

Código completo do programa java

```
package ConexaoBD;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

/**
 *
 * @author Eduardo
 */
public class Conexao {

    private static final String url = "jdbc:mysql://localhost:3306/mybd";
    private static final String user = "root";
    private static final String senha = "root";

    private static Connection conec;

    public static Connection Conexao() {
        try {
            if (conec == null) {
                conec = DriverManager.getConnection(url, user, senha);
                return conec;
            } else {
                return conec;
            }
        } catch (SQLException ex) {
            System.out.println("Erro de conexão "+ ex);
            return null;
        }
    }
}

/**
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this
 * license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package DAO;

import entidade.fornecedores;
import java.sql.PreparedStatement;
import ConexaoBD.Conexao;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;

/**
 *
 * @author Eduardo
 */
```

```

*/
public class fornecedoreDAO {
    PreparedStatement ps1;
    ResultSet resul;
    ArrayList<fornecedores> lista = new ArrayList<>();
    public void cadastrarFornecedores(fornecedores fornecedor){

        String sql1 = "insert into fornecedores( nomefornecedor, telefone, empresa, produto)values
        (?, ?, ?, ?)";

        try {

            ps1 = Conexao.Conexao().prepareStatement(sql1);
            ps1.setString(1, fornecedor.getNome());
            ps1.setLong(2, fornecedor.getTelefone());
            ps1.setString(3, fornecedor.getEmpresa());
            ps1.setString(4, fornecedor.getProduto());
            ps1.execute();
            ps1.close();
        }
        catch (SQLException ex) {

            System.out.println("\nErro de inserção de dados \n"+ ex);
        }
    }

    public ArrayList<fornecedores> listarFornecedores(){
        String sql = "SELECT * FROM fornecedores" ;
        try {

            ps1 = Conexao.Conexao().prepareStatement(sql);
            resul = ps1.executeQuery();

            while (resul.next()){
                fornecedores forne = new fornecedores();
                forne.setIdfornecedor(resul.getInt("idfornecedor"));
                forne.setNome(resul.getString("nomefornecedor"));
                forne.setTelefone(resul.getLong("telefone"));
                forne.setEmpresa(resul.getString("empresa"));
                forne.setProduto(resul.getString("produto"));

                lista.add(forne);
            }

        } catch (SQLException e) {
            System.out.println("Erro na listagem"+ e);
        }
        return lista;
    }
}

```

```

import ConexaoBD.Conecxao;
import entidade.produtos;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;

/**
 *
 * @author Eduardo
 */
public class produtosDAO {

    PreparedStatement prep;
    ResultSet resul;
    ArrayList<produtos> lista = new ArrayList<>();

    public void cadastrarProdutos(produtos prod) {

        String sql = "insert into produtos(dataCadastro, parafusos, saidaParafusos, aruelas,
saidaAruelas,alicates, saidaAlicates,martelos, saidaMartelos )values (?,?,?,?,?,?,?,?,?)";

        try {

            prep = Conexao.Conecxao().prepareStatement(sql);
            prep.setString(1, prod.getDataCadastro());
            prep.setInt(2, prod.getParafusos());
            prep.setInt(3, prod.getSaidaParafusos());
            prep.setInt(4, prod.getAruelas());
            prep.setInt(5, prod.getSaidaAruelas());
            prep.setInt(6, prod.getAlicate());
            prep.setInt(7, prod.getSaidaAlicates());
            prep.setInt(8, prod.getMartelo());
            prep.setInt(9, prod.getSaidaMartelos());
            prep.execute();
            prep.close();

        } catch (SQLException ex) {

            System.out.println("\nErro de inserção de dados produtos \n" + ex);
        }
    }

    public ArrayList<produtos> listarProdutos() {
        String sql = "SELECT * FROM produtos";
        try {

            prep = Conexao.Conecxao().prepareStatement(sql);
            resul = prep.executeQuery();

```

```

while (resul.next()) {
    produtos prod = new produtos();
    prod.setId_produtos(resul.getInt("id_produtos"));
    prod.setDataCadastro(resul.getString("dataCadastro"));
    prod.setParafusos(resul.getInt("parafusos"));
    prod.setSaidaParafusos(resul.getInt("saidaParafusos"));
    prod.setAruelas(resul.getInt("aruelas"));
    prod.setSaidaAruelas(resul.getInt("saidaAruelas"));
    prod.setAlicate(resul.getInt("alicates"));
    prod.setSaidaAlicates(resul.getInt("saidaAlicates"));
    prod.setMartelo(resul.getInt("martelos"));
    prod.setSaidaMartelos(resul.getInt("saidaMartelos"));

    lista.add(prod);

}

} catch (SQLException e) {
    System.out.println("Erro na listagem de produtos" + e);
}
return lista;

}

public void qnt_parafusos() {
    String sql = "select(sum(parafusos) - (select sum(saidaParafusos) from produtos)) as resultado
from produtos ";
    try {
        prep = Conecxao.Conecxao().prepareStatement(sql);

        resul = prep.executeQuery();
        resul.next();
        int qnt_parafusos = resul.getInt(1);
        System.out.print("A quantidade de parafusos é: " + qnt_parafusos + "\n \n");

    } catch (SQLException e) {
        System.out.println("Erro na Quantidade de parafusos" + e);
    }
}

public void qnt_aruelas() {
    String sql = "select(sum(aruelas) - (select sum(saidaAruelas)from produtos))as resultado from
produtos";
    try {
        prep = Conecxao.Conecxao().prepareStatement(sql);
        resul = prep.executeQuery();
        resul.next();
        int qnt_aruelas = resul.getInt(1);
        System.out.print("A quantidade de aruelas é: " + qnt_aruelas + "\n \n");
    } catch (SQLException e) {

```

```

        System.out.println("Erro no resulta de aruelas \n " + e);
    }

}

public void qnt_alicate() {
    String sql = "select(sum(alicates) - (select sum(saidaAlicates)from produtos)) as resultado
from produtos";
    try {
        prep = Conexao.Conexao().prepareStatement(sql);
        resul = prep.executeQuery();
        resul.next();
        int qnt_alicates = resul.getInt(1);
        System.out.print("A quantidade de alicates é: " + qnt_alicates + "\n \n");
    } catch (SQLException e) {
        System.out.println("Erro no resulta de alicates \n " + e);
    }

}

public void qnt_martelos() {
    String sql = "select(sum(martelos) - (select sum(saidaMartelos)from produtos)) as resultado
from produtos";
    try {
        prep = Conexao.Conexao().prepareStatement(sql);
        resul = prep.executeQuery();
        resul.next();
        int qnt_martelos = resul.getInt(1);
        System.out.print("A quantidade de martelos é: " + qnt_martelos+ "\n \n");
    } catch (SQLException e) {
        System.out.println("Erro no resulta de martelos \n " + e);
    }
}
}

```

```

package DAO;
import entidade.Usuario;
import java.sql.PreparedStatement;
import ConexaoBD.Conexao;
import java.sql.SQLException;

```

```

/**
 *
 * @author Eduardo
 */

```

```

public class usuarioDAO {
    public void cadastrarUsuario(Usuario usuario){

        String sql = "INSERT INTO USUÁRIO(nome, cargo, registroDefuncionario) VALUES (?, ?, ?)";

        PreparedStatement ps;
        try {

```

```

        ps = Conecxao.Conecxao().prepareStatement(sql);
        ps.setString(1, usuario.getNome());
        ps.setString(2, usuario.getCargo());
        ps.setString(3, usuario.getRegistroDeFuncionario());
        ps.execute();
        ps.close();
    } catch (SQLException ex) {
        ex.printStackTrace();
        System.out.println("Erro de inserção de dados");
    }
}
}
}
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this
license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package LoginAndSenha;

import java.util.Scanner;

/**
 *
 * @author Eduardo
 */
public class Login {

    /**
     * @param entrada the entrada to set
     */
    public void setEntrada(String entrada) {
        int cont=1;
        while(!entrada.equals("unicesumar")&& cont <=2){
            System.out.println("\nLogin errado.\nTentativas restante: "+(2-cont)+"\n");
            System.out.println("Login.");
            Scanner t = new Scanner(System.in);
            cont++;
            entrada = t.next();
        }
        if (entrada.equals("unicesumar")) {
            System.out.println("Login correto\n");
        }
        else{
            System.out.println("Tentativas excidas.\n!!!!!!Bloqueado!!!!!!");
            System.exit(0);
        }
    }
}

/**
 * @param entrada1

```

```

*/
public void setEntrada1(String entrada1) {
    int cont1=1;
    //usei meu Ra como senha.
    while(!entrada1.equals("20051512-5")&& cont1 <=2){
        System.out.println("\nSenha errada.\nTentativas restante: "+(2-cont1)+"\n");
        System.out.println("senha.");
        Scanner t = new Scanner(System.in);
        cont1++;
        entrada1 = t.next();
    }
    if (entrada1.equals("20051512-5")) {
        System.out.println("Senha correta\n");
    }
    else{
        System.out.println("Tentativas excidas.\n!!!!!!Bloqueado!!!!!!");
        System.exit(0);
    }
}

/**
 * @param senhaAdmim the senhaAdmim to set
 */
public void setSenhaAdmim(String senhaAdmim) {
    int cont2 = 1;
    while(!senhaAdmim.equals("1234")&& cont2 <=2){
        System.out.println("\nSenha do administrador errada.\nTentativas restante: "+(2-cont2)+"\n");
        Scanner t = new Scanner(System.in);
        cont2++;
        senhaAdmim = t.next();
    }
    if (senhaAdmim.equals("1234")) {
        System.out.println("Senha do administrador correta");
    }
    else{
        System.out.println("Tentativas excidas.\n!!!!!!Bloqueado!!!!!!");
        System.exit(0);
    }
}
}

```

package camel;

```

/**
 *
 * @author Eduardo
 */
public class Camel {

```

```

/**
 *
 * @param nomeCamel
 * @return
 */
public String tranforCamelCase(String nomeCamel){
    nomeCamel = nomeCamel.toLowerCase();
    String[] partes = nomeCamel.split("_");
    String camelCase = "";
    for (String parte : partes) {
        char primeiraLetra = parte.charAt(0);
        String primeira = Character.toString(primeiraLetra);
        String resto = parte.substring(1);
        camelCase = camelCase.concat(primeira.toUpperCase()).concat(resto).concat("_");
    }
    return camelCase;
}
}
package entidade;

/**
 *
 * @author Eduardo
 */
public class Usuario {
    private String nome;
    private String cargo;
    private String registroDeFuncionario;

    /**
     * @return the nome
     */
    public String getNome() {
        return nome;
    }

    /**
     * @param nome the nome to set
     */
    public void setNome(String nome) {
        this.nome = nome;
    }

    /**
     * @return the cargo
     */
    public String getCargo() {
        return cargo;
    }

    /**
     * @param cargo the cargo to set

```



```

    */
    public void setCargo(String cargo) {
        this.cargo = cargo;
    }

    /**
     * @return the registroDeFuncionario
     */
    public String getRegistroDeFuncionario() {
        return registroDeFuncionario;
    }

    /**
     * @param registroDeFuncionario the registroDeFuncionario to set
     */
    public void setRegistroDeFuncionario(String registroDeFuncionario) {
        this.registroDeFuncionario = registroDeFuncionario;
    }
}
package entidade;

/**
 *
 * @author Eduardo
 */
public class fornecedores {

    private String nome;
    private long telefone;
    private String empresa;
    private String produto;
    private int idfornecedor;
    /**
     * @return the nome
     */
    public String getNome() {
        return nome;
    }

    /**
     * @param nome the nome to set
     */
    public void setNome(String nome) {
        this.nome = nome;
    }

    /**
     * @return the empresa
     */

```

```
public String getEmpresa() {
    return empresa;
}

/**
 * @param empresa the empresa to set
 */
public void setEmpresa(String empresa) {
    this.empresa = empresa;
}

/**
 * @return the produto
 */
public String getProduto() {
    return produto;
}

/**
 * @param produto the produto to set
 */
public void setProduto(String produto) {
    this.produto = produto;
}

/**
 * @return the idfornecedor
 */
public int getIdfornecedor() {
    return idfornecedor;
}

/**
 * @param idfornecedor the idfornecedor to set
 */
public void setIdfornecedor(int idfornecedor) {
    this.idfornecedor = idfornecedor;
}

/**
 * @return the telefone
 */
public long getTelefone() {
    return telefone;
}

/**
 * @param telefone the telefone to set
 */
public void setTelefone(long telefone) {
    this.telefone = telefone;
}}
```

```
package entidade;
```

```
/**
```

```
*
```

```
* @author Eduardo
```

```
*/
```

```
public class produtos {
```

```
    private int id_produtos;
```

```
    private String dataCadastro;
```

```
    private int parafusos;
```

```
    private int aruelas;
```

```
    private int alicate;
```

```
    private int martelo;
```

```
    private int saidaParafusos;
```

```
    private int saidaAruelas;
```

```
    private int saidaAlicates;
```

```
    private int saidaMartelos;
```

```
/**
```

```
* @return the parafusos
```

```
*/
```

```
public int getParafusos() {
```

```
    return parafusos;
```

```
}
```

```
/**
```

```
* @param parafusos the parafusos to set
```

```
*/
```

```
public void setParafusos(int parafusos) {
```

```
    this.parafusos = parafusos;
```

```
}
```

```
/**
```

```
* @return the ruelas
```

```
*/
```

```
public int getAruelas() {
```

```
    return aruelas;
```

```
}
```

```
/**
```

```
* @param ruelas the ruelas to set
```

```
*/
```

```
public void setAruelas(int ruelas) {
```

```
    this.aruelas = ruelas;
```

```
}
```

```
/**
```

```
* @return the alicate
```

```
*/
```

```

public int getAlicate() {
    return alicate;
}

/**
 * @param alicate the alicate to set
 */
public void setAlicate(int alicate) {
    this.alicate = alicate;
}

/**
 * @return the martelo
 */
public int getMartelo() {
    return martelo;
}

/**
 * @param martelo the martelo to set
 */
public void setMartelo(int martelo) {
    this.martelo = martelo;
}

/**
 * @return the dataCadastro
 */
public String getDataCadastro() {
    return dataCadastro;
}

/**
 * @param dataCadastro the dataCadastro to set
 */
public void setDataCadastro(String dataCadastro) {
    this.dataCadastro = dataCadastro;
}

/**
 * @return the id_produtos
 */
public int getId_produtos() {
    return id_produtos;
}

/**
 * @param id_produtos the id_produtos to set
 */
public void setId_produtos(int id_produtos) {
    this.id_produtos = id_produtos;
}

```

```

/**
 * @return the saidaParafusos
 */
public int getSaidaParafusos() {
    return saidaParafusos;
}

/**
 * @param saidaParafusos the saidaParafusos to set
 */
public void setSaidaParafusos(int saidaParafusos) {
    this.saidaParafusos = saidaParafusos;
}

/**
 * @return the saidaAruelas
 */
public int getSaidaAruelas() {
    return saidaAruelas;
}

/**
 * @param saidaAruelas the saidaAruelas to set
 */
public void setSaidaAruelas(int saidaAruelas) {
    this.saidaAruelas = saidaAruelas;
}

/**
 * @return the saidaAlicates
 */
public int getSaidaAlicates() {
    return saidaAlicates;
}

/**
 * @param saidaAlicates the saidaAlicates to set
 */
public void setSaidaAlicates(int saidaAlicates) {
    this.saidaAlicates = saidaAlicates;
}

/**
 * @return the saidaMartelos
 */
public int getSaidaMartelos() {
    return saidaMartelos;
}

/**
 * @param saidaMartelos the saidaMartelos to set

```

```

    */
    public void setSaidaMartelos(int saidaMartelos) {
        this.saidaMartelos = saidaMartelos;
    }
}
package javacadastro;

import LoginAndSenha.Login;
import entidade.Usuario;
import java.util.Scanner;
import DAO.usuarioDAO;
import entidade.fornecedores;
import DAO.fornecedoreDAO;
import DAO.produtosDAO;
import camel.Camel;
import entidade.produtos;
import java.text.SimpleDateFormat;
import java.util.Date;

/**
 *
 * @author Eduardo
 */
public class JavaCadastro {

    public static void main(String[] args) {
        Date hoje = new Date();
        SimpleDateFormat dataFormatar = new SimpleDateFormat("dd/MM/YYYY");
        String dataformt = dataFormatar.format(hoje);

        Scanner t = new Scanner(System.in);
        Login login = new Login();
        Camel camel = new Camel();
        System.out.println("Login");
        String entrada = t.next();

        login.setEntrada(entrada);

        System.out.println("Senha");
        String entrada1 = t.next();

        //login.setEntrada1(entrada1);
        boolean rodando = true;
        while (rodando) {
            produtos prod = new produtos();
            System.out.println("[1] Cadastrado de usuarios");
            System.out.println("[2] Cadastro de fornecedores");
            System.out.println("[3] Cadastro de produtos/estoque e saida de produtos");
            System.out.println("[4] Listar fornecedores");
            System.out.println("[5] Listar produtos cadastrados");
            System.out.println("[6] Lista quantidade de produtos");
            System.out.println("[7] Sair\n");

```

```

System.out.println("Digite sua opção");
int op = t.nextInt();

switch (op) {

    case 1 -> {

        Usuario u = new Usuario();
        System.out.println("Digite o nome do usuário");
        String NomeUser = t.next();
        NomeUser = camel.tranforCamelCase(NomeUser);
        u.setNome(NomeUser);

        System.out.println("Digite o Cargo do usuário");
        String cargoUser = t.next();
        cargoUser = camel.tranforCamelCase(cargoUser);
        u.setCargo(cargoUser);

        System.out.println("Digite o registro do Usuario");
        String registroUser = t.next();
        u.setRegistroDeFuncionario(registroUser);

        new usuarioDAO().cadastarUsuario(u);
        break;
    }
    case 2 -> {
        System.out.println("Digite a senha do administrador");
        String senhaAdmim = t.next();
        login.setSenhaAdmim(senhaAdmim);

        fornecedores f = new fornecedores();

        System.out.println("Digite o nome do fornecedor");
        String nomefor = t.next();
        nomefor = camel.tranforCamelCase(nomefor);
        f.setNome(nomefor);

        System.out.println("Digite o telefone do fornecedor");
        long telFor = t.nextLong();
        f.setTelefone(telFor);

        System.out.println("Digite o nome da empresa");
        String nomeEmp = t.next();
        nomeEmp = camel.tranforCamelCase(nomeEmp);
        f.setEmpresa(nomeEmp);

        System.out.println("Digite o produto da empresa Parafuso, aruela, alicates ou martelo");
        String produto = t.next();
        produto = camel.tranforCamelCase(produto);
        f.setProduto(produto);

        new fornecedorDAO().cadastrarFornecedores(f);
    }
}

```

```

        break;
    }
    case 3 -> {
        boolean run = true;
        while (run) {

            System.out.println("[1] Cadastrar parafusos");
            System.out.println("[2] Cadastrar aruelas");
            System.out.println("[3] Cadastrar alicates");
            System.out.println("[4] Cadastrar martelos");
            System.out.println("[5] Saida de parafusos");
            System.out.println("[6] Saida de aruelas");
            System.out.println("[7] Saida de alicates");
            System.out.println("[8] Saida de martelos");
            System.out.println("[9] Voltar\n");
            System.out.println("Qual produto deseja cadastrar?");
            int opç = t.nextInt();
            switch (opç) {

                case 1 -> {

                    System.out.println("Digite a quantidade de parafusos");
                    int paraf = t.nextInt();
                    prod.setParafusos(paraf);

                }
                case 2 -> {

                    System.out.println("Digite a quantidade de aruelas");
                    int aru = t.nextInt();
                    prod.setAruelas(aru);

                }
                case 3 -> {

                    System.out.println("Digite a quantidade de alicates");
                    int ali = t.nextInt();
                    prod.setAlicate(ali);

                }
                case 4 -> {

                    System.out.println("Digite a quantidade de martelos");
                    int mar = t.nextInt();
                    prod.setMartelo(mar);

                }
                case 5 -> {
                    System.out.println("Digite quantidade de reirada de parafusos");
                    int saidaPara = t.nextInt();
                    prod.setSaidaParafusos(saidaPara);
                }
            }
        }
    }
}

```



```

    }
    case 6 -> {
        System.out.println("Digite a quantidade de retirada de aruelas");
        int saidaAru = t.nextInt();
        prod.setSaidaAruelas(saidaAru);
    }
    case 7 -> {
        System.out.println("Digite a quantidade de retirada de alicates");
        int saidaAli = t.nextInt();
        prod.setSaidaAlicates(saidaAli);
    }
    case 8 -> {
        System.out.println("Digite a quantidade de retirada de martelos");
        int saidaMarte = t.nextInt();
        prod.setSaidaMartelos(saidaMarte);
    }
    case 9 -> {
        run = false;
    }
    default ->
        System.out.println("Opção invalida");
}

}

prod.setDataCadastro(dataformt);
new produtosDAO().cadastrarProdutos(prod);
}
case 4 -> {
    System.out.println("Digite a senha do administrador");
    String senhaAdmim = t.next();
    login.setSenhaAdmim(senhaAdmim);

    listaFornecedor listafor = new listaFornecedor();
    listafor.setLocationRelativeTo(null);
    listafor.setVisible(true);
}
case 5 -> {
    listarProdutos listarPro = new listarProdutos();
    listarPro.setLocationRelativeTo(null);
    listarPro.setVisible(true);
}
case 6 -> {
    produtosDAO produ = new produtosDAO();
    boolean run = true;
    while (run) {
        System.out.println("[1] Quantidade de parafusos");
        System.out.println("[2] Quantidade de aruelas");
        System.out.println("[3] Quantiade de alicates");
    }
}

```

```
System.out.println("[4] Quantidade de martelos");
System.out.println("[5] Voltar");
```

```
System.out.println("Digite sua opção");
int opc = t.nextInt();
switch (opc) {
    case 1 -> {
        produ.qnt_parafusos();
    }
    case 2 -> {
        produ.qnt_aruelas();
    }
    case 3 -> {
        produ.qnt_alicate();
    }
    case 4 -> {
        produ.qnt_martelos();
    }
    case 5 -> {
        run = false;
    }
    default ->
        System.out.println("Opção invalida");
}
```

```
}
```

```
}
case 7 -> {
    rodando = false;
}
default ->
    System.out.println("Opção invalida");
```

```
}
```

```
}
```

```
}
```

```
}
```