

Action trong Struts 2

Trong Struts 2, lớp Action là POJO (viết tắt của Plain Old Java Object). POJO nghĩa là bạn không thể bắt buộc triển khai bất cứ Interface hoặc kế thừa bất cứ lớp nào. Nói chung, phương thức **execute** nên được xác định để biểu trình logic cho riêng bạn. Lớp Action đơn giản có dạng như sau:

Welcome.java

```
package com.vietjack;

public class Welcome {

    public String execute(){

        return "success";

    }

}
```

Action Interface trong Struts 2

Một phương pháp tiếp cận tiện lợi là để triển khai `com.opensymphony.xwork2.Action` interface mà định nghĩa 5 hằng và một phương thức `execute`. 5 hằng này có thể được trả về từ lớp Action, chúng là:

- **SUCCESS** chỉ rằng sự thực thi action là thành công và một success result nên được hiển thị người dùng.
- **Error** chỉ rằng sự thực thi action là thất bại và một error result nên được hiển thị tới người dùng.
- **LOGIN** chỉ rằng người dùng đã đăng nhập và một login result nên được hiển thị tới người dùng.
- **INPUT** chỉ rằng trình validation là thất bại và một input result nên được hiển thị tới người dùng.
- **NONE** chỉ rằng sự thực thi action là thành công nhưng không có result nên được hiển thị tới người dùng.

Sau đây, bạn theo dõi các giá trị nào được gán cho các hằng này:

```
public interface Action {  
    public static final String SUCCESS = "success";  
    public static final String NONE = "none";  
    public static final String ERROR = "error";  
    public static final String INPUT = "input";  
    public static final String LOGIN = "login";  
    public String execute() throws Exception;  
}
```

Phương thức của Action Interface

Action Interface chỉ gồm một phương thức execute mà nên được triển khai, được ghi đè bởi lớp action ngay cả khi bạn không bắt buộc.

```
public String execute();
```

Ví dụ của Struts Action mà triển khai Action Interface

Nếu bạn triển khai Action Interface, bạn có thể trực tiếp sử dụng các hằng thay cho các giá trị.

Welcome.java

```
package com.vietjack;  
import com.opensymphony.xwork2.Action;  
public class Welcome implements Action{  
    public String execute(){  
        return SUCCESS;  
    }  
}
```

Tiếp theo, bạn theo dõi phương thức action trong ví dụ Hello World.

```
package com.vietjack.struts2;  
  
public class HelloWorldAction{  
    private String name;  
  
    public String execute() throws Exception {
```

```
        return "success";
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
}
```

Lớp ActionSupport trong Struts 2

Nó là một lớp tiện lợi mà triển khai nhiều Interface như Action, Validateable, ValidationAware, TextProvider, LocaleProvider và Serializable. Vì thế, hầu như nó được sử dụng thay cho Action.

Ví dụ Struts Action mà kế thừa lớp ActionSupport

Welcome.java

```
package com.vietjack;

import com.opensymphony.xwork2.ActionSupport;

public class Welcome extends ActionSupport{

    public String execute(){

        return SUCCESS;

    }

}
```

Để minh họa cho quan điểm rằng phương thức action điều khiển view, bạn tạo sự thay đổi tới phương thức execute và kế thừa lớp ActionSupport như sau:

```
package com.vietjack.struts2;

import com.opensymphony.xwork2.ActionSupport;

public class HelloWorldAction extends ActionSupport{
```

```
private String name;

public String execute() throws Exception {
    if ("SECRET".equals(name))
    {
        return SUCCESS;
    }else{
        return ERROR;
    }
}

public String getName() {
    return name;
}

public void setName(String name) {
    this.name = name;
}
}
```

Trong ví dụ này, chúng ta có cùng trình logic trong phương thức execute để nhìn vào thuộc tính name. Nếu thuộc tính là bằng với chuỗi SECRET, chúng ta trả về SUCCESS như là kết quả, nếu không chúng ta trả về kết quả là ERROR. Bởi vì chúng ta đã kế thừa lớp ActionSupport, vì thế chúng ta sử dụng hằng SUCCESS và ERROR. Bây giờ sửa đổi struts.xml file, như sau:

```
<?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>
    <constant name="struts.devMode" value="true" />
    <package name="helloworld" extends="struts-default">
        <action name="hello"
            class="com.vietjack.struts2.HelloWorldAction">
```

```
        method="execute">
        <result name="success">/HelloWorld.jsp</result>
        <result name="error">/AccessDenied.jsp</result>
    </action>
</package>
</struts>
```

Tạo một View trong Struts 2

Bây giờ, chúng ta tạo HelloWorld.jsp file. Để tạo file này, bạn nhấn chuột phải trên thư mục WebContent và chọn **New > JSP file**. File này sẽ được gọi trong trường hợp kết quả trả về là SUCCESS mà là một hằng chuỗi "success" như được định nghĩa trong Action Interface.

```
<%@ page contentType="text/html; charset=UTF-8" %>
<%@ taglib prefix="s" uri="/struts-tags" %>
<html>
<head>
<title>Hello World</title>
</head>
<body>
    Hello World, <s:property value="name"/>
</body>
</html>
```

Dưới đây là file sẽ được triệu hồi bởi Framework trong trường hợp kết quả là ERROR, tương đương với hằng chuỗi "error". Dưới đây là nội dung của AccessDenied.jsp.

```
<%@ page contentType="text/html; charset=UTF-8" %>
<%@ taglib prefix="s" uri="/struts-tags" %>
<html>
<head>
<title>Access Denied</title>
</head>
<body>
    You are not authorized to view this page.
</body>
```

```
</html>
```

Chúng ta cũng cần tạo index.jsp trong WebContent folder. File này sẽ đóng vai trò như là URL ban đầu, là nơi người dùng có thể nhấn vào để nói cho Struts 2 Framework để gọi phương thức execute của lớp HelloWorldAction và truyền HelloWorld.jsp view.

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags"%>
    <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Hello World</title>
</head>
<body>
    <h1>Hello World From Struts2</h1>
    <form action="hello">
        <label for="name">Nhập tên của bạn</label><br/>
        <input type="text" name="name"/>
        <input type="submit" value="Xin chào!"/>
    </form>
</body>
</html>
```

Giữ web.xml không thay đổi như trong chương Ví dụ Hello World, sau đó bạn khởi động Server và kiểm tra kết quả.