

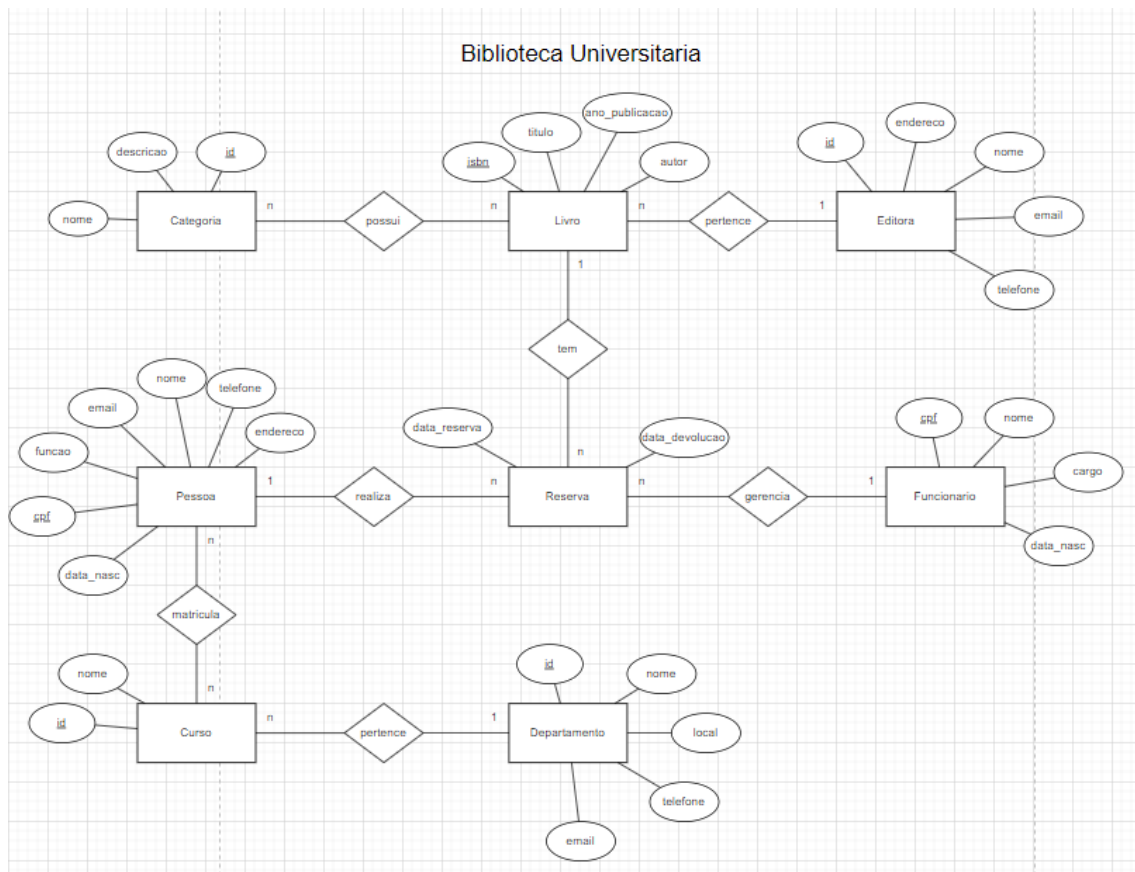
Minimundo de uma Biblioteca Universitária

A biblioteca universitária mantém um cadastro de livros. Uma categoria possui muitos livros assim como um livro tem várias categorias. Cada livro pertence uma editora, mas uma editora pode ter muitos livros. Os livros são identificados por seu ISBN e possuem título, autor, ano de publicação. Uma categoria é identificada por um id e possuem nome e descrição. Uma editora é identificada por seu id e possuem nome, endereço, telefone e email.

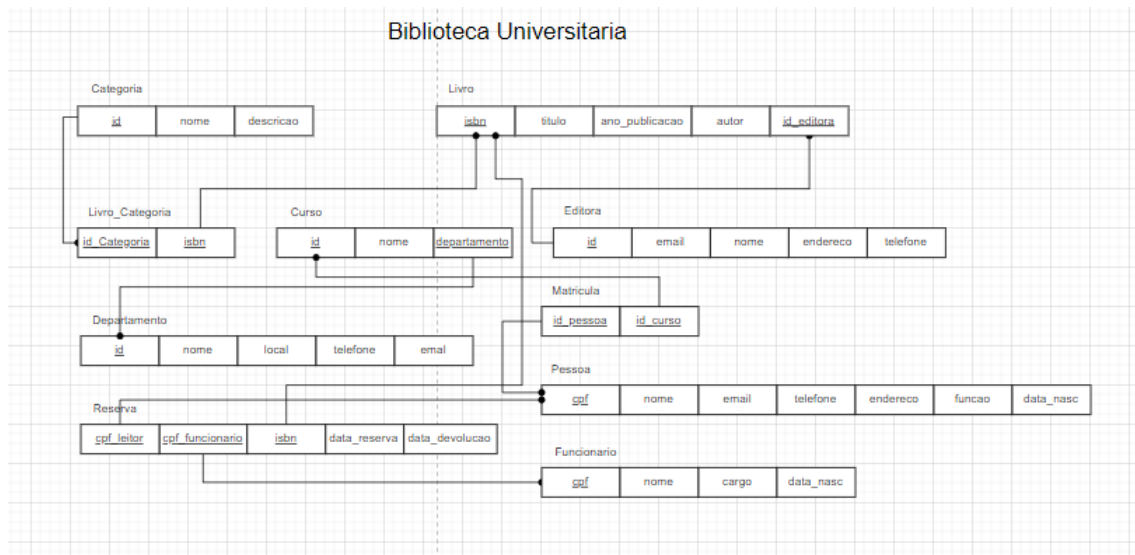
Qualquer pessoa da universidade pode reservar um livro. Para reservar um livro é necessário conhecer data de reserva e data de devolução, sendo que uma reserva está relacionada a um único livro, mas um livro pode ter muitas reservas. Uma pessoa pode fazer muitas reservas, mas cada reserva é feita por uma única pessoa. Uma pessoa é identificada pelo cpf e possui nome, função, data de nascimento, endereço, telefone, email.

Um curso pode ter várias pessoas matriculados assim como uma pessoa pode se matricular em mais de um curso. Um curso pertence obrigatoriamente a um departamento mas um departamento pode ter vários cursos. Curso é identificados por um id e possuem um nome. Departamento é identificado por um id e possuem nome, localização, telefone e email.

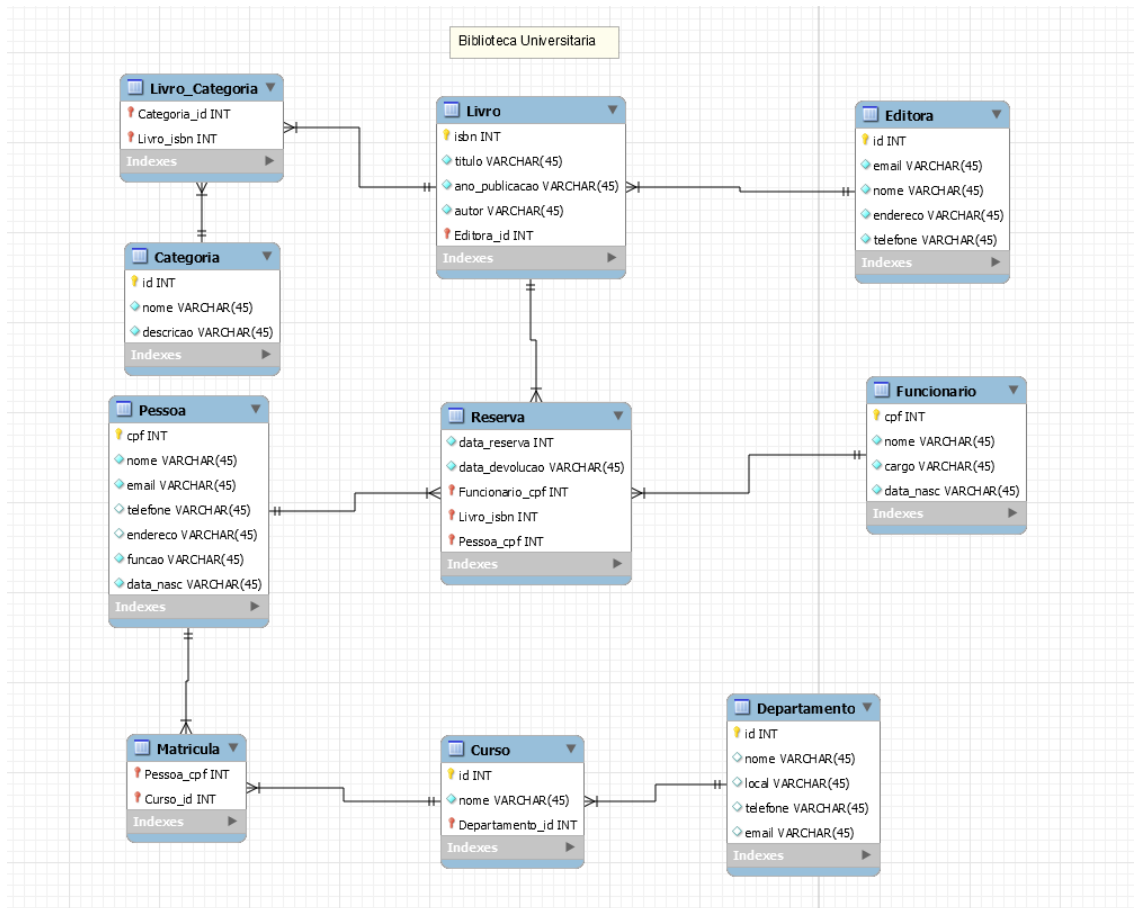
A biblioteca também possui funcionários que gerenciam as operações de reserva. Cada funcionário pode gerenciar muitas reservas, e cada reserva só pode ser gerida por um funcionário. Um funcionário é identificado pelo seu cpf e possui nome, cargo, data de nascimento.



3 -



4 -



5 –

-- MySQL Script generated by MySQL Workbench

-- Thu Jun 27 10:50:51 2024

-- Model: Sakila Full Version: 2.0

-- MySQL Workbench Forward Engineering

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;

SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;

SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='TRADITIONAL,ALLOW_INVALID_DATES';

-- Schema sakila

-- Schema sakila

CREATE SCHEMA IF NOT EXISTS `sakila` ;
USE `sakila` ;

-- Table `sakila`.`Categoria`

CREATE TABLE IF NOT EXISTS `sakila`.`Categoria` (
 `id` INT NOT NULL AUTO_INCREMENT,
 `nome` VARCHAR(45) NOT NULL,
 `descricao` VARCHAR(45) NOT NULL,
 PRIMARY KEY (`id`))
ENGINE = InnoDB;

-- Table `sakila`.`Editora`

CREATE TABLE IF NOT EXISTS `sakila`.`Editora` (
 `id` INT NOT NULL AUTO_INCREMENT,
 `email` VARCHAR(45) NOT NULL,
 `nome` VARCHAR(45) NOT NULL,
 `endereco` VARCHAR(45) NOT NULL,
 `telefone` VARCHAR(45) NOT NULL,
 PRIMARY KEY (`id`))
ENGINE = InnoDB;

-- Table `sakila`.`Livro`

```
-----  
  
CREATE TABLE IF NOT EXISTS `sakila`.`Livro` (  
  `isbn` INT NOT NULL AUTO_INCREMENT,  
  `titulo` VARCHAR(45) NOT NULL,  
  `ano_publicacao` VARCHAR(45) NOT NULL,  
  `autor` VARCHAR(45) NOT NULL,  
  `Editora_id` INT NOT NULL,  
  PRIMARY KEY (`isbn`, `Editora_id`),  
  INDEX `fk_Livro_Editora_idx` (`Editora_id` ASC),  
  CONSTRAINT `fk_Livro_Editora`  
    FOREIGN KEY (`Editora_id`)  
    REFERENCES `sakila`.`Editora` (`id`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

-- Table `sakila`.`Departamento`

```
-----  
  
CREATE TABLE IF NOT EXISTS `sakila`.`Departamento` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `nome` VARCHAR(45) NULL,  
  `local` VARCHAR(45) NULL,  
  `telefone` VARCHAR(45) NULL,  
  `email` VARCHAR(45) NULL,  
  PRIMARY KEY (`id`))  
ENGINE = InnoDB;
```

-- Table `sakila`.`Curso`

```
CREATE TABLE IF NOT EXISTS `sakila`.`Curso` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `nome` VARCHAR(45) NOT NULL,  
  `Departamento_id` INT NOT NULL,  
  PRIMARY KEY (`id`, `Departamento_id`),  
  INDEX `fk_Curso_Departamento1_idx` (`Departamento_id` ASC),  
  CONSTRAINT `fk_Curso_Departamento1`  
    FOREIGN KEY (`Departamento_id`)  
    REFERENCES `sakila`.`Departamento` (`id`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

-- Table `sakila`.`Funcionario`

```
CREATE TABLE IF NOT EXISTS `sakila`.`Funcionario` (  
  `cpf` INT NOT NULL,  
  `nome` VARCHAR(45) NOT NULL,  
  `cargo` VARCHAR(45) NOT NULL,  
  `data_nasc` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`cpf`))  
ENGINE = InnoDB;
```

-- Table `sakila`.`Pessoa`

```

-----
CREATE TABLE IF NOT EXISTS `sakila`.`Pessoa` (
  `cpf` INT NOT NULL,
  `nome` VARCHAR(45) NOT NULL,
  `email` VARCHAR(45) NOT NULL,
  `telefone` VARCHAR(45) NULL,
  `endereco` VARCHAR(45) NULL,
  `funcao` VARCHAR(45) NOT NULL,
  `data_nasc` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`cpf`))
ENGINE = InnoDB;

```

```

-----
-- Table `sakila`.`Reserva`
-----

```

```

CREATE TABLE IF NOT EXISTS `sakila`.`Reserva` (
  `data_reserva` INT NOT NULL,
  `data_devolucao` VARCHAR(45) NOT NULL,
  `Funcionario_cpf` INT NOT NULL,
  `Livro_isbn` INT NOT NULL,
  `Pessoa_cpf` INT NOT NULL,
  PRIMARY KEY (`Funcionario_cpf`, `Livro_isbn`, `Pessoa_cpf`),
  INDEX `fk_Reserva_Livro1_idx` (`Livro_isbn` ASC),
  INDEX `fk_Reserva_Pessoa1_idx` (`Pessoa_cpf` ASC),
  CONSTRAINT `fk_Reserva_Funcionario1`
    FOREIGN KEY (`Funcionario_cpf`)
    REFERENCES `sakila`.`Funcionario` (`cpf`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Reserva_Livro1`

```

```

FOREIGN KEY (`Livro_isbn`)
REFERENCES `sakila`.`Livro` (`isbn`)
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `fk_Reserva_Pessoa1`
FOREIGN KEY (`Pessoa_cpf`)
REFERENCES `sakila`.`Pessoa` (`cpf`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `sakila`.`Livro_Categoria`
-----

```

```

CREATE TABLE IF NOT EXISTS `sakila`.`Livro_Categoria` (
  `Categoria_id` INT NOT NULL,
  `Livro_isbn` INT NOT NULL,
  PRIMARY KEY (`Categoria_id`, `Livro_isbn`),
  INDEX `fk_Livro_Categoria_Livro1_idx` (`Livro_isbn` ASC),
  CONSTRAINT `fk_Livro_Categoria_Categoria1`
    FOREIGN KEY (`Categoria_id`)
    REFERENCES `sakila`.`Categoria` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Livro_Categoria_Livro1`
    FOREIGN KEY (`Livro_isbn`)
    REFERENCES `sakila`.`Livro` (`isbn`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```



```
-----  
-- Table `sakila`.`Matricula`  
-----
```

```
CREATE TABLE IF NOT EXISTS `sakila`.`Matricula` (  
  `Pessoa_cpf` INT NOT NULL,  
  `Curso_id` INT NOT NULL,  
  PRIMARY KEY (`Pessoa_cpf`, `Curso_id`),  
  INDEX `fk_Matricula_Curso1_idx` (`Curso_id` ASC),  
  CONSTRAINT `fk_Matricula_Pessoa1`  
    FOREIGN KEY (`Pessoa_cpf`)  
      REFERENCES `sakila`.`Pessoa` (`cpf`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Matricula_Curso1`  
    FOREIGN KEY (`Curso_id`)  
      REFERENCES `sakila`.`Curso` (`id`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
SET SQL_MODE=@OLD_SQL_MODE;
```

```
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
```

```
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

```
6-
```

```
-- Reset and use sakila schema
```

```
USE `sakila`;
```

```
-- Insert data into Categoria table
```

```
INSERT INTO `Categoria` (`nome`, `descricao`) VALUES
('Fiction', 'Fictional books'),
('Non-Fiction', 'Non-fictional books'),
('Science', 'Scientific books'),
('Art', 'Books about art');
```

-- Insert data into Editora table

```
INSERT INTO `Editora` (`email`, `nome`, `endereco`, `telefone`) VALUES
('info@editora1.com', 'Editora 1', '123 Main St', '123-456-7890'),
('info@editora2.com', 'Editora 2', '456 Elm St', '234-567-8901');
```

-- Insert data into Livro table

```
INSERT INTO `Livro` (`titulo`, `ano_publicacao`, `autor`, `Editora_id`) VALUES
('The Great Gatsby', '1925', 'F. Scott Fitzgerald', 1),
('A Brief History of Time', '1988', 'Stephen Hawking', 2);
```

-- Insert data into Departamento table

```
INSERT INTO `Departamento` (`nome`, `local`, `telefone`, `email`) VALUES
('Humanities', 'Building A', '555-1234', 'humanities@university.com'),
('Sciences', 'Building B', '555-5678', 'sciences@university.com');
```

-- Insert data into Curso table

```
INSERT INTO `Curso` (`nome`, `Departamento_id`) VALUES
('English Literature', 1),
('Physics', 2);
```

-- Insert data into Funcionario table

```
INSERT INTO `Funcionario` (`cpf`, `nome`, `cargo`, `data_nasc`) VALUES
(123456789, 'John Doe', 'Librarian', '1980-01-01'),
(987654321, 'Jane Smith', 'Assistant Librarian', '1985-05-05');
```

-- Insert data into Pessoa table

```
INSERT INTO `Pessoa` (`cpf`, `nome`, `email`, `telefone`, `endereco`, `funcao`,  
`data_nasc`) VALUES
```

```
(123123123, 'Alice Johnson', 'alice@example.com', '555-6789', '789 Pine St', 'Student',  
'1995-09-09'),
```

```
(321321321, 'Bob Brown', 'bob@example.com', '555-9876', '101 Oak St', 'Student', '1996-  
08-08');
```

-- Insert data into Reserva table

```
INSERT INTO `Reserva` (`data_reserva`, `data_devolucao`, `Funcionario_cpf`,  
`Livro_isbn`, `Pessoa_cpf`) VALUES
```

```
(20240101, '2024-01-15', 123456789, 1, 123123123),
```

```
(20240201, '2024-02-15', 987654321, 2, 321321321);
```

-- Insert data into Livro_Categoria table

```
INSERT INTO `Livro_Categoria` (`Categoria_id`, `Livro_isbn`) VALUES
```

```
(1, 1),
```

```
(2, 2);
```

-- Insert data into Matricula table

```
INSERT INTO `Matricula` (`Pessoa_cpf`, `Curso_id`) VALUES
```

```
(123123123, 1),
```

```
(321321321, 2);
```