

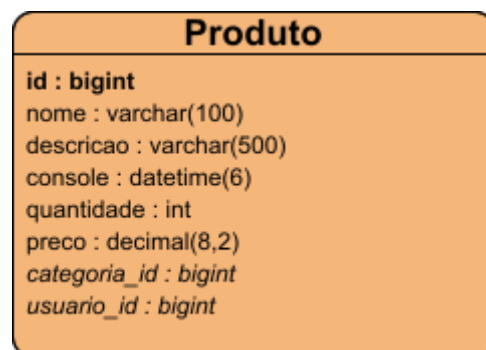
Diagrama Entidade Relacionamento (DER) no MySQL Workbench

Um **Diagrama Entidade Relacionamento** (DER) é um Fluxograma que ilustra como as “**Entidades**” (Usuários, Produtos, Postagens), se relacionam entre si dentro de um sistema de Banco de dados Relacional.

Os Diagramas DER são usados principalmente para modelar e criar bancos de dados relacionais, em termos de regras lógicas e regras de negócio dentro de um modelo lógico de dados.

1.1. Componentes

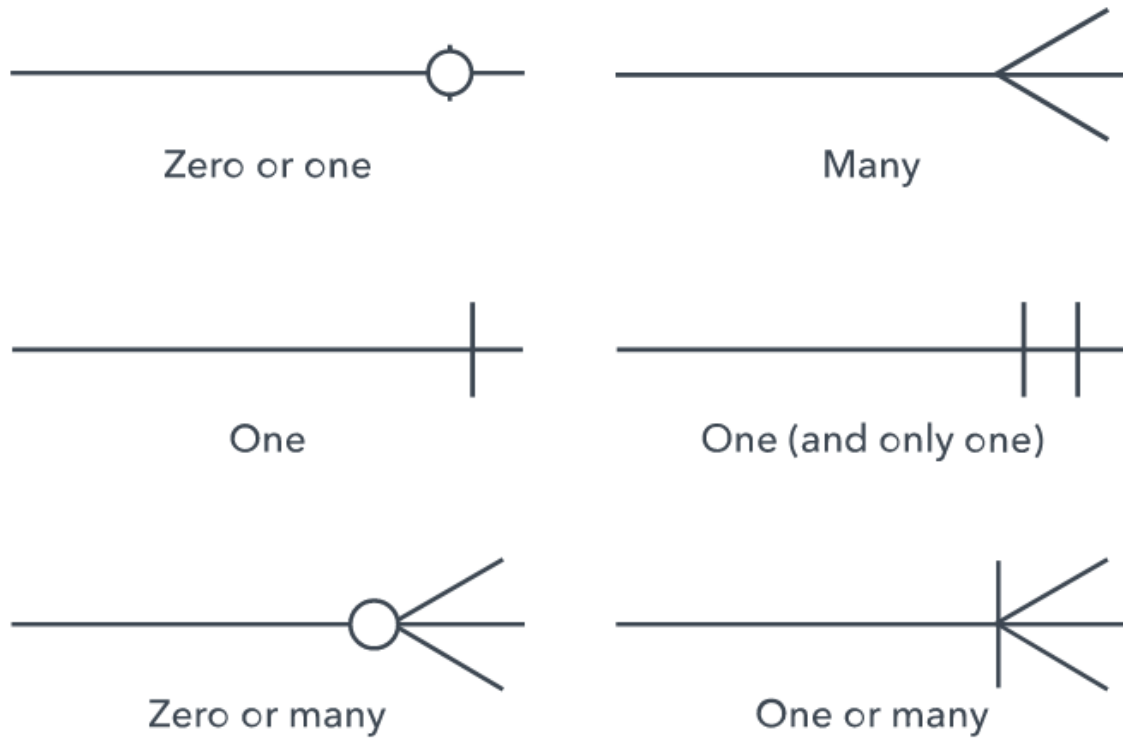
Entidade: Algo que pode ser definido e que pode ter dados armazenados sobre ele — como uma pessoa, um objeto, conceito ou evento. Pense em entidades como substantivos. **Exemplos:** um cliente, estudante, carro ou produto. Normalmente representado por um retângulo.



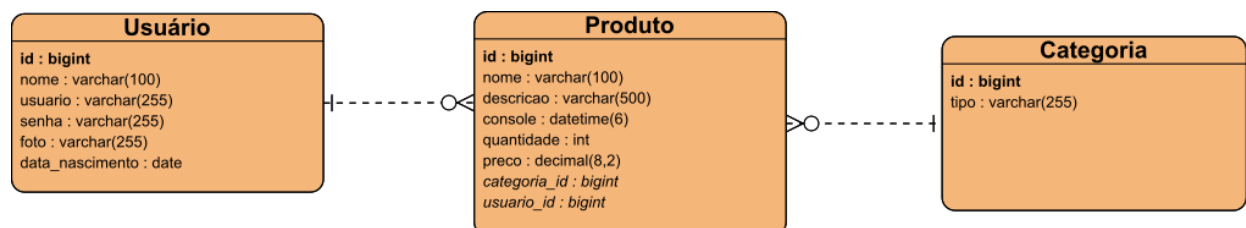
Relacionamento: Como entidades atuam umas sobre as outras ou estão associadas uma com a outra. Pense em relacionamentos como verbos. **Exemplo:** o estudante pode se inscrever em um curso. As duas entidades seriam o aluno e o curso, e o relacionamento descrito é o ato de matricular-se, assim conectando as duas entidades. Relacionamentos são tipicamente representados por linhas de ligação.

Cardinalidade: Define os atributos numéricos da relação entre duas entidades ou conjuntos de entidades. Os três principais relacionamentos cardinais são **um-para-um (1:1)**, **um-para-muitos (1:N)** e **muitos-para-muitos (N:M)**. Um **exemplo de um-para-um** seria um estudante associado a um endereço de correspondência. Um **exemplo de um-**

para-muitos (ou muitos-para-um, dependendo do sentido da relação): um estudante se inscreve para vários cursos, mas todos esses cursos têm uma única linha que leva de volta ao aluno. **Exemplo de muitos-para-muitos:** estudantes como um grupo são associados a vários membros do corpo docente, e membros do corpo docente, por sua vez, são associados a vários alunos. Abaixo temos os símbolos que representam a cardinalidade:

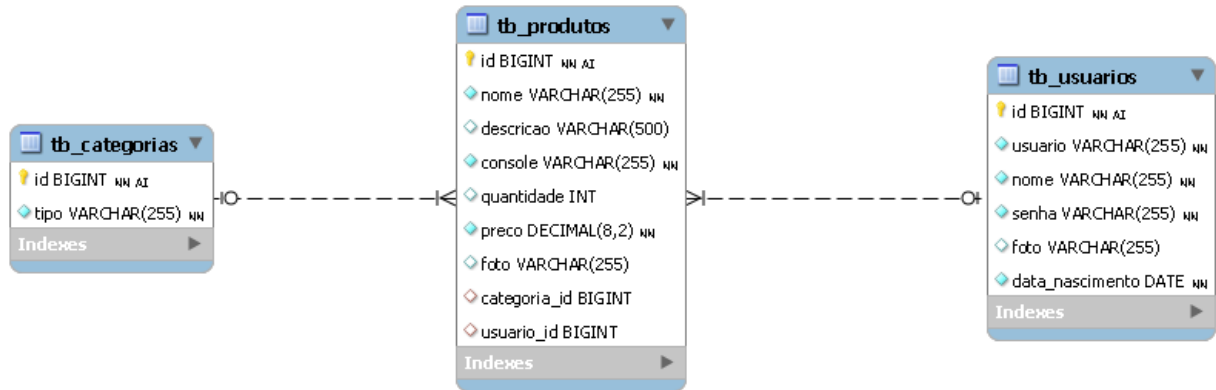


Na figura abaixo, temos um exemplo de Diagrama DER com Relacionamento:



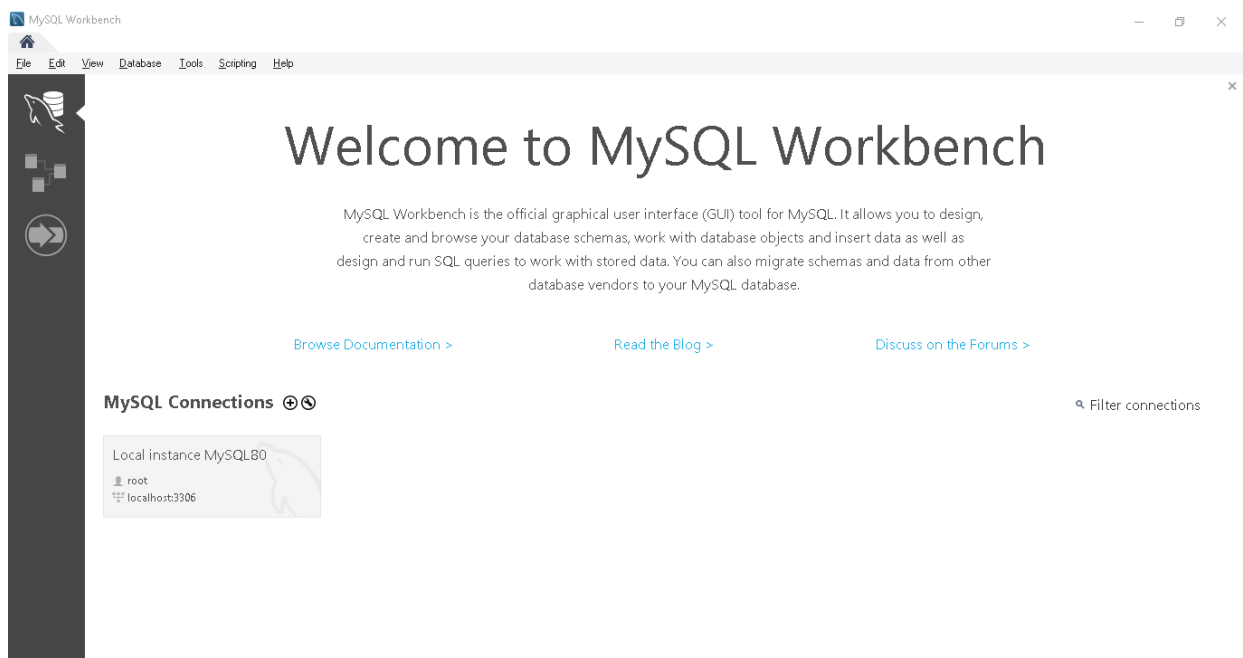
2. Criando o DER no MySQL Workbench

Vamos Criar o Diagrama Entidade Relacionamento de uma Loja de Games, seguindo o modelo abaixo, no MySQL Workbench:

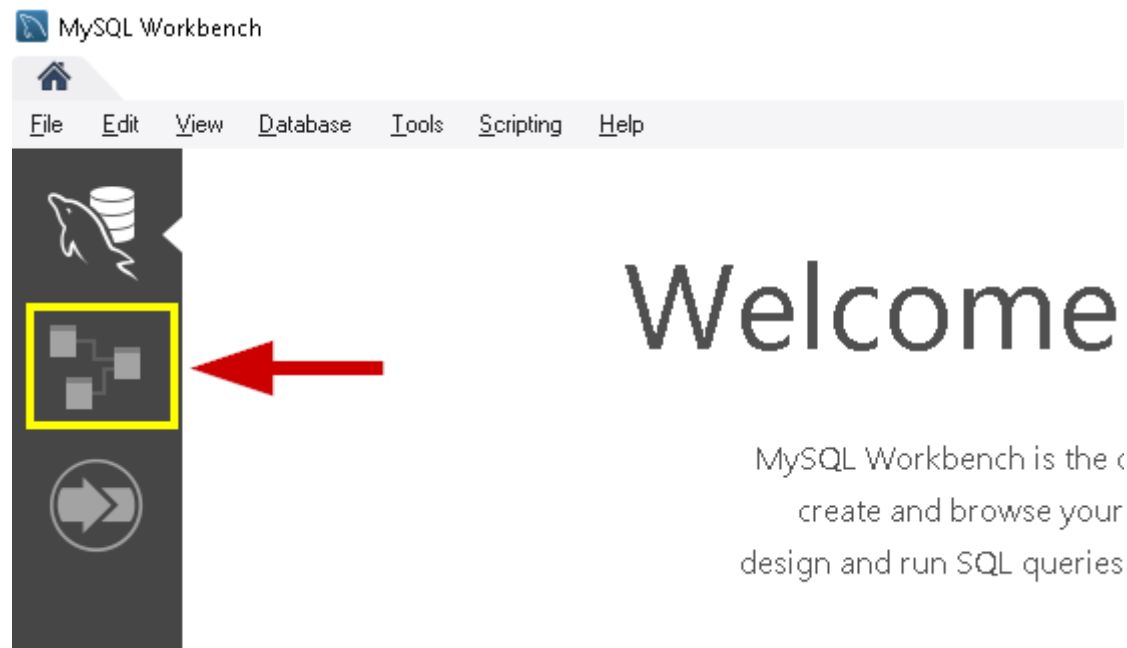



Passo 01 - Iniciando o Modelo

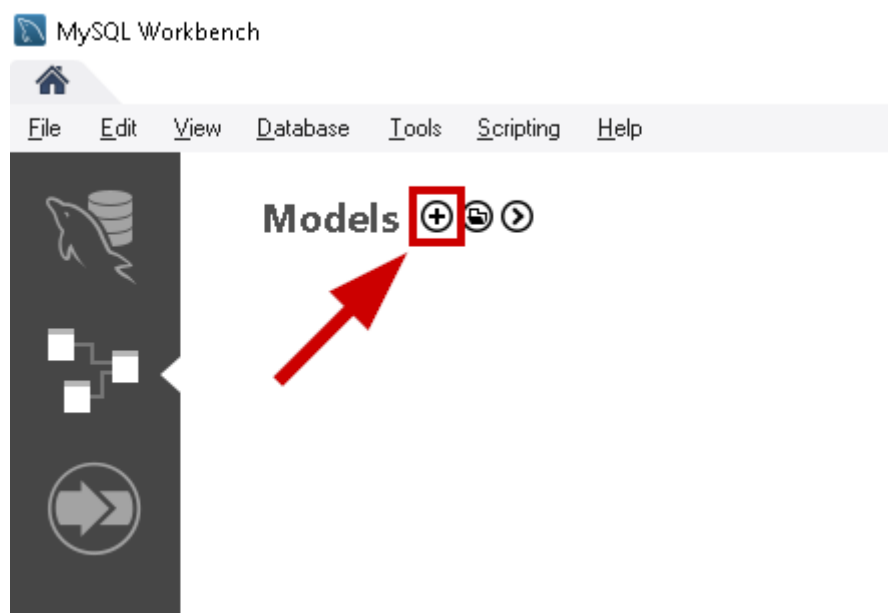
1. Abra o **MySQL Workbench**



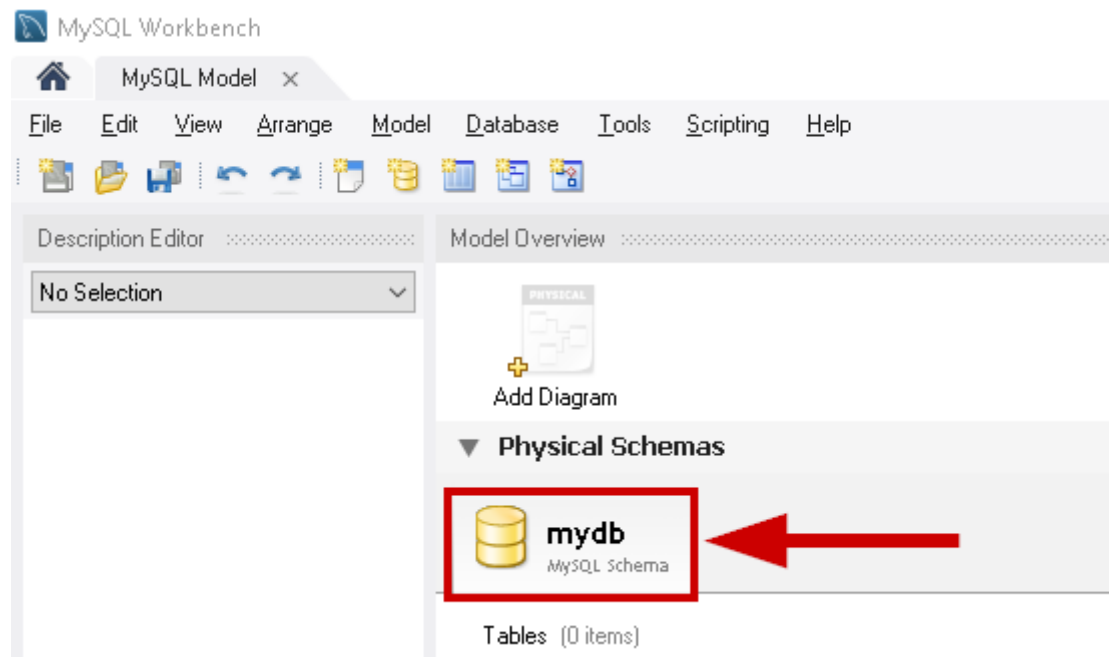
2. No menu lateral, do lado esquerdo superior do Workbench, clique no 2º ícone (**Models**)



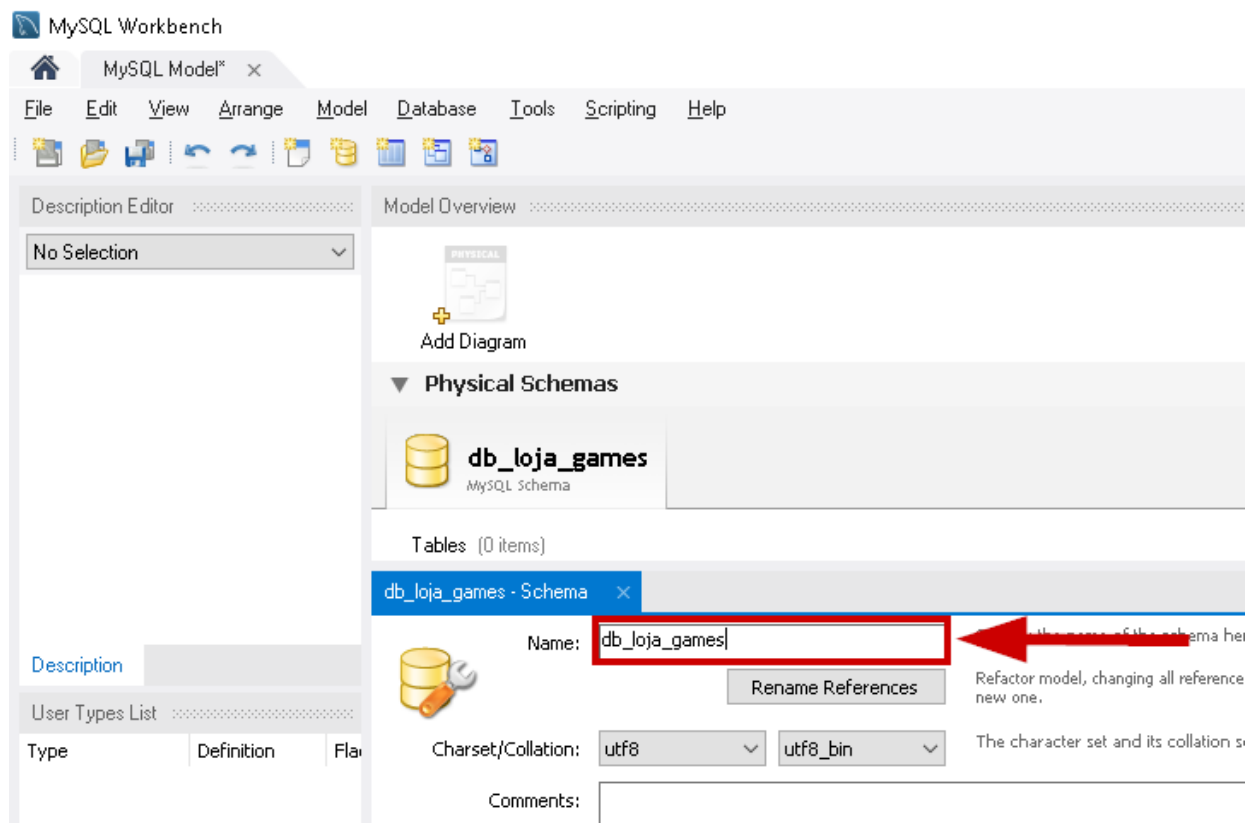
3. Será aberta a janela **Models**. Clique no botão  para adicionar um novo **Modelo**.



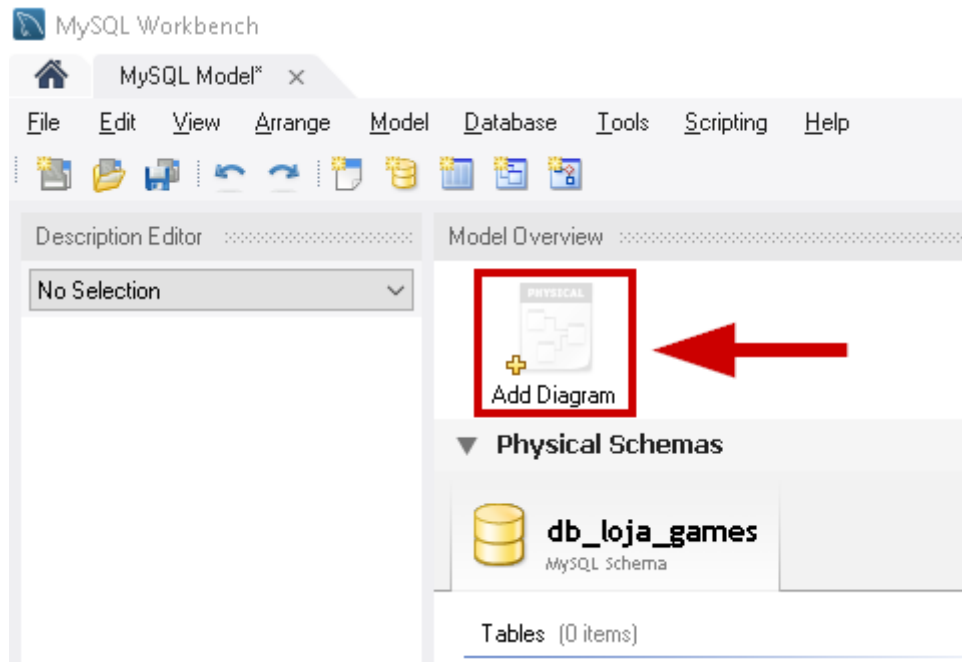
4. Na Guia **Physical of Schemas**, dê um duplo clique sobre **mydb** para **alterar o nome do Banco de dados**.



5. Vamos alterar o nome do Banco de dados para **db_loja_games**.

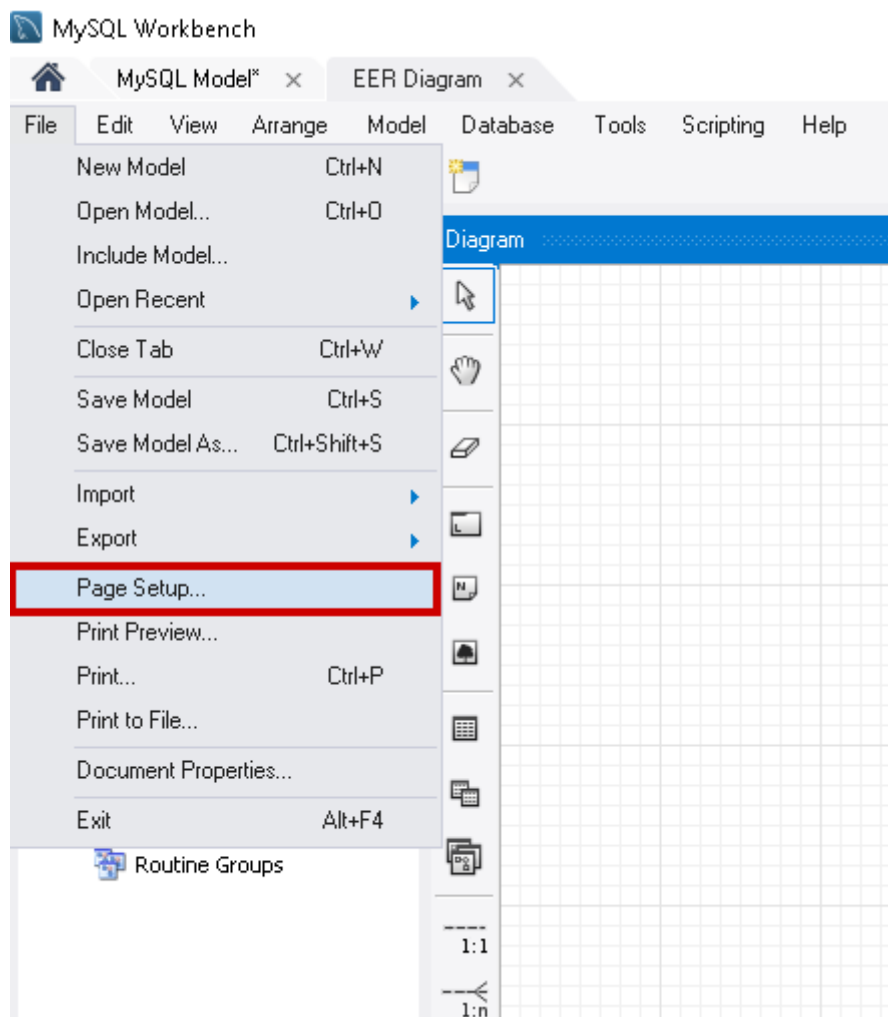


6. Vamos adicionar um novo Diagrama no Modelo. Clique no ícone **Add Diagram**.

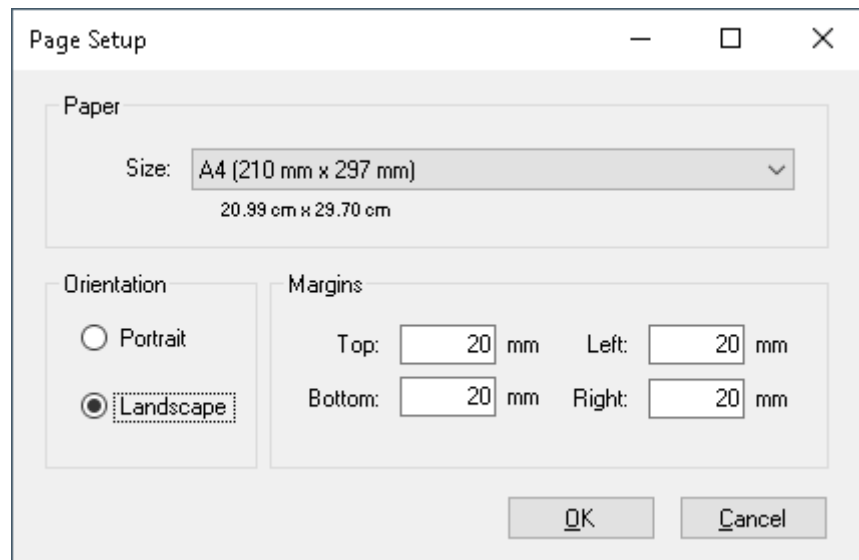


Antes de começar a criar o DER, vamos configurar alguns itens do Workbench.

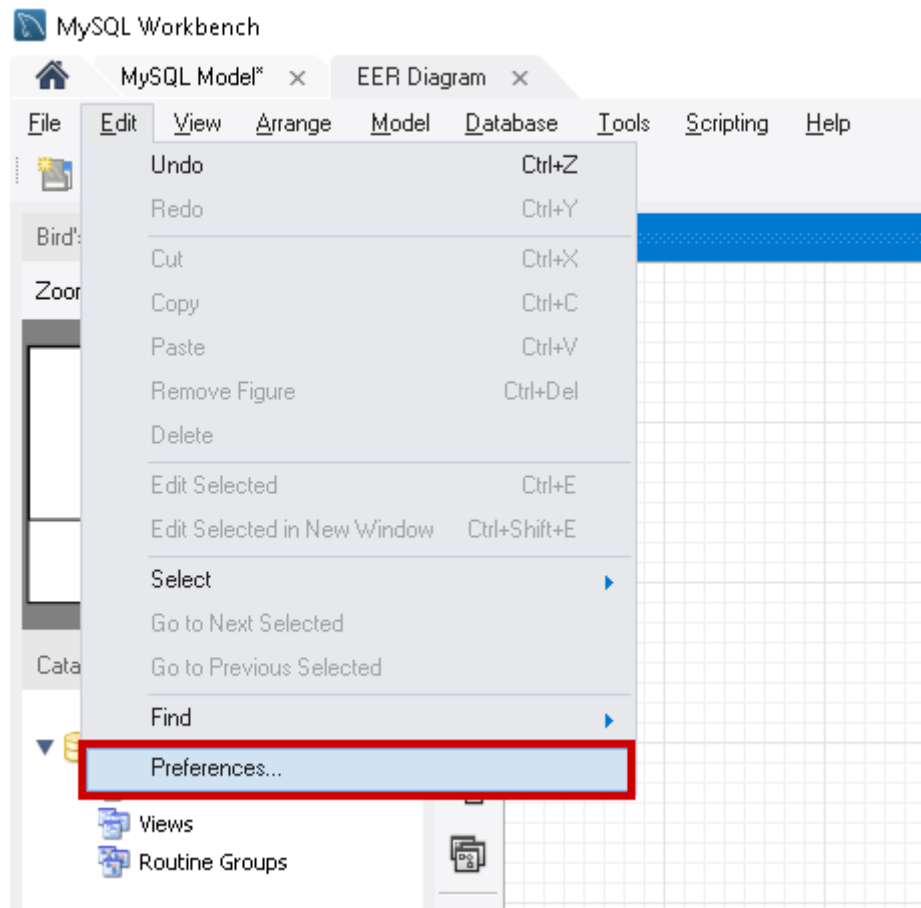
1. No menu **File**, clique na opção **Page Setup...** para configurar a página



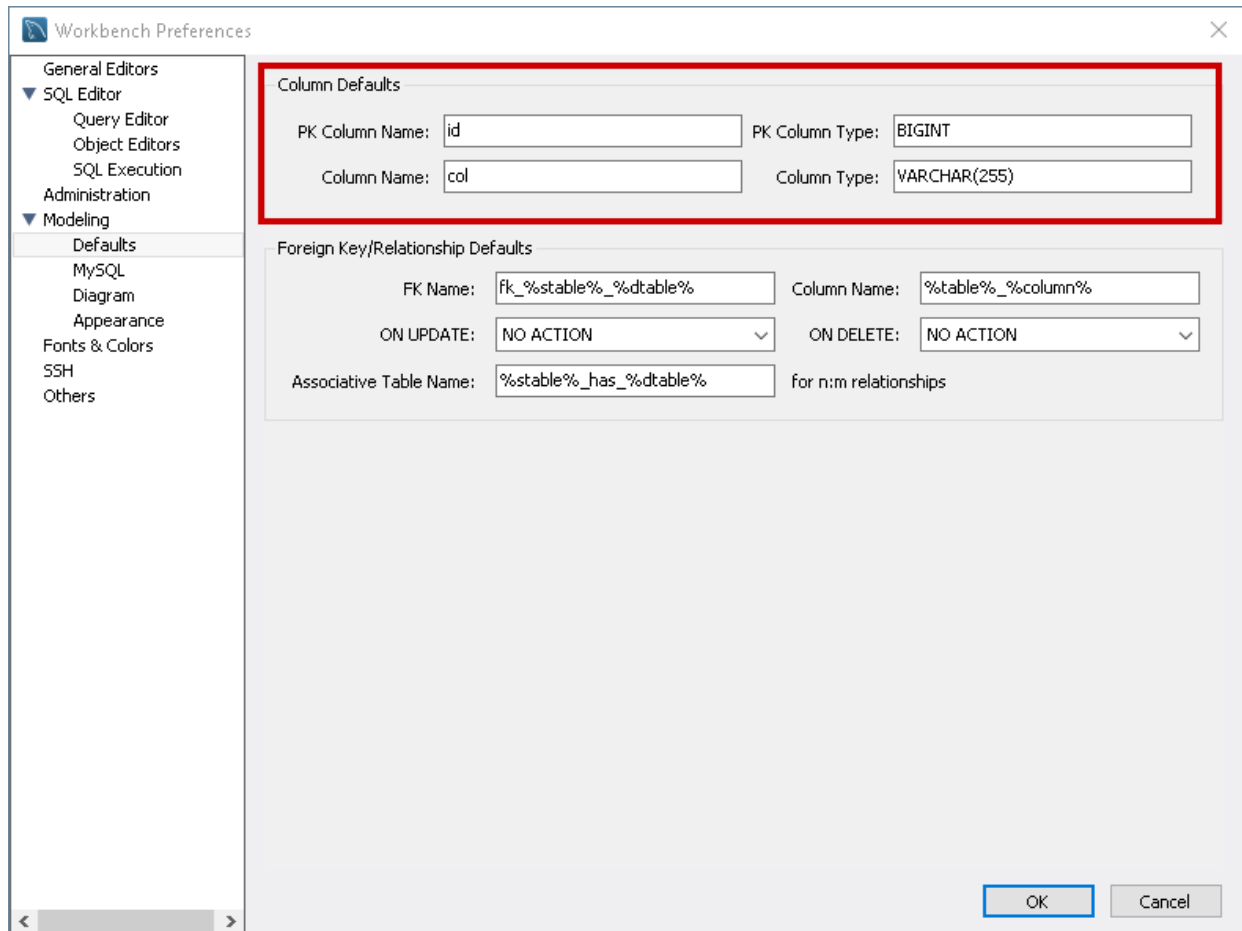
2. Configure igual a figura abaixo e clique em **OK** para concluir:



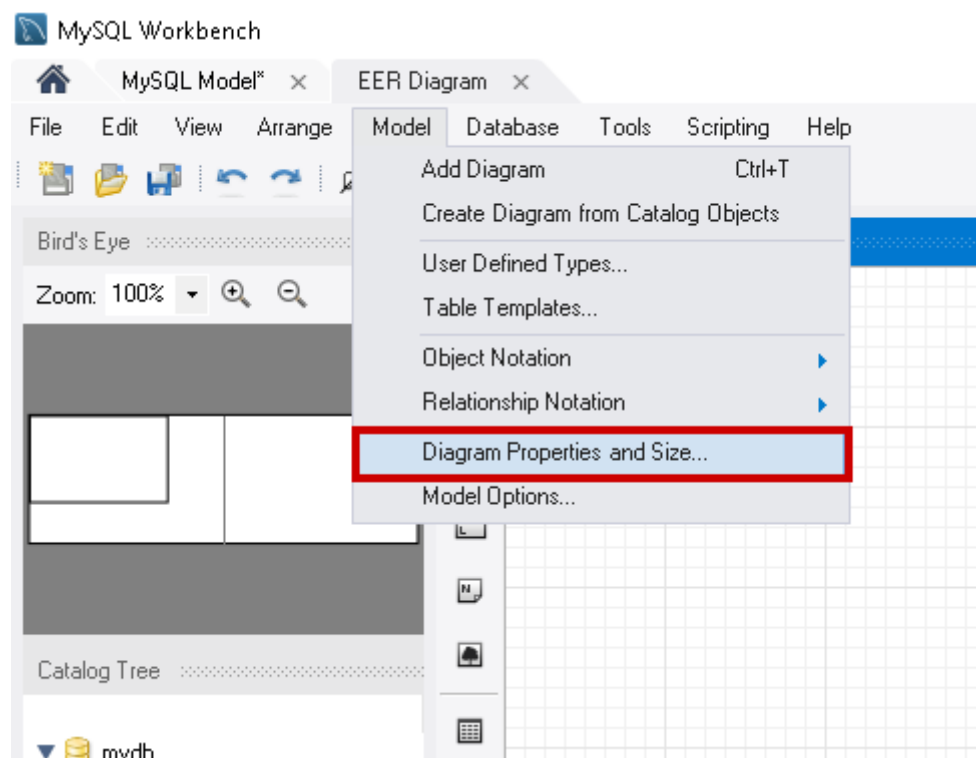
3. No menu **Edit**, clique na opção **Preferences** para configurar o Modelo de dados



4. Na guia **Modeling** → **Defaults**, configure igual a figura abaixo e clique em **OK** para concluir.



5. No menu **Model**, clique na opção **Diagram Properties and Size...** para configurar Diagrama




6. No item **Name**, informe o nome do Diagrama (**loja_games**) e as propriedades **Width e Height**, vamos configurar ambas com o valor **1** (numero de páginas). Clique em **OK** para concluir.

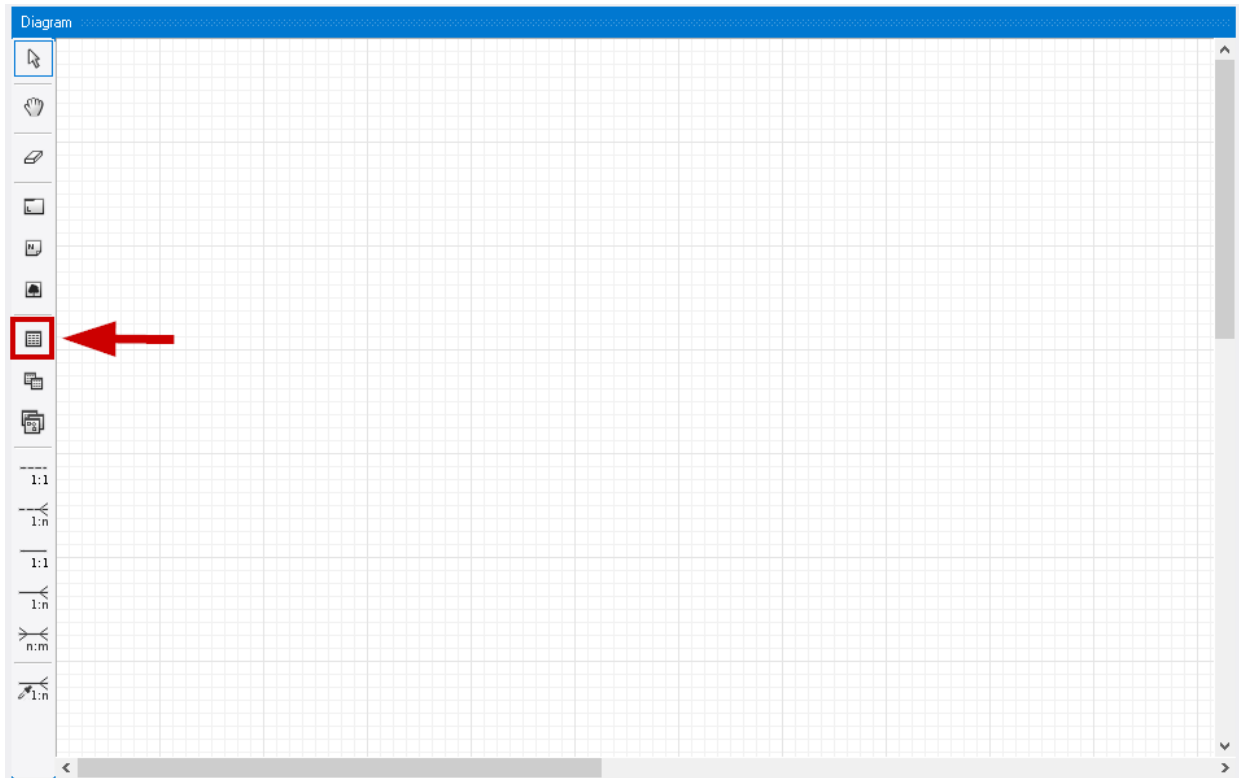
Diagram Properties

Name:

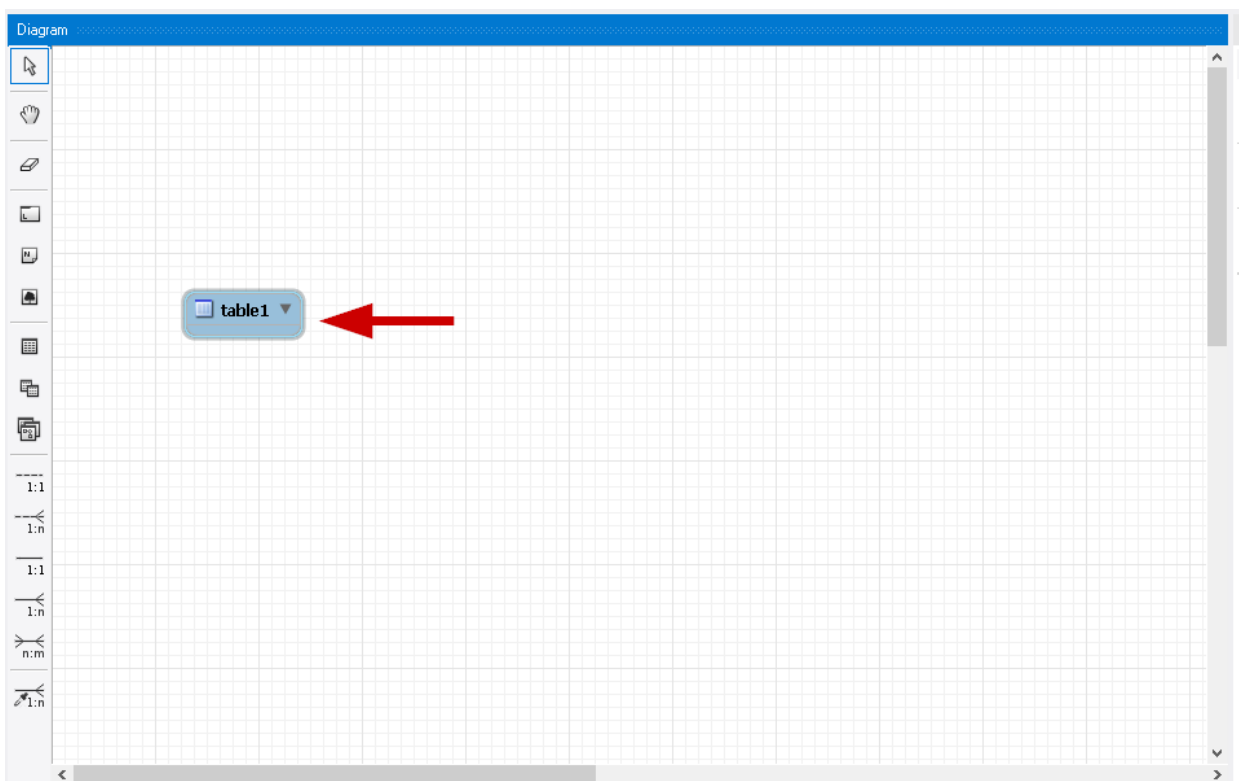
Width: pages Height: pages

Passo 02 - Criar as Tabelas

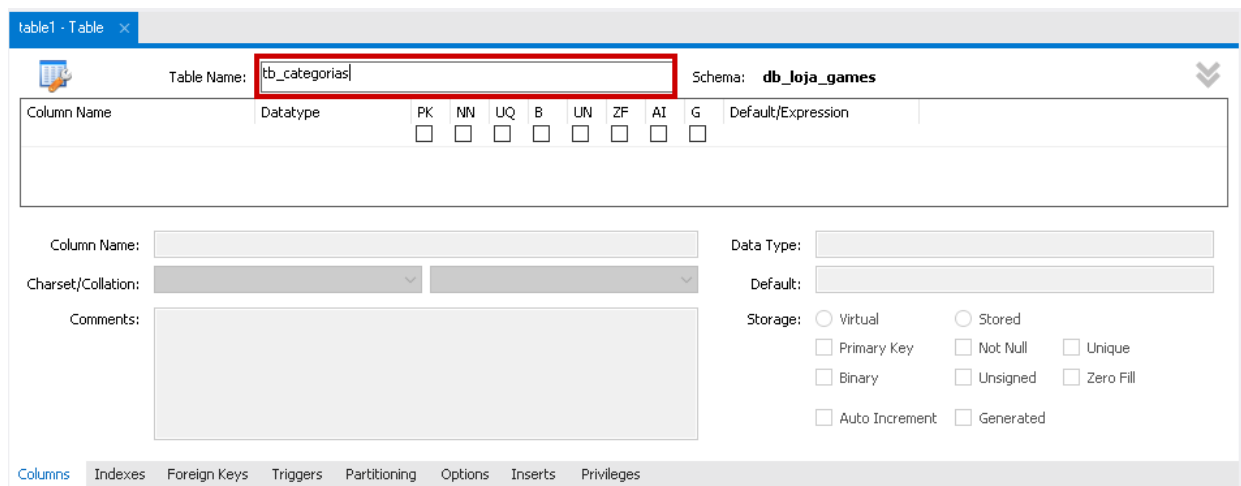
1. Na janela **Diagram**, clique no botão  **Place a New Table**, para adicionar uma nova tabela (Entidade) no Diagrama.



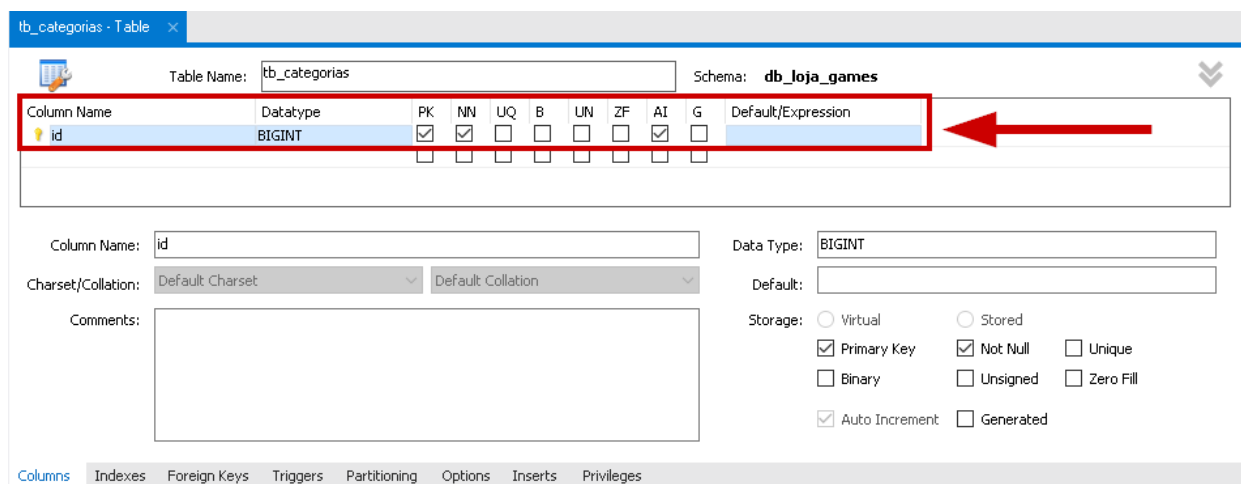
2. Dê um clique sobre a tela do Diagrama para adicionar a tabela. Para **Editar a tabela**, dê um duplo clique sobre ela.



3. No item **Table Name**, informe o nome da tabela (**tb_categorias**)



4. Para inserir o primeiro atributo da tabela, clique abaixo da coluna Column Name. Observe que o Workbench irá sugerir o atributo **id** (Chave primária) no formato **BIGINT**. Marque as opções **PK**, **NN** e **AI**.



Opções do atributo:

Opção	Descrição
PK	Primary Key → Chave Primária
NN	Not Null → Não pode ser Nulo
UQ	Unique → Index Impõe a exclusividade de valores em uma ou mais colunas, além da Chave Primária
B	Binary → Armazena atributos binários (0 1)
UN	Unsigned Data Type → Permite apenas numeros positivos inteiros

Opção	Descrição
ZF	Zero Fill → Preencher numeros inteiros com zeros. Exemplo: int(5) = 00001
AI	Auto Increment → Configurar a Chave Primária como Auto Incremento
G	Generated Column → Gerar colunas com cálculos ou outros valores específicos.
Expression Default	Valor Padrão ou a Expressão da opção Generated Column.

5. Clique na linha de baixo para inserir o segundo atributo. Observe que será sugerido o atributo **col** no formato **varchar(255)**, como mostra a figura abaixo:

Table Name: **tb_categoria** Schema: **db_loja_games**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
id	BIGINT	<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"/>	
col	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	

Column Name: Data Type:

Charset/Collation:

Comments:

Storage: ☐ Virtual ☐ Stored

☐ Primary Key ☐ Not Null ☐ Unique

☐ Binary ☐ Unsigned ☐ Zero Fill

☐ Auto Increment ☒ Generated

6. Vamos alterar o nome do atributo para **tipo** e manter o formato. Marque apenas a opção **NN**.

Table Name: **tb_categoria** Schema: **db_loja_games**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
id	BIGINT	<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"/>	
tipo	VARCHAR(255)	<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: Data Type:

Charset/Collation:

Comments:

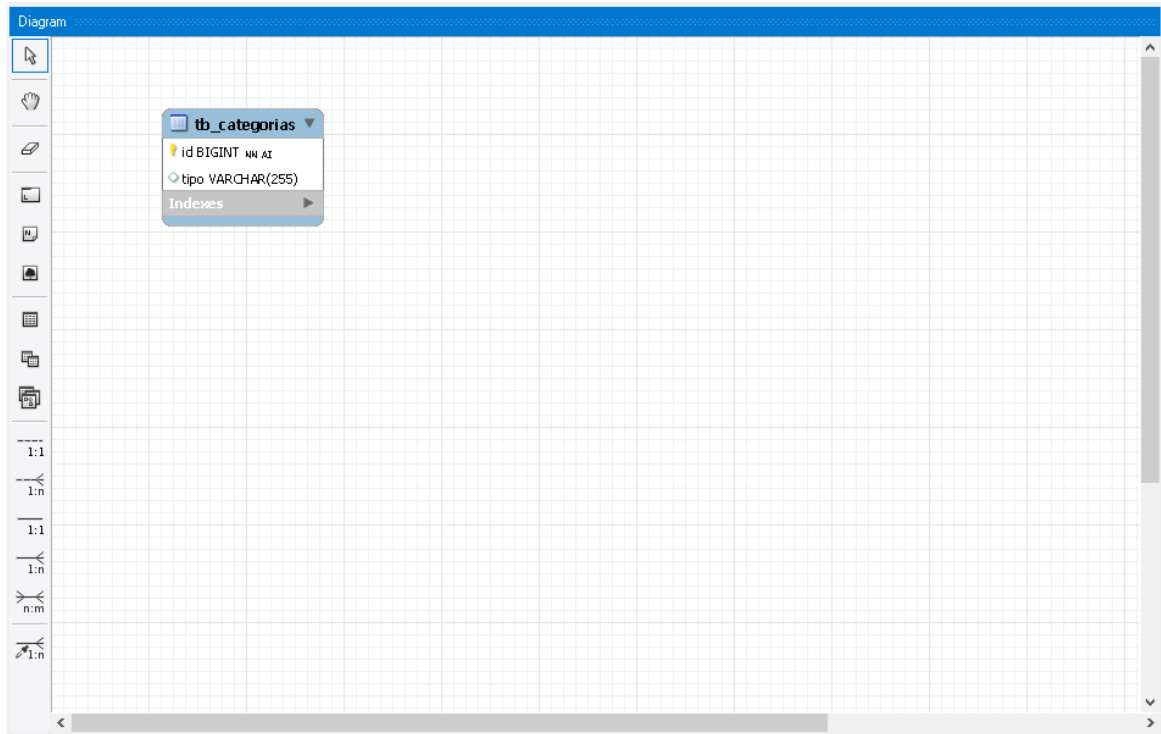
Storage: ☐ Virtual ☐ Stored

☐ Primary Key ☒ Not Null ☐ Unique

☐ Binary ☐ Unsigned ☐ Zero Fill

☐ Auto Increment ☐ Generated

7. Primeira Tabela finalizada, vamos fechar a guia da tabela `tb_categorias`. Veja o nosso DER com a primeira tabela na figura abaixo:



8. Vamos criar a segunda tabela (`tb_usuarios`), igual a figura abaixo. Siga os passos de 1 a 7 para construir a tabela.

Table Name: `tb_usuarios` Schema: `db_loja_games`

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
<code>id</code>	BIGINT	<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"/>	
<code>usuario</code>	VARCHAR(255)	<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"/>	
<code>nome</code>	VARCHAR(255)	<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"/>	
<code>senha</code>	VARCHAR(255)	<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"/>	
<code>foto</code>	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
<code>data_nascimento</code>	DATE	<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: `id` Data Type: `BIGINT`

Charset/Collation: `Default Charset` `Default Collation`

Comments:

Storage: ☐ Virtual ☐ Stored

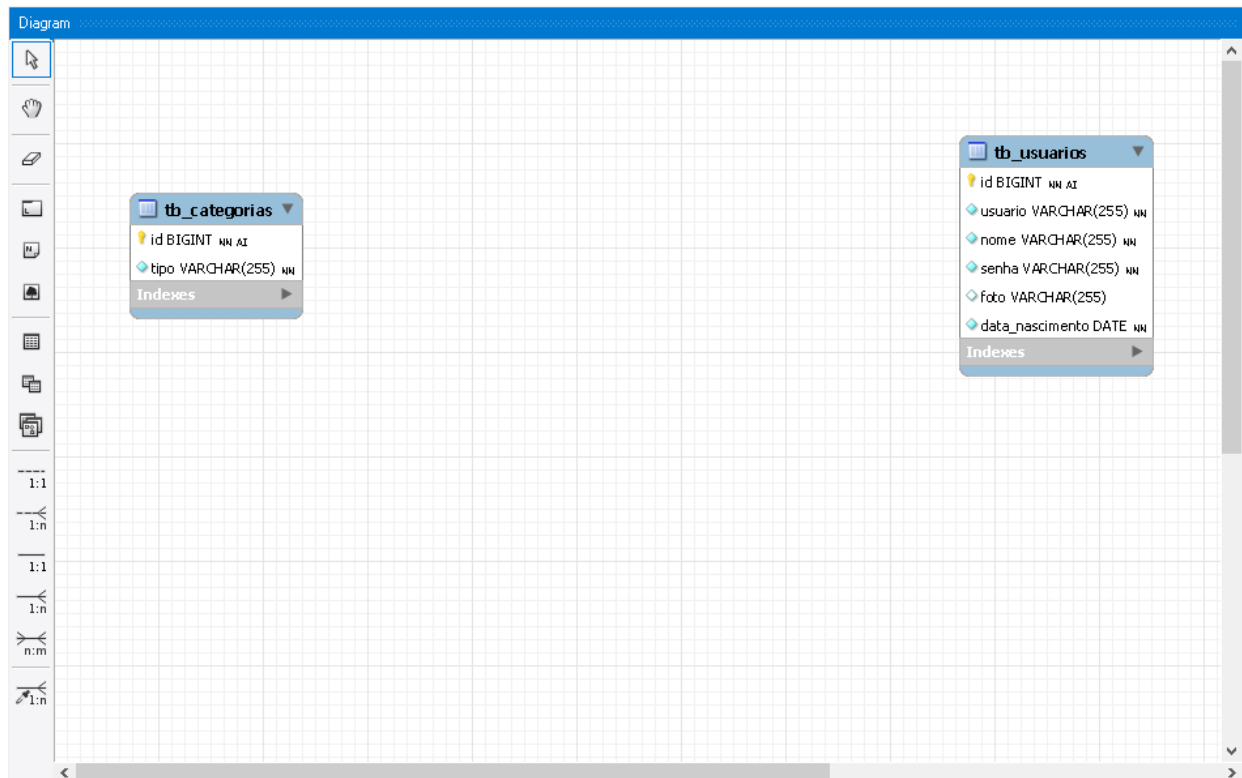
☒ Primary Key ☒ Not Null ☐ Unique

☐ Binary ☐ Unsigned ☐ Zero Fill

☒ Auto Increment ☐ Generated

Columns Indexes Foreign Keys Triggers Partitioning Options Inserts Privileges

9. Segunda Tabela finalizada, vamos fechar a guia da tabela `tb_usuarios`. Veja o nosso DER com as duas tabelas na figura abaixo:



10. Vamos criar a terceira tabela (**tb_produtos**), igual a figura abaixo. Siga os passos de 1 a 7 para construir a tabela.

A interface de criação da tabela `tb_produtos` no schema `db_loja_games`. A tabela é definida com as seguintes colunas:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
id	BIGINT	<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"/>	
nome	VARCHAR(255)	<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"/>	
descricao	VARCHAR(500)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
console	VARCHAR(255)	<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"/>	
quantidade	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
preco	DECIMAL(8,2)	<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"/>	
foto	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

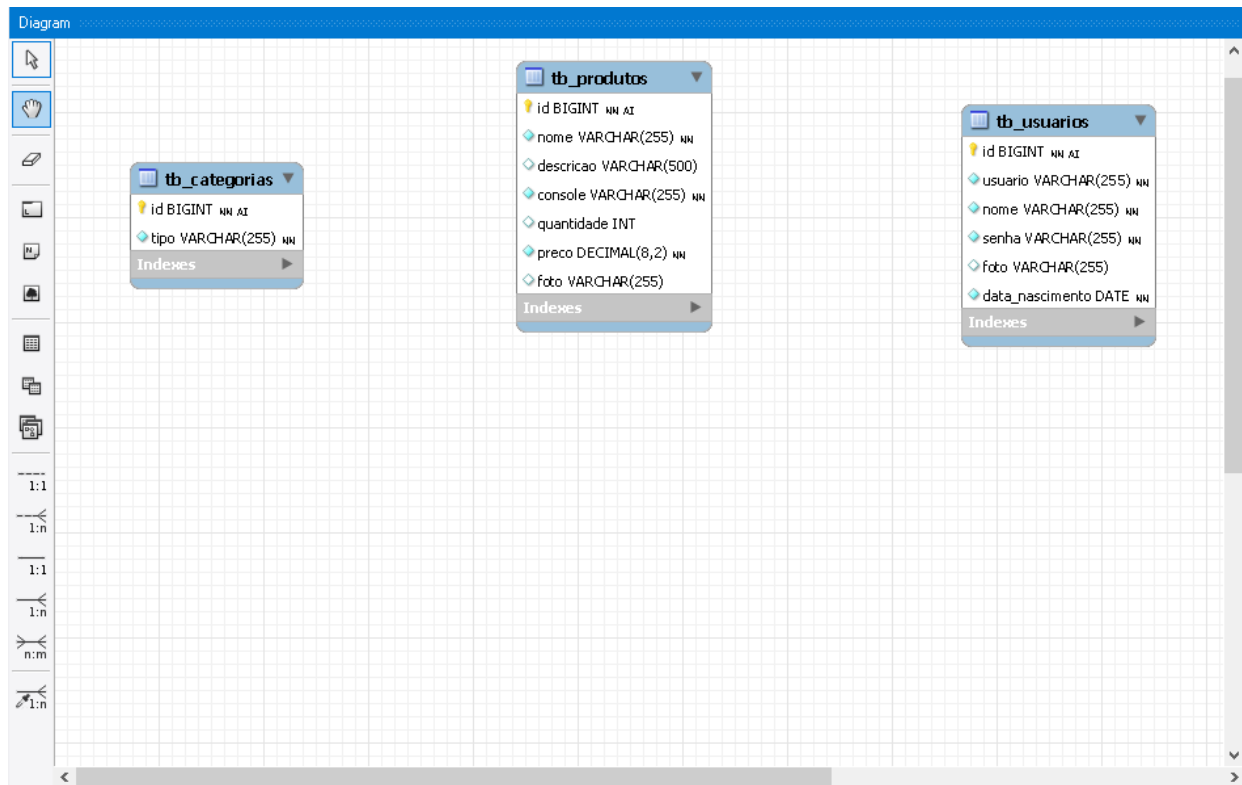
Abas de configuração:

- Columns**: Campos para definir a coluna (nome, tipo de dados, charset/collation, comentários).
- Indexes**: Índices.
- Foreign Keys**: Chaves estrangeiras.
- Triggers**: Gatilhos.
- Partitioning**: Particionamento.
- Options**: Opções.
- Inserts**: Inserções.
- Privileges**: Permissões.

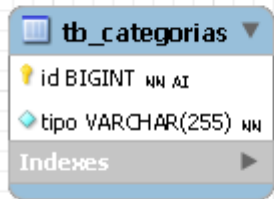
Configurações adicionais:

- Data Type**: Campo para definir o tipo de dados.
- Default**: Campo para definir o valor padrão.
- Storage**:
 - ☐ Virtual ☐ Stored
 - ☐ Primary Key ☐ Not Null ☐ Unique
 - ☐ Binary ☐ Unsigned ☐ Zero Fill
 - ☐ Auto Increment ☐ Generated

11. Observe que não criamos os atributos **categoria_id** e **usuario_id**, que são as **chaves estrangeiras** da tabela tb_produtos. Faremos isso no próximo passo. Veja o nosso DER com as 3 tabelas na figura abaixo:




- 

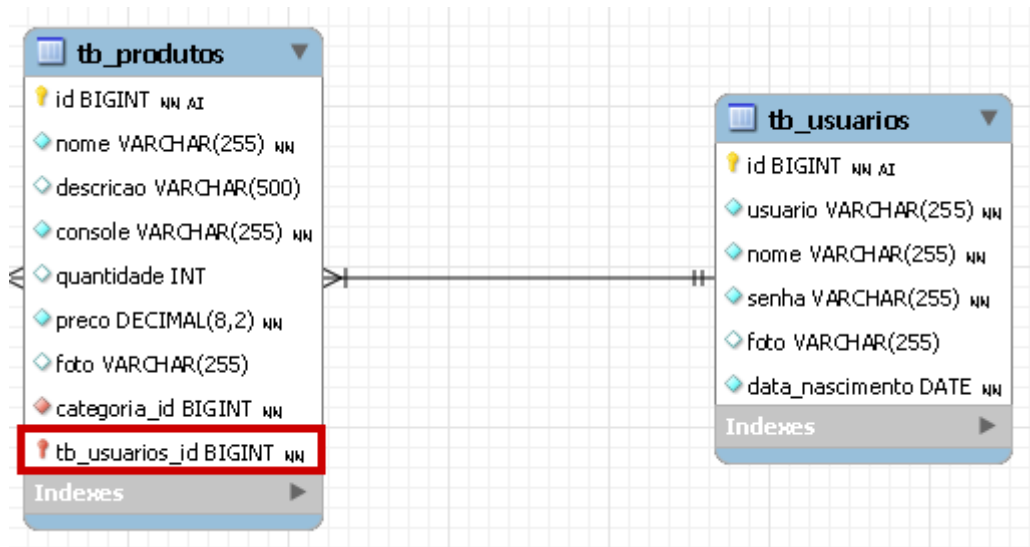


- Observe apenas que o nome do atributo Chave estrangeira está um pouco diferente. Dê um duplo clique sobre a Tabela tb_produtos e altere o nome do atributo **Chave Estrangeira** para **categoria_id** e desmarque a opção **PK e NN**, como mostra a figura abaixo:

[illegible]

3. Vamos Criar o segundo relacionamento (**usuario_id** → **id**). Este relacionamento

será do tipo um para muitos (1:N). Clique no botão  (**Place new 1:N Relationship Identify**). Clique sobre a tabela **tb_produtos** e depois clique sobre a tabela **tb_usuarios**.

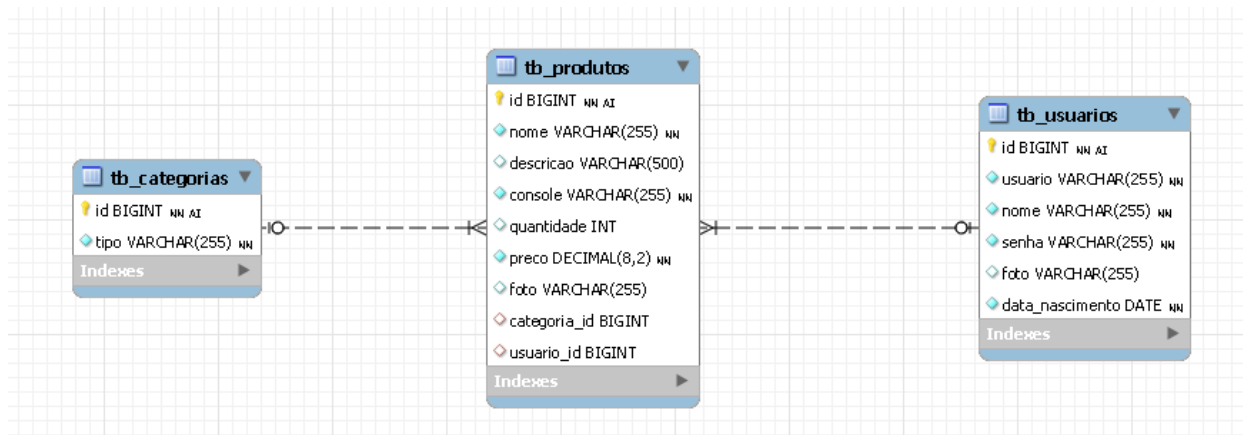


4. A Chave Estrangeira e o Relacionamento foram criados automaticamente.

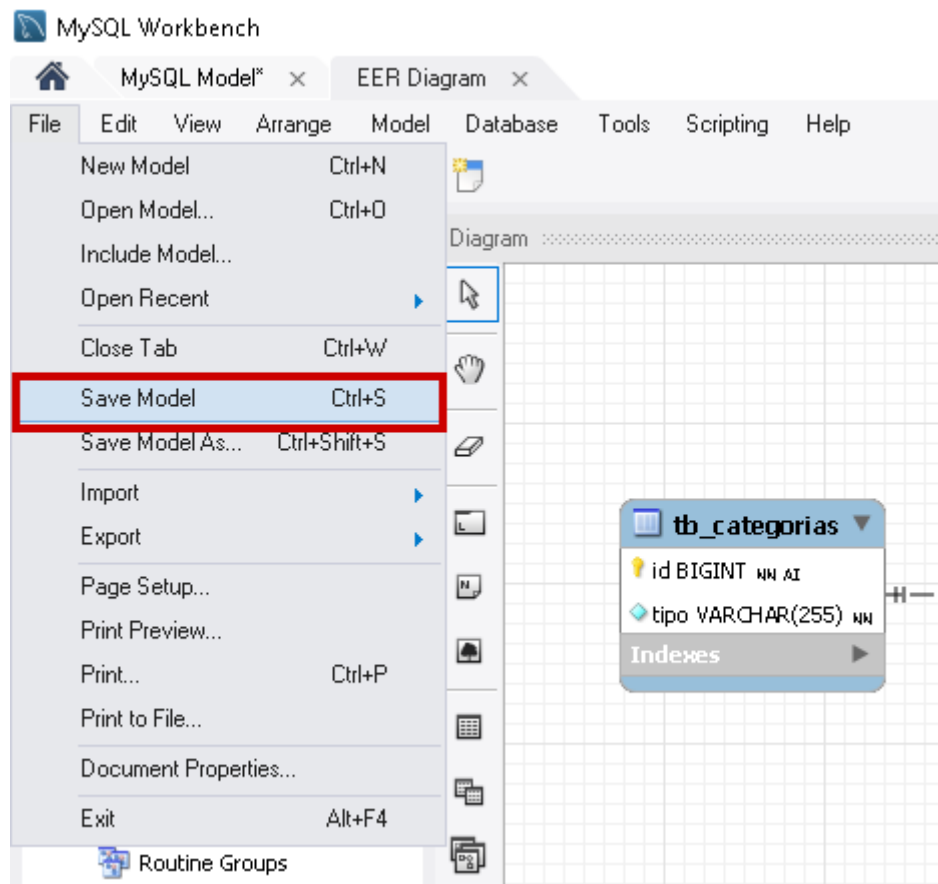
Observe apenas que o nome do atributo Chave estrangeira está um pouco diferente. Dê um duplo clique sobre a Tabela tb_produtos e altere o nome do atributo **Chave Estrangeira** para **usuario_id** e desmarque a opção **PK e NN**, como mostra a figura abaixo:

[illegible]

5. Na figura abaixo você confere o resultado final. Observe que as linhas do Relacionamento estão pontilhadas, o que indica que o atributo chave estrangeira pode ser nulo.



6. Para finalizar, Salve o Modelo. No menu **File**, clique na opção **Save Model**

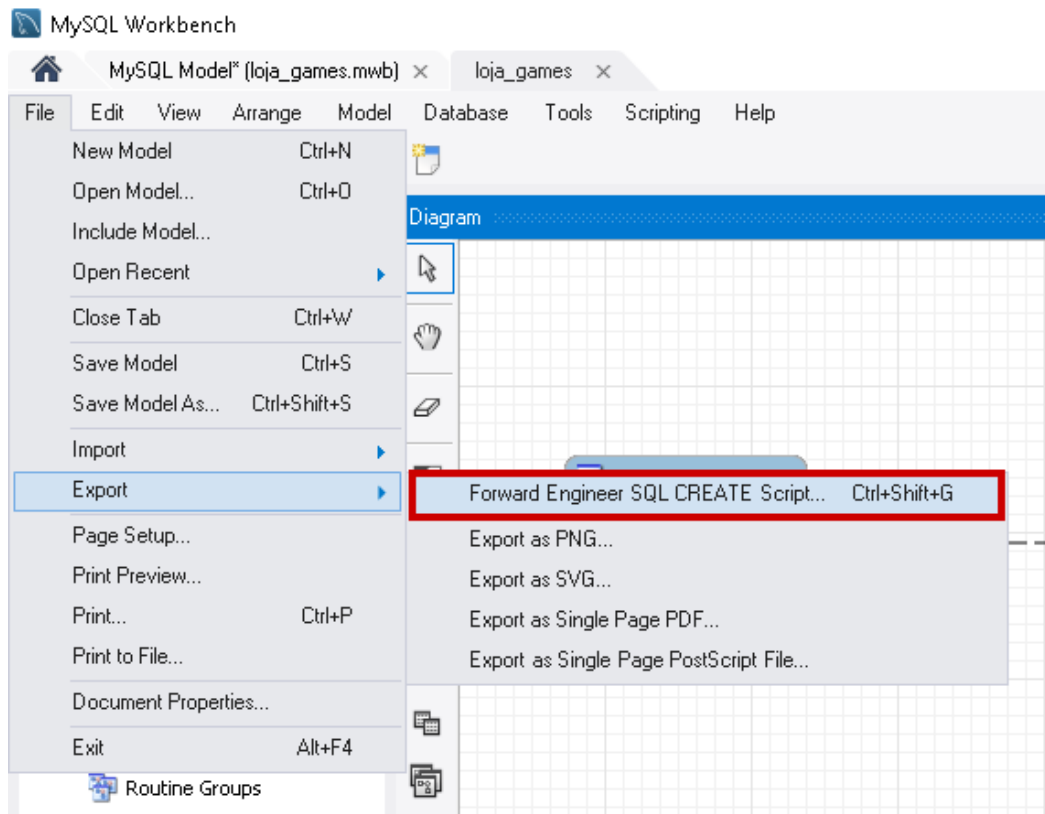


7. Na próxima janela, informe onde você deseja Salvar e clique no botão **Salvar**.

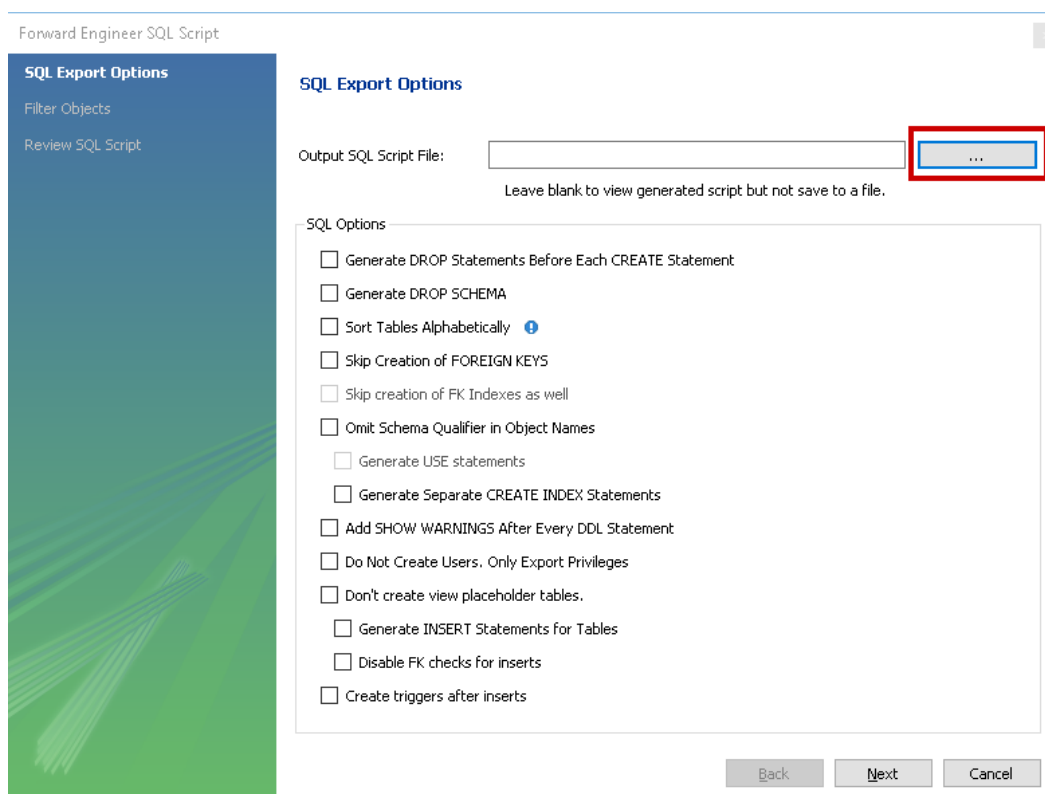


Passo 04 - Exportar como Script SQL

1. No menu **File**, clique na opção **Export** → **Forward Engineer SQL CREATE Script...**



2. No item **Output SQL Script File**, Clique no botão com 3 pontos (...) e informe o **nome do arquivo SQL** e onde deseja Salvar.



3. Clique em **Next** para continuar

Forward Engineer SQL Script

SQL Export Options

Filter Objects
Review SQL Script

SQL Export Options

Output SQL Script File: ...

Leave blank to view generated script but not save to a file.

SQL Options

- ☐ Generate DROP Statements Before Each CREATE Statement
- ☐ Generate DROP SCHEMA
- ☐ Sort Tables Alphabetically ⓘ
- ☐ Skip Creation of FOREIGN KEYS
- ☐ Skip creation of FK Indexes as well
- ☐ Omit Schema Qualifier in Object Names
 - ☐ Generate USE statements
 - ☐ Generate Separate CREATE INDEX Statements
- ☐ Add SHOW WARNINGS After Every DDL Statement
- ☐ Do Not Create Users. Only Export Privileges
- ☐ Don't create view placeholder tables.
 - ☐ Generate INSERT Statements for Tables
 - ☐ Disable FK checks for inserts
 - ☐ Create triggers after inserts

Back Next Cancel

4. Clique em **Next** para continuar

Forward Engineer SQL Script




SQL Export Options

Filter Objects

Review SQL Script

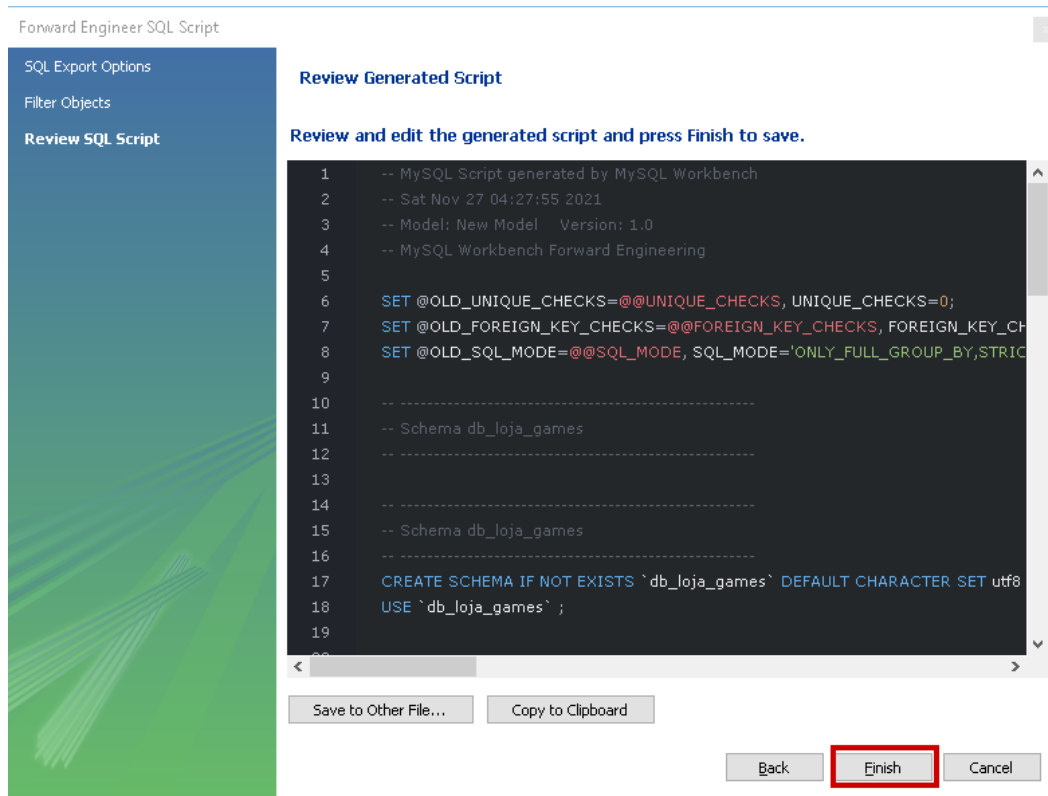
SQL Object Export Filter

To exclude objects of a specific type from the SQL Export, disable the corresponding checkbox. Press Show Filter and add objects or patterns to the ignore list to exclude them from the export.

	<input checked="" type="checkbox"/> Export MySQL Table Objects 3 Total Objects, 3 Selected	Show Filter
	<input type="checkbox"/> Export MySQL View Objects 0 Total Objects, 0 Selected	Show Filter
	<input type="checkbox"/> Export MySQL Routine Objects 0 Total Objects, 0 Selected	Show Filter
	<input type="checkbox"/> Export MySQL Trigger Objects 0 Total Objects, 0 Selected	Show Filter
	<input type="checkbox"/> Export User Objects 0 Total Objects, 0 Selected	Show Filter

Back Next Cancel

5. Clique em **Finish** para concluir

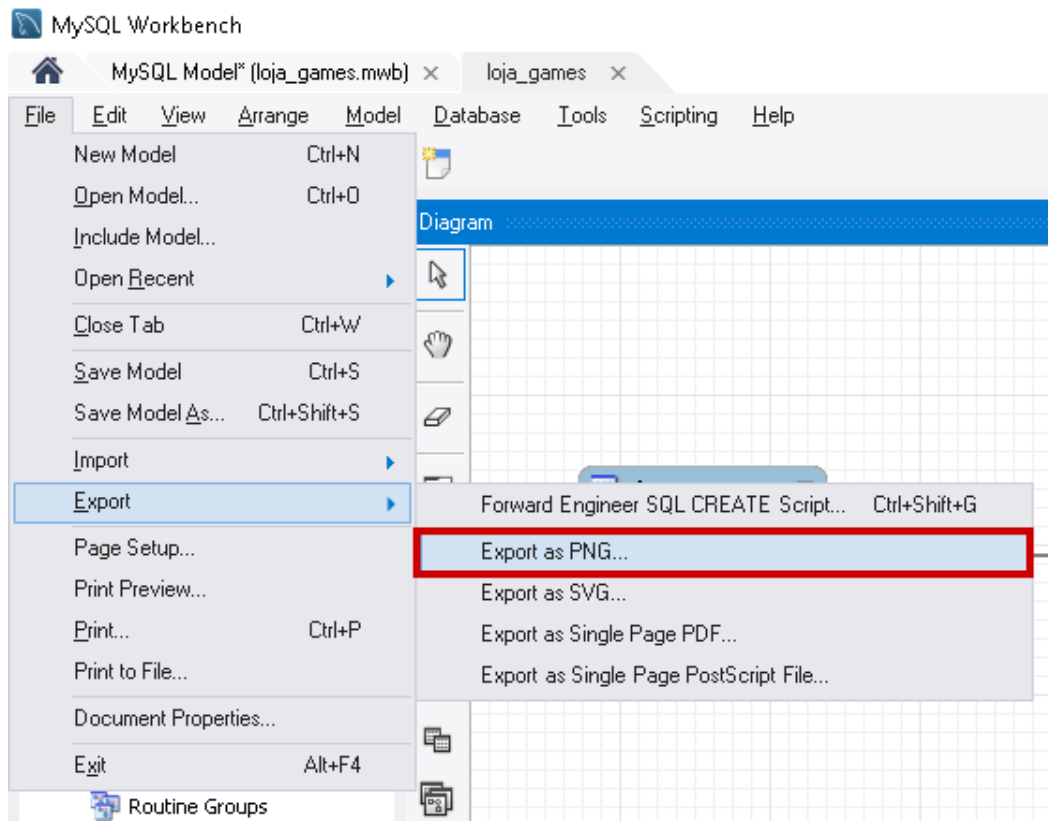


6. O Script gerado será semelhante à imagem abaixo:

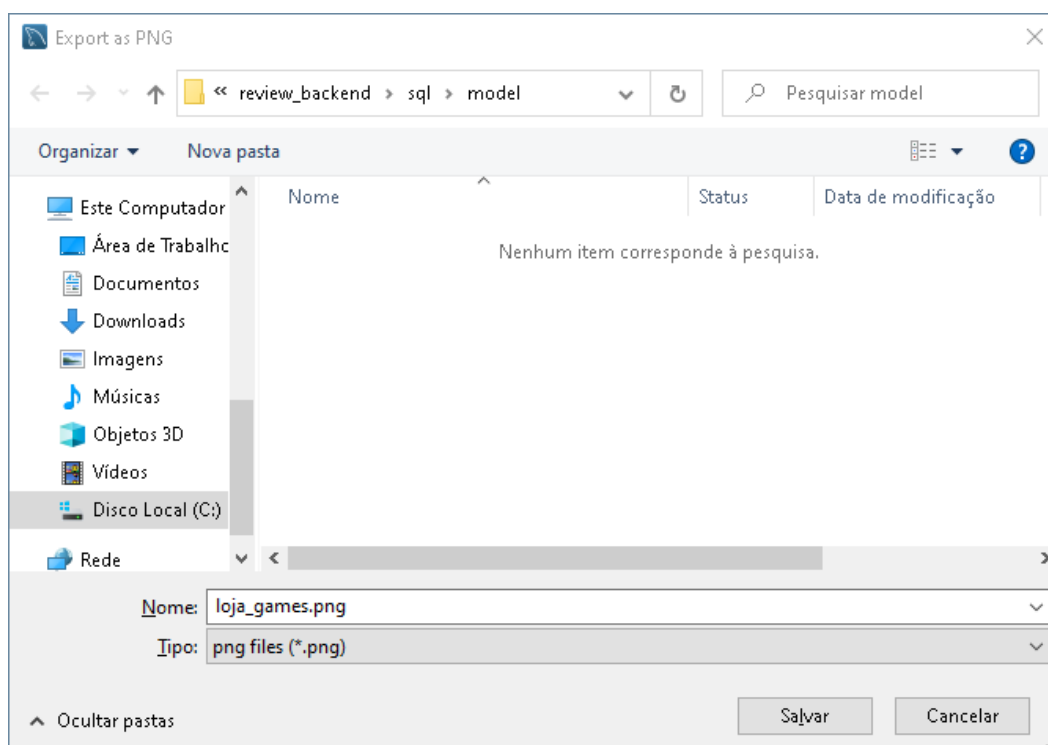
```
13
14
15 -- Schema db_loja_games
16
17 CREATE SCHEMA IF NOT EXISTS `db_loja_games` DEFAULT CHARACTER SET utf8 ;
18 USE `db_loja_games` ;
19
20
21 -- Table `db_loja_games`.`tb_categorias`
22
23 CREATE TABLE IF NOT EXISTS `db_loja_games`.`tb_categorias` (
24   `id` BIGINT NOT NULL AUTO_INCREMENT,
25   `tipo` VARCHAR(255) NOT NULL,
26   PRIMARY KEY (`id`))
27 ENGINE = InnoDB;
28
29
30
31 -- Table `db_loja_games`.`tb_usuarios`
```

Passo 05 - Exportar como Imagem PNG

1. No menu **File**, clique na opção **Export** → **Export as PNG...**

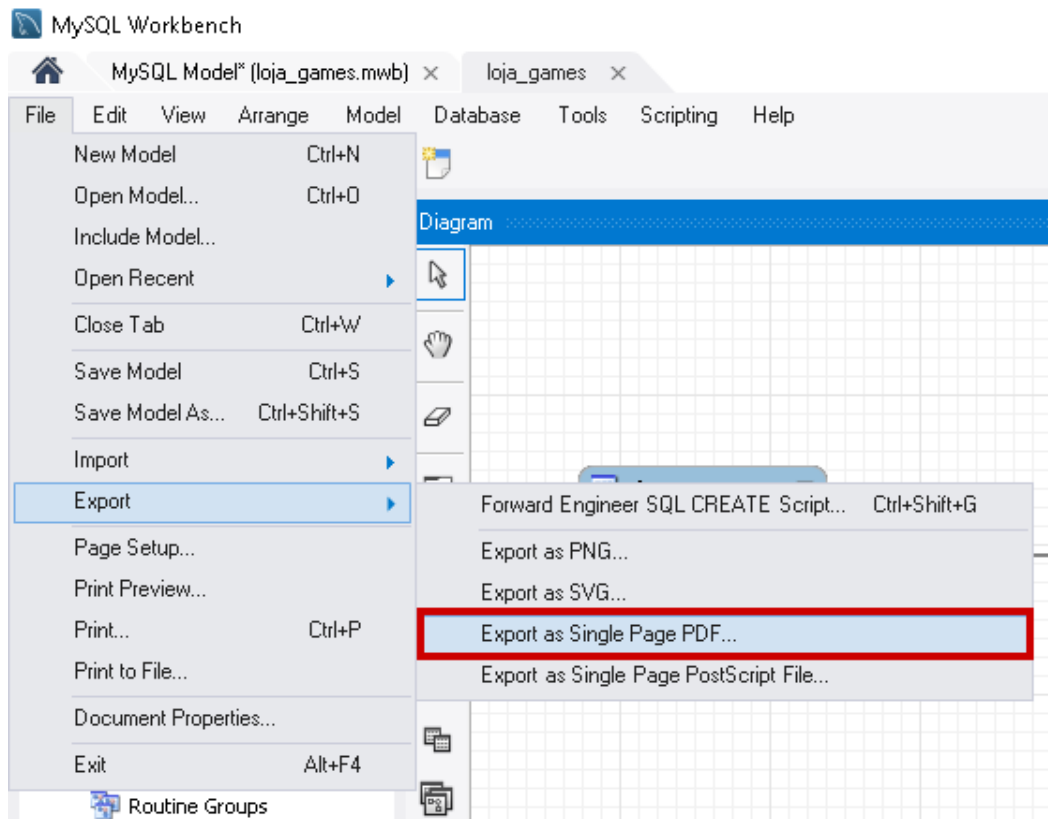


2. Informe o **nome do arquivo PNG** e onde deseja Salvar. Clique no botão **Salvar**.

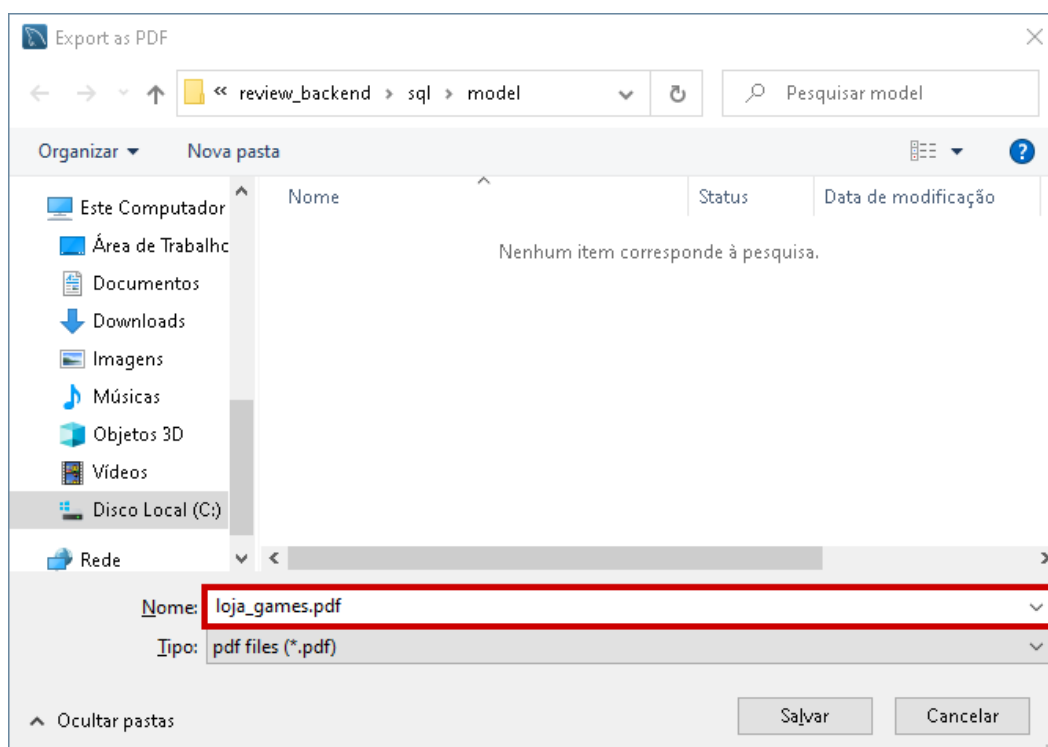


Passo 06 - Exportar como PDF

1. No menu **File**, clique na opção **Export** → **Export as Single Page PDF...**

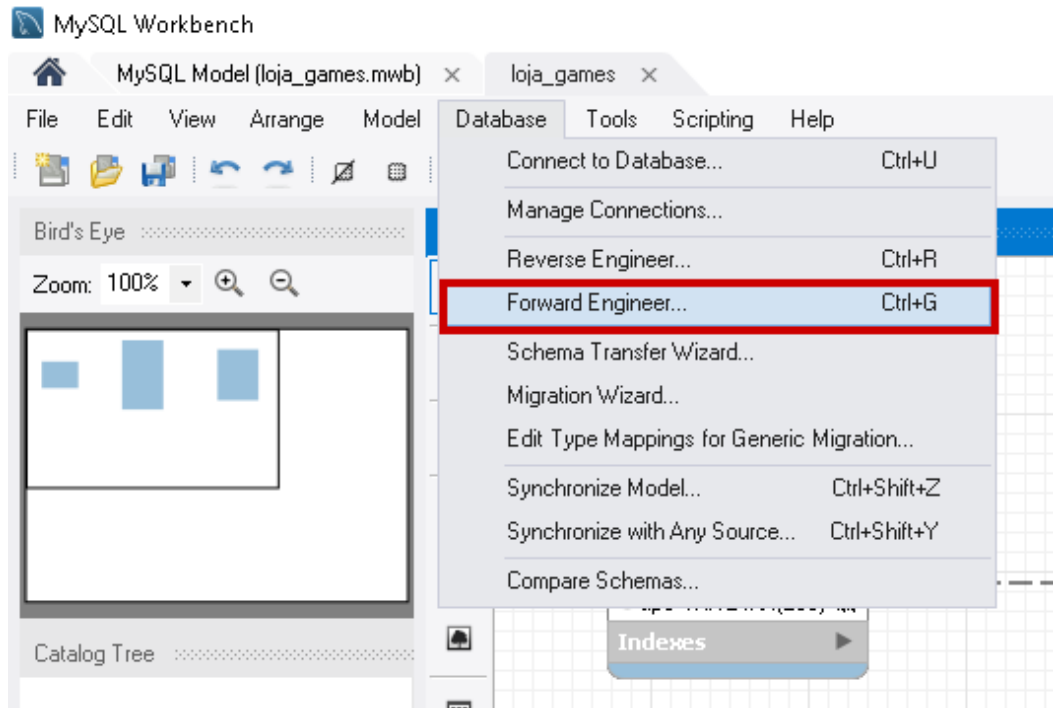


2. Informe o **nome do arquivo PDF** e onde deseja Salvar. Clique no botão **Salvar**.

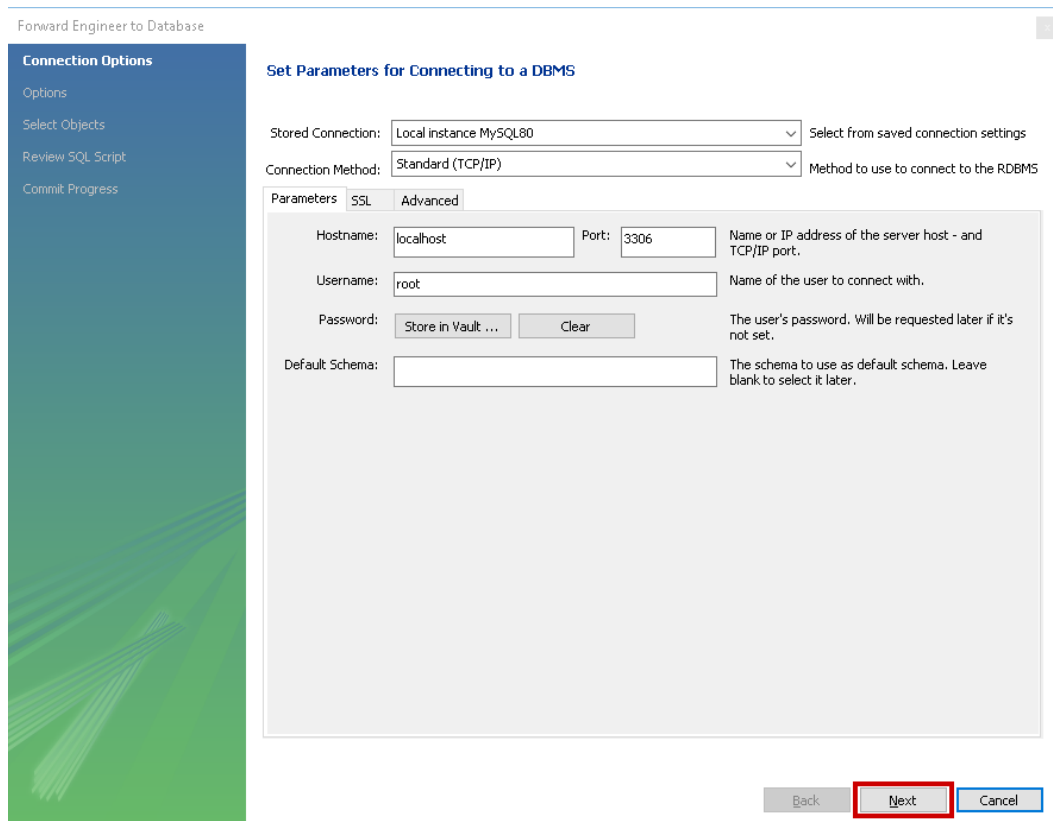


Passo 07 - Exportar para o MySQL

1. No menu **Database**, clique na opção **Forward Engineer...**



2. Como estamos trabalhando localmente (Localhost), clique em **Next** para continuar. Se estivéssemos trabalhando com um Banco de dados na Nuvem seria necessário configurar a conexão nesta etapa.



3. Clique em **Next** para continuar

Forward Engineer to Database

Connection Options

Options

Select Objects

Review SQL Script

Commit Progress

Set Options for Database to be Created

Tables

- ☐ Skip creation of FOREIGN KEYS
- ☐ Skip creation of FK Indexes as well
- ☐ Generate separate CREATE INDEX statements
- ☐ Generate INSERT statements for tables
- ☐ Disable FK checks for INSERTs

Other Objects

- ☐ Don't create view placeholder tables
- ☐ Do not create users. Only create privileges (GRANTS)

Code Generation

- ☐ DROP objects before each CREATE object
- ☒ Generate DROP SCHEMA
- ☐ Omit schema qualifier in object names
- ☐ Generate USE statements
- ☐ Add SHOW WARNINGS after every DDL statement
- ☒ Include model attached scripts

Back Next Cancel

4. Mantenha a primeira opção selecionada (**Criar todas as tabelas do modelo**) e clique em **Next** para continuar

Forward Engineer to Database

Connection Options

Options

Select Objects

Review SQL Script

Commit Progress

Select Objects to Forward Engineer

To exclude objects of a specific type from the SQL Export, disable the corresponding checkbox. Press Show Filter and add objects or patterns to the ignore list to exclude them from the export.

☒ Export MySQL Table Objects
3 Total Objects, 3 Selected Show Filter

☐ Export MySQL View Objects
0 Total Objects, 0 Selected Show Filter

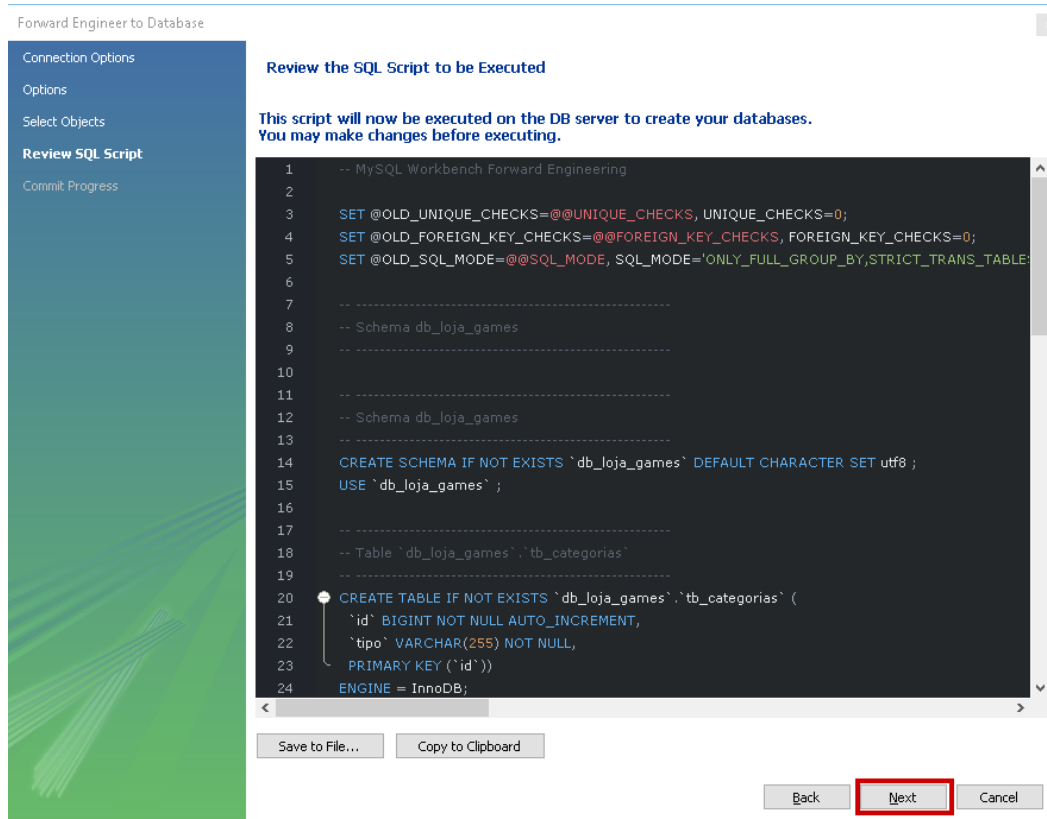
☐ Export MySQL Routine Objects
0 Total Objects, 0 Selected Show Filter

☐ Export MySQL Trigger Objects
0 Total Objects, 0 Selected Show Filter

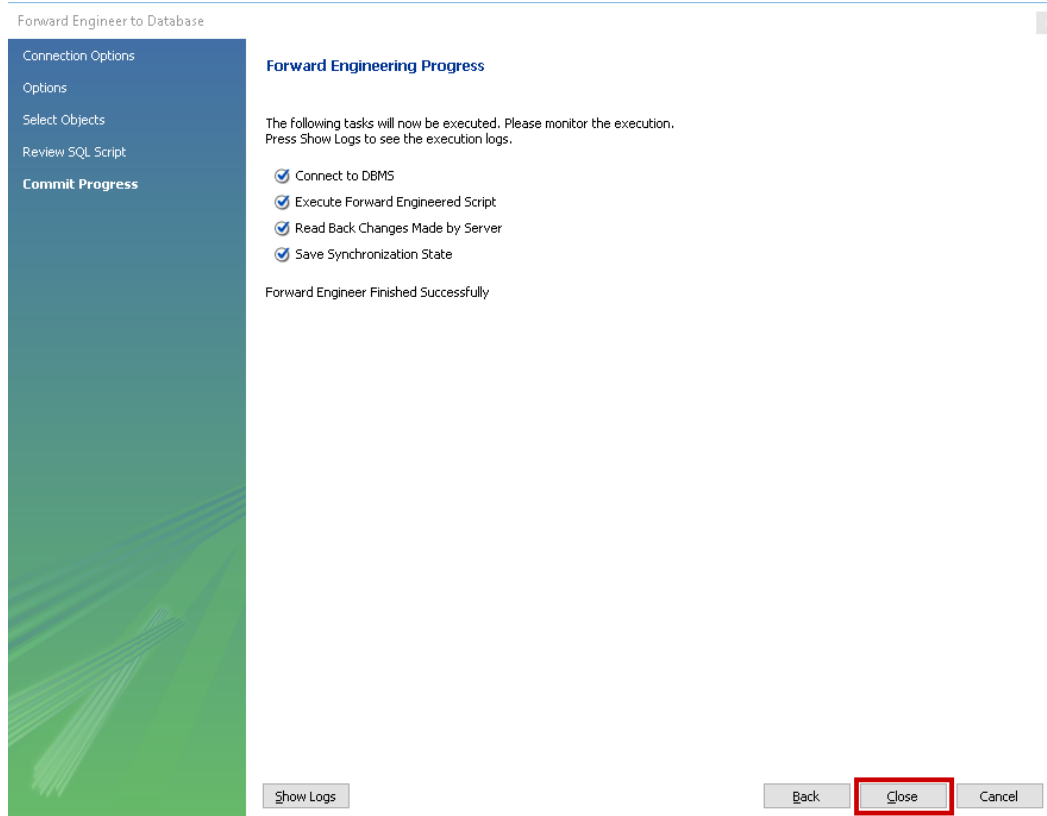
☐ Export User Objects
0 Total Objects, 0 Selected Show Filter

Back Next Cancel

5. Clique em **Next** para continuar



6. Clique em **Close** para concluir



7. Abra a conexão com o MySQL e verifique se o Banco de dados **db_loja_games** com as 3 Tabelas foi criado.

