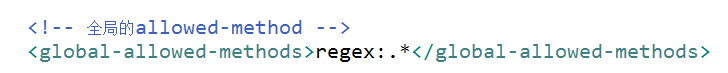
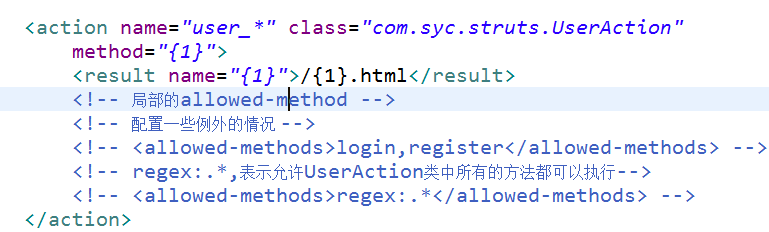
# Day03 Struts2框架

## 通配符

通配符的出现是为了简化struts.xml文件中action的配置.

在struts 2.0版本之后,\*作为通配符,{1}作为占位符.但是必须设置一个局部或者全局的allowed-methods方法.

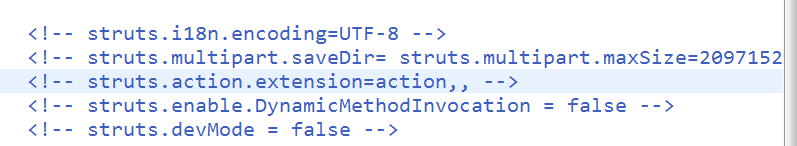


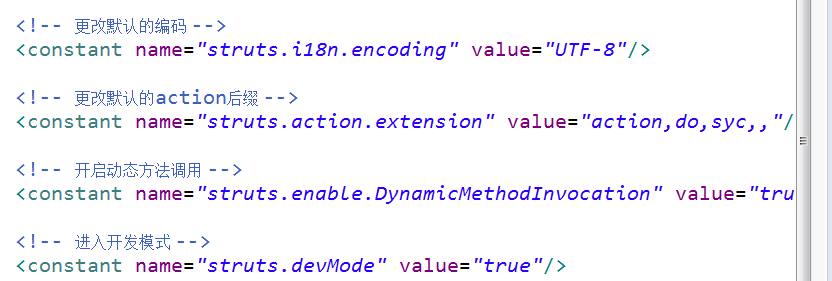


注意:

当同时配置了局部的和全局的allowed-methods之后,局部的allowed-methods生效,因为就近原则!

## Struts2框架的常量配置





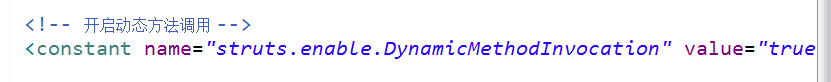
## 三.动态方法调用

### 1.动态方法的作用

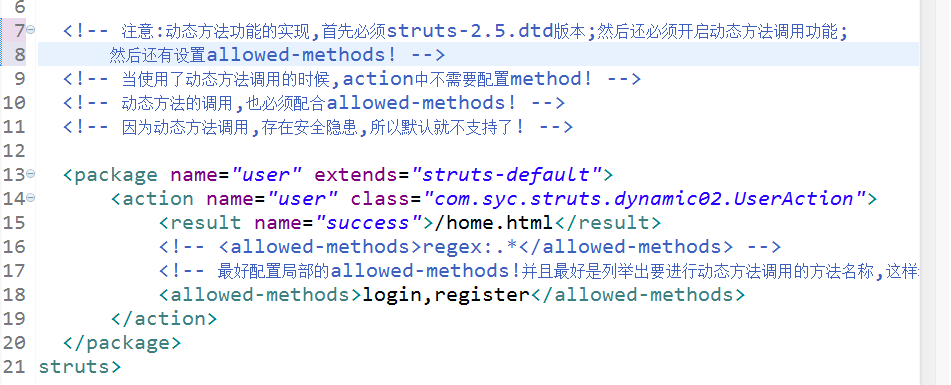
动态方法的出现,也是为了简化action的配置,与通配符的作用类似.

但是在struts2.0之后,动态方法调用的功能,默认是没有开启的,所以在使用动态方法调用之前,需要手动开启.

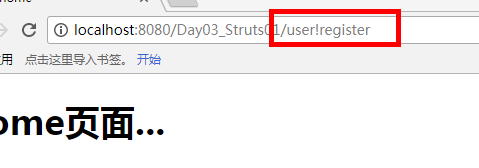
### 2.动态方法的实现



### 3.注意事项



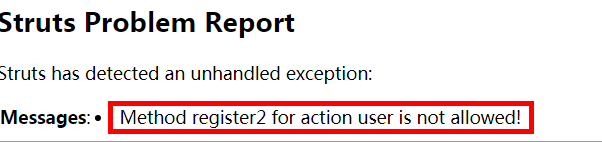
### 4.如何访问动态方法调用的action



动态方法的访问格式:

action的name!方法名!

### 5.动态方法调用时可能出现的错误



错误原因:

请求的方法method不被该action所允许,注意查看allowed-methods配置是否有问题!

## 四.namespace的作用

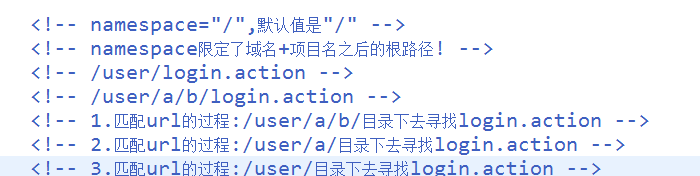
<http://www.baidu.com:8080/>index.html

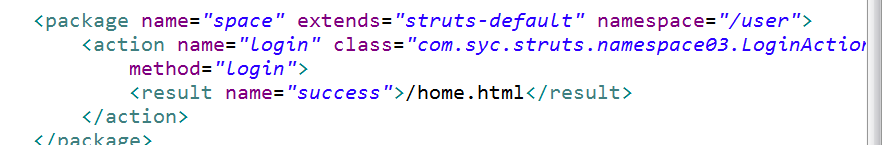
<http://news.baidu.com:8080/>{变量}/index.html

http://tiezhi.baidu.com:8080/index.html

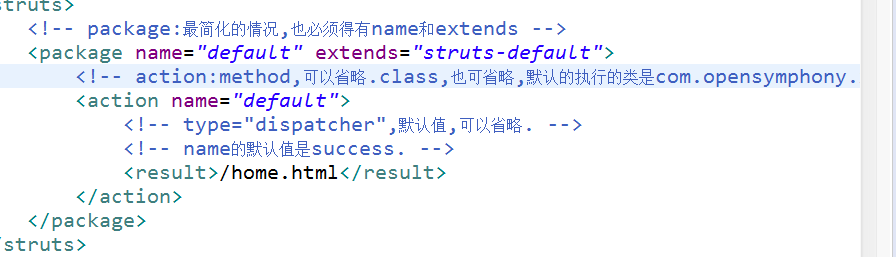
http://zhidao.baidu.com:8080/index.html

namespace的作用,可以给一个项目中不同的模块分配不同的url地址!





## 五.struts2框架的默认值



## 六.请求转发和重定向的区别?

1.请求转发地址栏里面的url内容不变,重定向会变化;

2.请求转发相当于发起一次请求,会有一个request对象,

重定向相当于发起2次请求,有2个request对象;

3.请求转发的时候可以传递数据,重定向的时候不可以;

4.请求转发只针对服务器本身,只在服务器内部实现跳转,

重定向可以跳转到其他的服务器中;

5.请求转发这个方法是request的api方法,重定向属于response对象.

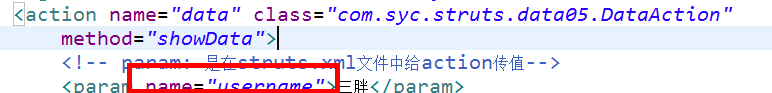
6.两个方法中路径不同:

request.getRequestDispatcher(相对路径).forward(request,response);

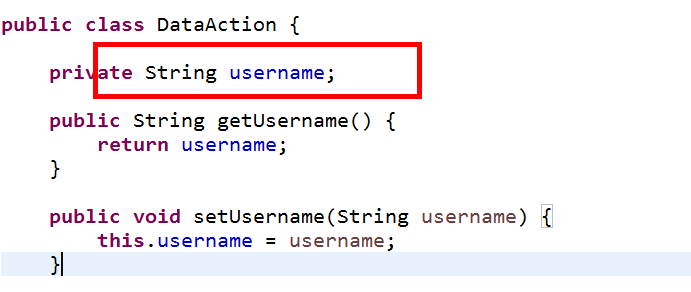
response.sendRedirect(完整路径);

## 七.struts.xml文件给action类传值

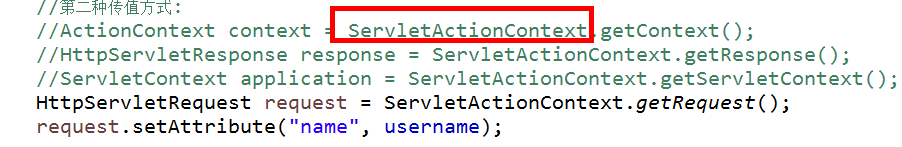
struts.xml文件中;

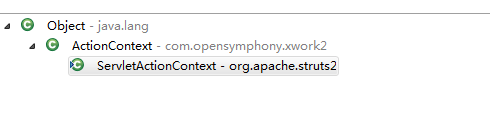


Action类中:



### 2.如何获取request和response对象?





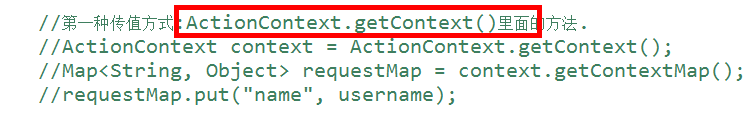
## 八.4个作用域和9个内置对象?

## 九.global-result

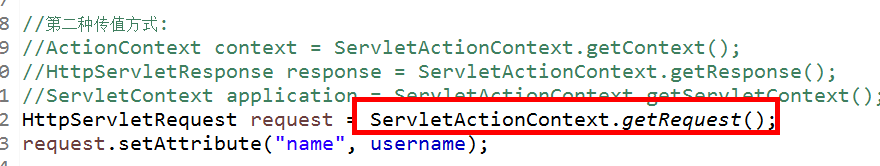


## 十.Action类给jsp传值的方式总结

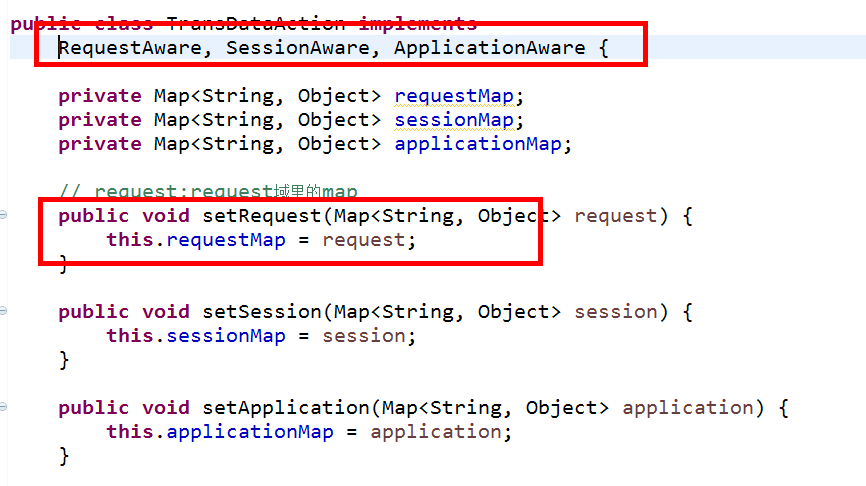
### 1. 方式一:



### 2.方式二:



### 3.方式三:



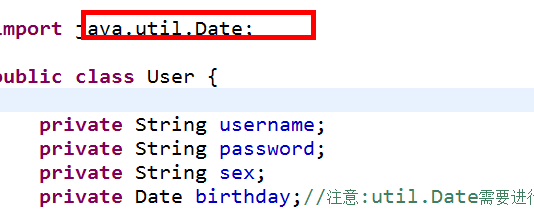
## 十一. util.Date转换问题

### 1.转换原因:

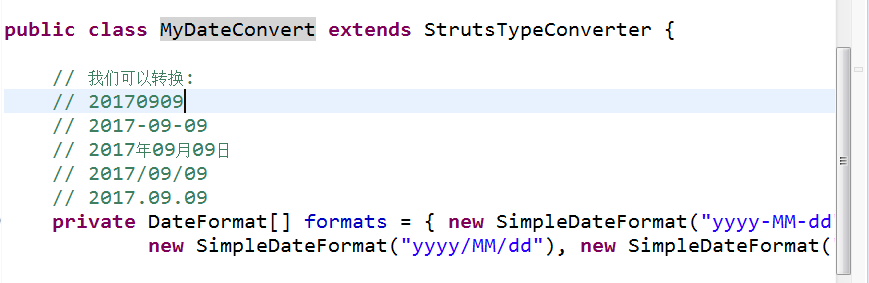
默认情况下,struts框架不能转换所有的util.Date类的数据,需要我们自己手动编写一个转换器类来进行转换.

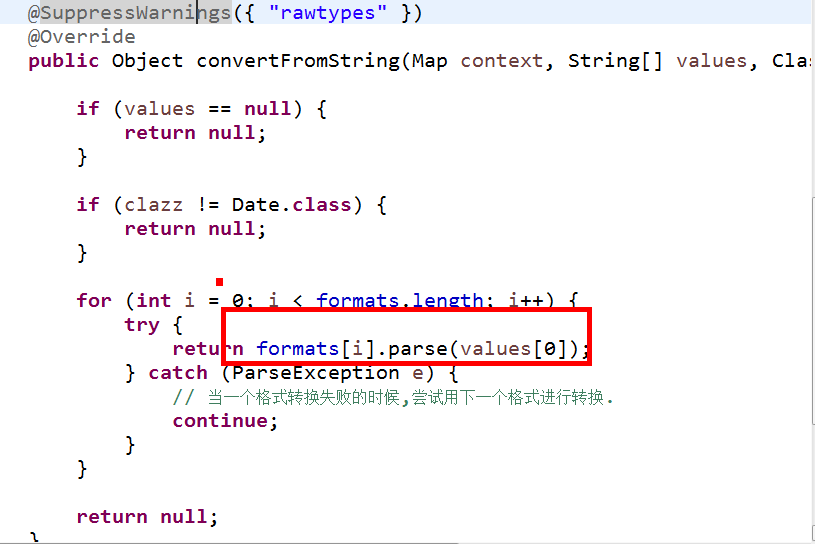
### 2.具体实现:

#### ①.定义JavaBean



#### ②.自定义转换器类





#### ③.用我们自己的转换器替换struts框架自带的转换器

