## Struts2创建步骤：

1. 创建Action，并声明服务方法
2. 配置action访问路径，需要在src下创建struts.xml文件

<?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>

<package name=*"demo"* extends=*"struts-default"* namespace=*"/"*><!--name: -->

<action name=*"hello"* class=*"com.jun.demo.HelloAction"*><!--hello为地址栏访问地址，class中内容为需要映射的类 -->

<result name=*"ok"*>index.jsp</result><!--ok为返回的字符串 index.jsp为访问上面地址跳转的页面 -->

</action>

</package>

</struts>

1. Action配置

### Struts2配置：

1. xml中3个最基本的标签

* **package**:类似与代码中的包，配置action需要写package

name：表示package的名称，多个package中的name不能重复

extend:默认为 struts-defaut

namespace:默认为 /

<package name=*"demo"* extends=*"struts-default"* namespace=*"/"*></package>

* **action**:配置action的访问路径

name：访问路径，与namespace构成访问路径，同一个package中action的name不能重复

class:对应的功能类路径

method:执行的方法，默认为execute，一个action可以有多个方法，若要写其他方法，可以用该参数进行配置

<action name=*"hello"* class=*"com.jun.demo.HelloAction"*> </action>

* **result**：根据action的返回值，配置到不同的路径中去

name：和**方法的返回值一样**，指定页面名称

type：配置如何到另外一个路径中去（转发或重定向），**默认转发**

<result name=*"ok"*>index.jsp</result><!--ok为返回的字符串 index.jsp为访问上面地址跳转的页面 -->

## Struts2内置拦截器：

不是所有的拦截器都被添加到默认拦截器栈里，只有被添加到默认拦截器栈才起作用

常见拦截器：

params：将请求的的参数设置到相应的Action的属性中，并自动类型转换

servletConfig:感知接口封装

fileUpload:用于文件上传，及封装检验

token:使用其作为核对当前请求的有效表示，解决重复提交问题

validation：使用action-validation.xml文件中定义的内容校验提交的数据

exception:程序抛出异常是，该拦截器开始对异常进行处理，建议放在第一位，让它补货所有异常

### 自定义拦截器

自定义一个登陆栏截器步骤

1、自定义拦截器类

实现com.opensymphony.xwork2 interceptor.nterceptor接口

继承**com.opensymphony.xwork2 .interceptor.Abstractinterceptor**类

2、自定拦截器类需要重写**public String intercept(Actionlnvocation a) throws Exception**

3.在struts .xml中**注册**该拦截器

拦截器只有注册后才能使用，注册时需要描述该拦截器与action的关系

在需要拦截的action所在的package中，通过<interceptor>元素来定义拦截器在具体的action标签里面通过<interceptor-ref>元素来使用拦截器

4、如果为Action指定了一个拦截器，则系统默认的拦截器栈将会失去作用。为了继续使用默认拦截器，所以需要手动声明默认拦截器，如代码<interceptor-ref name="defaultStack > </interceptor- ref>如果定义的Action没有使用到struts2里面的功能，则不需要考虑该问题

5.该拦截器会对该action中所有的方法进行拦截，这就会产生一些不必要的拦截过程而影响系统功能。所以，我们要将无需拦截的方法进行不拦截配置

在<interceptor-ref>标签下加入代码:

<param name="excludeMethods" >login</param>, 如有多个方法使用逗号隔开

6. 扩展系统默认的拦截器栈：将系统拦截器栈和自定义拦截器栈重组成一个新的拦截器栈

<!-- 扩展系统默认的拦截器栈 -->

<interceptor-stack name=*"myStack"*>

<interceptor-ref name=*"defaultStack"*></interceptor-ref>

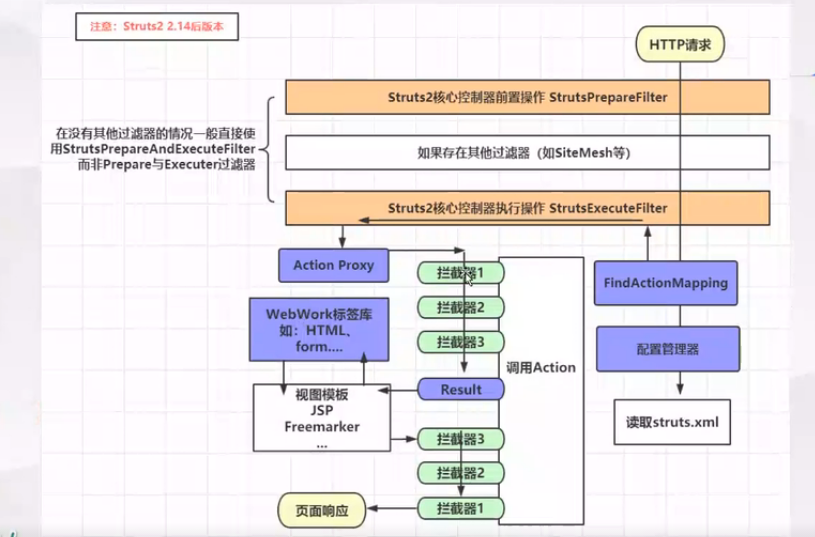
<interceptor-ref name=*"loginInterc"*></interceptor-ref>

</interceptor-stack>

<!-- 将自定义拦截器栈映射到所有的action中 -->

<default-interceptor-ref name=*"myStack"*/>

## Struts2工作流程图



### Struts2文件上传

1. 文件上传的**条件**：
2. 使用post方式
3. 使用二进制编码：multipart/form-date;
4. <input type = “file” name=””/>
5. 文件上传编写动作类注意
6. Struts2提供的上传文件名命名规则：上传的字段名（form表单中的）+FileName
7. Struts2提供MIME类型命名规则：上传的字段名（form表单中的）+ContenType
8. 没有使用模型驱动是：form表单提供的属性和struts2提供的属性要生成set，get方法
9. 在Action类中提供上传的方法：获取要保存文件的位置；
10. 在目标文件夹内，创建一个与上传文件同名的文件
11. 通过FileUtils工具类提供copyFile()方法，将临时文件内容拷贝到目标文件夹下的那个同名的文件
12. 调用保存临时文件的属性的delete()方法删除临时文件（也就是File类型属性的delete方法）

## 文件上传配置：

1. 编写struts.xml配置文件注意：
2. 上传文件默认为 2兆，修改上传文件限制时，valie的值应该算出来**不能**写出1024\*1024\*5（5兆）

设置文件上传总大小，对所有的form有效，只想对当前form进行设置，可以设置fileUpload拦截器属性超过文件总大小，跳转input视图，通过debug取出回显的错误信息

1. 在fileUpload拦截器中有是三个属性：

Protected Long maximumSize:可以设置每一个上传文件的大小

Protected Set<String> allowedTypeSet:设置允许文件上传的类型

Protected Set<String> allowedExtendsionsSet:设置文件后缀名

<!--设置指定action上传文件大小最大为10k -->

<interceptor-ref name=*"fileUpload"*>

<param name=*"fileUpload.maximumSize"*>10240</param>

<!--设置上传文件类型 -->

<param name=*"fileUpload.allowedTypes"*>image/jpeg,application/msword/msword</param>

<!--设置文件上传后缀 -->

<param name=*"fileUpload.allowedExtensions"*>.jpg,.doc</param>

</interceptor-ref>

## 文件下载

步骤：

1. 在Action中定义一个inputStream方法处理下载文件请求
2. 指定当前方法的结果类型为：stream
3. 可以通过修改inputName属性的值，修改处理下载请求的方法名称
4. 设置结果类型的contentDisposition的值修改下载文件名称
5. 通过设置bufferSize指定下载文件时的缓冲大小

<result type=*"stream"*>

<!--对应的是Action中的方法，返回流的属性 -->

<param name=*"inputName"*>inputStream</param>

<!-- 允许下载的文件类型：指定为所有的二进制文件类型，去tongcat的web.xml里搜索 -->

<param name=*"contentType"*>application/octet-stream</param>

<!-- 下载头，包括：浏览器显示的文件名，attachment不直接打开 -->

<param name=*"contentDisposition"*>attachment;filename="${downFileName}"</param>

<!-- 缓冲区大小设置 -->

<param name=*"bufferSize"*>1024</param>

</result>