



**Sumário**

1. Projeto Conceitual do sistema

1.1 Especificação Textual Completa

1.2 Modelo Entidade Relacionamento (MER)

2. Projeto Lógico do sistema

2.1 (MR)

2.2 MER implementado no Workbench

3. Projeto Físico do sistema

4. Consultas Implementadas

**1. Projeto conceitual do sistema**

**1.1 Especificação Textual Completa**

**Controle de Empréstimos e Utilização de Serviços de uma Biblioteca:**

Uma biblioteca deseja implementar um sistema de banco de dados, com o intuito de controlar e relacionar os diversos tipos de dados manipulados em seu funcionamento.

Dito isso, projetamos um sistema de banco de dados que irá guardar informações de todos os tipos de obras fornecidas pela biblioteca. Podendo ser elas, livros, revistas e filmes.

Todas as obras terão obrigatoriamente uma quantidade de cópias, um ano de publicação, categoria, título, edição, volume, tipo e código como atributos, sendo este último o atributo identificador da obra. Além disso as obras possuirão assuntos, categorizados por nome e código, bem como exemplares, estes que terão como características, TOMBO, código, capa e estado de conservação. Sendo que tanto para assuntos como exemplares, o código será identificador e no caso dos exemplares também poderão ser identificados pelo TOMBO.

Além disso, as obras se relacionam com outras entidades, sendo elas Autores-Editores-Diretores e Editoras-Produtoras. Estas, por sua vez, apresentam os seguintes atributos, autores são compostos por nome, data de nascimento e nacionalidade. Já as editoras por nome e endereço, composto por rua, bairro, cidade, cep, estado, número e país. Em ambas, o nome é o atributo identificador, bem como CPF para Autores-Editores-Diretores e CNPJ para Editoras-Produtoras.

Todas as obras, podem ser reservadas ou ter seus exemplares emprestados por leitores. Os leitores são obrigatoriamente compostos por nome, código, sexo, data de nascimento, telefones (mais de um), e-mail, número de RG, estado emissor do RG, CPF e endereço (rua, bairro, cidade, estado, CEP e número).

Cada leitor pode ser classificado entre duas entidades, exclusivamente, alunos ou servidores.

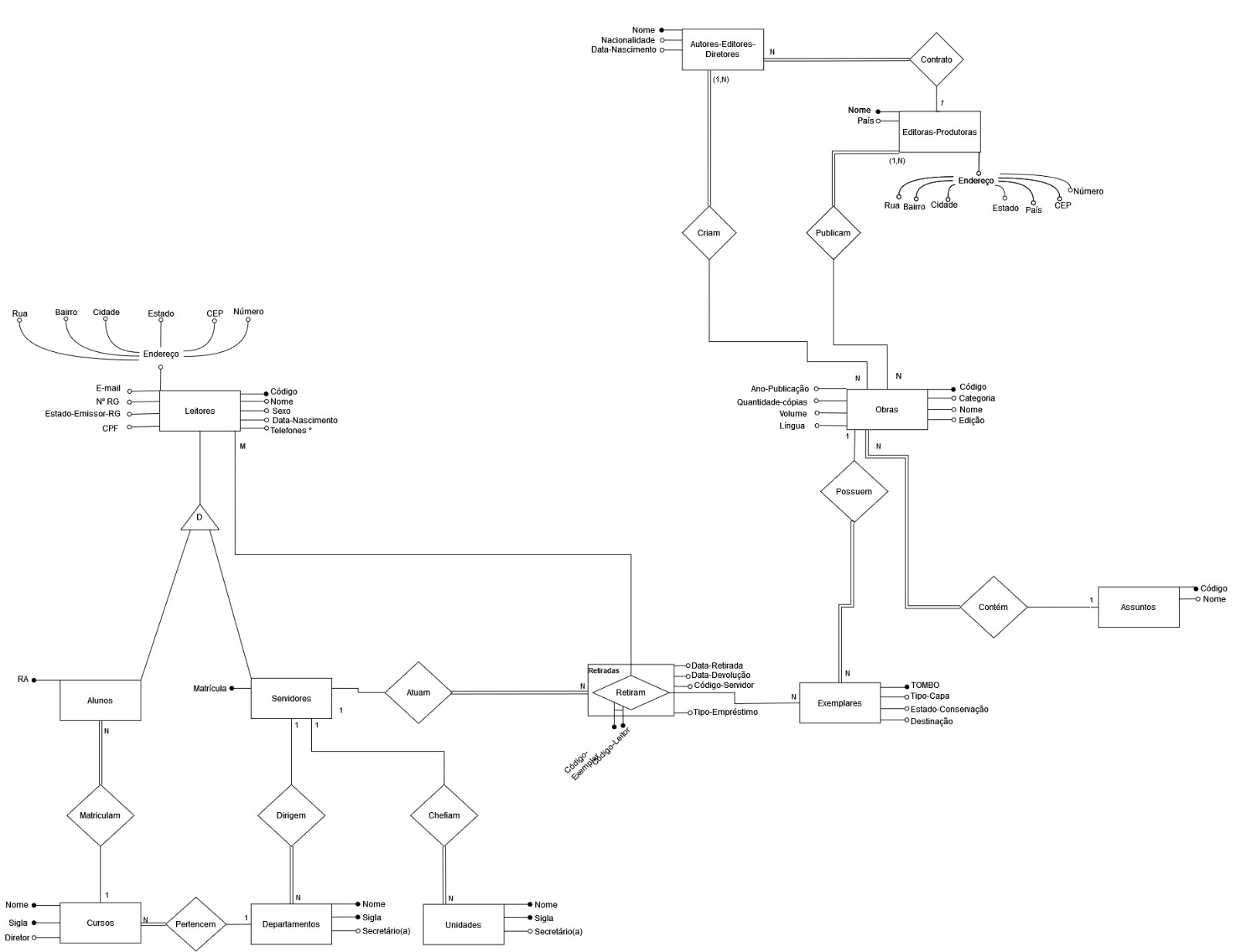
Cada aluno, além dos atributos herdados de leitores, também é composto por RA, curso e departamento, onde RA é identificador. Esses alunos podem estar matriculados em cursos, que pertencerão à um departamento.

Já os servidores, além dos atributos que também herda de leitores, possuem como atributo identificador um número de matrícula e se relacionam com departamentos, unidades e com as reservas, visto que são responsáveis por efetuar os empréstimos das obras aos leitores.

Tanto os departamentos como as unidades, apresentam como atributo nome, sigla e secretário(a), porém departamento ainda terá um diretor, enquanto a unidade terá um chefe. Ambas as entidades possuem o nome como identificador.

As reservas proporcionam informações de interesse dos leitores, sendo assim elas podem se relacionar com assuntos das obras que são retiradas. Juntando informações do código do leitor e o assunto da obra é possível criar um valor único e identificador que será utilizado no envio de e-mails para os leitores.

Também nas reservas é determinado a data de retirada, código da reserva, código do exemplar, código do leitor e código do servidor. Os quatro primeiros atributos são identificadores das reservas, sendo a data da retirada, código do exemplar e código do leitor, unificados em um único identificador, capaz de proporcionar o cálculo da data de devolução.

**1.2 Modelo Entidade Relacionamento (MER)**

**2. Projeto Lógico do sistema**

**2.1 (MR)**

Obras (**Código**, Categoria, Nome, Edição, Volume, Ano-Publicação, Quantidade-Cópias, Língua, Nome-Editoras-Produtoras, Nome-Autores-Editores-Diretores, Código-Assunto) Nome-Editoras-Produtoras referencia Editoras-Produtoras, Nome-Autores-Editores-Diretores referencia Autores-Editores-Diretores, Código-Assunto referencia Assuntos.

Exemplares (**TOMBO**, Tipo-Capa, Estado-Conservação, Destinação, Código-Obra) Código-Obra referencia Obras.

Retiradas (**Codigo-Exemplar, Codigo-Leitor**, Data-Retirada, Data-Devolução, Codigo-Servidor, Tipo-Emprestimo) Codigo-Exemplar referencia Exemplares, Codigo-Leitor referencia Leitores, Codigo-Servidor referencia Servidores.

Assuntos (**Codigo**, Nome)

Autores-Editores-Diretores (**Nome**, Nacionalidade, Data-Nascimento, Nome-Editoras-Produtoras) Nome-Editoras-Produtoras referencia Editoras-Produtoras.

Editoras-Produtoras (**Nome**, País)

Leitores (**Código**, Nome, Sexo, Data-Nascimento, Telefones, Endereço, E-mail, Num-RG, Estado-Emissor-RG, CPF)

Alunos (**Código**, **RA**, Nome-Curso) Código referencia Leitores, Nome-Curso referencia Curso

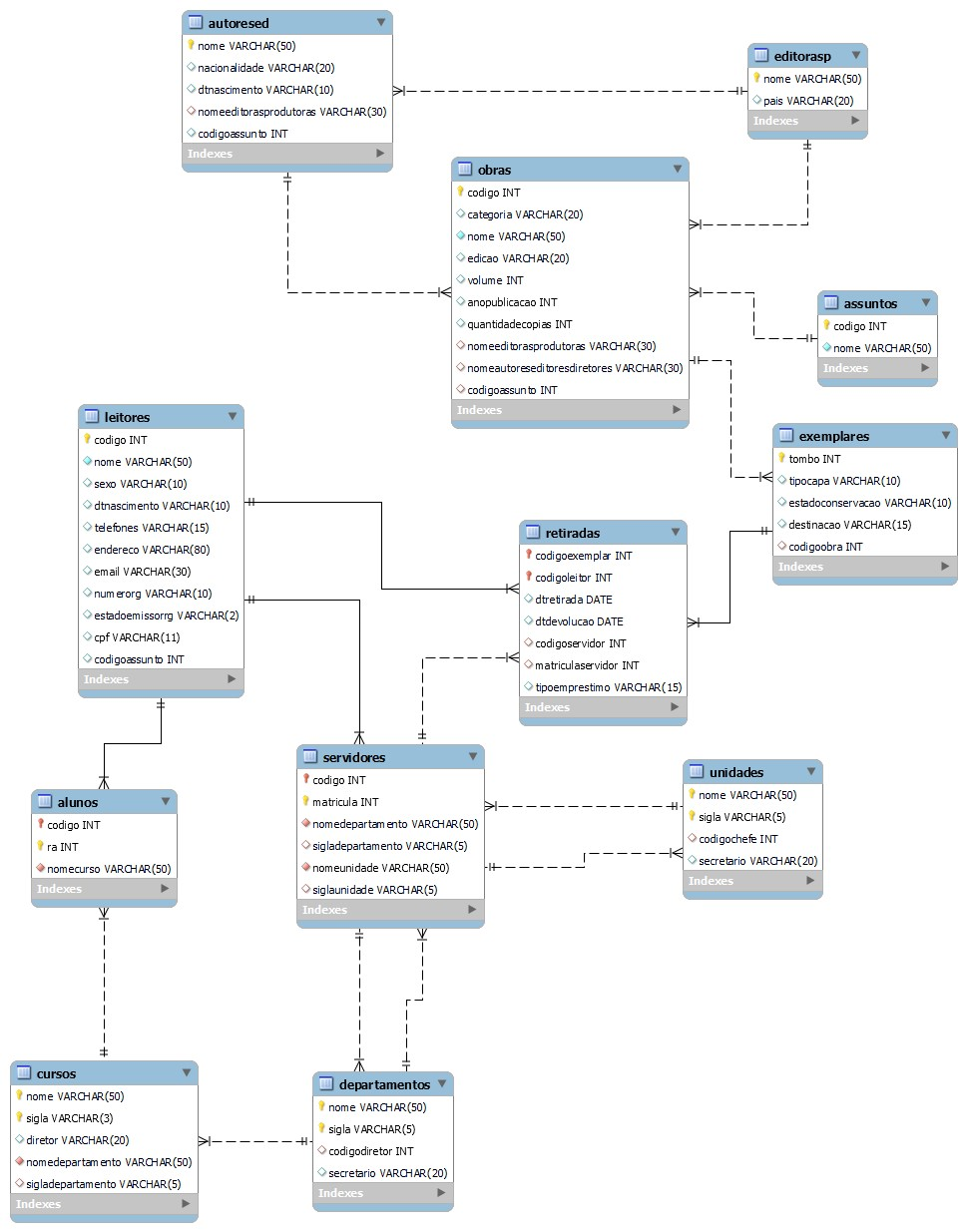
Servidores (**Código**, **Matrícula**, Nome-Departamento-Sigla-Departamento, Nome-Unidade-Sigla-Unidade) Código referencia Leitores, Nome-Departamento-Sigla-Departamento referencia Departamentos, Nome-Unidade-Sigla-Unidade referencia Unidades

Cursos (**Nome**, **Sigla**, Diretor, Nome-Departamento-Sigla-Departamento) Nome-Departamento-Sigla-Departamento referencia Departamentos

Departamentos (**Nome**, **Sigla**, Código-Diretor, Secretário) Código-Diretor referencia Servidores.

Unidades (**Nome**, **Sigla**, Código-Chefe, Secretário) Código-Chefe referencia Servidores.

**2.2 MER implementado no Workbench**

****

**3. Projeto Físico do sistema**

/\* Cria tabela Obras \*/

CREATE TABLE Obras(

codigo int(3) AUTO\_INCREMENT,

categoria varchar(20),

nome varchar(50) NOT NULL,

edicao varchar(20),

volume int(2),

anopublicacao int(4),

quantidadecopias int(3),

nomeeditorasprodutoras varchar(30),

nomeautoreseditoresdiretores varchar(30),

codigoassunto int(2),

PRIMARY KEY (codigo)

)ENGINE = innodb;

/\* Insere na tabela Obras \*/

INSERT INTO Obras(codigo, categoria, nome, edicao, volume, anopublicacao, quantidadecopias, nomeeditorasprodutoras, nomeautoreseditoresdiretores, codigoassunto) VALUES (101, "Livro", "O Hobbit", "padrão", 01, 1937, 10, "HarperCollins", "John Ronald Reuel Tolkien", 01);

INSERT INTO Obras(codigo, categoria, nome, edicao, volume, anopublicacao, quantidadecopias, nomeeditorasprodutoras, nomeautoreseditoresdiretores, codigoassunto) VALUES (102, "Filme", "De volta para o futuro", "padrão", 01, 1985, 3, "Universal Pictures", "Robert Zemeckis", 02);

INSERT INTO Obras(codigo, categoria, nome, edicao, volume, anopublicacao, quantidadecopias, nomeeditorasprodutoras, nomeautoreseditoresdiretores, codigoassunto) VALUES (103, "Filme", "Titanic", "especial", 01, 1998, 2, "Paramount Pictures", "James Cameron", 03);

INSERT INTO Obras(codigo, categoria, nome, edicao, volume, anopublicacao, quantidadecopias, nomeeditorasprodutoras, nomeautoreseditoresdiretores, codigoassunto) VALUES (104, "Livro", "Harry Potter e o Prisioneiro de Azkaban", "padrão", 03, 1999, 5, "Rocco", "J. K. Rowling", 01);

INSERT INTO Obras(codigo, categoria, nome, edicao, volume, anopublicacao, quantidadecopias, nomeeditorasprodutoras, nomeautoreseditoresdiretores, codigoassunto) VALUES (105, "Livro", "O Pequeno Principe", "padrão", 01, 1943, 1, "Rocco", "Antoine de SaintExupéry", 04);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Cria tabela Exemplares \*/

CREATE TABLE Exemplares(

tombo int(5) AUTO\_INCREMENT,

tipocapa varchar(10),

estadoconservacao varchar(10),

destinacao varchar(15),

codigoobra int(3),

PRIMARY KEY (tombo)

)ENGINE = innodb;

/\* Insere na tabela Exemplares \*/

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10101,"Dura","Bom","Reserva",101);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10102,"Dura","Regular","Reserva",101);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10103,"Dura","Regular","Empréstimo",101);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10104,"Dura","Bom","Empréstimo",101);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10105,"Dura","Bom","Empréstimo",101);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10106,"Dura","Bom","Empréstimo",101);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10107,"Dura","Bom","Empréstimo",101);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10108,"Dura","Bom","Empréstimo",101);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10109,"Dura","Bom","Empréstimo",101);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10110,"Dura","Bom","Empréstimo",101);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10201,"Normal","Bom","Reserva",102);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10202,"Normal","Regular","Empréstimo",102);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10203,"Normal","Bom","Empréstimo",102);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10301,"Normal","Ruim","Reserva",103);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10302,"Normal","Bom","Empréstimo",103);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10401,"Normal","Bom","Reserva",104);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10402,"Normal","Bom","Empréstimo",104);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10403,"Normal","Regular","Empréstimo",104);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10404,"Normal","Bom","Empréstimo",104);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10405,"Normal","Regular","Empréstimo",104);

INSERT INTO Exemplares(tombo, tipocapa, estadoconservacao, destinacao, codigoobra) VALUES (10501,"Normal","Bom","Reserva",104);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Cria tabela AutoresED \*/

CREATE TABLE AutoresEd(

nome varchar(50) NOT NULL,

nacionalidade varchar(20),

dtnascimento varchar(10),

nomeeditorasprodutoras varchar(30),

codigoassunto int(2),

PRIMARY KEY (nome)

)ENGINE = innodb;

/\* Insere na tabela AutoresED \*/

INSERT INTO AutoresEd(nome, nacionalidade, dtnascimento, nomeeditorasprodutoras, codigoassunto) VALUES ("John Ronald Reuel Tolkien","Inglaterra","03/01/1893","HarperCollins",01);

INSERT INTO AutoresEd(nome, nacionalidade, dtnascimento, nomeeditorasprodutoras, codigoassunto) VALUES ("Robert Zemeckis","Estados Unidos","14/05/1952","Universal Pictures",02);

INSERT INTO AutoresEd(nome, nacionalidade, dtnascimento, nomeeditorasprodutoras, codigoassunto) VALUES ("James Cameron","Canadá","16/08/1954","Paramount Pictures",03);

INSERT INTO AutoresEd(nome, nacionalidade, dtnascimento, nomeeditorasprodutoras, codigoassunto) VALUES ("J. K. Rowling","Inglaterra","31/07/1965","Rocco",01);

INSERT INTO AutoresEd(nome, nacionalidade, dtnascimento, nomeeditorasprodutoras, codigoassunto) VALUES ("Antoine de SaintExupéry","França","29/06/1900","Rocco",04);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Cria tabela EditorasP \*/

CREATE TABLE EditorasP(

nome varchar(50) NOT NULL,

pais varchar(20),

PRIMARY KEY (nome)

)ENGINE = innodb;

/\* Insere na tabela EditorasP \*/

INSERT INTO EditorasP(nome, pais) VALUES ("HarperCollins","Inglaterra");

INSERT INTO EditorasP(nome, pais) VALUES ("Universal Pictures","Estados Unidos");

INSERT INTO EditorasP(nome, pais) VALUES ("Paramount Pictures","Estados Unidos");

INSERT INTO EditorasP(nome, pais) VALUES ("Rocco","Brasil");

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Cria tabela Assuntos \*/

CREATE TABLE Assuntos(

codigo int(2) AUTO\_INCREMENT,

nome varchar(50) NOT NULL,

PRIMARY KEY (codigo)

)ENGINE = innodb;

/\* Insere na tabela Assuntos \*/

INSERT INTO Assuntos(codigo, nome) VALUES (01, "Fantasia");

INSERT INTO Assuntos(codigo, nome) VALUES (02, "Ficção Científica");

INSERT INTO Assuntos(codigo, nome) VALUES (03, "Romance");

INSERT INTO Assuntos(codigo, nome) VALUES (04, "InfantoJuvenil");

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Cria tabela Leitores \*/

CREATE TABLE Leitores(

codigo int(3) AUTO\_INCREMENT,

nome varchar(50) NOT NULL,

sexo varchar(10),

dtnascimento varchar(10),

telefones varchar(15),

endereco varchar(80),

email varchar(30),

numerorg varchar(10),

estadoemissorrg varchar(2),

cpf varchar(11),

codigoassunto int(2),

PRIMARY KEY (codigo)

)ENGINE = innodb;

/\* Insere na tabela de Leitores \*/

INSERT INTO Leitores(codigo, nome, sexo, dtnascimento, telefones, endereco, email, numerorg, estadoemissorrg, cpf, codigoassunto) VALUES (901,"Adanrond","Masculino","27/04/1998","(11) 2622-0436","Travessa Naum Gabo,1515","adanrond@unesp.br","442755521","SP","67105404051",01);

INSERT INTO Leitores(codigo, nome, sexo, dtnascimento, telefones, endereco, email, numerorg, estadoemissorrg, cpf, codigoassunto) VALUES (902,"Tarpen","Masculino","18/02/1964","(16) 3242-1817","Rua Professor Azambuja Suzano,1232","tarpen@unesp.br","138551935","SP","02801638005",02);

INSERT INTO Leitores(codigo, nome, sexo, dtnascimento, telefones, endereco, email, numerorg, estadoemissorrg, cpf, codigoassunto) VALUES (903,"Rivaepen","Feminino","15/06/1990","(15) 3253-4577","Rua Padre Montoya,673","rivaepen@unesp.br","420847200","SP","30332729087",04);

INSERT INTO Leitores(codigo, nome, sexo, dtnascimento, telefones, endereco, email, numerorg, estadoemissorrg, cpf, codigoassunto) VALUES (904,"Arinit","Feminino","04/01/1982","(14) 3577-8637","Travessa Bernardino de Andrade,23","arinit@unesp.br","226458283","RJ","05137301094",03);

INSERT INTO Leitores(codigo, nome, sexo, dtnascimento, telefones, endereco, email, numerorg, estadoemissorrg, cpf, codigoassunto) VALUES (905,"Vonusir","Masculino","23/02/1915","(11) 3468-3580","Rua Tenente-Coronel Cunha,532","vonusir@unesp.br","349159385","RJ","41190609002",01);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Cria tabela Alunos \*/

CREATE TABLE Alunos(

codigo int(3) AUTO\_INCREMENT,

ra int(10),

nomecurso varchar(50) NOT NULL,

PRIMARY KEY (codigo, ra)

)ENGINE = innodb;

/\*Insere na tabela de Alunos \*/

INSERT INTO Alunos(codigo, ra, nomecurso) VALUES (901, 99991, "Ciencias da Computação");

INSERT INTO Alunos(codigo, ra, nomecurso) VALUES (902, 99992, "Ciencias da Computação");

INSERT INTO Alunos(codigo, ra, nomecurso) VALUES (904, 99994, "Biologia");

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Cria tabela Servidores \*/

CREATE TABLE Servidores(

codigo int(3) AUTO\_INCREMENT,

matricula int(3),

nomedepartamento varchar(50) NOT NULL,

sigladepartamento varchar(5),

nomeunidade varchar(50) NOT NULL,

siglaunidade varchar(5),

PRIMARY KEY (codigo, matricula)

)ENGINE = innodb;

/\* Insere na tabela de Servidores \*/

INSERT INTO Servidores(codigo, matricula, nomedepartamento, sigladepartamento, nomeunidade, siglaunidade) VALUES (903, 881, "Departamento de Matemática aplicada à Computação", "Demac", "Universidade Estadual Paulista", "Unesp");

INSERT INTO Servidores(codigo, matricula, nomedepartamento, sigladepartamento, nomeunidade, siglaunidade) VALUES (905, 882, "Departamento de Biologia", "Debio", "Universidade Estadual Paulista", "Unesp");

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Cria tabela Cursos \*/

CREATE TABLE Cursos(

nome varchar(50) NOT NULL,

sigla varchar(3),

diretor varchar(20),

nomedepartamento varchar(50) NOT NULL,

sigladepartamento varchar (5),

PRIMARY KEY (nome, sigla)

)ENGINE = innodb;

/\* Insere na tabela de Cursos \*/

INSERT INTO Cursos(nome, sigla, diretor, nomedepartamento, sigladepartamento) VALUES ("Ciencias da Computação", "BCC", "Lehzeucuo", "Departamento de Matemática aplicada à Computação", "Demac");

INSERT INTO Cursos(nome, sigla, diretor, nomedepartamento, sigladepartamento) VALUES ("Biologia", "BIO", "Heyvliopya", "Departamento de Biologia", "Debio");

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Cria tabela Departamento \*/

CREATE TABLE Departamentos(

nome varchar(50) NOT NULL,

sigla varchar(5),

codigodiretor int(3),

secretario varchar (20),

PRIMARY KEY (nome, sigla)

)ENGINE = innodb;

/\* Insere na tabela de Departamento \*/

INSERT INTO Departamentos(nome, sigla, codigodiretor, secretario) VALUES ("Departamento de Matemática aplicada à Computação", "Demac", "903", "Reopo");

INSERT INTO Departamentos(nome, sigla, codigodiretor, secretario) VALUES ("Departamento de Biologia", "Debio", "905", "Saemo");

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Cria tabela Unidades \*/

CREATE TABLE Unidades(

nome varchar(50) NOT NULL,

sigla varchar(5),

codigochefe int(3),

secretario varchar (20),

PRIMARY KEY (nome, sigla)

)ENGINE = innodb;

/\* Insere na tabela de Unidades \*/

INSERT INTO Unidades(nome, sigla, codigochefe, secretario) VALUES ("Universidade Estadual Paulista", "Unesp", "903", "Reopo");

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Cria tabela Retiradas \*/

CREATE TABLE Retiradas(

codigoexemplar int(5),

codigoleitor int(3),

dtretirada date,

dtdevolucao date,

codigoservidor int(3),

matriculaservidor int(3),

tipoemprestimo varchar(15),

PRIMARY KEY (codigoexemplar, codigoleitor)

)ENGINE = innodb;

/\* Insere na tabela de Unidades \*/

INSERT INTO Retiradas(codigoexemplar, codigoleitor, dtretirada, dtdevolucao, codigoservidor, matriculaservidor, tipoemprestimo) VALUES (10101, 901, "2023-10-10", "2023-10-15", 903, 881, "Reserva");

INSERT INTO Retiradas(codigoexemplar, codigoleitor, dtretirada, dtdevolucao, codigoservidor, matriculaservidor, tipoemprestimo) VALUES (10201, 902, "2023-11-01", "2023-11-11", 905, 882, "Reserva");

INSERT INTO Retiradas(codigoexemplar, codigoleitor, dtretirada, dtdevolucao, codigoservidor, matriculaservidor, tipoemprestimo) VALUES (10301, 903, "2023-01-27", "2023-02-01", 903, 881, "Reserva");

INSERT INTO Retiradas(codigoexemplar, codigoleitor, dtretirada, dtdevolucao, codigoservidor, matriculaservidor, tipoemprestimo) VALUES (10202, 901, "2023-12-15", "2023-12-20", 905, 882, "Empréstimo");

INSERT INTO Retiradas(codigoexemplar, codigoleitor, dtretirada, dtdevolucao, codigoservidor, matriculaservidor, tipoemprestimo) VALUES (10103, 901, "2023-06-12", "2023-06-19", 905, 882, "Empréstimo");

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

ALTER TABLE Obras ADD CONSTRAINT fk\_nomeeditorasprodutoras FOREIGN KEY ( nomeeditorasprodutoras ) REFERENCES EditorasP ( nome ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Obras ADD CONSTRAINT fk\_nomeautoreseditoresdiretores FOREIGN KEY ( nomeautoreseditoresdiretores ) REFERENCES AutoresEd ( nome ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Obras ADD CONSTRAINT fk\_codigoassunto FOREIGN KEY ( codigoassunto ) REFERENCES Assuntos ( codigo ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Exemplares ADD CONSTRAINT fk\_codigoobra FOREIGN KEY ( codigoobra ) REFERENCES Obras ( codigo ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Retiradas ADD CONSTRAINT fk\_codigoexemplar FOREIGN KEY ( codigoexemplar ) REFERENCES Exemplares ( tombo ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Retiradas ADD CONSTRAINT fk\_codigoleitor3 FOREIGN KEY ( codigoleitor ) REFERENCES Leitores ( codigo ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Retiradas ADD CONSTRAINT fk\_codigoservidor FOREIGN KEY ( codigoservidor, matriculaservidor ) REFERENCES Servidores ( codigo, matricula ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE AutoresED ADD CONSTRAINT fk\_nomeeditorasprodutoras2 FOREIGN KEY ( nomeeditorasprodutoras ) REFERENCES EditorasP ( nome ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Alunos ADD CONSTRAINT fk\_codigoleitor FOREIGN KEY ( codigo ) REFERENCES Leitores ( codigo ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Alunos ADD CONSTRAINT fk\_nomecurso FOREIGN KEY ( nomecurso ) REFERENCES Cursos ( nome ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Servidores ADD CONSTRAINT fk\_codigoleitor2 FOREIGN KEY ( codigo ) REFERENCES Leitores ( codigo ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Servidores ADD CONSTRAINT fk\_sigladepartamento FOREIGN KEY ( nomedepartamento, sigladepartamento ) REFERENCES Departamentos ( nome, sigla ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Servidores ADD CONSTRAINT fk\_siglaunidade FOREIGN KEY ( nomeunidade, siglaunidade ) REFERENCES Unidades ( nome, sigla ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Cursos ADD CONSTRAINT fk\_sigladepartamento2 FOREIGN KEY ( nomedepartamento, sigladepartamento ) REFERENCES Departamentos ( nome, sigla ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Departamentos ADD CONSTRAINT fk\_codigodiretor FOREIGN KEY ( codigodiretor ) REFERENCES Servidores ( codigo ) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE Unidades ADD CONSTRAINT fk\_codigochefe FOREIGN KEY ( codigochefe ) REFERENCES Servidores ( codigo ) ON DELETE NO ACTION ON UPDATE NO ACTION;

**4. Consultas Implementadas**

1. Deseja-se ter uma relação dos leitores, onde para cada leitor devem aparecer as seguintes

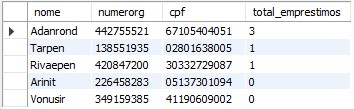
Informações: Nome, seu RG e CPF e o número total de empréstimos realizados pelo leitor.

SELECT l.nome, l.numerorg, l.cpf, COUNT(r.codigoleitor) AS total\_emprestimos

FROM Leitores l

LEFT JOIN Retiradas r ON l.codigo = r.codigoleitor

GROUP BY l.codigo, l.nome, l.numerorg, l.cpf;



2. Deseja-se saber quais são os leitores (CPF, Nome) que pegaram um determinado livro, tendo

disponível somente o nome do livro.

SELECT DISTINCT l.cpf, l.nome

FROM Leitores l

JOIN Retiradas R ON l.codigo = r.codigoleitor

JOIN Exemplares e ON r.codigoexemplar = e.tombo

JOIN Obras o ON e.codigoobra = o.codigo

WHERE o.nome = 'De volta para o futuro';



3. Deseja-se saber quais são os leitores (CPF, Nome), residentes em uma cidade específica,

que pegaram um determinado livro, tendo disponível somente o nome do livro e o nome da

cidade.

SELECT DISTINCT l.cpf, l.nome

FROM Leitores l

JOIN Retiradas r ON l.codigo = r.codigoleitor

JOIN Exemplares e ON r.codigoexemplar = e.tombo

JOIN Obras o ON e.codigoobra = o.codigo

WHERE o.nome = 'O Hobbit' AND l.endereco = 'Travessa Naum Gabo,1515';



4. Deseja-se saber quais são os livros de um determinado autor, publicados por uma

determinada editora, tendo disponível o nome do autor e o nome da editora desejada. Além

disso, para cada livro precisa-se conhecer quem é o tradutor (nome), se for o caso.

Como não temos tradutor, consultamos o assunto da obra.

SELECT o.nome AS livro, a.nome AS assunto

FROM Obras o

JOIN Assuntos a ON o.codigoassunto = a.codigo

JOIN AutoresEd au ON o.nomeautoreseditoresdiretores = au.nome

JOIN EditorasP e ON o.nomeeditorasprodutoras = e.nome

WHERE au.nome = 'J. K. Rowling' AND e.nome = 'Rocco';



5. Deseja-se saber quais são os empréstimos (data, identificação do livro e o período) realizados por um determinado leitor (CPF conhecido), num período específico.

Período: 2023-06-01 a 2023-11-01

SELECT r.dtretirada, r.dtdevolucao, o.nome AS livro, DATEDIFF(r.dtdevolucao, r.dtretirada) AS periodo

FROM Retiradas r

JOIN Exemplares e ON r.codigoexemplar = e.tombo

JOIN Obras o ON e.codigoobra = o.codigo

JOIN Leitores l ON r.codigoleitor = l.codigo

WHERE l.cpf = '67105404051'

AND r.dtretirada >= '2023-06-01'

AND r.dtdevolucao <= '2023-11-01';



6. Deseja-se saber quais são os 5 livros mais procurados, num determinado período.

SELECT o.nome AS livro, COUNT(\*) AS total\_retiradas

FROM Retiradas r

JOIN Exemplares e ON r.codigoexemplar = e.tombo

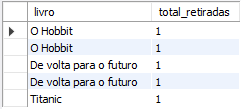
JOIN Obras o ON e.codigoobra = o.codigo

WHERE r.dtretirada >= '2023-01-01' AND r.dtretirada <= '2023-12-31'

GROUP BY r.codigoexemplar, o.nome

ORDER BY total\_retiradas DESC

LIMIT 5;



7. Deseja-se saber quais são os 10 leitores (CPF, Nome) mais frequentes (com maior número de

empréstimos).

SELECT l.cpf, l.nome, COUNT(r.codigoleitor) AS totalemprestimos

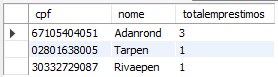
FROM Leitores l

JOIN Retiradas r ON l.codigo = r.codigoleitor

GROUP BY l.codigo

ORDER BY totalemprestimos DESC

LIMIT 10;



8. Procura-se conhecer os 3 autores mais procurados, as 3 editoras com maior número de

empréstimos e os 3 assuntos mais consultados (em termos de número de empréstimos).

SELECT a.nome AS assunto, au.nome AS autor, ep.nome AS editora, COUNT(\*) AS total\_retiradas

FROM Retiradas r

JOIN Exemplares e ON r.codigoexemplar = e.tombo

JOIN Obras o ON e.codigoobra = o.codigo

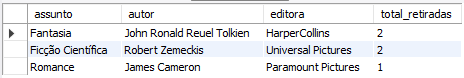
JOIN Assuntos a ON o.codigoassunto = a.codigo

JOIN AutoresED au ON o.nomeautoreseditoresdiretores = au.nome

JOIN Editorasp ep ON o.nomeeditorasprodutoras = ep.nome

GROUP BY a.nome, au.nome, ep.nome

ORDER BY total\_retiradas DESC LIMIT 3;

****