

Projeto: IngreSISEmissorIngresso

Arquivos:

- Pacotes de Códigos-fonte
 - Imagens
 - controller
 - EventoDao.java
 - model.bean
 - EventoModel.java
 - model.conexao
 - Conexao.java
 - principal
 - IngreSISEmissorIngresso.java
 - view
 - EventoCadastroView.java
 - IngressoCadastroView.java
 - LoginView.java
 - LoteCadastroView.java
 - MenuView.java
 - UsuarioCadastroView.java
 - VendaIngressoView.java

Navegador:

Membros:

- EventoDao
 - alterar(EventoModel eventoModel)
 - cadastrar(EventoModel eventoModel)
 - consultar(String nomeEvento): List<EventoModel>
 - excluir(EventoModel eventoModel)
- con: Connection
- ps: PreparedStatement
- rs: ResultSet
- sql: String

Código-Fonte:

```
19 public void cadastrar(EventoModel eventoModel) {
20
21     con = Conexao.getConexao();
22     sql = "insert into evento(nomeEvento, localEvento, dataEvento, horarioEvento, statusEvento) values (?, ?, ?, ?, 'ATIVO')";
23     try {
24         ps = (PreparedStatement) con.prepareStatement(sql);
25         ps.setString(1, eventoModel.getNomeEvento());
26         ps.setString(2, eventoModel.getLocalEvento());
27         ps.setString(3, eventoModel.getDataEvento());
28         ps.setString(4, eventoModel.getHorarioEvento());
29         ps.execute();
30         JOptionPane.showMessageDialog(null, "Novo Evento cadastrado com sucesso.");
31     } catch (Exception e) {
32         e.getMessage();
33         JOptionPane.showMessageDialog(null, "Não foi possível cadastrar o Evento.");
34     } finally {
35         Conexao.closeConexao(con);
36     }
37 }
38
39 public List<EventoModel> consultar(String nomeEvento) {
40
41     con = Conexao.getConexao();
42     sql = "select *from evento where nomeEvento = '" + nomeEvento + "' and statusEvento = 'ATIVO'";
43     List<EventoModel> lista = new ArrayList<EventoModel>();
44     try {
45         ps = (PreparedStatement) con.prepareStatement(sql);
46         rs = ps.executeQuery();
47         while (rs.next()) {
48             EventoModel evento = new EventoModel();
49             evento.setIdEvento(rs.getInt("idEvento"));
50             evento.setNomeEvento(rs.getString("nomeEvento"));
51             evento.setLocalEvento(rs.getString("localEvento"));
52             evento.setDataEvento(rs.getString("dataEvento"));
53             evento.setHorarioEvento(rs.getString("horarioEvento"));
54             lista.add(evento);
55         }
56     } catch (Exception e) {
```

Projetos x Arquivos Serviços

IngreSISEmissorIngresso

- Pacotes de Códigos-fonte
 - Imagens
 - controller
 - EventoDao.java
 - model.bean
 - EventoModel.java
 - model.conexao
 - Conexao.java
 - principal
 - IngreSISEmissorIngresso.java
 - view
 - EventoCadastroView.java
 - IngressoCadastroView.java
 - LoginView.java
 - LoteCadastroView.java
 - MenuView.java
 - UsuarioCadastroView.java

Navegador x

Membros

EventoModel

- EventoModel()
- getDataEvento(): String
- getHorarioEvento(): String
- getIdEvento(): Integer
- getLocalEvento(): String
- getNomeEvento(): String
- setDataEvento(String dataEvento)
- setHorarioEvento(String horarioEvento)
- setIdEvento(Integer idEvento)
- setLocalEvento(String localEvento)
- setNomeEvento(String nomeEvento)
- dataEvento: String
- horarioEvento: String
- idEvento: Integer
- localEvento: String
- nomeEvento: String

Página Inicial x MenuView.java x EventoDao.java x EventoModel.java x Conexao.java x

Código-Fonte Histórico

```
4
5
6 private Integer idEvento;
7 private String nomeEvento;
8 private String localEvento;
9 private String dataEvento;
10 private String horarioEvento;
11
12 public EventoModel() {
13
14 }
15
16 public Integer getIdEvento() {
17     return idEvento;
18 }
19
20 public void setIdEvento(Integer idEvento) {
21     this.idEvento = idEvento;
22 }
23
24 public String getNomeEvento() {
25     return nomeEvento;
26 }
27
28 public void setNomeEvento(String nomeEvento) {
29     this.nomeEvento = nomeEvento;
30 }
31
32 public String getLocalEvento() {
33     return localEvento;
34 }
35
36 public void setLocalEvento(String localEvento) {
37     this.localEvento = localEvento;
38 }
39
40 public String getDataEvento() {
41     return dataEvento;
42 }
```

Projeto: IngreSISEmissorIngresso

Arquivos:

- Pacotes de Códigos-fonte
 - Imagens
 - controller
 - EventoDao.java
 - model.bean
 - EventoModel.java
 - model.conexao
 - Conexao.java
 - principal
 - IngreSISEmissorIngresso.java
 - view
 - EventoCadastroView.java
 - IngressoCadastroView.java
 - LoginView.java
 - LoteCadastroView.java
 - MenuView.java
 - UsuarioCadastroView.java

Navegador

Membros

- Conexao
 - closeConexao(Connection con)
 - getConexao(): Connection
 - main(String[] args)

Código-Fonte

```
6
7
8 public class Conexao {
9
10
11 public static Connection getConexao() {
12
13     Connection con = null;
14
15     String host = "localhost";
16     String porta = "3306";
17     String nomeBanco = "dbingresis";
18     String usuario = "root";
19     String senha = "1234";
20
21     String urlBanco = "jdbc:mysql://" + host + ":" + porta + "/" + nomeBanco;
22
23     try {
24         Class.forName("com.mysql.jdbc.Driver").newInstance();
25         con = DriverManager.getConnection(urlBanco, usuario, senha);
26         System.out.println("Conexão OK!");
27     } catch (Exception erro) {
28         erro.getMessage();
29         System.out.println("Não foi possível conectar ao banco de dados!");
30     }
31     return con;
32 }
33
34 public static void closeConexao(Connection con) {
35     try {
36         con.close();
37         System.out.println("Conexão com o Banco finalizado com sucesso!");
38     } catch (Exception e) {
39         e.getMessage();
40     }
41 }
42
43 public static void main(String[] args) {
```

Projetos x Arquivos Serviços

IngreSISEmissorIngresso

- Pacotes de Códigos-fonte
 - Imagens
 - controller
 - EventoDao.java
 - model.bean
 - EventoModel.java
 - model.conexao
 - Conexao.java
 - principal
 - IngreSISEmissorIngresso.java
 - view
 - EventoCadastroView.java

Navegador x

Membros <vazio>

EventoCadastroView :: JFrame

- EventoCadastroView()
- Novo()
- alterar()
- btAlterarActionPerformed(ActionEvent evt)
- btCancelarActionPerformed(ActionEvent evt)
- btConsultarActionPerformed(ActionEvent evt)
- btExcluirActionPerformed(ActionEvent evt)
- btNovoActionPerformed(ActionEvent evt)
- btSalvarActionPerformed(ActionEvent evt)
- carregaRegistroSelecionado()
- carregaListagemTeste(String eventoPesquisa)
- consultar()
- desabilitaBotoesCrud()
- desabilitaBotoesManterCrud()
- desabilitaCampos()
- desabilitaGridListagem()
- excluir()
- habilitaBotoesCrud()
- habilitaBotoesManterCrud()
- habilitaCampos()
- habilitaGridListagem()
- iniciarTela()

Página Inicial x MenuView.java x EventoDao.java x EventoModel.java x Conexao.java x EventoCadastroView.java x

Código-Fonte Projeto Histórico

```
1 package view;
2
3 import controller.EventoDao;
4 import java.util.ArrayList;
5 import java.util.List;
6 import javax.swing.JOptionPane;
7 import javax.swing.JTable;
8 import javax.swing.table.DefaultTableModel;
9 import model.bean.EventoModel;
10
11
12 public class EventoCadastroView extends javax.swing.JFrame {
13
14     public EventoCadastroView() {
15         initComponents();
16         this.setLocationRelativeTo(null);
17         iniciarTela();
18     }
19
20     int statusAcao = 0; //1 - Novo / 2 - Alterar
21
22     private void habilitaBotoesCrud() {
23         btAlterar.setEnabled(true);
24         btExcluir.setEnabled(true);
25         btConsultar.setEnabled(true);
26         btExcluir.setEnabled(true);
27         btNovo.setEnabled(true);
28     }
29
30     private void desabilitaBotoesCrud() {
31         btAlterar.setEnabled(false);
32         btExcluir.setEnabled(false);
33         btConsultar.setEnabled(false);
34         btExcluir.setEnabled(false);
35         btNovo.setEnabled(false);
36     }
37
38     private void habilitaBotoesManterCrud() {
```

