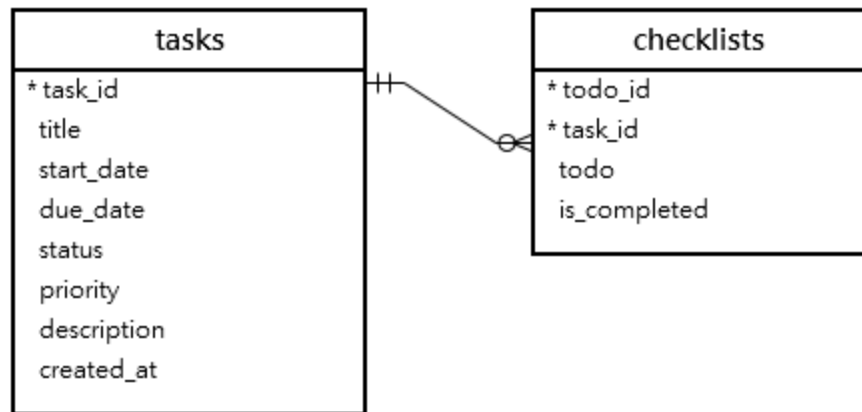
# Create Table

- Exemplo com **chave estrangeira**:



# Create Table

- Exemplo com **chave estrangeira**:



```
CREATE TABLE IF NOT EXISTS checklists (  
  todo_id INT AUTO_INCREMENT,  
  task_id INT,  
  todo VARCHAR(255) NOT NULL,  
  is_completed BOOLEAN NOT NULL DEFAULT FALSE,  
  PRIMARY KEY (todo_id , task_id),  
  FOREIGN KEY (task_id)  
    REFERENCES tasks (task_id)  
    ON UPDATE RESTRICT ON DELETE CASCADE  
);
```





# Apagar Tabela

---

# Drop Table

# Drop Table

- Sintaxe:

# Drop Table

- Sintaxe:

```
DROP [TEMPORARY] TABLE [IF EXISTS] table_name [, table_name] ...  
[RESTRICT | CASCADE]
```

# Drop Table

- Sintaxe:

```
DROP [TEMPORARY] TABLE [IF EXISTS] table_name [, table_name] ...  
[RESTRICT | CASCADE]
```

- A opção *temporary* permite remover somente as tabelas temporárias.

# Drop Table

- Sintaxe:

```
DROP [TEMPORARY] TABLE [IF EXISTS] table_name [, table_name] ...  
[RESTRICT | CASCADE]
```

- A opção *temporary* permite remover somente as tabelas temporárias.
- As opções *restrict* e *cascade* estão reservadas para futuras opções do MySQL.

# Drop Table

- Exemplo:

```
DROP TABLE CarAccessories, CarGadgets;
```



# Alterar Tabela

---



# Alter Table

# Alter Table

- Sintaxe:

# Alter Table

- Sintaxe:

```
ALTER TABLE table_name
ADD
    new_column_name column_definition
    [FIRST | AFTER column_name]
```

# Alter Table

- Sintaxe:

```
ALTER TABLE table_name  
ADD  
    new_column_name column_definition  
    [FIRST | AFTER column_name]
```

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

# Alter Table

- Sintaxe:

```
ALTER TABLE table_name  
ADD  
    new_column_name column_definition  
    [FIRST | AFTER column_name]
```

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

```
ALTER TABLE table_name  
RENAME TO new_table_name;
```

# Alter Table

- Sintaxe:

```
ALTER TABLE table_name
ADD
    new_column_name column_definition
    [FIRST | AFTER column_name]
```

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

```
ALTER TABLE table_name
RENAME TO new_table_name;
```

```
ALTER TABLE table_name
CHANGE COLUMN original_name new_name column_definition
    [FIRST | AFTER column_name];
```

# Alter Table

- Exemplo:

```
ALTER TABLE vehicles  
ADD model VARCHAR(100) NOT NULL;
```



# Inserir Dados

---



# Insert

- Sintaxe:

```
INSERT INTO table(c1,c2,...)  
VALUES (v1,v2,...);
```

# Insert

# Insert

- Exemplo:

# Insert

- Exemplo:

```
INSERT INTO tasks(title,priority)
VALUES('Learn MySQL INSERT Statement',1);
```

# Insert

- Exemplo:

```
INSERT INTO tasks(title,priority)
VALUES('Learn MySQL INSERT Statement',1);
```

```
1 row(s) affected
```

# Insert

- Exemplo:

```
INSERT INTO tasks(title,priority)
VALUES('Learn MySQL INSERT Statement',1);
```

1 row(s) affected

	task_id	title	start_date	due_date	priority	description
▶	1	Learn MySQL INSERT Statement	NULL	NULL	1	NULL



# Apagar Dados

---

# Delete

- Sintaxe:

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



# Delete

- Exemplo:

```
DELETE FROM employees  
WHERE officeCode = 4;
```



# Alterar Dados

---

# Update

- Sintaxe:

```
UPDATE [LOW_PRIORITY] [IGNORE] table_name
SET
    column_name1 = expr1,
    column_name2 = expr2,
    ...
[WHERE
    condition];
```

# Views

---



# View

# View

# View

# View



# View

# View

- As *views* (visões) são usadas para criar uma tabela virtual baseada em uma consulta SQL previamente definida.
- As *views* se parecem bastante com uma tabela.
- As *views* têm linhas e colunas que podem ter dados de diversas colunas e de múltiplas tabelas de um banco de dados.
- Vantagens:
  - Aumento de segurança por definir uma visão limitada e controlada dos dados que podem ser acessados no banco de dados.
  - Melhoria da performance por utilizar uma consulta previamente otimizada.

# View

- As *views* (visões) são usadas para criar uma tabela virtual baseada em uma consulta SQL previamente definida.
- As *views* se parecem bastante com uma tabela.
- As *views* têm linhas e colunas que podem ter dados de diversas colunas e de múltiplas tabelas de um banco de dados.
- Vantagens:
  - Aumento de segurança por definir uma visão limitada e controlada dos dados que podem ser acessados no banco de dados.
  - Melhoria da performance por utilizar uma consulta previamente otimizada.

# View

# View

- Sintaxe:

# View

- Sintaxe:

```
CREATE VIEW view_name AS  
SELECT column1, column2, ...  
FROM table_name  
WHERE condition;
```

# View

- Sintaxe:

```
CREATE VIEW view_name AS  
SELECT column1, column2, ...  
FROM table_name  
WHERE condition;
```

- Exemplo:

# View

- Sintaxe:

```
CREATE VIEW view_name AS
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```

- Exemplo:

```
CREATE VIEW [Brazil Customers] AS
SELECT CustomerName, ContactName
FROM Customers
WHERE Country = 'Brazil';
```



# Exercícios

---



# Perguntas

- Qual é a diferença entre o comando *drop* e *delete*?
- O que ocorre quando não usamos a cláusula *where* na operação de apagar dados (*delete*)?
- Qual é a diferença entre o comando *alter table* e *update*?
- É correto afirmar que o uso de *views* aumenta a redundância dos dados no banco de dados, uma vez que *views* podem ser consideradas tabelas?