



### MAPA – Material de Avaliação Prática da Aprendizagem

<b>Acadêmico: Douglas Marcelo Monquero</b>	<b>R.A. 23343540-5</b>
<b>Curso: Engenharia de Software</b>	
<b>Disciplina: Banco de Dados</b>	
<b>Valor da atividade: 3,50</b>	<b>Prazo: 15/09/2024</b>

#### Instruções para Realização da Atividade

- 1) Todos os campos acima deverão ser devidamente preenchidos;
- 2) É obrigatória a utilização deste formulário para a realização do MAPA;
- 3) Esta é uma atividade individual. Caso identificado cópia de colegas, o trabalho de ambos sofrerá decréscimo de nota;
- 4) Utilizando este formulário, realize sua atividade, salve em seu computador, renomeie e envie em forma de anexo no campo de resposta da atividade MAPA;
- 5) Formatação exigida para esta atividade: documento Word, Fonte Arial ou Times New Roman tamanho 12, Espaçamento entre linhas 1,5, texto justificado;
- 6) Ao utilizar quaisquer materiais de pesquisa referência conforme as normas da ABNT;
- 7) No campo “Material da disciplina”, no ambiente virtual da disciplina, você encontrará orientações importantes para elaboração desta atividade. Confira!
- 8) Critérios de avaliação: Utilização do template; atendimento ao Tema; Constituição dos argumentos e organização das Ideias; Correção Gramatical e atendimento às normas ABNT.
- 9) Procure argumentar claramente e objetiva, conforme o conteúdo da disciplina.

**Em caso de dúvidas, entre em contato com seu Professor Mediador.**

**Bons estudos!**

## 1- Introdução

Segundo o nosso livro da disciplina, Banco de Dados, pagina 46, durante nossa disciplina estaremos utilizando a notação de Peter Chen (1990), essa notação foi criada em 1976 pelo Doutor Peter Pin-Shan Chen conhecido por ser o criador do modelo entidade relacionamento.

C. J. Date nos ensina que “a base da moderna tecnologia de bancos de dados é, sem dúvida, o modelo relacional: é essa a base que faz da área uma ciência.” (Introdução ao Sistema de Banco de Dados, pagina 93). Date continua sua ministração citando três aspectos principais do modelo relacional que são a estrutura dos dados, a manipulação dos dados e a integridade dos dados.

O professor especialista Vitor de Marqui Pedroso complementa que o modelo relacional tem uma base sólida formal construída sob a teoria dos conjuntos e trata-se de um modelo de tabelas e alguns conceitos, sendo que esse modelo nos permite construir uma representação clara e simples dos dados do mundo real com objetos que denominamos entidades ou conjuntos de entidades.

## 2 - Criar o Diagrama Entidade Relacionamento (DER)

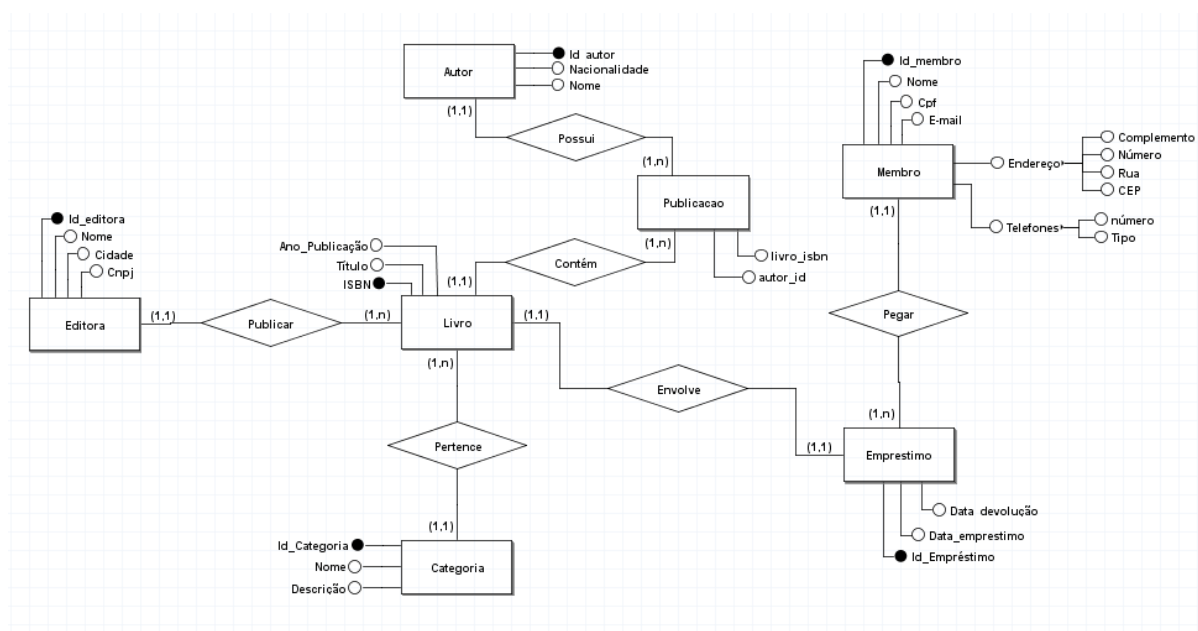


Figura 01 – DER (Diagrama Entidade Relacionamento)

Seguimos agora para o Modelo Entidade Relacionamento (MER), que é o principal padrão utilizado atualmente para a modelagem de dados pois nele está contido conceitos importantes do universo do Banco de Dados, dentre eles entidade, relacionamento e atributos.

Uma vez conhecido o conceito de modelo relacional e construído o Diagrama Entidade Relacionamento o passo seguinte é fazer seu mapeamento para o Modelo Relacional. Esse padrão de modelo foi introduzido por Codd em 1970, e é implementada pela maioria dos Sistemas de Gerenciamento de Banco de Dados comerciais pois uma de suas características primordiais é a sua eficiência.

Esse modelo é uma representação por tabelas, linhas, colunas e domínios/tipos dentro de uma única estrutura de dados.

### 3 - Criar o Modelo Entidade Relacionamento (MER)

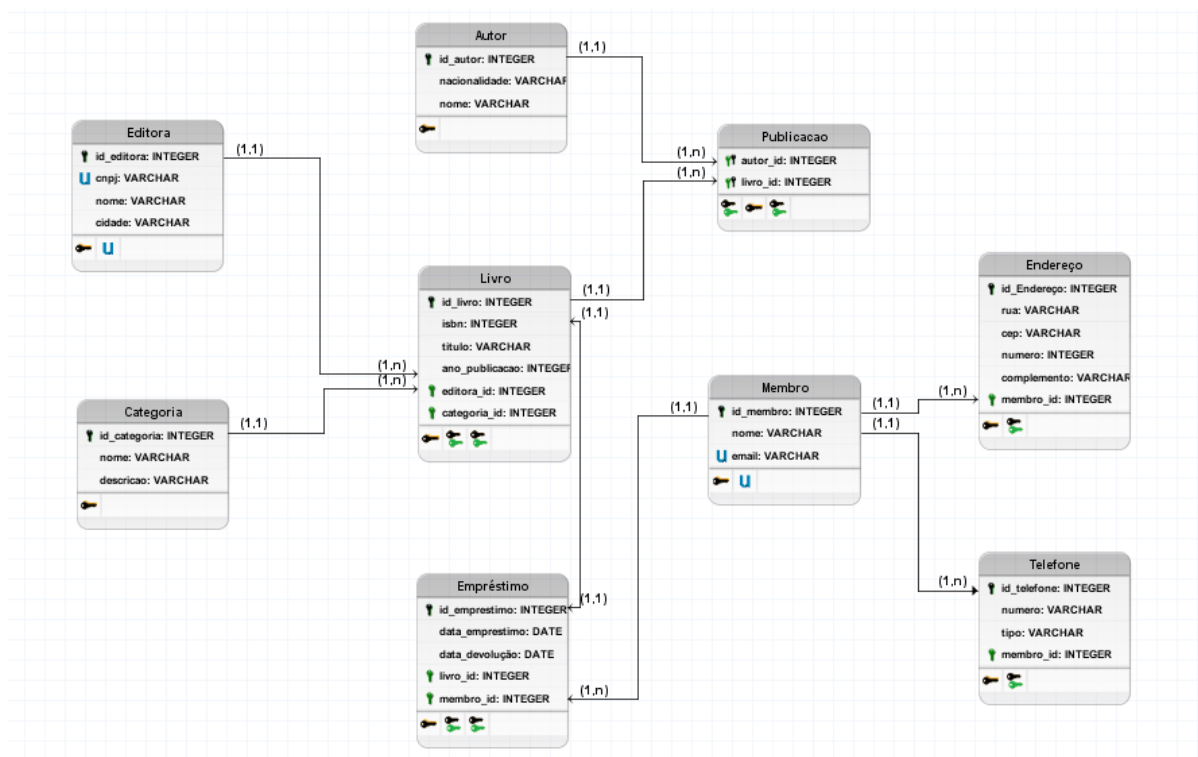


Figura 02 – MER (Modelo Entidade Relacionamento)



#### 4 - Modelo Lógico Normalizado – Representação Textual do MER

1. Autor (id\_autor, nacionalidade, nome).
2. Categoria (id\_categoria, nome, descricao ).
3. Editora (id\_editora, cnpj, nome, cidade).
4. Livro (id\_livro, isbn, titulo, ano\_publicacao, editora\_id, categoria\_id).  
**editora\_id referência Editora(id\_editora)**  
**categoria\_id referência Categoria(id\_categoria)**
5. Publicacao (autor\_id, livro\_id).  
**autor\_id referência Livro(id\_livro)**  
**livro\_id referência Autor(id\_autor)**
6. Membro (id\_membro, nome, email).
7. Emprestimo (id\_emprestimo, membro\_id, livro\_id, data\_emprestimo, data\_devolucao)  
**livro\_id referência Livro(id\_livro)**  
**membro\_id referência Membro(id\_membro)**
8. Endereco (id\_endereco, rua, cep, numero, complemento, membro\_id).  
**membro\_id referência Membro(id\_membro)**
9. Telefone (id\_telefone, numero, tipo, membro\_id).  
**membro\_id referência Membro(id\_membro)**

#### 5 - Criar o projeto físico, por meio de código SQL

##### 5.1 - Conversao direta feita pelo BRModelo:

/\* MAPA\_Lógico: \*/

```
CREATE TABLE Livro (  
    id_livro INTEGER PRIMARY KEY,  
    isbn INTEGER,  
    titulo VARCHAR,  
    ano_publicacao INTEGER,  
    editora_id INTEGER,  
    categoria_id INTEGER
```



);

```
CREATE TABLE Autor (  
    id_autor INTEGER PRIMARY KEY,  
    nacionalidade VARCHAR,  
    nome VARCHAR  
);
```

```
CREATE TABLE Editora (  
    id_editora INTEGER PRIMARY KEY,  
    cnpj VARCHAR UNIQUE,  
    nome VARCHAR,  
    cidade VARCHAR  
);
```

```
CREATE TABLE Categoria (  
    id_categoria INTEGER PRIMARY KEY,  
    nome VARCHAR,  
    descricao VARCHAR  
);
```

```
CREATE TABLE Empréstimo (  
    id_emprestimo INTEGER PRIMARY KEY,  
    data_emprestimo DATE,  
    data_devolução DATE,  
    livro_id INTEGER,  
    membro_id INTEGER  
);
```

```
CREATE TABLE Membro (  
    id_membro INTEGER PRIMARY KEY,  
    nome VARCHAR,
```

```
email VARCHAR UNIQUE  
);
```

```
CREATE TABLE Publicacao (  
    autor_id INTEGER,  
    livro_id INTEGER,  
    PRIMARY KEY (autor_id, livro_id)  
);
```

```
CREATE TABLE Endereço (  
    id_Endereço INTEGER PRIMARY KEY,  
    rua VARCHAR,  
    cep VARCHAR,  
    numero INTEGER,  
    complemento VARCHAR,  
    membro_id INTEGER  
);
```

```
CREATE TABLE Telefone (  
    id_telefone INTEGER PRIMARY KEY,  
    numero VARCHAR,  
    tipo VARCHAR,  
    membro_id INTEGER  
);
```

```
ALTER TABLE Livro ADD CONSTRAINT FK_Livro_2  
    FOREIGN KEY (editora_id)  
    REFERENCES Editora (id_editora);
```

```
ALTER TABLE Livro ADD CONSTRAINT FK_Livro_3  
    FOREIGN KEY (categoria_id)  
    REFERENCES Categoria (id_categoria);
```



```
ALTER TABLE Empréstimo ADD CONSTRAINT FK_Empréstimo_2
FOREIGN KEY (livro_id)
REFERENCES Livro (id_livro);
```

```
ALTER TABLE Empréstimo ADD CONSTRAINT FK_Empréstimo_3
FOREIGN KEY (membro_id)
REFERENCES Membro (id_membro);
```

```
ALTER TABLE Publicacao ADD CONSTRAINT FK_Publicacao_1
FOREIGN KEY (livro_id)
REFERENCES Livro (id_livro);
```

```
ALTER TABLE Publicacao ADD CONSTRAINT FK_Publicacao_3
FOREIGN KEY (autor_id)
REFERENCES Autor (id_autor);
```

```
ALTER TABLE Endereço ADD CONSTRAINT FK_Endereço_2
FOREIGN KEY (membro_id)
REFERENCES Membro (id_membro);
```

```
ALTER TABLE Telefone ADD CONSTRAINT FK_Telefone_2
FOREIGN KEY (membro_id)
REFERENCES Membro (id_membro);
```

## **5. 2 Código desenvolvido pelo autor no Programa Workbench:**

#Criacao de Schema e Tabelas

```
CREATE SCHEMA if not exists biblioteca;
USE biblioteca;
```

#Drop Schema biblioteca



#Autor (id\_autor, nacionalidade, nome)

#DROP TABLE Autor

```
CREATE TABLE if not exists Autor(  
    id_autor INT AUTO_INCREMENT,  
    nome VARCHAR(30) NOT NULL,  
    nacionalidade VARCHAR(45) NOT NULL,  
    PRIMARY KEY(id_autor)  
);
```

#Editora (id\_editora, cnpj, nome, cidade)

#DROP TABLE Editora

```
CREATE TABLE if not exists Editora(  
    id_editora INT AUTO_INCREMENT,  
    cnpj VARCHAR(30) NOT NULL,  
    nome VARCHAR(30) NOT NULL,  
    cidade VARCHAR(45) NOT NULL,  
    PRIMARY KEY(id_editora)  
);
```

#Categoria (id\_categoria, nome, descricao)

#DROP TABLE Categoria

```
CREATE TABLE if not exists Categoria(  
    id_categoria INT AUTO_INCREMENT,  
    nome VARCHAR(30) NOT NULL,  
    descricao VARCHAR(150) NOT NULL,  
    PRIMARY KEY(id_categoria)  
);
```

#Livro (id\_livro, isbn, titulo, ano\_publicacao, editora\_id, categoria\_id)

#DROP TABLE Livro

```
CREATE TABLE if not exists Livro(  
    id_livro INT AUTO_INCREMENT,
```





```
isbn INT NOT NULL,  
titulo VARCHAR(50) NOT NULL,  
ano_publicacao YEAR NOT NULL,  
editora_id INT NOT NULL,  
categoria_id INT NOT NULL,  
PRIMARY KEY(id_livro),  
FOREIGN KEY(editora_id) REFERENCES Editora(id_editora) ON DELETE  
CASCADE ON UPDATE CASCADE,  
FOREIGN KEY(categoria_id) REFERENCES Categoria(id_categoria) ON DELETE  
CASCADE ON UPDATE CASCADE  
);
```

#Publicacao (autor\_id, livro\_id)

#DROP TABLE Publicacao

```
CREATE TABLE if not exists Publicacao(  
    autor_id INT,  
    livro_id INT,  
    PRIMARY KEY(autor_id, livro_id),  
    FOREIGN KEY(autor_id) REFERENCES autor(id_autor) ON DELETE CASCADE  
ON UPDATE CASCADE,  
    FOREIGN KEY(livro_id) REFERENCES livro(id_livro) ON DELETE CASCADE  
ON UPDATE CASCADE  
);
```

#Membro (id\_membro, nome, email)

#DROP TABLE Membro

```
CREATE TABLE if not exists Membro(  
    id_membro INT AUTO_INCREMENT,  
    nome VARCHAR(50) NOT NULL,  
    email VARCHAR(50) NOT NULL,  
    PRIMARY KEY(id_membro)  
);
```



#Emprestimo (id\_emprestimo, membro\_id, livro\_id, data\_emprestimo, data\_devolucao)

#DROP TABLE Emprestimo

```
CREATE TABLE if not exists Emprestimo(  
    id_emprestimo INT AUTO_INCREMENT,  
    membro_id INT NOT NULL,  
    livro_id INT NOT NULL,  
    data_emprestimo DATE NOT NULL,  
    data_devolucao DATE NOT NULL,  
    PRIMARY KEY (Id_emprestimo),  
    FOREIGN KEY(livro_id) REFERENCES livro(id_livro) ON DELETE CASCADE  
    ON UPDATE CASCADE,  
    FOREIGN KEY(membro_id) REFERENCES Membro(id_membro) ON DELETE  
    CASCADE ON UPDATE CASCADE  
);
```

#Endereco (id\_endereco, membro\_id, rua, cep, numero, complemento).

#DROP TABLE Endereco

```
CREATE TABLE if not exists Endereco(  
    id_endereco INT AUTO_INCREMENT,  
    membro_id INT NOT NULL,  
    rua VARCHAR(30) NOT NULL,  
    cep VARCHAR(30) NOT NULL,  
    numero INT NOT NULL,  
    complemento VARCHAR(45) NOT NULL,  
    PRIMARY KEY(id_endereco),  
    FOREIGN KEY(membro_id) REFERENCES membro(id_membro) ON DELETE  
    CASCADE ON UPDATE CASCADE  
);
```

#Telefone (id\_telefone, membro\_id, numero, tipo)

#DROP TABLE Telefone

```
CREATE TABLE if not exists Telefone(
    id_telefone INT AUTO_INCREMENT,
    membro_id INT NOT NULL,
    numero VARCHAR(20) NOT NULL,
    tipo VARCHAR(15) NOT NULL,
    PRIMARY KEY(id_telefone),
    FOREIGN KEY(membro_id) REFERENCES membro(id_membro) ON DELETE
    CASCADE ON UPDATE CASCADE
);
```

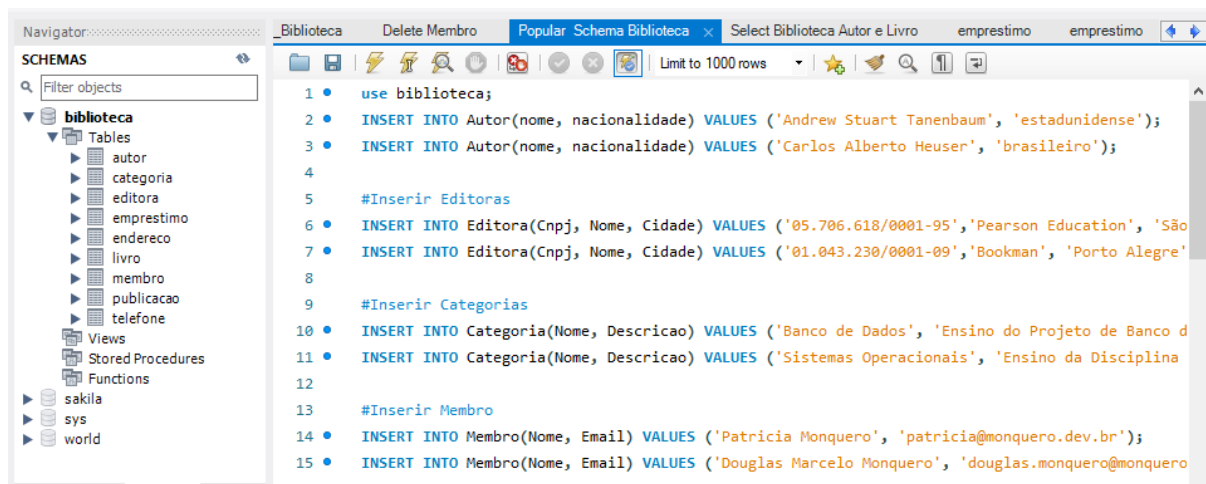


Figura 03. Visão do Banco de Dados e Código Popular BD

## 6 - Código teste para popular as tabelas da biblioteca

```
use biblioteca;
INSERT INTO Autor(nome, nacionalidade) VALUES ('Andrew Stuart Tanenbaum',
'estadunidense');
INSERT INTO Autor(nome, nacionalidade) VALUES ('Carlos Alberto Heuser', 'brasileiro');

#Inserir Editoras
INSERT INTO Editora(Cnpj, Nome, Cidade) VALUES ('05.706.618/0001-95','Pearson
Education', 'São Paulo');
```

```
INSERT INTO Editora(Cnpj, Nome, Cidade) VALUES ('01.043.230/0001-09','Bookman',
'Porto Alegre');
```

#### #Inserir Categorias

```
INSERT INTO Categoria(Nome, Descricao) VALUES ('Banco de Dados', 'Ensino do Projeto
de Banco de Dados - Conceitual e Lógico');
```

```
INSERT INTO Categoria(Nome, Descricao) VALUES ('Sistemas Operacionais', 'Ensino da
Disciplina de Sistemas Operacionais');
```

#### #Inserir Membro

```
INSERT INTO Membro(Nome, Email) VALUES ('Patricia Monquero',
'patricia@monquero.dev.br');
```

```
INSERT INTO Membro(Nome, Email) VALUES ('Douglas Marcelo Monquero',
'douglas.monquero@monquero.dev.br');
```

#### #Inserir Endereço

```
INSERT INTO Endereco(Rua, Cep, Numero, Complemento, Membro_id) VALUES ('Rua
Monsenor Tanaka', '87010-255', 300, 'Bloco 03 - apt 21', 1);
```

```
INSERT INTO Endereco(Rua, Cep, Numero, Complemento, Membro_id) VALUES ('Rua dos
Ipes', '87020-080', 412, 'Casa', 2);
```

#### #Inserir Telefone

```
INSERT INTO Telefone(Numero, Tipo, Membro_id) VALUES (999013434, 'Celular', 2);
```

```
INSERT INTO Telefone(Numero, Tipo, Membro_id) VALUES (30253434, 'Residencial', 2);
```

```
INSERT INTO Telefone(Numero, Tipo, Membro_id) VALUES (99823047, 'Celular', 1);
```

#### #Inserir Livro

```
INSERT INTO Livro(ISBN, Titulo, Ano_publicacao, Editora_id, Categoria_id) VALUES
(00515, 'Sistemas Operacionais Modernos', 2010, 1, 2);
```

```
INSERT INTO Livro(ISBN, Titulo, Ano_publicacao, Editora_id, Categoria_id) VALUES
(00574, 'Projeto de Banco de Dados', 2010, 2, 1);
```

### #Inserir Emprestimo

```
INSERT INTO Emprestimo(data_emprestimo, data_devolucao, Livro_id, Membro_id)
VALUES ('2024-09-22', '2024-10-07', 1, 1);
```

```
INSERT INTO Emprestimo(data_emprestimo, data_devolucao, Livro_id, Membro_id)
VALUES ('2024-09-22', '2024-10-07', 2, 1);
```

```
INSERT INTO Emprestimo(data_emprestimo, data_devolucao, Livro_id, Membro_id)
VALUES ('2024-08-21', '2024-09-15', 1, 2);
```

```
INSERT INTO Emprestimo(data_emprestimo, data_devolucao, Livro_id, Membro_id)
VALUES ('2024-08-21', '2024-09-15', 2, 2);
```

### #Inserir Publicacao

```
INSERT INTO publicacao(Autor_id, livro_id) VALUES (1, 1);
```

```
INSERT INTO publicacao(Autor_id, livro_id) VALUES (2, 2);
```

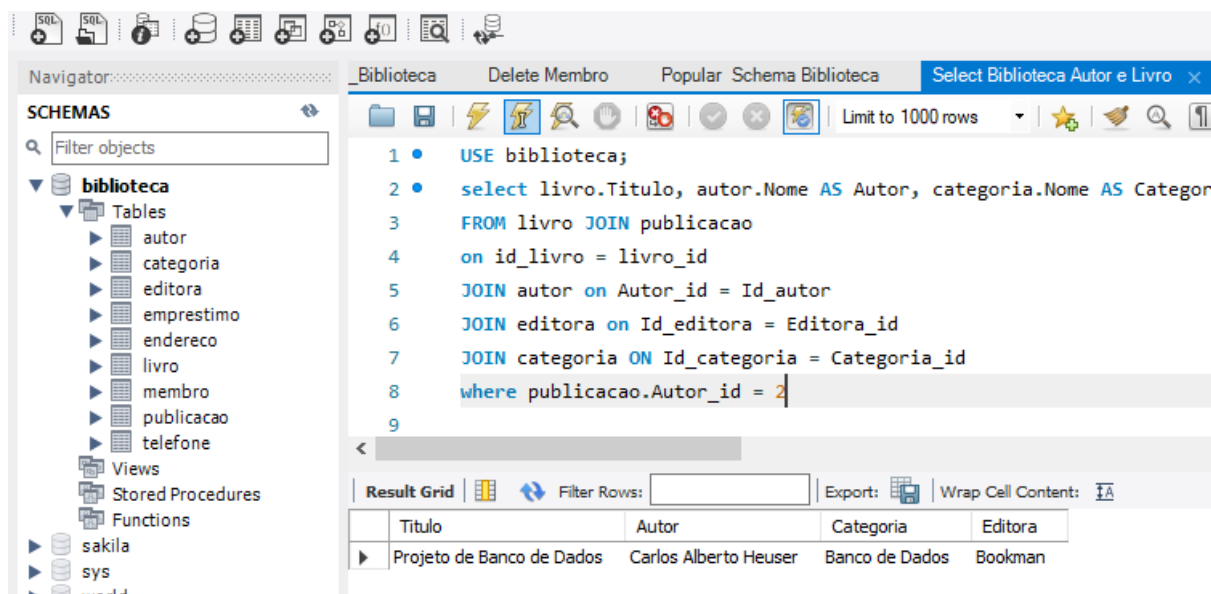


Figura 04 – Exemplo de um Select executado

## 7 - Select fazendo relação entre as tabelas

USE biblioteca;

```
select livro.Titulo, autor.Nome AS Autor, categoria.Nome AS Categoria, editora.Nome as
Editora
```

```
FROM livro JOIN publicacao
```

```
on id_livro = livro_id  
JOIN autor on Autor_id = Id_autor  
JOIN editora on Id_editora = Editora_id  
JOIN categoria ON Id_categoria = Categoria_id  
where publicacao.Autor_id = 2
```

### **Referências**

Date, C. J. Introdução a sistemas de banco de dados – 8ª ed. - Rio de Janeiro: Elsevier, 2003.

Heuser, C. A. Projeto de banco de dados – 6ª ed. - Porto Alegre: Bookman, 2009.

Pedroso, V. de M., Yanaga, E. – Banco de Dados – reimpresso em 2023 – Maringá: Unicesumar, 2016.