# 拦截器（interceptor）一般在action前执行

# 定义拦截器（interceptor）：

**package** com.yang.strutsinterceptor.interceptors;

**import** java.util.Map;

**import** javax.servlet.http.HttpServletRequest;

**import** org.apache.struts2.ServletActionContext;

**import** com.opensymphony.xwork2.ActionContext;

**import** com.opensymphony.xwork2.ActionInvocation;

**import** com.opensymphony.xwork2.interceptor.AbstractInterceptor;

**public** **class** LoginCheck **extends** AbstractInterceptor{

@Override

**public** String intercept(ActionInvocation invocation) **throws** Exception {

Map<String,Object> session = ActionContext.*getContext*().getSession();

HttpServletRequest request = ServletActionContext.*getRequest*();

String username = request.getParameter("user.username");

String password = request.getParameter("user.password");

System.***out***.println(username);

System.***out***.println(password);

/\*\*

\* 如username为"yang"且password为"5354"，则继续执行action所调用方法

\*/

**if**( username.equals("yang") && password.equals("5354")){

session.put("username", username);

session.put("password", password);

//执行该代码后，将继续执行action所调用方法

**return** invocation.invoke();

}

**else**{

**return** "input";

}

}

}

# 在xml中配置及使用拦截器（interceptor）：

<?xml version=*"1.0"* encoding=*"UTF-8"* ?>

<!DOCTYPE struts PUBLIC

"-//Apache Software Foundation//DTD Struts Configuration 2.3//EN"

"http://struts.apache.org/dtds/struts-2.3.dtd">

<struts>

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

<interceptors>

<!-- intercptor的name在使用拦截器是要用到 -->

<interceptor name=*"loginCheck"* class=*"com.yang.strutsinterceptor.interceptors.LoginCheck"*></interceptor>

</interceptors>

<action name=*"login"* class=*"com.yang.strutsinterceptor.controller.UserAction"* method=*"login"* >

<result name=*"success"*>/WEB-INF/page/loginSuccess.jsp</result>

<result name=*"input"*>/WEB-INF/page/loginFail.jsp</result>

<!-- 使用自定义拦截器前，应先调用默认的拦截器，这很重要 -->

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

<!-- 使用自定义拦截器 -->

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

</action>

</package>

</struts>

# Note：

使用自定义拦截器前，应先调用默认的拦截器，这很重要

在类中定义拦截器时，执行**return** invocation.invoke();后将继续执行action所调用方法