

# struts2

---

今天是w 学 struts2 的第二天，但真正来说这只算是第一天的内容

## 分模块开发

---

单独写配置文件，将配置文件最终引入到核心配置文件【struts2.xml】中

引入外部配置文件

```
<include file="外部配置文件路径"></include>
```

## Action 编写方式

---

action的编写方式有三种.

1. 写一个action的类并在类中写上execute()方法
2. 实现Action接口【接口中有一些常量供我们使用】，实现execute方法【execute方法必须实现【不实用】】
3. 继承ActionSupprot类【可以不重写，ActionSupprot内部已经实现了Action接口，可以使用Action的常量】

## 编写方式

第一种

```
package action;

public class MyAction {
    public String execute() {
        // TODO Auto-generated method stub
        return "跳转吧靓仔";
    }
}
```

第二种【注意导入的 Action 包】

```

package action;
import com.opensymphony.xwork2.Action;

public class ActionImpl implements Action {
    @Override
    public String execute() throws Exception {
        // TODO Auto-generated method stub
        return "跳转吧靓仔";
    }
}

```

第三种【可以不重写，演示需要】

```

package action;
import com.opensymphony.xwork2.ActionSupport;

public class ActionExtends extends ActionSupport {
    @Override
    public String execute() throws Exception {
        // TODO Auto-generated method stub
        return "跳转吧靓仔";
    }
}

```

## 根据路径使用不同的方法

若不想任何提交都执行execute()方法则可以设置.

方式有三种：

1. 在struts.xml 中action内加上method="指定执行方法".
2. 使用通配符的方式实现
3. 动态访问实现

案例一【method】

```

<package name="test_package" extends="struts-default" namespace="/">
    <action name="addAction" class="action.myAction" method="add">
        <result name="ok">/index.jsp</result>
    </action>
    <action name="deleteAction" class="action.myAction" method="delete">
        <result name="ok">/index.jsp</result>
    </action>
</package>

```

### 【优点】

通过不同的访问路径执行同一个类中对应的方法，因此可以省略许多servlet类的冗余写法

### 【缺陷】

若此处的myAction内的方法过多则需写很多action

### 案例二【通配符 \*】

```
<package name="test_package" extends="struts-default" namespace="/">
  <action name="Action_*" class="action.myAction" method="{1}">
    <result name="ok">/index.jsp</result>
    <result name="!ok">/logon.jsp</result>
  </action>
</package>
```

### 【代码解析】

**Action\_\***：name中可以直接写\*号，但为了美观加上Action\_，用户想访问action哪个方法就直接Action\_方法名

**{1}**：代表若name出现多个\*取第一个

### 【过程推敲】

当用户输入完访问的路径时，服务器将解析完毕的action的name值根据method的规则将name截取，然后看截取请求的资源是否包含截取的值，若全都包含就将请求资源的\*代替的剩余部分拿出来去myaction类中找是否有该方法，若有则给予指定反馈。

### 【演示错误】

1. 当action执行的方法返回值若在result中不存在则会报404错误
2. 在action中默认执行的方法可以没有返回值，若有则必须为String
3. action默认执行的方法可以没有返回值，若没有则可以不用配置result
  1. 直接在方法声明void【有其他方法需要返回值则推荐使用】
  2. return "none"【推荐使用/前提是action所有执行的方法都不需要返回值否则404】

**来源于 ----- 【通配符】**