

Struts 2 – CRUD com Hibernate

Passos Iniciais

1. Preparar o ambiente para o Hibernate e Struts 2 (veja anteriores)
2. Verificar os arquivos web.xml e hibernate.cfg.xml se estão condizentes com o projeto que está trabalhando.

Criar a Base de Dados

1. Criar a base chamada basealuno: CREATE DATABASE basecurso;
2. Entrar nesta base de dados: USE basecurso;
3. Criar a tabela de aluno: CREATE TABLE aluno (matricula int not null, nome varchar(80), endereco varchar(80), primary key (matricula));
4. Lembre-se de adicionar o mapeamento da classe Aluno para o arquivo hibernate.cfg.xml.

Fontes dos Arquivos

mapa\Aluno.java

```
package mapa;

import javax.persistence.Entity;
import javax.persistence.Id;

@Entity
public class Aluno {

    @Id
    private int matricula;
    private String nome;
    private String endereco;

    public int getMatricula() {
        return matricula;
    }
    public void setMatricula(int matricula) {
        this.matricula = matricula;
    }

    public String getNome() {
        return nome;
    }
    public void setNome(String nome) {
        this.nome = nome;
    }

    public String getEndereco() {
        return endereco;
    }
    public void setEndereco(String endereco) {
        this.endereco = endereco;
    }
}
```

actions\conexao.java

```
package actions;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;
import com.opensymphony.xwork2.ActionSupport;

@SuppressWarnings("serial")
public class Conexao extends ActionSupport {
```

```

protected SessionFactory sessionFactory;
protected Session session;

protected void conectar() throws Throwable {
    sessionFactory = new Configuration().configure().buildSessionFactory();
    session = sessionFactory.openSession();
    session.beginTransaction();
}
protected void desconectar() throws Throwable {
    session.getTransaction().commit();
    session.close();
}
}

```

actions/ListagemAction.java

```

package actions;

import java.util.ArrayList;
import java.util.List;
import org.hibernate.Query;
import mapa.Aluno;

@SuppressWarnings("serial")
public class ListagemAction extends Conexao {

    public List<Aluno> getObterLista() {
        List<Aluno> alunos = new ArrayList<Aluno>();
        try {
            conectar();
            Query query = session.createQuery("from Aluno");
            alunos = query.list();
            desconectar();
        } catch (Throwable e) {}
        return alunos;
    }
}

```

actions/ModificarAction.java

```

package actions;

import mapa.Aluno;

@SuppressWarnings("serial")
public class ModificarAction extends Conexao {

    private Aluno aluno;

    public void setAluno(Aluno aluno) {
        this.aluno = aluno;
    }
    public Aluno getAluno() {
        return aluno;
    }

    public String inserir() {
        try {
            conectar();
            session.save(aluno);
            desconectar();
            return "sucesso";
        } catch (Throwable e) {}
        return "fracasso";
    }
}

```

WEB-INF/classes/struts.xml

```

<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE struts PUBLIC

```

```

"-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"
"http://struts.apache.org/dtds/struts-2.0.dtd">

<struts>
  <constant name="struts.enable.DynamicMethodInvocation" value="false" />
  <constant name="struts.devMode" value="false" />

  <package name="default" namespace="/" extends="struts-default">
    <default-action-ref name="inicio" />
    <global-results>
      <result name="error">/error.jsp</result>
    </global-results>
    <global-exception-mappings>
      <exception-mapping exception="java.lang.Exception" result="error"/>
    </global-exception-mappings>

    <action name="inicio" class="actions.ListagemAction">
      <result>/pages/pagina01.jsp</result>
    </action>

    <action name="cadastrar" class="actions.ModificarAction" method="inserir">
      <result name="sucesso" type="redirect">/inicio.action</result>
      <result name="fracasso">/pages/naoCadastrado.jsp</result>
    </action>
  </package>
</struts>

```

index.jsp

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
  <META HTTP-EQUIV="Refresh" CONTENT="0;URL=inicio.action">
</head>
<body>
<p>Loading ...</p>
</body>
</html>

```

/pages/pagina01.jsp

```

<%@ taglib prefix="s" uri="/struts-tags" %>

<h2>Novo Aluno</h2>
<s:form action="cadastrar">
  <s:textfield name="aluno.matricula" label="Matrícula" />
  <s:textfield name="aluno.nome" label="Nome" />
  <s:textfield name="aluno.endereco" label="Endereço" />
  <s:submit />
</s:form>

<h2>Alunos já Cadastrados</h2>
<table>
  <tr>
    <th>Matrícula</th>
    <th style="width:250px">Nome</th>
    <th style="width:250px">Endereço</th>
    <th>Ação</th>
  </tr>
  <s:iterator value="obterLista">
    <tr>
      <td><s:property value="matricula" /></td>
      <td><s:property value="nome" /></td>
      <td><s:property value="endereco" /></td>
      <td>
        <s:a action="alterar">
          <s:param name="matricula"><s:property value="matricula" /></s:param>Alt.
        </s:a>
        <s:a action="excluir">
          <s:param name="matricula"><s:property value="matricula" /></s:param>Exc.
        </s:a>
      </td>
    </tr>
  </s:iterator>
</table>

```

```
</td>
</tr>
</s:iterator>
</table>
```

pages\naoCadastrado.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<%@ taglib prefix="s" uri="/struts-tags" %>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
</head>
<body>
Registro não cadastrado
</body>
</html>
```