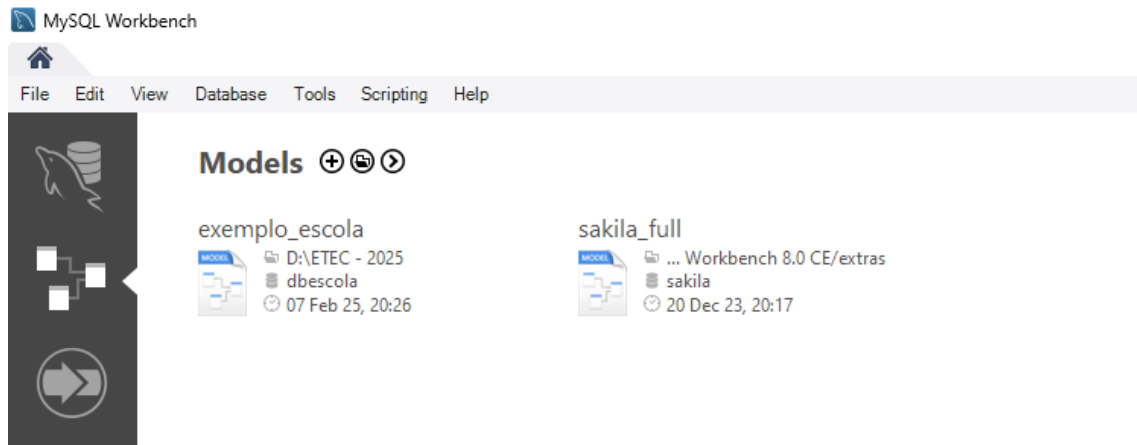
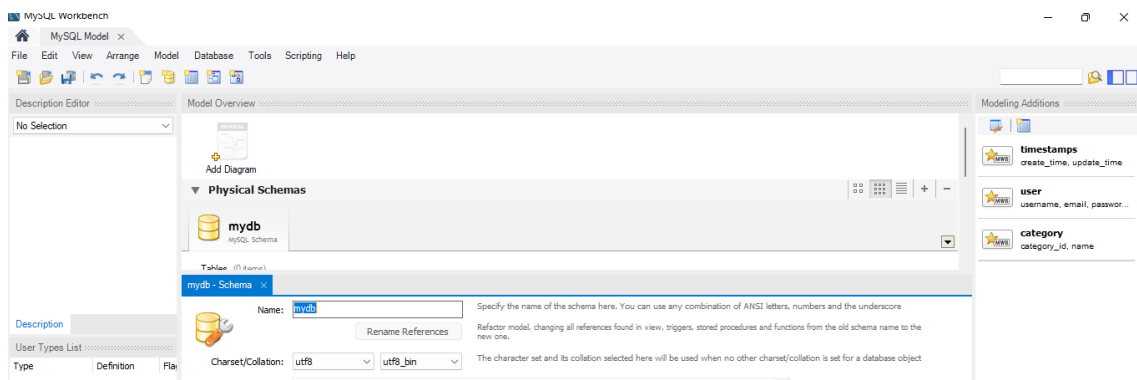


Criando base de Dados usando MySQL Workbench – Diagram

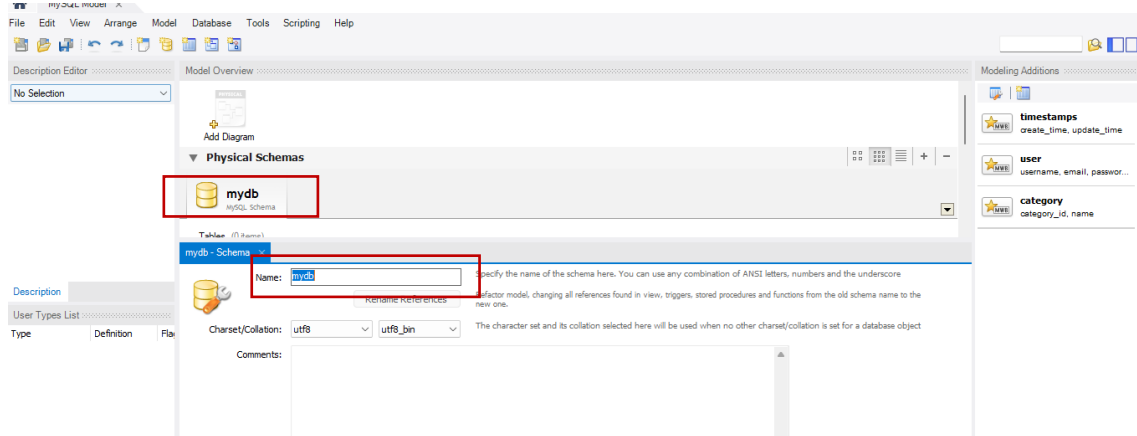
Ao abrir o MySQL Workbench selecione o ícone Diagrama e clique em criar novo diagrama



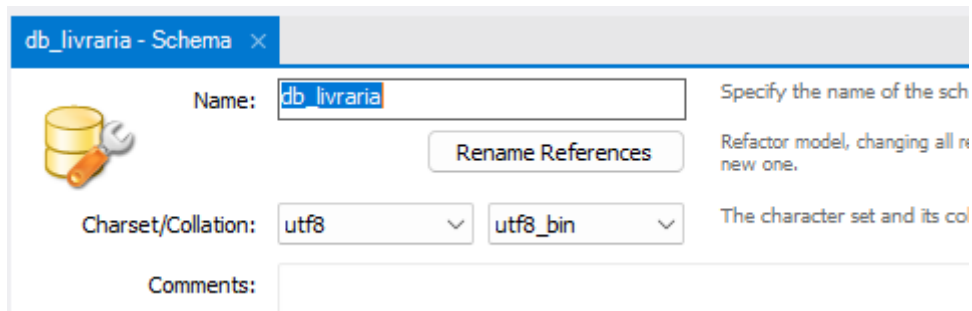
Agora dê um duplo clique no mysql



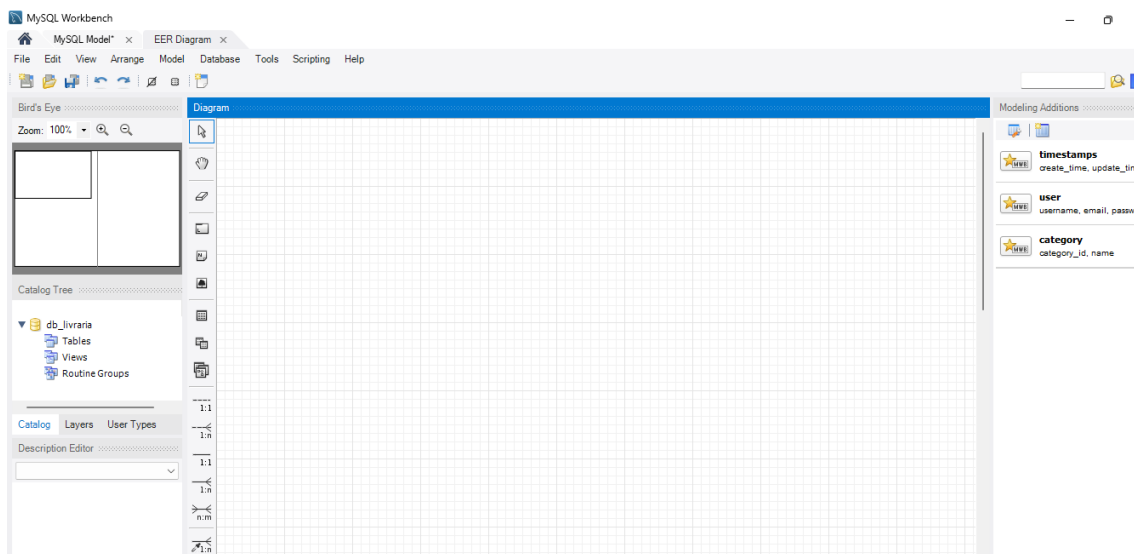
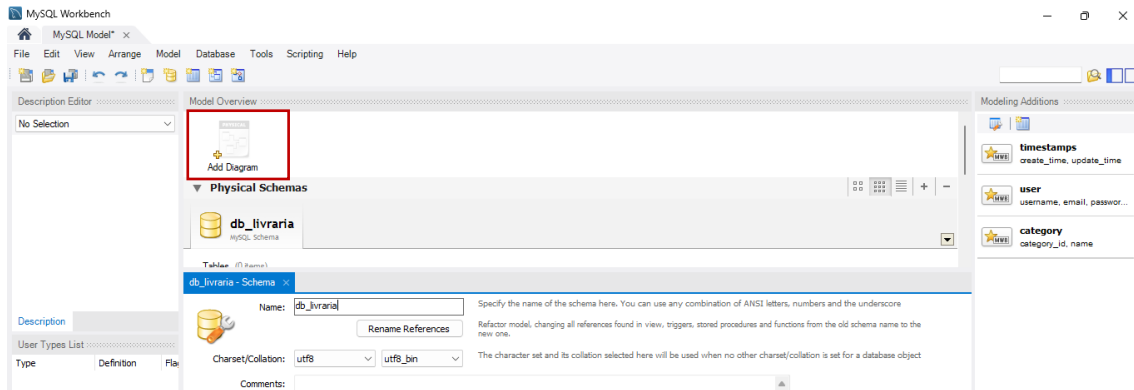
Digite o nome para a base de dados.



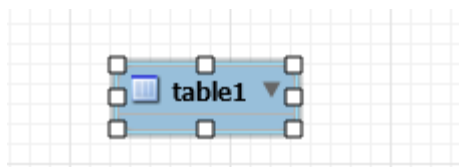
Digite o nome: db_livraria



Clique em ADD Diagram



Clique no ícone tabela para adicionar uma nova tabela ao diagrama



- Dê duplo clique na tabela1 aberta
- No nome coloque: autores
- Nos campos utilize:

- [illegible]

-
- The diagram shows a table named **autores** with the following columns:
- id_autores** INT (Primary Key)
 - nome** VARCHAR(45)
 - dataNascimento** DATE
 - morada** VARCHAR(45)
- Below the columns is a section labeled **Indexes** with a right-pointing arrow.

- Clique no ícone da tabela e adicione ao diagrama
- Com duplo clique altere o nome para: **editora**

- **id_editora**, que será a chave da tabela (PK), o campo não poderá ser nulo (NN) e a cada registo o valor da chave é incrementado + 1 (AI – AutoIncrement)
- **nomeEditora** - Nome da editora que será do tipo VARCHAR, com 45 caracteres

[illegible]

Relações entre tabelas

Numa base de dados relacional, as relações permitem-lhe impedir a existência de dados redundantes. Existem 3 tipos de relações:

1) 1 para 1 Tal como o nome indica uma relação do tipo 1 para 1, é uma relação em que a uma ocorrência da tabela A, corresponde uma e só uma ocorrência da tabela B e vice-versa.

Exemplo: Uma pessoa só pode ter um número de BI, e um BI só pode pertencer a uma pessoa.

Numa relação do tipo um para um, cabe ao “criador” do modelo entidade-relação a escolha de qual a tabela que irá receber a chave estrangeira.



2) 1 para N (em que N significa vários) Uma relação do tipo 1 para n, é uma relação de um para vários, ou seja, entre duas tabelas A e B, a uma ocorrência da tabela A podem corresponder várias ocorrências da tabela B, enquanto que a uma ocorrência da tabela B corresponde só uma da tabela A.

Exemplo: Um leitor pode fazer várias requisições, mas uma requisição só pode ser feita por um leitor, quer isto dizer que entre a tabela Leitor e a tabela Requisições existe uma relação do tipo 1 para n.

A chave principal é adicionada ao lado que tem n, transformando-se assim numa chave estrangeira.



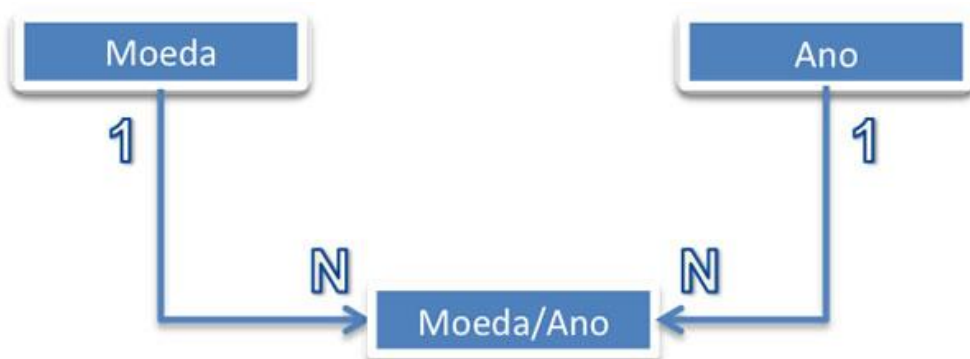
3. N para M (em que N e M significam vários) Uma relação do tipo n para m, é uma relação de vários para vários, ou seja, entre duas tabelas A e B, a várias ocorrências da tabela A podem corresponder várias ocorrências da tabela B, e vice-versa.

Exemplo: Uma moeda pode ser emitida durante vários anos, mas um ano pode emitir várias moedas, quer isto dizer que entre a tabela Moeda e a tabela Ano existe uma relação do tipo n para m.

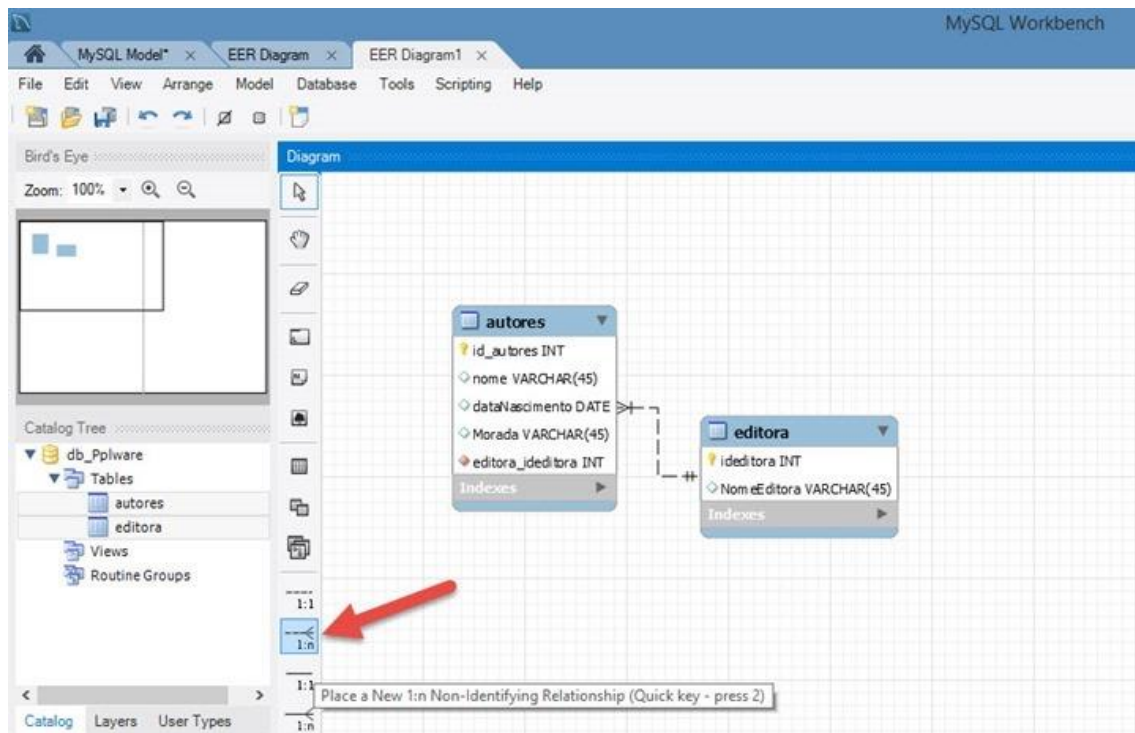
Para toda e qualquer relação do tipo n para m, há que decompor a relação em duas do tipo 1 para n, ou seja, irá ser necessário criar uma nova tabela, com o nome que o “criador” do modelo entidade-relação bem entender, onde a mesma irá conter as chaves principais das tabelas envolvidas, chaves estas que se irão tornar numa chave composta da nova tabela.



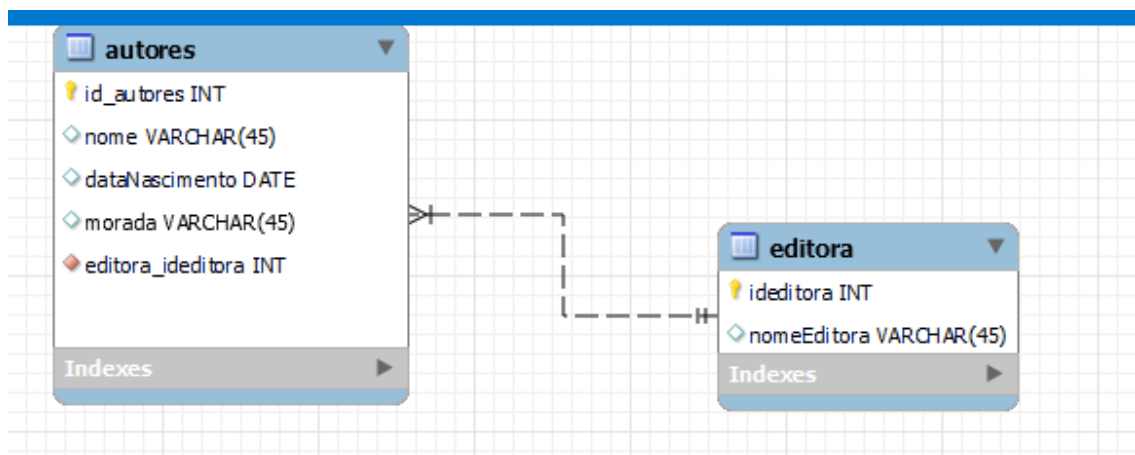
Decomposição final:



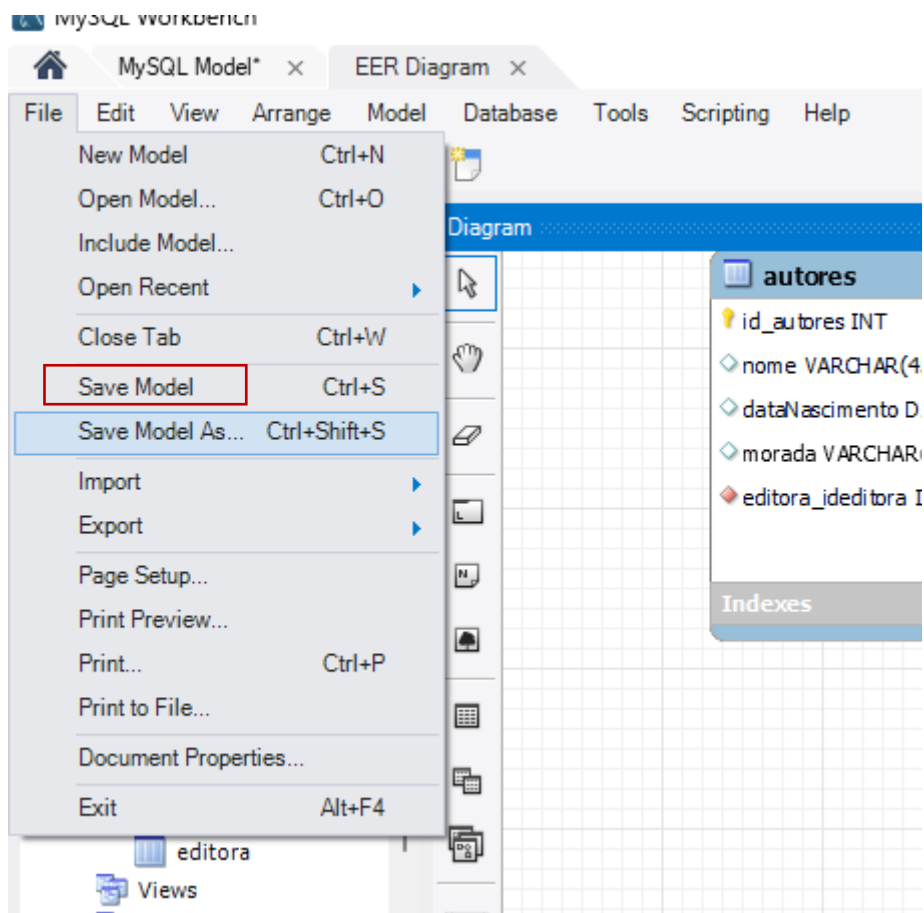
Criando relações entre tabelas no Workbench **Nota:** Como podem ver, nessa barra lateral há dois tipos de botões para definir uma relação de 1:N. O --- significa que a chave primária da tabela com que vamos fazer a relação passa a estrangeira na outra tabela. No caso de escolher o botão com __ essa chave fica também como primária da tabela para onde vamos fazer a relação.



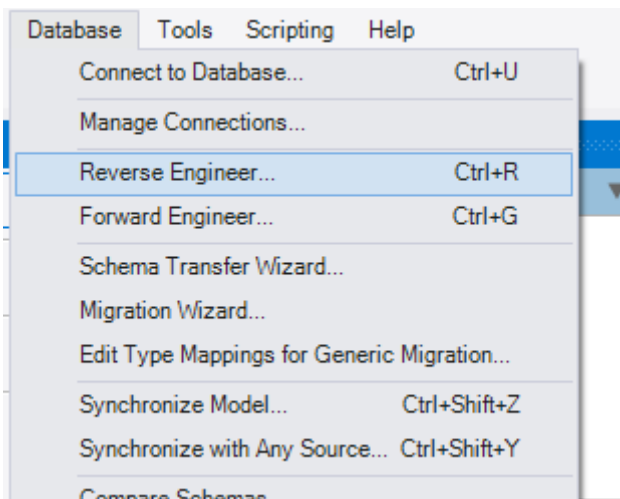
Clique na tabela autores e depois clique sob a tabela editora



Agora salve o diagrama



Para exportar clique no menu Database – Forward Engineer



Forward Engineer to Database

Connection Options

Options

Select Objects

Review SQL Script

Commit Progress

Set Parameters for Connecting to a DBMS

Stored Connection: Select from saved connection settings

Connection Method: Method to use to connect to the RDBMS

Parameters SSL Advanced

Hostname: Port: Name or IP address of the server host - and TCP/IP port.

Username: Name of the user to connect with.

Password: The user's password. Will be requested later if it's not set.

Default Schema: The schema to use as default schema. Leave blank to select it later.

Clique em Next

Na próxima tela aguarde a validação

Forward Engineer to Database

Connection Options

Options

Select Objects

Review SQL Script

Commit Progress

Set Parameters for Connecting to a DBMS

Stored Connection:

conexao

Select from saved connection settings

Connection Method:

Standard (TCP/IP)

Method to use to connect to the RDBMS

Parameters

SSL

Advanced

Hostname:

127.0.0.1

 Port:

3306

Name or IP address of the server host - and TCP/IP port.

Username:

root

Name of the user to connect with.

Password:

Store in Vault ...

Clear

The user's password. Will be requested later if it's not set.

Default Schema:

The schema to use as default schema. Leave blank to select it later.

Back

Next

Cancel

Clique em Next

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

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

2 Total Objects, 2 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

Connection Options

Options

Select Objects

Review SQL Script

Commit Progress

Review the SQL Script to be Executed

This script will now be executed on the DB server to create your databases.
You may make changes before executing.

```
1      -- MySQL Workbench Forward Engineering
2
3      SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
4      SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
5      SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES';
6
7      -----
8      -- Schema db_livraria
9      -----
10
11     -----
12     -- Schema db_livraria
13     -----
14     CREATE SCHEMA IF NOT EXISTS `db_livraria` DEFAULT CHARACTER SET utf8 ;
15     USE `db_livraria` ;
16
17     -----
18     -- Table `db_livraria`.`editora`
19     -----
20     CREATE TABLE IF NOT EXISTS `db_livraria`.`editora` (
21       `ideditora` INT NOT NULL AUTO_INCREMENT,
22       `nomeEditora` VARCHAR(45) NULL,
23       PRIMARY KEY (`ideditora`))
24     ENGINE = InnoDB;
```

Save to File...

Copy to Clipboard

Back

Next

Cancel

Atividade

- Acrescente as demais tabelas
- **Livros**
 - Para armazenar informações sobre os livros escritos pelos autores.
 - Exemplo de colunas:
 - id_livro (INT, PK)
 - titulo (VARCHAR)
 - ano_publicacao (YEAR)
 - genero (VARCHAR)
 - editora_id (INT, FK para editora)
- **Autores_Livros**
 - Uma tabela de relacionamento muitos-para-muitos entre autores e livros.
 - Exemplo de colunas:
 - id_autor (INT, FK para autores)
 - id_livro (INT, FK para livros)
- **Gêneros**
 - Para categorizar os livros por gênero.
 - Exemplo de colunas:
 - id_genero (INT, PK)
 - nome_genero (VARCHAR)
- **Usuários** (se for um sistema de biblioteca ou vendas)
 - Para gerenciar usuários que podem interagir com os livros.
 - Exemplo de colunas:
 - id_usuario (INT, PK)
 - nome (VARCHAR)
 - email (VARCHAR)
 - senha (VARCHAR)
- **Empréstimos** (se for uma biblioteca)
 - Para rastrear empréstimos de livros.
 - Exemplo de colunas:
 - id_emprestimo (INT, PK)
 - id_usuario (INT, FK para usuarios)
 - id_livro (INT, FK para livros)
 - data_emprestimo (DATE)
 - data_devolucao (DATE)

- **Vendas** (se for um sistema de e-commerce de livros)
 - Para registrar a compra de livros.
 - Exemplo de colunas:
 - id_venda (INT, PK)
 - id_usuario (INT, FK para usuarios)
 - id_livro (INT, FK para livros)
 - quantidade (INT)
 - preco_total (DECIMAL)
 - data_venda (DATE)