**javaBean**

**简单理解：**

**javaBean在MVC设计模型中是model，又称模型层，在一般的程序中，我们称它为数据层，就是用来设置数据的属性和一些行为，然后我会提供获取属性和设置属性的get/set方法**

**即**

**Javabean 就是一个类，这个类就定义一系列 get<Name> 和 set<Name> 方法。 So simple ！Javabean 就是为了和 jsp 页面传数据化简交互过程而产生的。**

**而使用 javabean 之后，优势也就是 java 的优势：组件技术，代码重用，易于维护**

**一、什么是JavaBean**

**JavaBean是一个遵循特定写法的Java类，它通常具有如下特点：**

**这个Java类必须具有一个无参的构造函数**

**属性必须私有化。**

**私有化的属性必须通过public类型的方法暴露给其它程序，并且方法的命名也必须遵守一定的命名规范。**

**1.1、javaBean范例：**



**JavaBean在J2EE开发中，通常用于封装数据，对于遵循以上写法的JavaBean组件，其它程序可以通过反射技术实例化JavaBean对象，并且通过反射那些遵守命名规范的方法，从而获知JavaBean的属性，进而调用其属性保存数据。**

**1.2、JavaBean 现实应用中的实例,写一张用户注册页**



**表单提交 请求地址 RequestDemo**

**//获取单一的请求参数的值(用户所有的输入都是String)**

**String username = request.getParameter("username");//不区分get还是post的**

**//获取重名请求参数的值**

**String passwords[] = request.getParameterValues("password");//获取重名的请求参数值**

**//获取所有的请求参数名和值**

**Enumeration<String> e = request.getParameterNames();//参数名**

**while(e.hasMoreElements()){**

**String paramName = e.nextElement();**

**String values [] = request.getParameterValues(paramName);**

**System.out.println(paramName+"="+Arrays.asList(values));**

**}**

**// 如果属性比较多的情况下一个属性一个属性的set 太麻烦**



**二、JavaBean的属性**

**JavaBean的属性可以是任意类型，并且一个JavaBean可以有多个属性。每个属性通常都需要具有相应的setter、 getter方法，**

**setter方法称为属性修改器，getter方法称为属性访问器。**

**属性修改器必须以小写的set前缀开始，后跟属性名，且属性名的第一个字母要改为大写，例如，name属性的修改器名称为setName，password属性的修改器名称为setPassword。**

**属性访问器通常以小写的get前缀开始，后跟属性名，且属性名的第一个字母也要改为大写，例如，name属性的访问器名称为getName，password属性的访问器名称为getPassword。**

**一个JavaBean的某个属性也可以只有set方法或get方法，这样的属性通常也称之为只写、只读属性。**

**三、在JSP中使用JavaBean(这种用法只是做为了解, 现在项目开发,基本不用了!)**

**JSP技术提供了三个关于JavaBean组件的动作元素，即JSP标签，它们分别为：**

**<jsp:useBean>标签：用于在JSP页面中查找或实例化一个JavaBean组件。**

**<jsp:setProperty>标签：用于在JSP页面中设置一个JavaBean组件的属性。**

**<jsp:getProperty>标签：用于在JSP页面中获取一个JavaBean组件的属性。**

**<jsp:useBean>标签用于在指定的域范围内查找指定名称的JavaBean对象，如果存在则直接返回该JavaBean对象的引用，如果不存在则实例化一个新的JavaBean对象并将它以指定的名称存储到指定的域范围中。**

**常用语法：<jsp:useBean id="beanName" class="package.class" scope="page|request|session|application"/>**

**"id"属性用于指定JavaBean实例对象的引用名称和其存储在域范围中的名称。**

**"class"属性用于指定JavaBean的完整类名（即必须带有包名）。**

**"scope"属性用于指定JavaBean实例对象所存储的域范围，其取值只能是page、request、session和application等四个值中的一个，其默认值是page。**

**<jsp:setProperty>标签用于设置和访问JavaBean对象的属性。**

**语法格式一：**

**<jsp:setProperty name="beanName" property="propertyName" value="string字符串"/>**

**语法格式二： <jsp:setProperty name="beanName" property="propertyName" param="parameterName"/>**

**语法格式三： <jsp:setProperty name="beanName" property= "\*" />**

**name属性用于指定JavaBean对象的名称。**

**property属性用于指定JavaBean实例对象的属性名。**

**使用jsp:setProperty标签设置person对象的属性值**

**jsp:setProperty在设置对象的属性值时会自动把字符串转换成8种基本数据类型**

**但是jsp:setProperty对于复合数据类型无法自动转换**

**<jsp:setProperty property="sex" name="person" value="男"/>**

**jsp:setProperty标签可以使用请求参数为bean的属性赋值**

**param="param\_name"用于接收参数名为param\_name的参数值，然后将接收到的值赋给name属性**

**<jsp:setProperty property="name" name="person" param="param\_name"/>**

**property="\*"代表bean的所有属性**

**<jsp:getProperty>标签用于读取JavaBean对象的属性，也就是调用JavaBean对象的getter方法，然后将读取的属性值转换成字符串后插入进输出的响应正文中。**

**<jsp:getProperty property="sex" name="person"/>**