

Introdução ao Desenvolvimento Web



PRIMEFACES
PROF. PABLO VARGAS

Tópicos Abordados



- Introdução
- Primefaces
- Cadastro de Produtos

Introdução



- “é uma biblioteca de componentes para JSF com mais de 100 componentes.”
- Auxilia na criação das páginas através dos recursos de seus componentes.
- Fornecedor líder de bibliotecas de componentes de interface do usuário de código aberto.
- A partir da versão 2.2 não é necessário configurar o primefaces.
 - basta adicionar o arquivo primefaces-5.0.jar à biblioteca do projeto.
- <https://www.primefaces.org/>

Primefaces



- https://www.youtube.com/watch?v=ygUK_OSRPt8

Cadastro de Produtos



- Partindo do projeto SpringSecurity. Copie o projeto com o nome Primefaces.
- Crie a classe java “Produto” dentro do pacote “produto”.

```
@Entity
@Table(name = "produto")
public class Produto {
    @Id
    @GeneratedValue
    @Column(name = "cod_produto")
    private Integer cod_produto;

    @Column(name = "quantidade")
    private Integer quantidade;

    @Column(name = "nome")
    private String nome;

    @Column(name = "preco")
    private BigDecimal preco;

    @Column(name = "tipo")
    private String tipo;

    //gere os metodos get e set
    //gere os hashCode e equals
}
```

Cadastro de Produtos



- Crie a classe java “ProdutoDAO” dentro do pacote “produto”.

```
public interface ProdutoDAO {  
  
    public List<Produto> listar();  
  
    public void salvar(Produto produto);  
  
    public void excluir (Produto produto);  
  
}
```

Cadastro de Produtos



- Crie a classe java “ProdutoDAOHibernate” dentro do pacote “produto”.

```
public class ProdutoDAOHibernate implements ProdutoDAO {  
  
    private Session session;  
  
    public void setSession(Session session) {  
        this.session = session;  
    }  
  
    public List<Produto> listar() {  
        return this.session.createCriteria(Produto.class).list();  
    }  
  
    public void salvar(Produto produto) {  
        this.session.save(produto);  
    }  
  
    public void excluir(Produto produto) {  
        this.session.delete(produto);  
    }  
}
```

Cadastro de Produtos



- Adicione o método abaixo dentro da classe “DAOFactory”.

```
public static ProdutoDAO criarProdutoDAO() {  
    ProdutoDAOHibernate produtoDAO = new ProdutoDAOHibernate();  
    produtoDAO.setSession(HibernateUtil.getSessionFactory().getCurrentSession());  
    return produtoDAO;  
}
```


Cadastro de Produtos



- Crie a classe “ProdutoRN” dentro do pacote “produto”.

```
public class ProdutoRN {  
  
    private ProdutoDAO produtoDAO;  
  
    public ProdutoRN() {  
        this.produtoDAO = DAOFactory.criarProdutoDAO();  
    }  
  
    public List<Produto> listar() {  
        return this.produtoDAO.listar();  
    }  
  
    public void salvar(Produto produto) {  
        this.produtoDAO.salvar(produto);  
    }  
  
    public void excluir(Produto produto) {  
        this.produtoDAO.excluir(produto);  
    }  
}
```

Cadastro de Produtos



- Adicione, no arquivo de configuração do hibernate, a linha abaixo dentro da tag “<session-factory>”.

```
<mapping class="produto.Produto"/>
```

Cadastro de Produtos



- Crie a classe “ProdutoBean” no pacote “web”.

```
@Named
@RequestScoped
public class ProdutoBean {
    private Produto produto = new Produto();
    private List<Produto> lista = null;

    public Produto getProduto() {
        return produto;
    }

    public void setProduto(Produto produto) {
        this.produto = produto;
    }

    public List<Produto> getLista() {
        if (this.lista == null) {
            ProdutoRN produtoRN = new ProdutoRN();
            this.lista = produtoRN.listar();
        }
        return this.lista;
    }
}
```

Cadastro de Produtos



- Adicione o método salvar() dentro da classe “ProdutoBean”.

```
public void salvar() {  
    FacesContext context = FacesContext.getCurrentInstance();  
    try {  
        ProdutoRN produtoRN = new ProdutoRN();  
        produtoRN.salvar(this.produto);  
        this.produto = new Produto();  
        this.lista = null;  
    } catch (Exception e) {  
        FacesMessage facesMessage = new FacesMessage  
(FacesMessage.SEVERITY_ERROR, e.getMessage(), e.getMessage());  
        context.addMessage(null, facesMessage);  
    }  
}
```

Cadastro de Produtos



- Adicione o método `excluir()` dentro da classe “ProdutoBean”.

```
public void excluir() {  
    FacesContext context = FacesContext.getCurrentInstance();  
    try {  
        ProdutoRN produtoRN = new ProdutoRN();  
        produtoRN.excluir(this.produto);  
        this.produto = new Produto();  
        this.lista = null;  
    } catch (Exception e) {  
        FacesMessage facesMessage = new FacesMessage  
(FacesMessage.SEVERITY_ERROR, e.getMessage(), e.getMessage());  
        context.addMessage(null, facesMessage);  
    }  
}
```

Cadastro de Produtos



- Crie a pagina “produto.xhtml” na pasta “Logado”.

```
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://xmlns.jcp.org/jsf/html"
      xmlns:f="http://xmlns.jcp.org/jsf/core"
      xmlns:ui="http://xmlns.jcp.org/jsf/facelets"
      xmlns:p="http://primefaces.org/ui">
  <ui:composition template="/templates/interna.xhtml">
    <ui:define name="titulo">
      Cadastro de Produtos
    </ui:define>
    <ui:define name="corpo">
      <h:form id="listagem">
        <p:dataTable emptyMessage="Sem produtos cadastrados!" value="#{produtoBean.lista}"
                    var="produto" rendered="#{!empty produtoBean.lista}">
          <f:facet name="header">Listagem de Produtos</f:facet>
```

Cadastro de Produtos



```
<p:column filterBy="nome">
  <f:facet name="header">Nome do Produto</f:facet>
  #{produto.nome}
</p:column>
<p:column filterBy="#{produto.quantidade}">
  <f:facet name="header">Qty.</f:facet>
  #{produto.quantidade}
</p:column>
<p:column filterBy="preco">
  <f:facet name="header">Preço</f:facet>
  <h:outputText value="R$ #{produto.preco}">
    <f:convertNumber type="currency" minFractionDigits="2" maxFractionDigits="2"/>
  </h:outputText>
</p:column>
```

Cadastro de Produtos



```
<p:column>
  <f:facet name="header">Excluir</f:facet>
  <h:commandLink action="#{produtoBean.excluir}"
    onclick="if (!confirm('Confirma a exclusão do produto #{produto.nome}?'))
      return false;">
    <f:ajax execute="@this" render=":listagem :edicao"/>
    <h:graphicImage library="imagens" name="excluir16.png"/>
    <f:setPropertyActionListener target="#{produtoBean.produto}" value="#{produto}"/>
  </h:commandLink>
</p:column>
</p:dataTable>
</h:form>
<br/>
```


Cadastro de Produtos



```
<h:form id="edicao">
  <p:messages/>
  <h:inputHidden value="#{produtoBean.produto.cod_produto}"/>
  Cadastro de Produtos
  <p:panelGrid columns="2">
    <p:outputLabel value="Nome: " for="nome"/>
    <p:inputText id="nome" label="nome" value="#{produtoBean.produto.nome}" maxlength="100"
      required="true" requiredMessage="Qual o nome do produto?"/>
    <p:outputLabel value="Quantidade: " for="quantidade"/>
    <p:spinner id="quantidade" label="quantidade" value="#{produtoBean.produto.quantidade}"
      required="true" requiredMessage="Qual a quantidade de produtos?" min="0"/>
    <p:outputLabel value="Preço:" for="preco"/>
    <p:inputText id="preco" label="Valor" value="#{produtoBean.produto.preco}" required="true"
      requiredMessage="Qual o valor do produto?" size="20">
      <f:convertNumber minFractionDigits="2" maxFractionDigits="2"/>
    </p:inputText>
  </p:panelGrid>
```

Cadastro de Produtos



```
        <p:commandButton action="#{produtoBean.salvar}" value="Salvar" update=":edicao :listagem">
        </p:commandButton>
    </h:form>
</ui:define>
</ui:composition>
</html>
```

Atividades



- Execute o projeto, cadastre um produto e depois exclua-os.
- a) Crie uma forma para editar o produto cadastrado.
- b) Crie uma forma que apenas usuários `ROLE_USUARIOVIP` consiga apagar e editar os produtos.
- c) Utilize 5 dos componentes do Primefaces que não foram utilizados nos projetos.
- d) Crie um página de cadastro de fornecedores:
 - Requisitos: cnpj, nome, telefone, email, data de cadastro, endereço.
 - Deve-se poder editar e apagar os fornecedores.

Extras



- Bootstrap vs Primefaces:
 - <https://www.youtube.com/watch?v=-Db7dRRv-As>