

Criando um CRUD com Eclipse e Hibernate

Passos Iniciais

1. Baixar o Hibernate de www.hibernate.org e a biblioteca JSTL em <http://jstl.java.net/>
2. No Eclipse após criar o projeto, ir em Properties | Java Build Path e aba Libraries
3. Botão "Add Library..."
4. Opção "User Library"
5. Selecionar o nome para a nova biblioteca
6. Adicionar as seguintes classes:
 1. Hibernate3.jar
 2. Todas da Pasta lib\required
 3. Todas da Pasta lib\jpa
 4. Todas da Pasta lib\bytecode\javassist
7. Adicionar o driver JDBC do banco

Mapeamentos no Arquivo de Configuração

1. Salvar um arquivo com o nome hibernate.cfg.xml na pasta src.
2. Criar o mapeamento DTO
3. Apontar esse mapeamento na tag MAPPING do hibernate.cfg.xml

Para usar a API do Hibernate

1. Criar a Session Factory
2. Criar um objeto Session para a Session Factory
3. Usar o objeto Session para salvar o objeto

Fontes dos Arquivos

hibernate.cfg.xml

```
<?xml version='1.0' encoding='utf-8'?>
<!DOCTYPE hibernate-configuration PUBLIC
"-//Hibernate/Hibernate Configuration DTD 3.0//EN"
"http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">

<hibernate-configuration>
  <session-factory>
    <property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
    <property name="hibernate.connection.url">jdbc:mysql://localhost:3306/hibernate?
useUnicode=true</property>
    <property name="hibernate.connection.username">root</property>
    <property name="hibernate.connection.password">root</property>
    <property name="connection.pool_size">1</property>
    <property name="dialect">org.hibernate.dialect.MySQLDialect</property>
    <property name="cache.provider_class">org.hibernate.cache.NoCacheProvider</property>
    <property name="show_sql">>false</property>
    <property name="hibernate.hbm2ddl.auto">create</property> <!-- update -->
    <!-- Mapeamentos -->
    <mapping class="org.dto.UserDetails" />
  </session-factory>
</hibernate-configuration>
```

org.dto.userDetails.java

```
package org.dto;

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

@Entity
```

```

public class UserDetails {
    @Id
    private int userId;
    private String userName;

    public int getUserId() {
        return userId;
    }
    public void setUserId(int userId) {
        this.userId = userId;
    }

    public String getUserName() {
        return userName;
    }
    public void setUserName(String userName) {
        this.userName = userName;
    }
}

```

org.dao.userDao.java

```

package org.dao;

import java.util.ArrayList;
import java.util.List;

import org.dto.UserDetails;
import org.hibernate.Query;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;

public class UserDAO {

    private SessionFactory sessionFactory;
    private Session session;
    private UserDetails user;
    private int codId;

    public void setCodId(int codId) {
        this.codId = codId;
    }

    public UserDetails getUser() {
        return user;
    }
    public void setUser(UserDetails user) {
        this.user = user;
    }

    private void abrirTransacao() {
        sessionFactory = new Configuration().configure().buildSessionFactory();
        session = sessionFactory.openSession();
        session.beginTransaction();
    }
    private void fecharTransacao() {
        session.getTransaction().commit();
        session.close();
    }

    public List<UserDetails> getListar() {
        List<UserDetails> lista = new ArrayList<UserDetails>();
        try {
            abrirTransacao();
            Query query = session.createQuery("from UserDetails");
            lista = query.list();
            fecharTransacao();
        } catch (Throwable e) {
        }
        return lista;
    }
}

```

```

}
public boolean getObter() {
    try {
        abrirTransacao();
        user = (UserDetails)session.get(UserDetails.class, codId);
        fecharTransacao();
        return true;
    } catch (Throwable e) {
        System.out.println("ID: " + e.getMessage());
        return false;
    }
}
public boolean getIncluir() {
    try {
        abrirTransacao();
        session.save(user);
        fecharTransacao();
        return true;
    } catch (Throwable e) {
        return false;
    }
}
public boolean getAlterar() {
    try {
        abrirTransacao();
        session.update(user);
        fecharTransacao();
        return true;
    } catch (Throwable e) {
        return false;
    }
}
public boolean getExcluir() {
    try {
        abrirTransacao();
        session.delete(user);
        fecharTransacao();
        return true;
    } catch (Throwable e) {
        return false;
    }
}
}
}

```

WEB-INF/jspf/cabec.jspf

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Controle de Usuários</title>

```

index.jsp

```

<%@include file="WEB-INF/jspf/cabec.jspf" %>
<body>
<h3>Cadastrar Usuário</h3>
<form action="userIC.jsp">
    Identificação: <input name="id" size="5"/><br/>
    Nome: <input name="nome" size="50"/><br/><br/>
    <input type="submit"/>
</form>

<h3>Usuários Cadastrados</h3>
<table style="width:80%" border=1>
<tr>
<th style="width:50px">ID</th>

```

```

<th>Nome</th>
<th style="width:80px">Ação</th>
</tr>
<jsp:useBean id="dao" class="org.dao.UserDAO" scope="page" />
<c:forEach var="col" items="${dao.listar}">
<tr>
<td>${col.userId}</td>
<td>${col.userName}</td>
<td>
<a href="userAE.jsp?id=${col.userId}">ALT</a>
<a href="userEC.jsp?id=${col.userId}">EXC</a>
</td>
</tr>
</c:forEach>
</table>
</body>
</html>

```

userIC.jsp

```

<jsp:useBean id="user" class="org.dto.UserDetails" scope="page" />
<jsp:setProperty name="user" property="userId" value="${param.id}"/>
<jsp:setProperty name="user" property="userName" value="${param.nome}"/>
<jsp:useBean id="dao" class="org.dao.UserDAO" scope="page" />
<jsp:setProperty name="dao" property="user" value="${user}"/>
${dao.incluir}

```

userAE.jsp

```

<%@include file="WEB-INF/jspf/cabec.jspf" %>
<jsp:useBean id="dao" class="org.dao.UserDAO" scope="page" />
<jsp:setProperty name="dao" property="codId" value="${param.id}"/>
<body>
<h3>Modificar as Usuário</h3>
<c:choose>
<c:when test="${dao.obter and dao.user != null}">
<form action="userAC.jsp">
Identificação: <input name="id" size="5" readonly value="${dao.user.userId}"/><br/>
Nome: <input name="nome" size="50" value="${dao.user.userName}"/><br/><br/>
<input type="submit"/>
</form>
</c:when>
<c:otherwise>Registro não existe!</c:otherwise>
</c:choose>
</body>
</html>

```

userAC.jsp

```

<jsp:useBean id="user" class="org.dto.UserDetails" scope="page" />
<jsp:setProperty name="user" property="userId" value="${param.id}"/>
<jsp:setProperty name="user" property="userName" value="${param.nome}"/>
<jsp:useBean id="dao" class="org.dao.UserDAO" scope="page" />
<jsp:setProperty name="dao" property="user" value="${user}"/>
${dao.alterar}

```

userEC.jsp

```

<jsp:useBean id="user" class="org.dto.UserDetails" scope="page" />
<jsp:setProperty name="user" property="userId" value="${param.id}"/>
<jsp:useBean id="dao" class="org.dao.UserDAO" scope="page" />
<jsp:setProperty name="dao" property="user" value="${user}"/>
${dao.excluir}

```