Jsf：视图层实现框架；| 类似，struts2；

参考：

Ibm三篇文章，

<http://www.ibm.com/developerworks/cn/java/j-jsf2fu1/index.html>

<http://www.ibm.com/developerworks/cn/java/j-jsf2fu0111/index.html>

-----------------------------------------------

内容：

页面；

配置文件；

业务逻辑中的相关；

Facelets-模版框架，之上，复合组件；

//////////////////////////////////////////////////////////////////////////////

页面控件：

.xhtml页面；

基本控件，ui:composition、c:set、h:ead、h:body；

<ui:composition xmlns=*"http://www.w3.org/1999/xhtml"*

xmlns:h=*"http://java.sun.com/jsf/html"*

xmlns:f=*"http://java.sun.com/jsf/core"*

xmlns:ui=*"http://java.sun.com/jsf/facelets"*

xmlns:a4j=*"http://richfaces.org/a4j"*

xmlns:rich=*"http://richfaces.org/rich"*

xmlns:shiro=*"http://xproject.flightCenter/tag"*

xmlns:c=*"http://java.sun.com/jsp/jstl/core"*>

<c:set var=*"path"* value=*"${request.contextPath}"* />

<h:head>

</h:head>

<h:body>

</h:body>

</ui:composition>

基本控件，

<h:graphicImage library="images" name="cloudy.gif"/>

<h:form prependId="false">

<h:panelGrid columns="2">

<h:inputText value="#{place.streetAddress}" size="15"/>

<h:commandButton value="#{msgs.goButtonText}"

style="font-family:Palatino;font-style:italic"

action="#{place.fetch}"/>

//图标组件，

<util:icon image="#{resource['images:back.jpg']}"

actionMethod="#{places.logout}"

style="border: thin solid lightBlue"/>

-----------------------------

引入js和css外部文件，

<h:outputScript library="javascript" name="util.js" target="head"/>

<h:outputStylesheet library="css" name="styles.css" target="body"/>

##### 模版：/templates/masterLayout.xhtml

<div class="pageHeading">

<**ui:insert** name="heading">

#{msgs.placesHeading}

</ui:insert>

</div>

<div class="menuLeft">

<**ui:insert** name="menuLeft"/>

</div>

<**ui:insert** name="menuRight">

<ui:include src="/sections/shared/sourceViewer.xhtml"/>

</ui:insert>

ui:define，覆盖内容；

login 视图

<ui:composition xmlns="http://www.w3.org/1999/xhtml"

xmlns:ui="http://java.sun.com/jsf/facelets"

template="/templates/masterLayout.xhtml">

<ui:define name="menuLeft">

<ui:include src="/sections/login/menuLeft.xhtml"/>

</ui:define>

<ui:define name="content">

<ui:include src="/sections/login/content.xhtml"/>

</ui:define>

</ui:composition>

~

也可以修改menuRight内容；

<ui:define name="menuRight">

<ui:include src="/sections/showSource/menuRight.xhtml"/>

</ui:define>

页面赋值：

-------------------------------------------------

**表 2. JSF 2 托管 bean 注释（@...Scoped 注释只对 @ManagedBean 有效）**

| **托管 bean 注释** | **描述** | **属性** |
| --- | --- | --- |
| @ManagedBean | 以托管 bean 的形式注册一个类实例，然后将其放入到使用其中一个 @...Scoped 注释指定的范围内。如果没有指定任何范围，JSF 将把此 bean 放入请求范围，如果没有指定任何名称，JSF 将把类名的第一个字母转换为小写，形成一个托管 bean 名称；例如，如果类名为UserBean，那么 JSF 将创建一个托管 bean，其名为 userBean。eager 和 name 属性都是可选的。  注释必须结合使用一个实现零参数构造器的 Java 类。 | eager，name |
| @ManagedProperty | 为托管 bean 设置一个属性。注释必须放在类成员变量的声明之前。name 属性指定特性的名称，默认情况下为成员变量的名称。value 属性是特性的值，可以是一个字符串，也可以是一个 JSF 表达式，比如 #{...}。 | value，name |
| @ApplicationScoped | 在应用程序范围内存储托管 bean。 |  |
| @SessionScoped | 在会话范围内存储托管 bean。 |  |
| @RequestScoped | 在请求范围内存储托管 bean。 |  |
| @ViewScoped | 在视图范围内存储托管 bean。 |  |
| @NoneScoped | 将托管 bean 指定为没有范围。无范围的托管 bean 在被其他 bean 引用时比较有用。 |  |
| @CustomScoped | 在定制范围内存储托管 bean。  定制范围就是指可以由页面创建者访问的地图。可以通过编程的方式控制定制范围内的 bean 的可视性和生命周期。value 属性指向一个地图。 | value |

项目业务逻辑类，

@ManagedBean(name = "productAction")

@ViewScoped

**public** **class** ProductAction **implements** Serializable{

//service类，

@ManagedProperty(name = "financialProductService", value = "#{financialProductService}")

**private** FinancialProductService financialProductService;

//实体类---理财产品对象

@ManagedProperty(name = "financialProduct", value = "#{financialProduct}")

**private** FinancialProduct financialProduct;