

Aluno: Tiago Gonçalves da Silva
RA: 2023644

Exercícios Livros DB - criação de banco de dados e interface

1- Criação da base de dados

```
DROP DATABASE IF EXISTS livrosdb;  
CREATE DATABASE IF NOT EXISTS livrosdb;  
USE livrosdb;
```

2- Criação da tabela livros

```
DROP TABLE IF EXISTS livros,  
autores;
```

```
CREATE TABLE livros (  
    livro_id    INT          NOT NULL,  
    titulo      VARCHAR(500) NOT NULL,  
    autor       VARCHAR(100) NOT NULL,  
    edicao       INT          NOT NULL,  
    ano         INT          NOT NULL,  
    editora     VARCHAR(100) NOT NULL,  
    PRIMARY KEY (livro_id),  
    UNIQUE KEY  (titulo)  
);
```

3- Inserção dos livros pelo csv

```
LOAD DATA INFILE '/var/lib/mysql-files/livros-db.csv'  
INTO TABLE livros  
FIELDS TERMINATED BY ';'   
ENCLOSED BY '"'   
LINES TERMINATED BY '\n'   
IGNORE 1 ROWS;
```

4- Inserção das capas

4.1 – Adicionar o campo blob na tabela

```
//Cria o campo blob  
try{  
    Connection con;  
    con = DriverManager.getConnection(url, user, password);  
  
    Statement update = con.createStatement();  
  
    String query = "ALTER TABLE livros "  
        + "    add column"  
        + "    capa mediumblob";  
    update.executeUpdate(query);  
  
    con.close();  
} catch(SQLException e1){  
    System.out.println("Erro criar conexao: " + e1);  
}
```

4.2 – Adicionar as capas na tabela

```
while(rs.next())
{
    System.out.println(rs.getInt("livro_id"));

    //Ler a capa a ser inserida
    String local = "/home/tiago145/Tudo/Desktop/UTFPR - 5 Semestre/Introdução a
    FileInputStream arq = new FileInputStream(local+rs.getString(1)+".jpg");

    //Preencher o byte array output stream
    int bytelido = arq.read();
    ByteArrayOutputStream buffer = new ByteArrayOutputStream();
    while(bytelido != -1)
    {
        buffer.write(bytelido);
        bytelido = arq.read();
    }

    //Arrumar os campos do prepared statement
    inserirCapa.setInt(2, rs.getInt(1));
    inserirCapa.setBytes(1, buffer.toByteArray());
    //Executar o comando
    inserirCapa.executeUpdate();
}
```

5- Criação da tabela autores

USE livrosdb;

SELECT 'CREATING autores' AS 'INFO';

DROP TABLE IF EXISTS autores;

```
CREATE TABLE autores(
    nome    VARCHAR(50)    NOT NULL,
    titulo  VARCHAR(500)NOT NULL,
    FOREIGN KEY (titulo) REFERENCES livros(titulo) ON DELETE CASCADE,
    PRIMARY KEY (nome,titulo)
);
```

6- Inserção dos autores na tabela

```
//Executa a leitura dos livros
String query = "SELECT título, autor FROM livros";
ResultSet rs = ler.executeQuery(query);

//Para cada livro na tabela livros
while(rs.next())
{
    //Coloca o título no prepared statement
    inserir_autor.setString(2, rs.getString("título"));
    String autores = rs.getString("autor");

    //Separa cada autor da virgula
    for(int i = 0; i < autores.length(); i++)
    {
        String autor_nome = "";
        while(i < autores.length() && autores.charAt(i) != ',')
        {
            autor_nome += autores.charAt(i);
            i++;
        }
        i++;
        //Insere o autor no prepared statement e executa o update
        inserir_autor.setString(1, autor_nome);
        inserir_autor.executeUpdate();
        //System.out.println("Livro: " + rs.getString("título") + " Autor: " + autor_nome);
    }
}
```

7- Observações

Para executar o código é necessário incluir o .jar do jdbc, editar a linha da string senha, e editar o caminho do diretório das capas.