

# Banco de Dados

Professor Maurício Buess

[mbuess@up.edu.br](mailto:mbuess@up.edu.br)

## **Objetivo:**

- Solução atividades;
- Compreender e conhecer os principais tipos de dados;
- MySql (instalação)
- DER no MySql

## Atividade 01

Uma livraria deseja organizar suas informações sobre os livros que possui em estoque. Cada livro é identificado por um código único, tem um título, uma data de publicação e pertence a um ou mais autores. Além disso, cada autor tem um código único e um nome. A livraria quer poder relacionar cada livro aos seus autores.

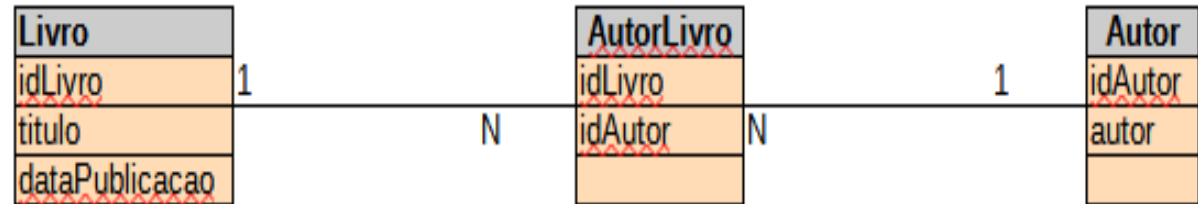
Dicas:

- 1) Identifique as entidades: Comece identificando os principais elementos que serão representados no diagrama, como Livro e Autor.
- 2) Defina os atributos: Para cada entidade, defina seus atributos, como código, título, data de publicação para o Livro e código, nome para o Autor.
- 3) Determine os relacionamentos: Identifique o relacionamento entre as entidades Livro e Autor. Qual a cardinalidade desse relacionamento?
- 4) Crie o DER: Use as entidades, atributos e relacionamentos identificados para construir o diagrama.

## Solução

### Exercício 1 - Livraria

Entidades			
Nome	Atributo		
Livro	idLivro	C.P.	
	titulo		
	dataPublicacao		
Autor	idAutor	C.P.	
	autor		
AutorLivro	idLivro	C.E.	C.P.
	idAutor	C.E.	C.P.



## Atividade 02

Uma empresa de transporte precisa organizar as informações de seus motoristas e os veículos que eles dirigem. Cada motorista tem um código único, um nome, uma data de contratação e pode dirigir vários veículos. Cada veículo é identificado por uma placa única e possui um modelo e uma capacidade de carga. A empresa também quer armazenar a data de início da utilização de cada veículo por cada motorista.

### Dicas

- 1) Identifique as entidades: Comece identificando as entidades Motorista e Veículo.
- 2) Defina os atributos: Para Motorista, você deve incluir atributos como `codigo_motorista`, `nome` e `data_contratacao`. Para Veículo, considere atributos como `placa`, `modelo`, e `capacidade_carga`.
- 3) Determine o relacionamento: O relacionamento entre Motorista e Veículo deve incluir um atributo adicional para a data de início da utilização.
- 4) Trate o relacionamento N-N: Como cada motorista pode dirigir vários veículos e cada veículo pode ser dirigido por vários motoristas ao longo do tempo, você precisará de uma tabela associativa.

## Solução

### Exercício 1 - Empresa de Transporte

Entidades			
Nome	Atributo		
Motorista	idMotorista	C.P.	
	motorista		
	dataContratacao		
Veiculo	placaVeiculo	C.P.	
	modelo		
	cargaCapacidade		
VeiculoMotorista	idMotorista	C.E.	C.P.
	placaVeiculo	C.E.	C.P.
	dataInicio		C.P.



## Atividade 03

Uma universidade deseja criar um sistema de gerenciamento para seus cursos, professores e alunos. Um curso é identificado por um código único, tem um nome e pode ser ministrado em vários semestres diferentes. Cada professor é identificado por um código único, tem um nome, um título (por exemplo, Doutor, Mestre) e pode ministrar várias disciplinas. Uma disciplina é identificada por um código único, pertence a um curso específico, tem um nome, uma carga horária e é ministrada por um ou mais professores. Além disso, cada aluno é identificado por um código único, tem um nome e pode estar matriculado em várias disciplinas. Considere que um professor pode ministrar várias disciplinas, e que um aluno pode se matricular em várias disciplinas ao longo de diferentes semestres.

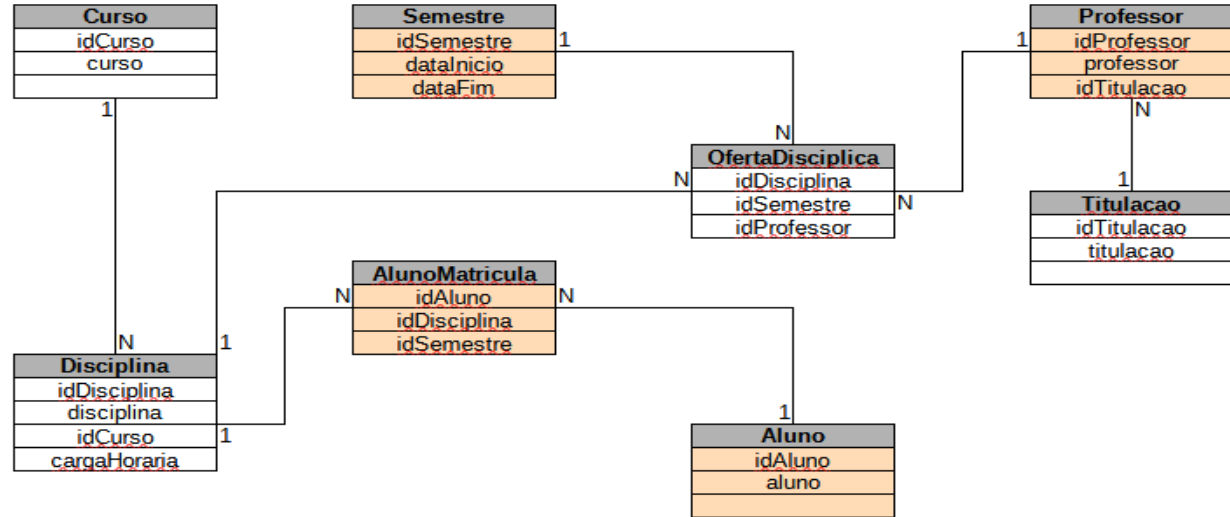
Dicas:

- 1) Comece identificando as entidades Curso, Professor, Disciplina, e Aluno.
- 2) Determine os relacionamentos entre Curso e Disciplina, Professor e Disciplina, e Aluno e Disciplina.
- 3) Trate relacionamentos complexos: Como existem relacionamentos muitos-para-muitos, use tabelas associativas para representá-los. Por exemplo, Ministra para relacionar Professor e Disciplina, e Matricula para relacionar Aluno e Disciplina.
- 4) Considere o tempo: Para disciplinas que ocorrem em diferentes semestres, você pode incluir um atributo para semestre na tabela associativa Matricula.

## Solução

### Exercício 3 - Universidade

Nome	Atributo		
Curso	<u>idCurso</u>	C.P.	
	curso		
Professor	<u>idProfessor</u>	C.P.	
	professor		
	<u>idTitulacao</u>		C.E.
Titulacao	<u>idTitulacao</u>	C.P.	
	titulacao		
Aluno	<u>idAluno</u>	C.P.	
	aluno		
Disciplina	<u>idDisciplina</u>	C.P.	
	disciplina		
	<u>idCurso</u>		C.E.
	cargaHoraria		
Semestre	<u>idSemestre</u>	C.P.	
	dataInicio		
	dataFim		
OfertaDisciplina	<u>idDisciplina</u>	C.P.	C.E.
	<u>idSemestre</u>	C.P.	C.E.
	<u>idProfessor</u>	C.P.	C.E.
AlunoMatricula	<u>idAluno</u>	C.P.	C.E.
	<u>idDisciplina</u>	C.P.	C.E.
	<u>idSemestre</u>	C.P.	C.E.





## MySql

- Sistema de gerenciamento de banco de dados relacional (SGBD) de código aberto amplamente utilizado em todo o mundo;
- Ele oferece uma plataforma robusta e confiável para armazenar, gerenciar e recuperar dados;
- Amplamente utilizado em aplicativos da web, aplicativos empresariais, servidores de banco de dados e muito mais devido à sua confiabilidade e desempenho.
- O MySQL é conhecido por sua escalabilidade, segurança e suporte à linguagem SQL (Structured Query Language), que é a linguagem padrão para consultar bancos de dados relacionais.

## Instalação do MySQL

- Duas opções de instalação do MySql:
  - MySQL Community Edition: É uma versão gratuita e de código aberto do MySQL, adequada para iniciantes e projetos menores.
  - MySQL Enterprise Edition: Esta é uma versão paga que oferece recursos adicionais, suporte premium e é mais adequada para empresas.

## Instalação do MySQL

- Instalação do MySQL Community Edition:
  - Baixe o instalador do MySQL no site oficial (<https://www.mysql.com/>).
  - Execute o instalador e siga as instruções para configurar o MySQL.
  - Durante o processo de instalação, você será solicitado a definir uma senha para o usuário "root" do MySQL.
  - Certifique-se de lembrar dessa senha, pois você a usará para acessar o servidor.

## Iniciar o MySQL Server:

- Após a instalação, o próximo passo é iniciar o servidor MySQL.
  - No Windows: Use o "MySQL Notifier" ou execute ``mysqlstart.exe`` no prompt de comando.
  - No macOS: Execute ``sudo /usr/local/mysql/support-files/mysql.server start``.
  - No Linux: Execute ``sudo service mysql start`` ou ``sudo systemctl start mysql``.

## Acesso ao MySQL:

- Depois de iniciar o servidor, você pode acessá-lo usando o cliente MySQL.
- O cliente MySQL padrão é chamado de "mysql."
- Para acessar o servidor, abra o terminal ou prompt de comando e execute o seguinte comando:
- `mysql -u root -p`
- Isso solicitará que você insira a senha que você definiu durante a instalação.

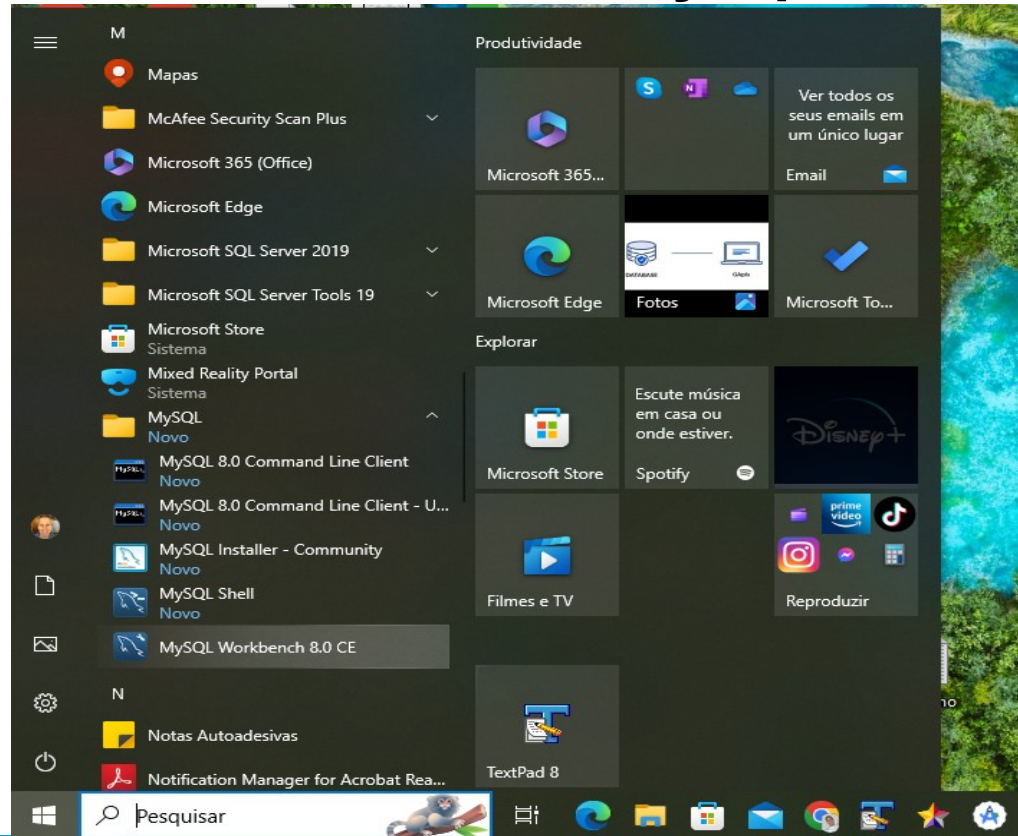
## Criação de um Banco de Dados:

- Depois de iniciar o servidor, você pode acessá-lo usando o cliente MySQL.
- O cliente MySQL padrão é chamado de "mysql."
- Para acessar o servidor, abra o terminal ou prompt de comando e execute o seguinte comando:
- `mysql -u root -p`
- Isso solicitará que você insira a senha que você definiu durante a instalação.

## Criação de um Banco de Dados:

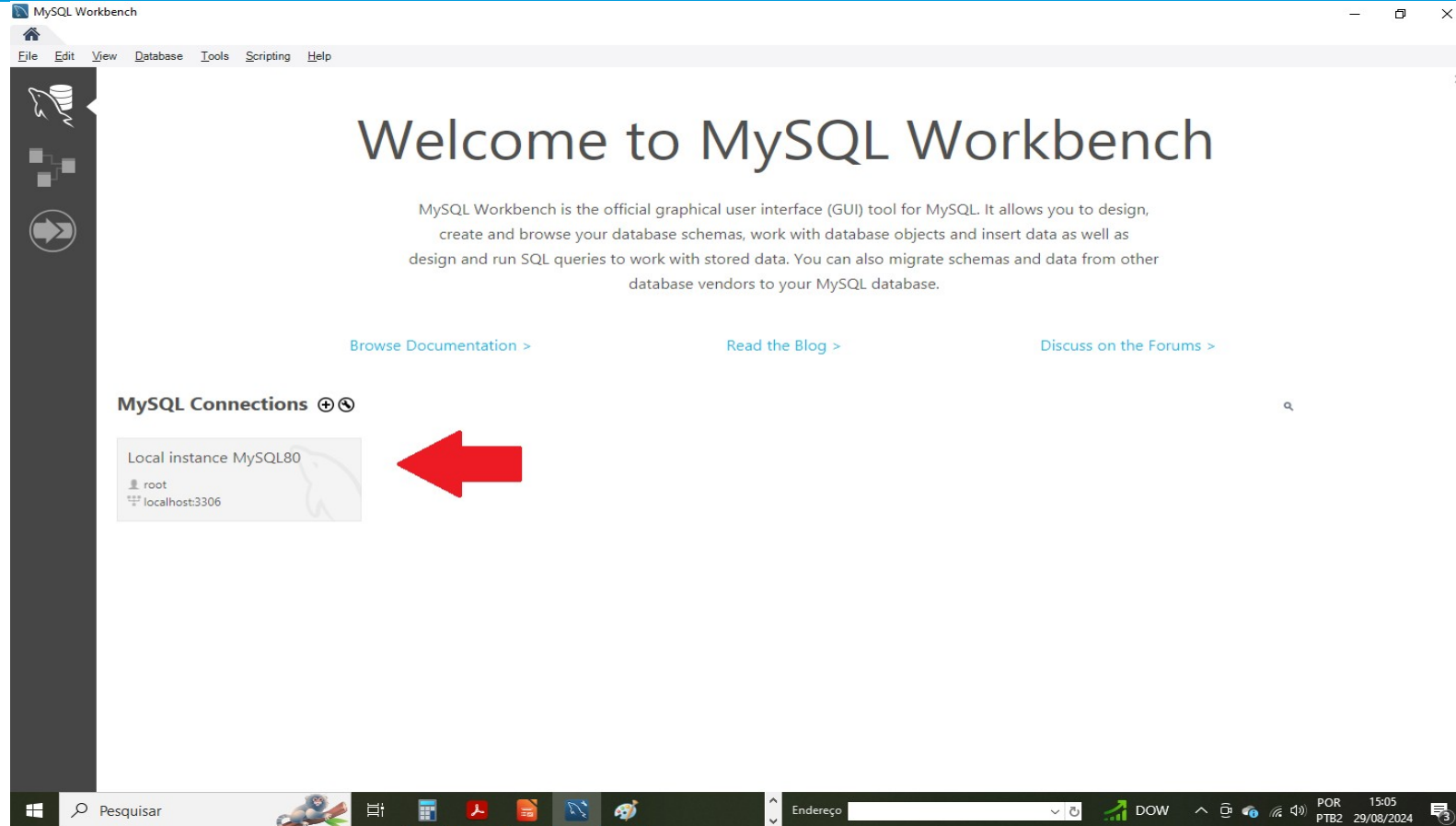
- Há vários tutoriais de instalação do MySql disponíveis na Internet;
- Sugiro acompanhar e seguir o seguinte tutorial:
- <https://www.youtube.com/watch?v=oi3UHWXLxLs>

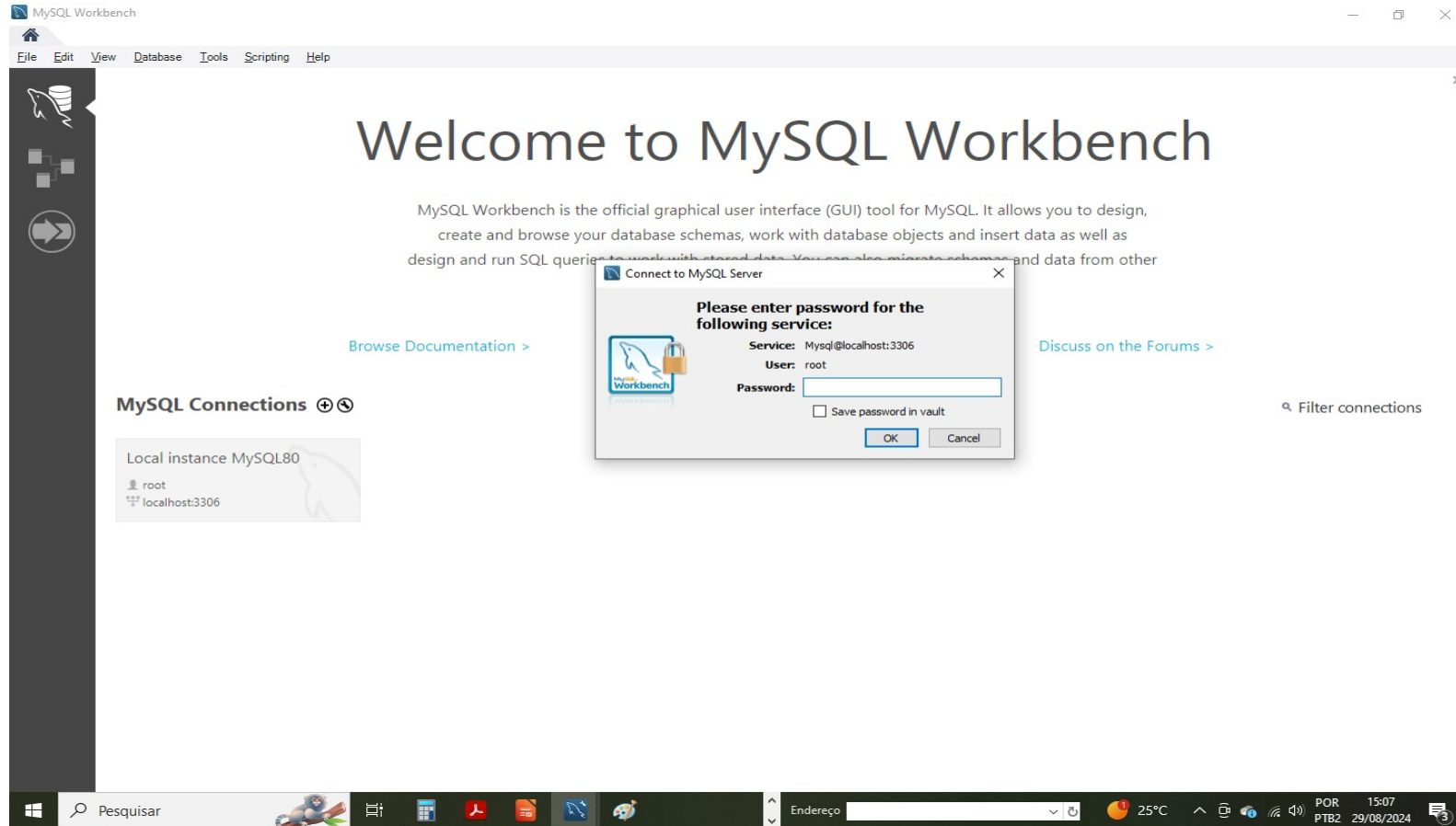
## Trabalhando com o MySQL

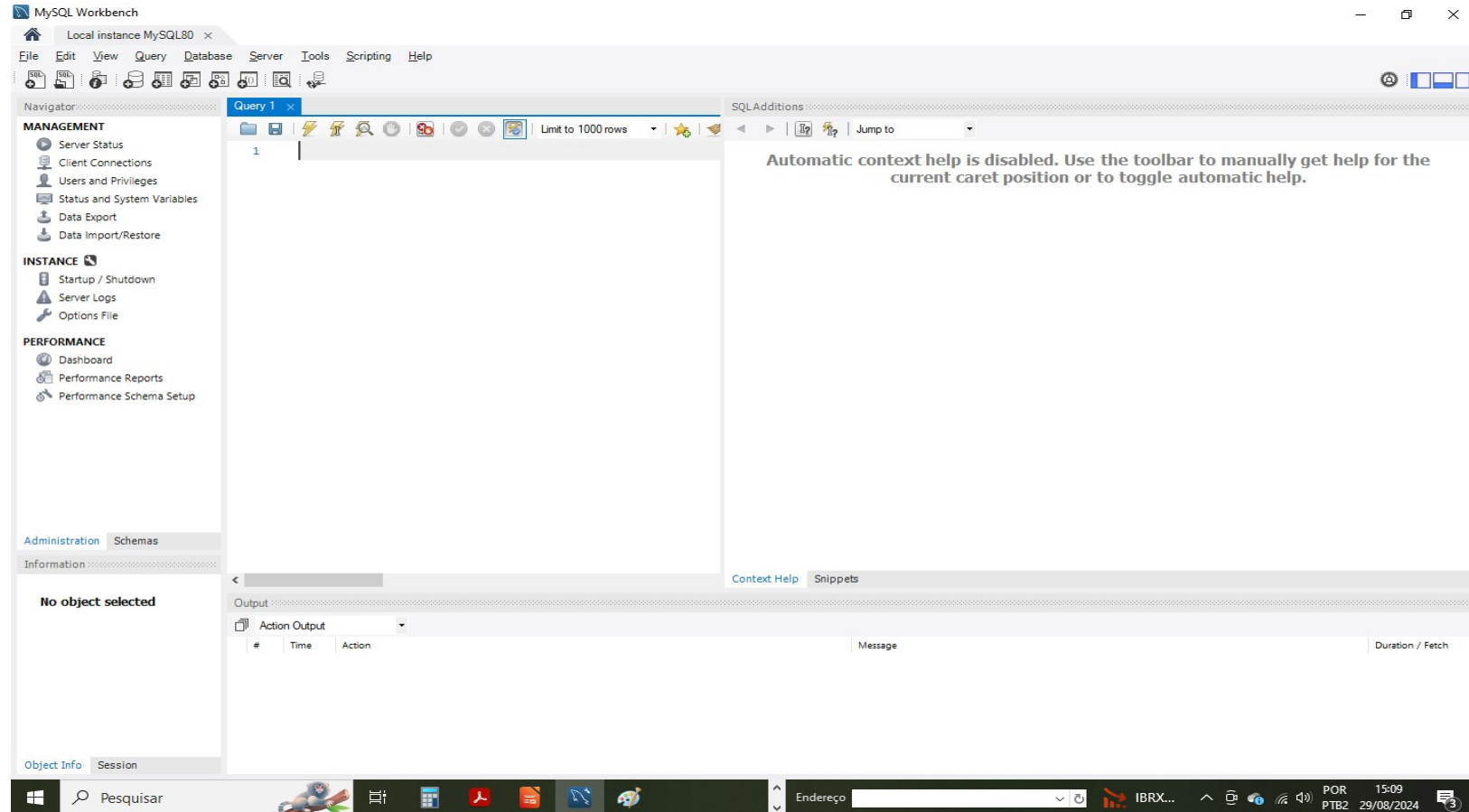


Acessar o MySQL Workbench









## Tipos de Dados no MySql

- Há variedade de tipos de dados para que os desenvolvedores possam armazenar dados de diferentes tipos e formatos;
- Um dos aspectos mais importantes da gestão de dados é a escolha do tipo de dado adequado para cada valor armazenado na base de dados;
- No MySql há tipos de dados de caractere, de data e hora, binários, geométricos, de texto completo e Unicode;
- Entendendo os diferentes tipos de dados disponíveis no MySQL, você poderá tomar decisões informadas sobre como armazenar e gerenciar dados em sua aplicação.

## Tipos de dados numéricos Inteiros (INT)

- **TINYINT** - Intervalo Sem Sinal: 0 a 255 / Intervalo com Sinal: -128 a 127
- **SMALLINT** - Intervalo Sem Sinal: 0 a 65,535 Intervalo com Sinal: -32,768 a 32,767
- **MEDIUMINT**
  - Intervalo Sem Sinal: 0 a 16,777,215
  - Intervalo com Sinal: -8,388,608 a 8,388,607
- **INT**
  - Intervalo Sem Sinal: 0 a 4,294,967,295
  - Intervalo com Sinal: -2,147,483,648 a 2,147,483,647
- **BIGINT**
  - Intervalo Sem Sinal: 0 a 18,446,744,073,709,551,615
  - Intervalo com Sinal: -9,223,372,036,854,775,808 a 9,223,372,036,854,775,807

## Detalhes Adicionais

- Sem Sinal (UNSIGNED): Se um campo inteiro for declarado como UNSIGNED, ele não pode armazenar valores negativos, e o intervalo de valores é ajustado para incluir apenas números não negativos.
- Valores Default e Limites: Se você não especificar a palavra-chave UNSIGNED, o campo INT pode armazenar valores negativos e positivos dentro dos intervalos mencionados.

## Tipos de Dados de Caractere

- **CHAR(n)**
  - Armazena uma string de comprimento fixo.
  - Se a string for menor que o comprimento especificado, o MySQL a preenche com espaços em branco à direita até atingir o comprimento total.
  - Comprimento: n pode variar de 0 a 255 caracteres.
  - Uso Típico: Utilizado para armazenar dados de comprimento fixo, como códigos postais ou números de telefone.
  - Exemplo: CHAR(10)

## Tipos de Dados de Caractere

- **VARCHAR(n)**
  - Armazena uma string de comprimento variável. Somente o espaço necessário para armazenar a string é utilizado, mais um byte ou dois para armazenar o comprimento da string.
  - Comprimento: n pode variar de 0 a 65,535 caracteres.
  - Uso Típico: Utilizado para armazenar dados de comprimento variável, como nomes, descrições, ou endereços de e-mail.
  - Exemplo: VARCHAR(255)



## Tipos de Dados de Caractere

- **TEXT**

- Armazena grandes quantidades de texto.
- Não é possível especificar um comprimento fixo.
- Subtipos:
  - TINYTEXT: Até 255 caracteres.
  - TEXT: Até 65,535 caracteres.
  - MEDIUMTEXT: Até 16,777,215 caracteres.
  - LONGTEXT: Até 4,294,967,295 caracteres.
- Uso Típico: Utilizado para armazenar textos longos, como descrições ou conteúdos de artigos.
- Exemplo: TEXT

## Tipos de Dados de Caractere

- **BINARY(n)**
  - Descrição: Armazena uma string de comprimento fixo de dados binários. Assim como CHAR, é preenchido com zeros à direita se a string for menor que o comprimento especificado.
  - Comprimento: n pode variar de 0 a 255 bytes.
  - Uso Típico: Utilizado para armazenar dados binários, como imagens ou arquivos de configuração.
  - Exemplo: BINARY(10)

## Tipos de Dados de Caractere

- **VARBINARY(n)**
  - Armazena uma string de comprimento variável de dados binários. Somente o espaço necessário para armazenar a string é utilizado, mais um byte ou dois para armazenar o comprimento da string.
  - Comprimento: n pode variar de 0 a 65,535 bytes.
  - Uso Típico: Utilizado para armazenar dados binários de comprimento variável.
  - Exemplo: VARBINARY(255)

## Tipos de Dados de Caractere

- **ENUM**

- Armazena uma lista de valores predefinidos. Cada valor é armazenado como um índice na lista de valores fornecidos.
- Uso Típico: Utilizado para armazenar um conjunto restrito de valores, como categorias ou estados de status.
- Exemplo: `ENUM('Pequeno', 'Médio', 'Grande')`

## Tipos de Dados de Caractere

- **SET**

- Armazena zero ou mais valores de uma lista predefinida de valores. Os valores são armazenados como um conjunto de bits.
- Uso Típico: Utilizado para armazenar um conjunto de opções, como permissões ou tags.
- Exemplo: SET('Leitura', 'Escrita', 'Execução')

## Tipos de Dados de Data

- **DATE**

- Armazena uma data no formato YYYY-MM-DD.
- Intervalo: De '1000-01-01' a '9999-12-31'.
- Uso Típico: Utilizado para armazenar apenas a data, sem informação de hora.
- Exemplo: DATE('2024-08-25')

- **DATETIME**

- Descrição: Armazena uma data e uma hora no formato YYYY-MM-DD HH:MM:SS.
- Intervalo: De '1000-01-01 00:00:00' a '9999-12-31 23:59:59'.
- Uso Típico: Utilizado para armazenar uma combinação de data e hora.
- Exemplo: DATETIME('2024-08-25 14:30:00')

## Tipos de Dados de Data

- **TIMESTAMP**

- Armazena uma data e uma hora no formato YYYY-MM-DD HH:MM:SS com informações sobre o fuso horário.
- Os valores são armazenados em UTC e convertidos para o fuso horário do cliente ao serem recuperados.
- Intervalo: De '1970-01-01 00:00:01' UTC a '2038-01-19 03:14:07' UTC.
- Uso Típico: Utilizado para rastrear quando uma linha foi criada ou modificada. Ideal para registros que precisam de marcação temporal precisa.
- Exemplo: `TIMESTAMP('2024-08-25 14:30:00')`

## Tipos de Dados de Data

- **TIME**

- Armazena uma hora no formato HH:MM:SS.
- Intervalo: De '-838:59:59' a '838:59:59'.
- Uso Típico: Utilizado para armazenar apenas a hora, sem a data.
- Exemplo: TIME('14:30:00')

- **YEAR**

- Armazena um ano no formato YYYY.
- Intervalo: De 1901 a 2155.
- Uso Típico: Utilizado para armazenar apenas o ano.
- Exemplo: YEAR(2024)



## Tipos de Dados Lógicos

- **BOOLEAN**

- O tipo BOOLEAN é um alias para TINYINT(1), que é um inteiro pequeno. Ele armazena valores binários (0 e 1), onde 0 representa FALSE e 1 representa TRUE.
- Utilizado para armazenar valores lógicos (booleanos), como flags e condições.

## Tipos de Dados de Imagem e Dados Binários

- **BLOB (Binary Large Object)**

- O tipo BLOB é usado para armazenar grandes quantidades de dados binários. Ele pode conter qualquer tipo de dados binários, como imagens, arquivos de áudio, vídeos, etc.

- **Subtipos de BLOB:**

- **TINYBLOB:** Armazena até 255 bytes.
  - **BLOB:** Armazena até 65.535 bytes (64 KB).
  - **MEDIUMBLOB:** Armazena até 16.777.215 bytes (16 MB).
  - **LONGBLOB:** Armazena até 4.294.967.295 bytes (4 GB).
- Usado para armazenar grandes quantidades de dados binários, como imagens e arquivos multimídia.

## Tipos de Dados de Imagem e Dados Binários

- **VARBINARY**

- O tipo VARBINARY é usado para armazenar dados binários de tamanho variável. Ao contrário do BLOB, que é para grandes quantidades de dados, VARBINARY é para dados binários que têm tamanho variável mas são relativamente menores.
- Usado para armazenar dados binários cujo tamanho pode variar, mas que não são grandes o suficiente para justificar o uso de BLOB.

- **BINARY**

- O tipo BINARY é usado para armazenar dados binários de comprimento fixo. É semelhante ao CHAR, mas para dados binários.
- Uso Típico: Usado para armazenar dados binários de tamanho fixo, como hashes ou códigos binários.

## Atividade 01 – DER com o Workbench

Uma livraria deseja organizar suas informações sobre os livros que possui em estoque. Cada livro é identificado por um código único, tem um título, uma data de publicação e pertence a um ou mais autores. Além disso, cada autor tem um código único e um nome. A livraria quer poder relacionar cada livro aos seus autores.

Dicas:

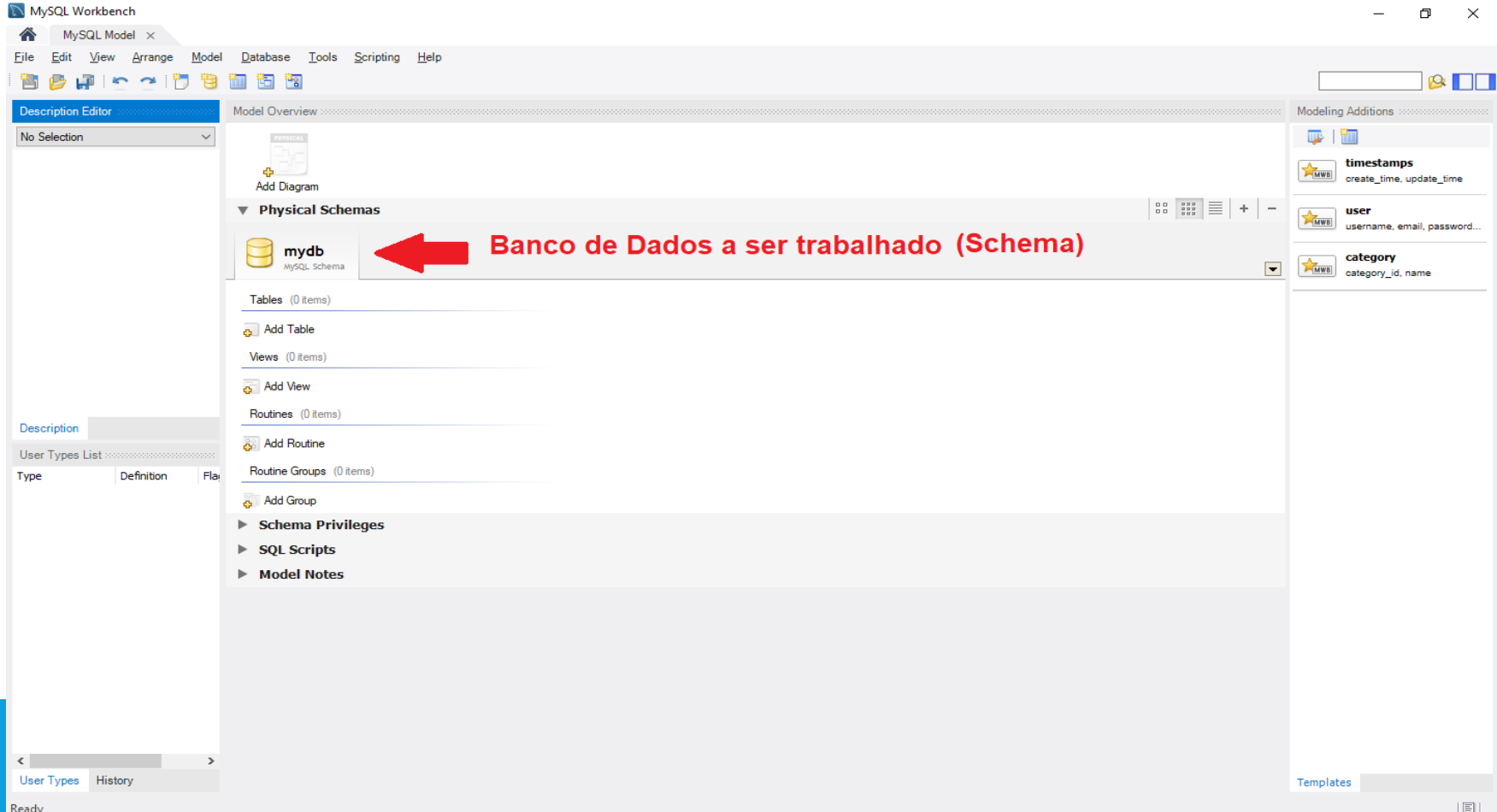
- 1) Identifique as entidades: Comece identificando os principais elementos que serão representados no diagrama, como Livro e Autor.
- 2) Defina os atributos: Para cada entidade, defina seus atributos, como código, título, data de publicação para o Livro e código, nome para o Autor.
- 3) Determine os relacionamentos: Identifique o relacionamento entre as entidades Livro e Autor. Qual a cardinalidade desse relacionamento?
- 4) Crie o DER: Use as entidades, atributos e relacionamentos identificados para construir o diagrama.

## Atividade 01 – DER com o Workbench

- Acesse o MySql Workbench;
- Escolha a opção Models (opção central do menu vertical ao lado esquerdo da tela - 1);
- Selecione a opção (+) - seta 2.



## Atividade 01 – DER com o Workbench



The screenshot shows the MySQL Workbench interface. The 'Physical Schemas' section is expanded, and the 'mydb' schema is selected. A red arrow points to the 'mydb' schema with the text 'Banco de Dados a ser trabalhado (Schema)'. The 'Model Overview' pane shows the 'mydb' schema and its associated objects: Tables (0 items), Views (0 items), Routines (0 items), and Routine Groups (0 items). The 'Schema Privileges', 'SQL Scripts', and 'Model Notes' sections are also visible. The 'Modeling Additions' pane on the right shows the 'timestamps' and 'user' objects.

MySQL Workbench

MySQL Model x

File Edit View Arrange Model Database Tools Scripting Help

Description Editor

No Selection

Model Overview

Physical Schemas

mydb  
MySQL Schema

Tables (0 items)

Add Table

Views (0 items)

Add View

Routines (0 items)

Add Routine

Routine Groups (0 items)

Add Group

Schema Privileges

SQL Scripts

Model Notes

Modeling Additions

timestamps  
create\_time, update\_time

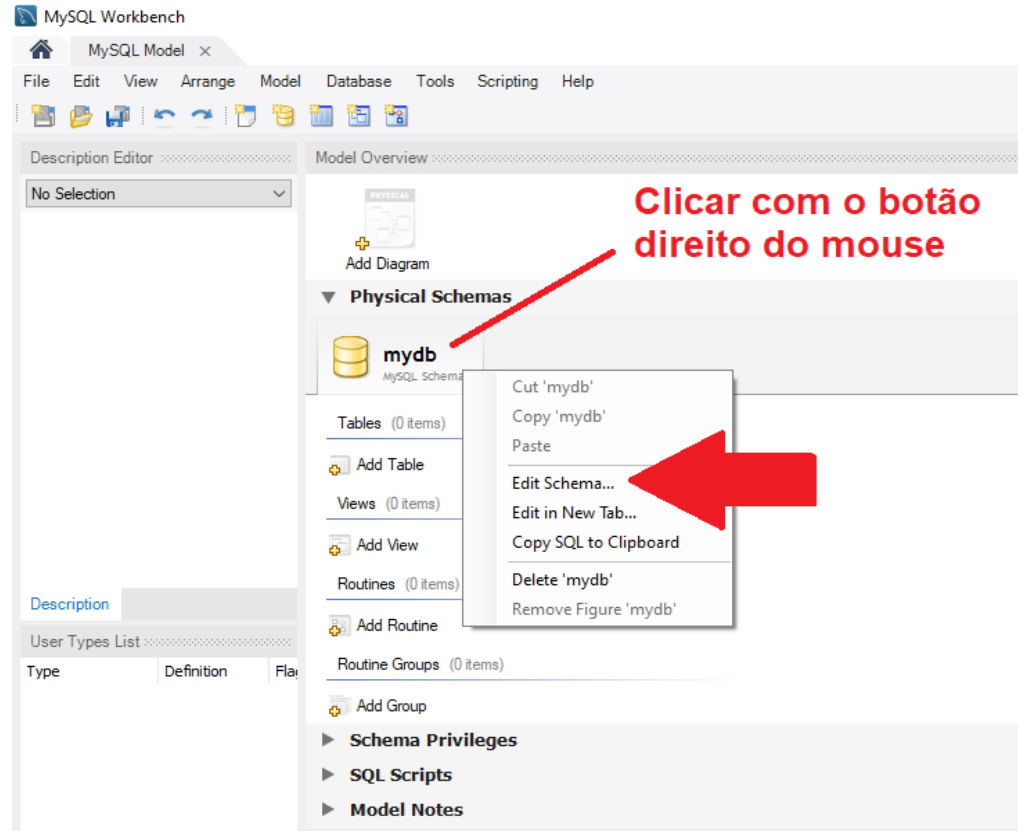
user  
username, email, password...

category  
category\_id, name

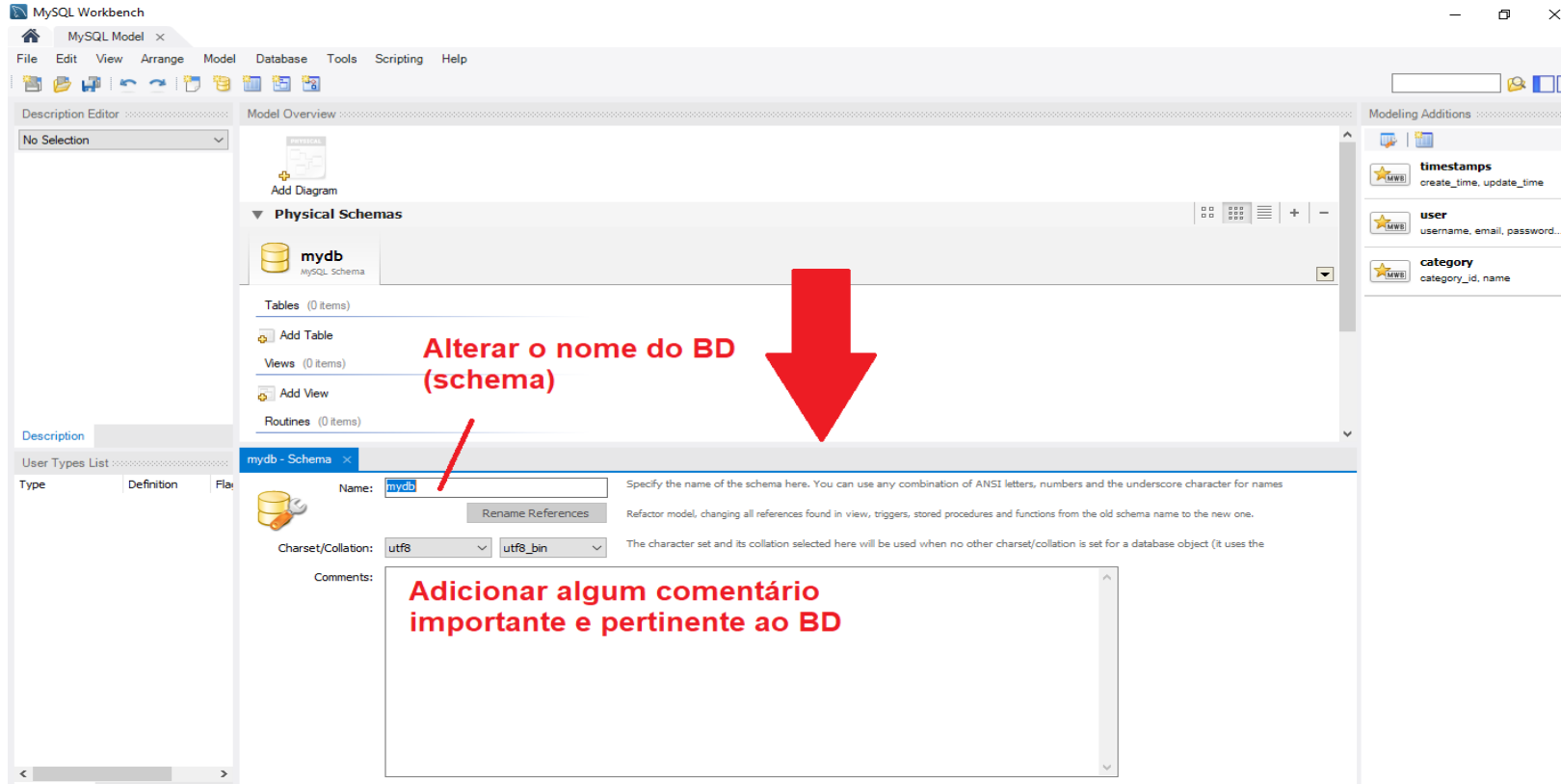
30/08/24

38

## Atividade 01 – DER com o Workbench



## Atividade 01 – DER com o Workbench



MySQL Workbench

File Edit View Arrange Model Database Tools Scripting Help

Description Editor: No Selection

Model Overview

Physical Schemas

mydb  
MySQL Schema

Tables (0 items)

Add Table

Views (0 items)

Add View

Routines (0 items)

mydb - Schema

Name: mydb

Charset/Collation: utf8 utf8\_bin

Comments: Adicionar algum comentário importante e pertinente ao BD


Modeling Additions

- timestamps  
create\_time, update\_time
- user  
username, email, password...
- category  
category\_id, name



## Atividade 01 – DER com o Workbench

Aula04\_Testes - Schema ×

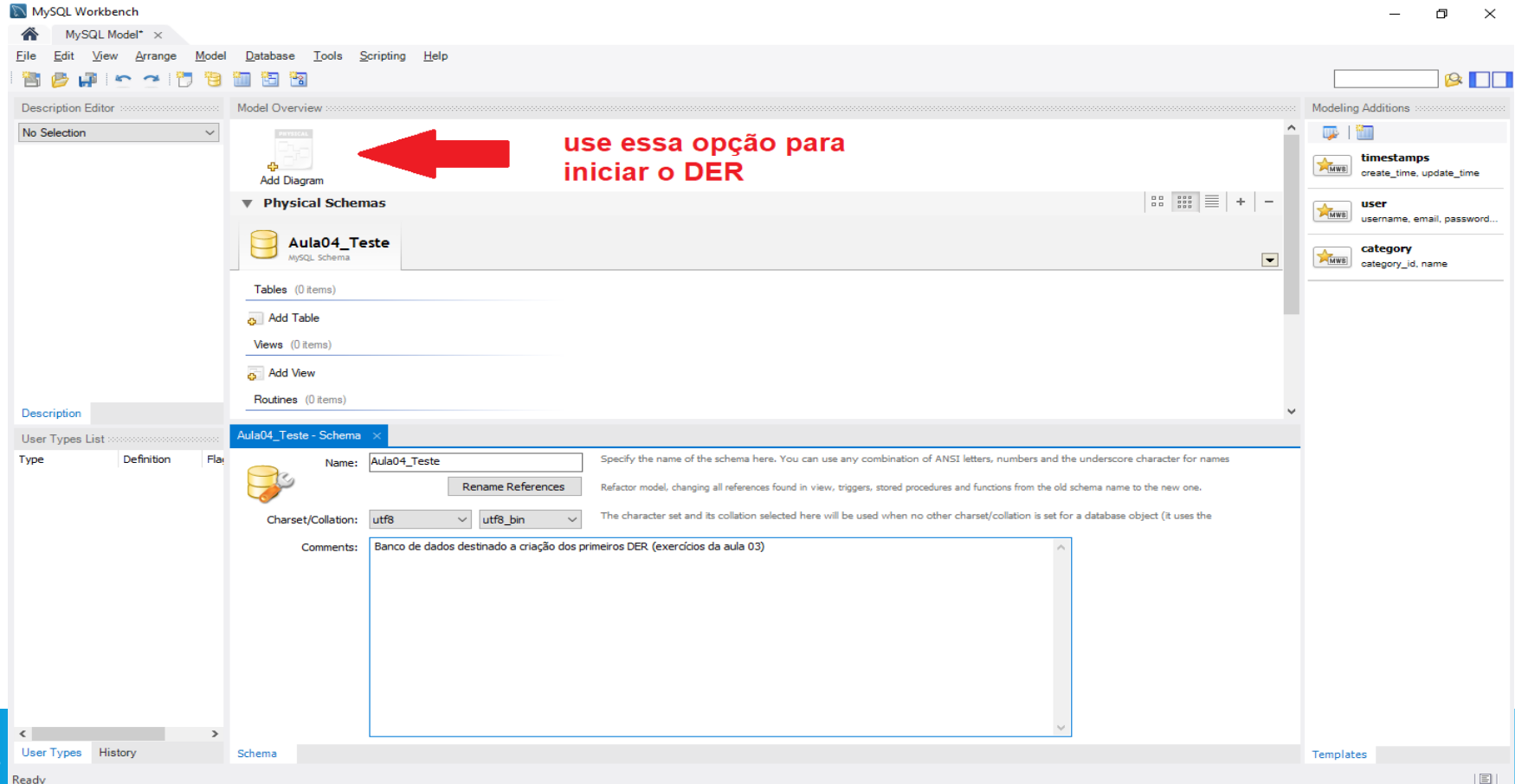
 Name:  Specify the name of the schema here. You can use any combination of ANSI letters, numbers and the underscore character for names

Refactor model, changing all references found in view, triggers, stored procedures and functions from the old schema name to the new one.

Charset/Collation:   The character set and its collation selected here will be used when no other charset/collation is set for a database object (it uses the

Comments:

## Atividade 01 – DER com o Workbench



The screenshot shows the MySQL Workbench interface. In the 'Model Overview' tab, the 'Add Diagram' button is highlighted with a red arrow. To the right of the arrow, the text 'use essa opção para iniciar o DER' is written in red. Below the 'Add Diagram' button, the 'Physical Schemas' section is visible, showing a schema named 'Aula04\_Testes'. The 'Description' tab is also visible, showing the 'Aula04\_Testes - Schema' details, including the name, charset/collation, and comments.

MySQL Workbench

MySQL Model\* x

File Edit View Arrange Model Database Tools Scripting Help

Description Editor: No Selection

Model Overview

Add Diagram

Physical Schemas

Aula04\_Testes  
MySQL Schema

Tables (0 items)

Add Table

Views (0 items)

Add View

Routines (0 items)

Description

User Types List

Type Definition Flag

Aula04\_Testes - Schema

Name: Aula04\_Testes

Rename References

Charset/Collation: utf8 utf8\_bin

Comments: Banco de dados destinado a criação dos primeiros DER (exercícios da aula 03)

Modeling Additions

timestamps  
create\_time, update\_time

user  
username, email, password...

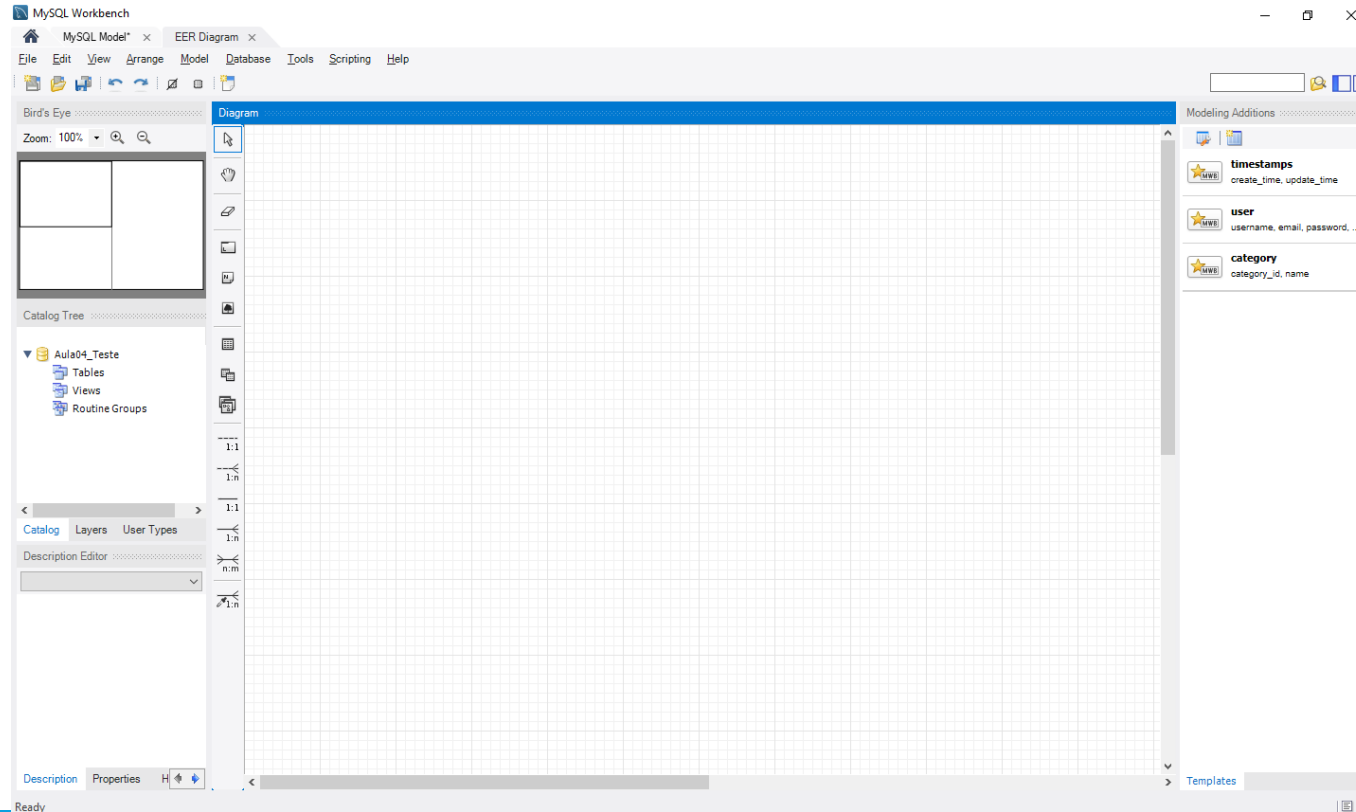
category  
category\_id, name

Ready

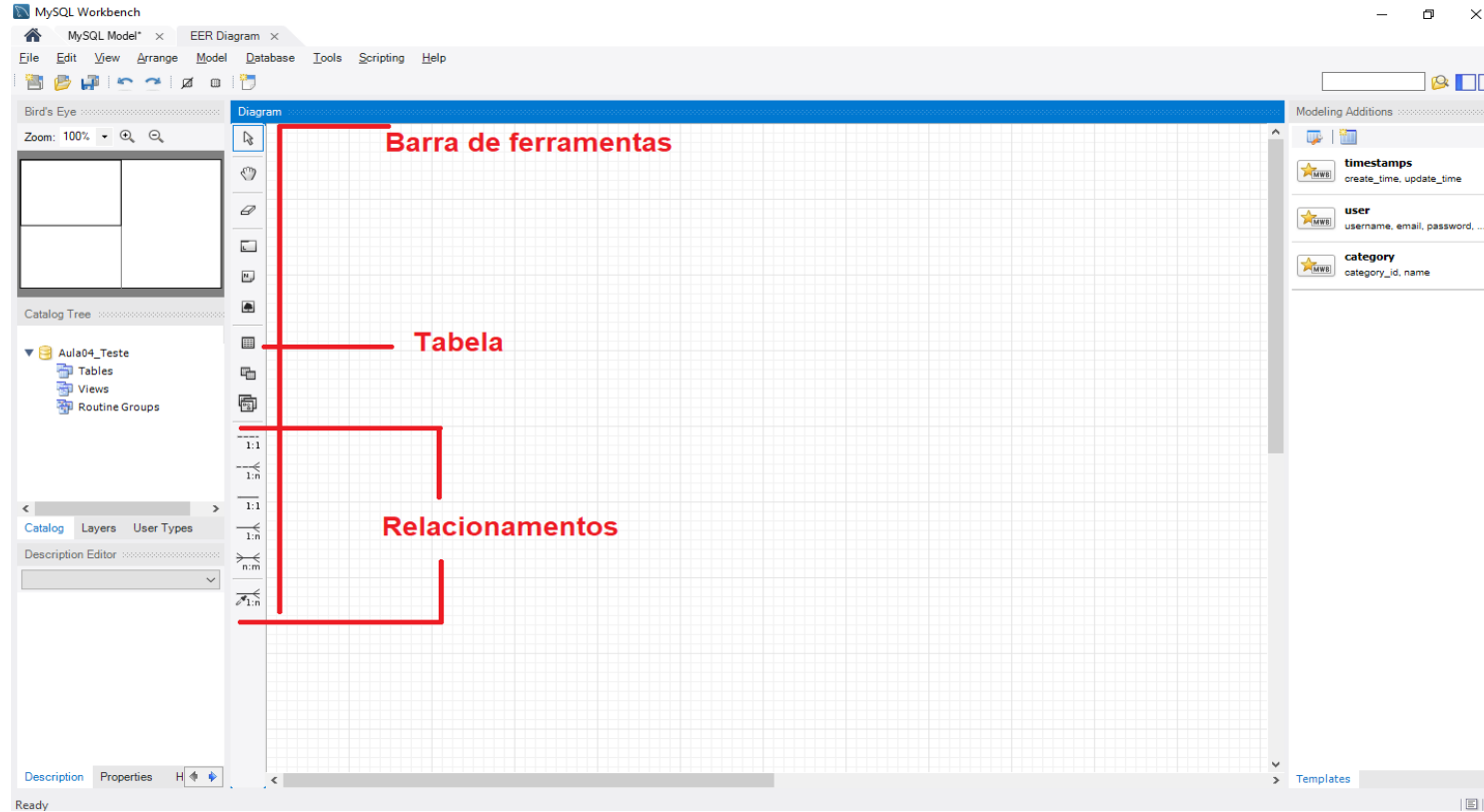
30/08/24

42

## Atividade 01 – DER com o Workbench



## Atividade 01 – DER com o Workbench



## Atividade 01 – DER com o Workbench

Diagram

**Duplo clique sobre Table para alterar características**

**Nome da tabela**

**Detalhamento do(s) atributo(s)**

Table Name: Livro Schema: Aula04\_Testes

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idLivro	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Column Name: idLivro

Charset/Collation: Default Charset Default Collation

Comments: Chave primária - garante a unicidade do registro da tabela Livro

Data Type: INT

Default:

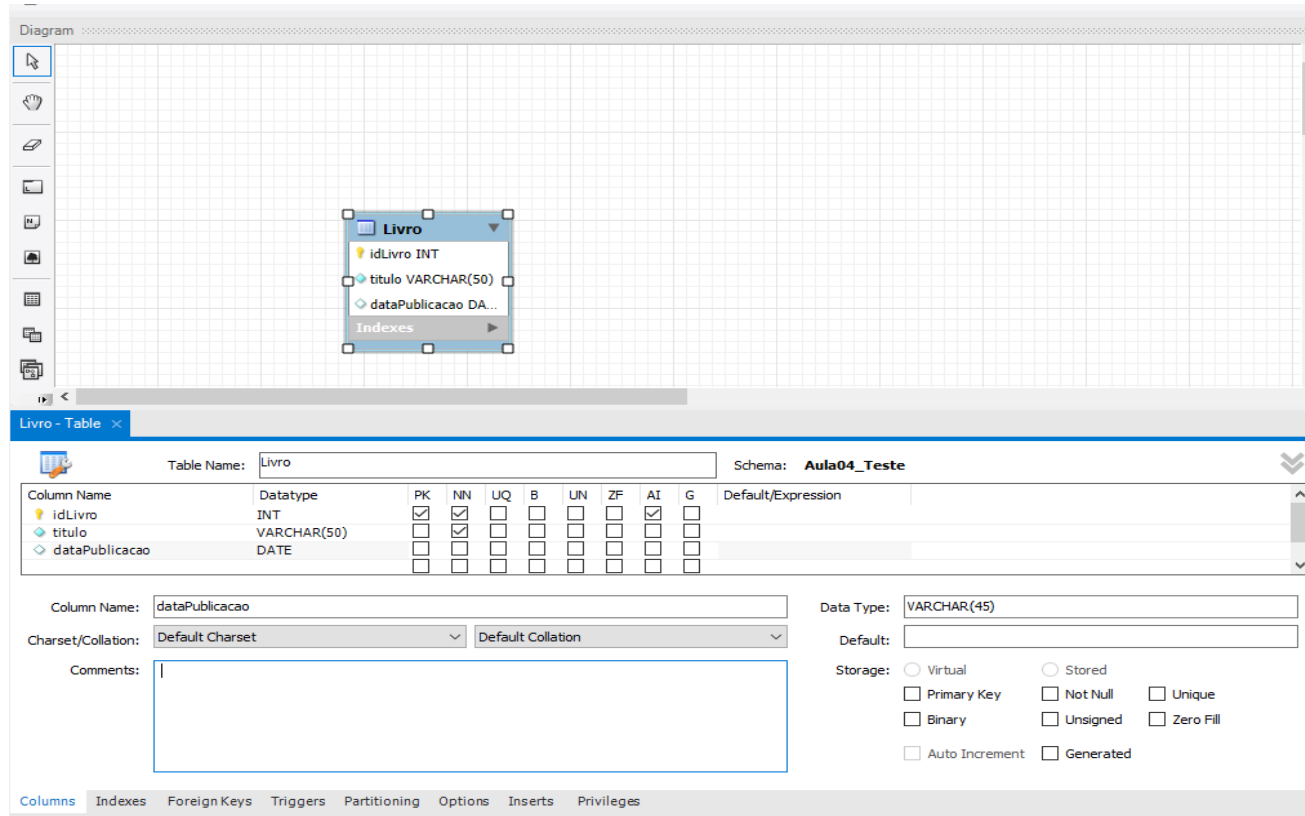
Storage: ☐ Virtual ☐ Stored

☒ Primary Key ☒ Not Null ☐ Unique

☐ Binary ☐ Unsigned ☐ Zero Fill

☒ Auto Increment ☐ Generated

## Atividade 01 – DER com o Workbench



Diagram

**Livro**

- idLivro INT
- titulo VARCHAR(50)
- dataPublicacao DA...

Indexes

**Livro - Table**

Table Name:  Schema: **Aula04\_Test**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idLivro	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
titulo	VARCHAR(50)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
dataPublicacao	DATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Column Name:

Charset/Collation:

Comments:

Data Type:

Default:

Storage: ☐ Virtual ☐ Stored

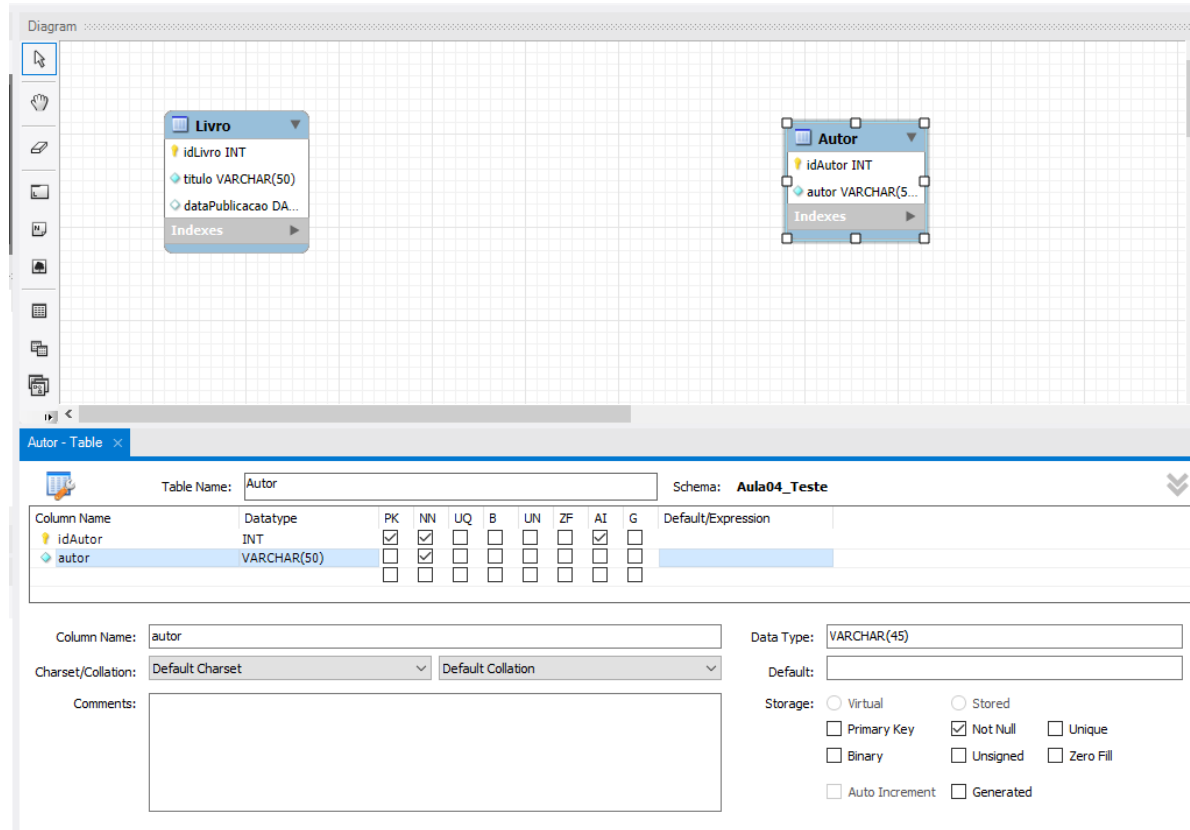
☐ Primary Key ☐ Not Null ☐ Unique

☐ Binary ☐ Unsigned ☐ Zero Fill

☐ Auto Increment ☐ Generated

Columns Indexes Foreign Keys Triggers Partitioning Options Inserts Privileges

## Atividade 01 – DER com o Workbench



The screenshot shows the MySQL Workbench interface. At the top, a diagram view displays two tables: 'Livro' and 'Autor'. The 'Livro' table has columns 'idLivro INT', 'titulo VARCHAR(50)', and 'dataPublicacao DA...'. The 'Autor' table has columns 'idAutor INT' and 'autor VARCHAR(50)'. Below the diagram, the 'Table' tab for the 'Autor' table is selected, showing the table's structure within the 'Aula04\_Testes' schema.

Table Name:  Schema: **Aula04\_Testes**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idAutor	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
autor	VARCHAR(50)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Column Name:  Data Type:

Charset/Collation:

Comments:

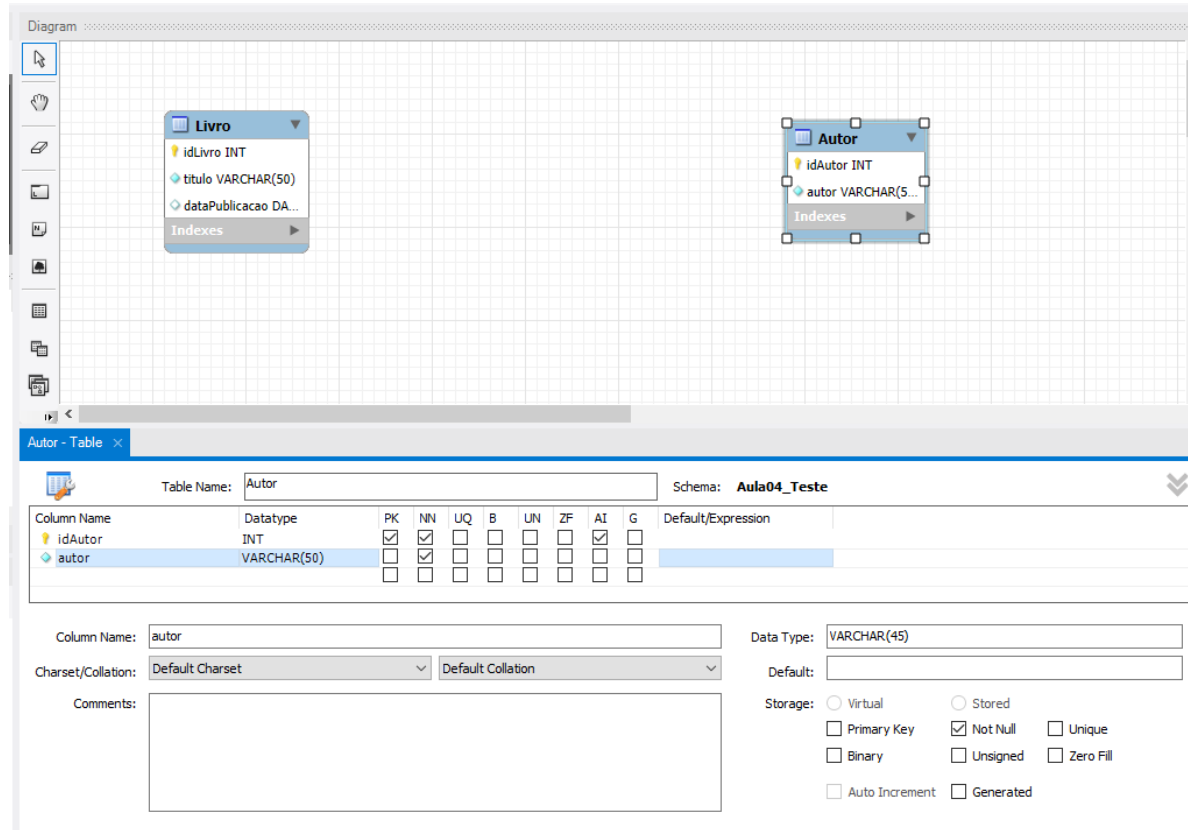
Storage: ☐ Virtual ☐ Stored

☐ Primary Key ☒ Not Null ☐ Unique

☐ Binary ☐ Unsigned ☐ Zero Fill

☐ Auto Increment ☐ Generated

## Atividade 01 – DER com o Workbench



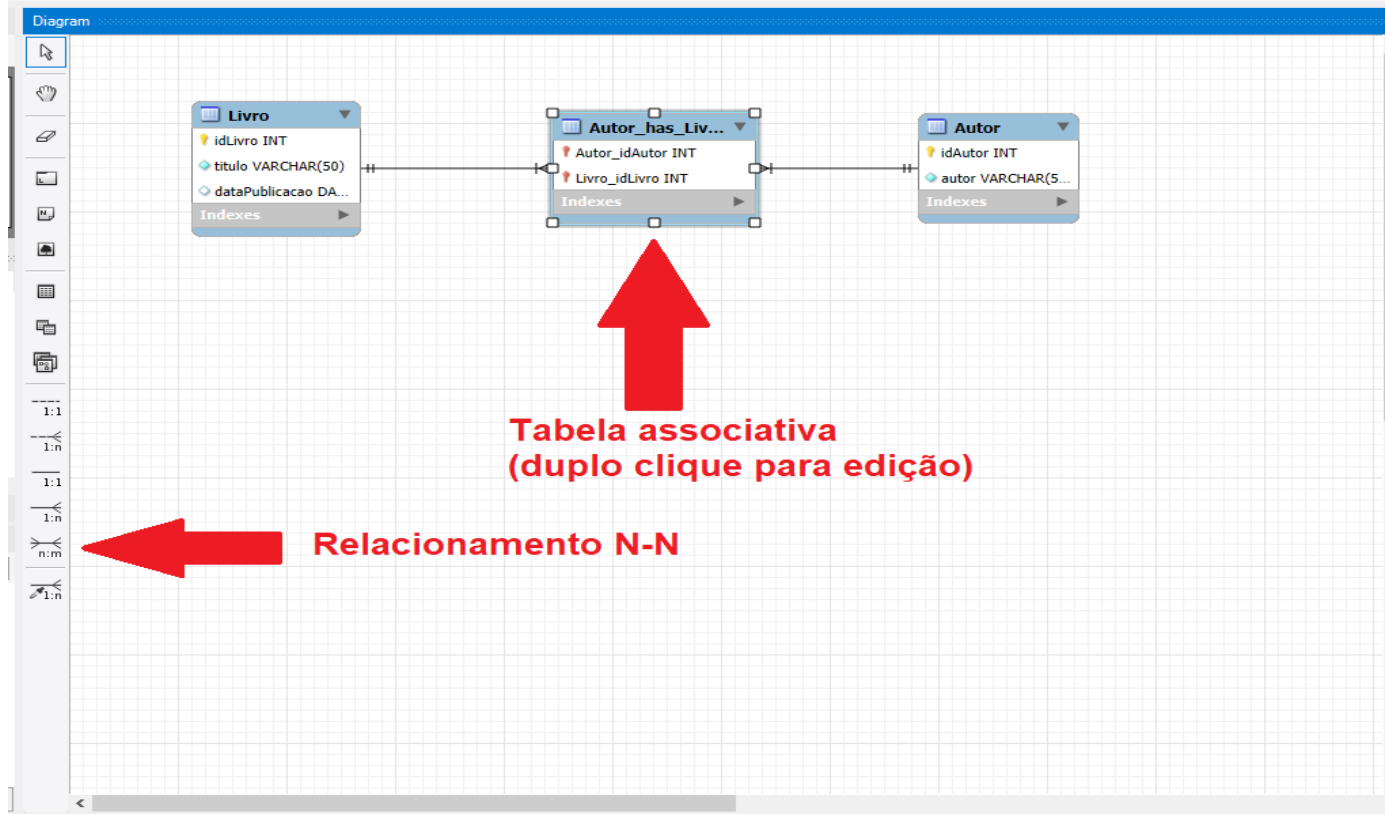
The screenshot shows the MySQL Workbench interface. At the top, a diagram view displays two tables: 'Livro' and 'Autor'. The 'Livro' table has columns 'idLivro INT', 'titulo VARCHAR(50)', and 'dataPublicacao DA...'. The 'Autor' table has columns 'idAutor INT' and 'autor VARCHAR(50)'. Below the diagram, the 'Table: Autor' tab is selected, showing the table structure for the 'Autor' table in the 'Aula04\_Testes' schema.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idAutor	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
autor	VARCHAR(50)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

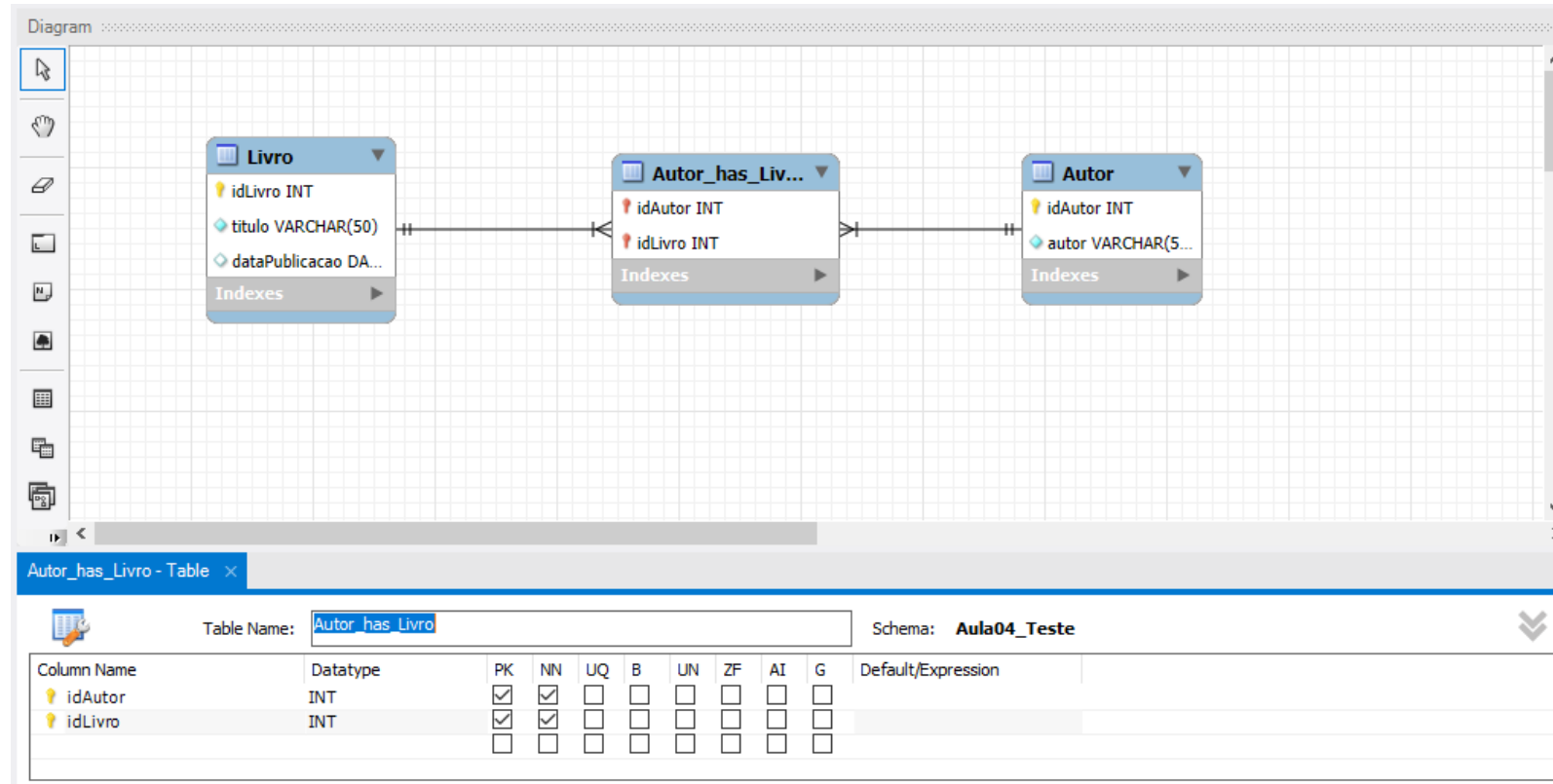
Below the table structure, the 'Column Name' is 'autor', the 'Data Type' is 'VARCHAR(45)', and the 'CharSet/Collation' is 'Default CharSet'. The 'Comments' field is empty. The 'Storage' options are: ☐ Virtual, ☐ Stored, ☐ Primary Key, ☒ Not Null, ☐ Unique, ☐ Binary, ☐ Unsigned, ☐ Zero Fill, ☐ Auto Increment, and ☐ Generated.



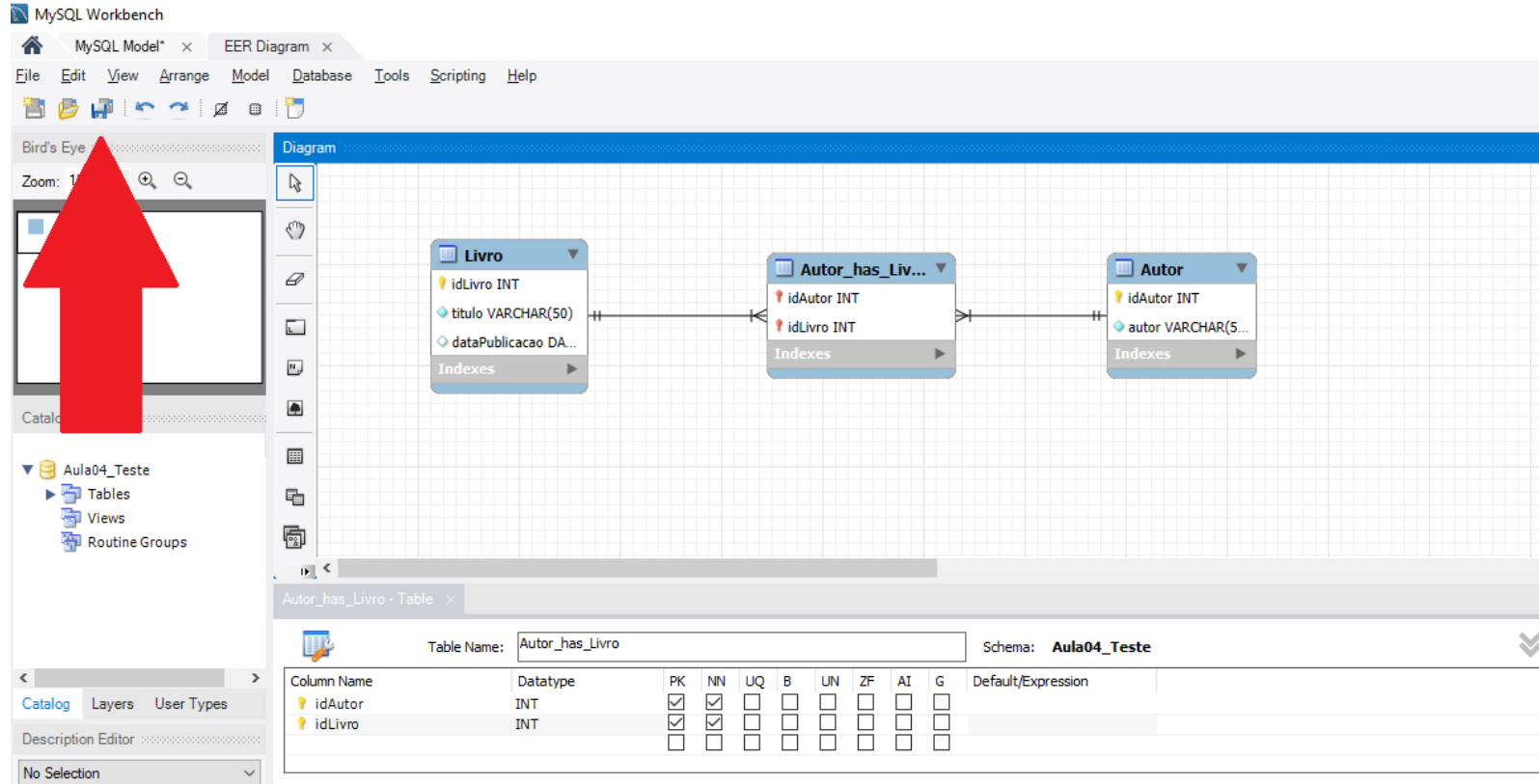
## Atividade 01 – DER com o Workbench

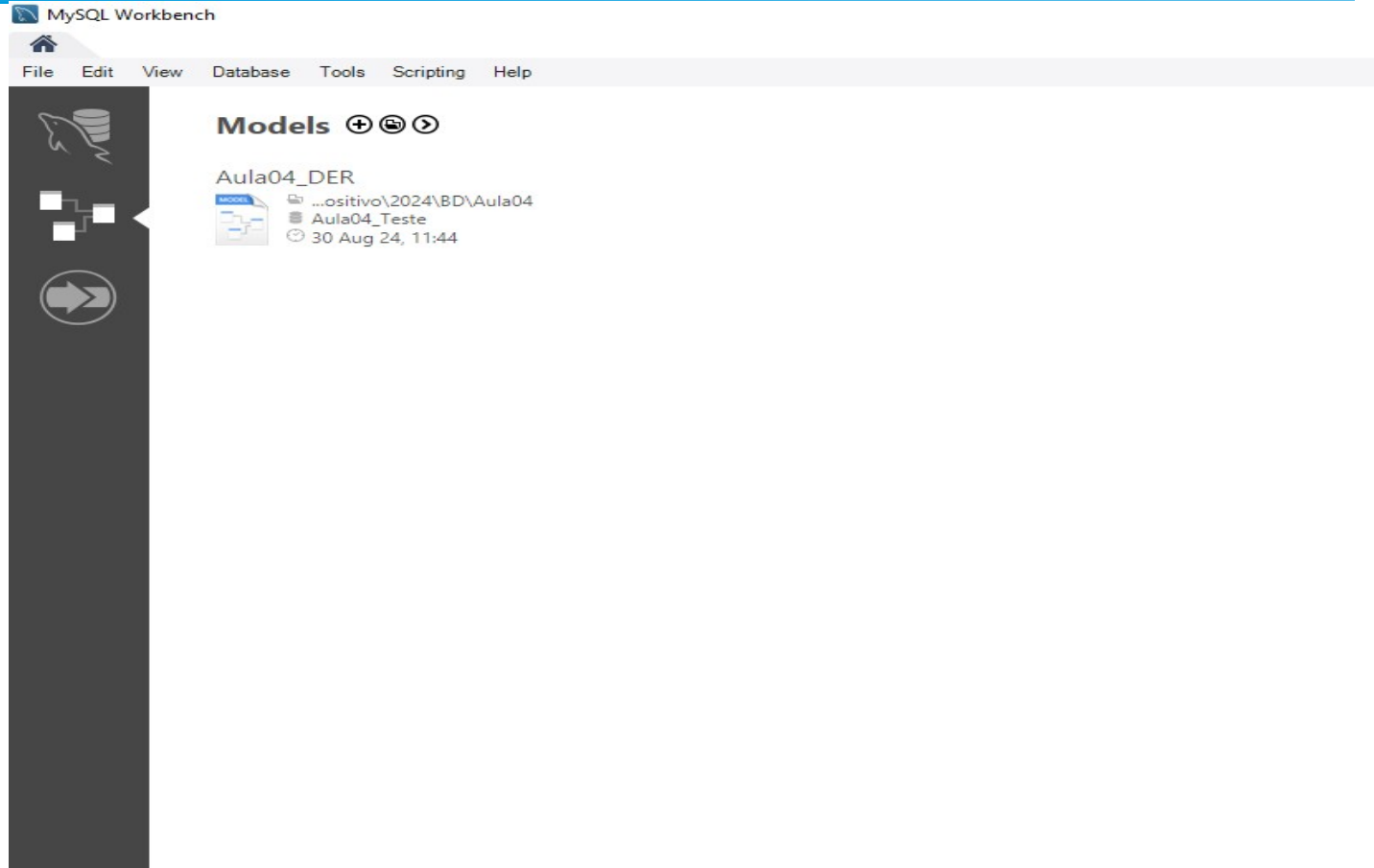


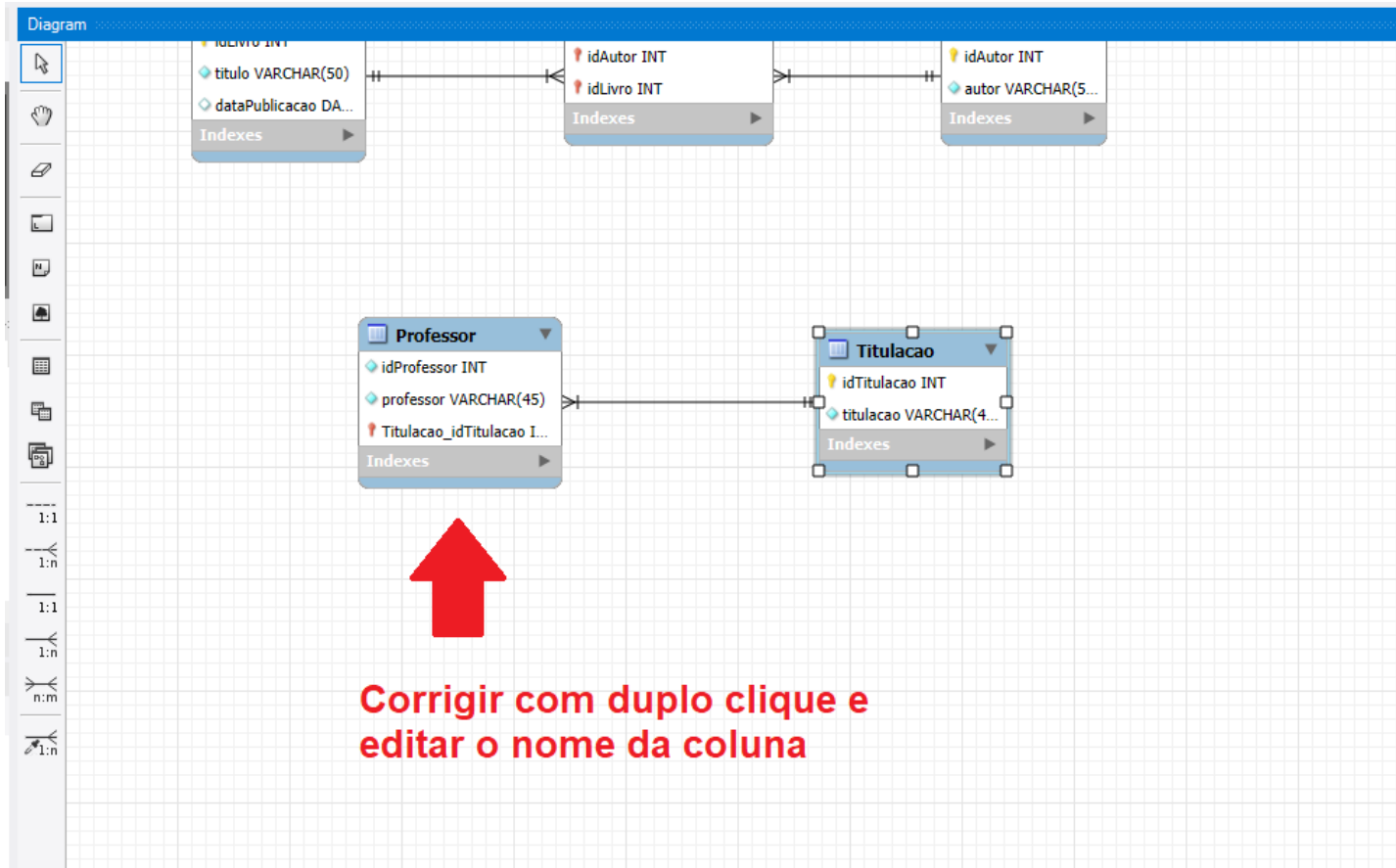
## Atividade 01 – DER com o Workbench



## Atividade 01 – DER com o Workbench







MySQL Workbench

MySQL Model\* (Aula04\_DER.mwb) x EER Diagram x

File Edit View Arrange Model Database Tools Scripting Help

Bird's Eye

Zoom: 100%

Catalog Tree

- Aula04\_Testes
  - Tables
  - Views
  - Routine Groups

Diagram

Professor

- idProfessor INT
- professor VARCHAR(45)
- idTitulacao INT

Indexes

Titulacao

- idTitulacao INT
- titulacao VARCHAR(45)

Indexes

Professor - Table

Table Name: Professor Schema: Aula04\_Testes

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idProfessor	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
professor	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
idTitulacao	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Column Name: idTitulacao Data Type: INT

Charset/Collation: Default Charset Default Collation

Comments:

Storage: ☐ Virtual ☐ Stored

☒ Primary Key ☒ Not Null ☐ Unique

☐ Binary ☐ Unsigned ☐ Zero Fill

☐ Auto Increment ☐ Generated

Description Properties H Columns Indexes Foreign Keys Triggers Partitioning Options Inserts Privileges