

# regex Validation trong Struts 2

Regex Validator xác nhận chuỗi đã cung cấp với Regular expression đã cho. Nó có thể được sử dụng trong password, security key, ... Có 4 tham số được định nghĩa cho Regex Validator, đó là:

- **fieldName**: xác định tên trường mà để được validate. Nó chỉ bắt buộc trong Plain-Validator.
- **expression**: xác định regular expression.
- **caseSensitive**: xác định rằng có nên so khớp theo cách phân biệt kiểu chữ không. Theo mặc định là true.
- **trim**: xác định xem có nên cắt tỉa chuỗi trước khi so khớp không. Theo mặc định là true.

## Ví dụ regex validation

Cho Plain Validator:

```
<validators>
  <!-- Ví dụ cho Plain Validator -->
  <validator type="regex">
    <param name="fieldName">data</param>
    <param name="expression">[A-Z,a-z,0-9]{5}</param>
    <message>Du lieu phai gom 5 ky so dang chu-so</message>
  </validator>
</validators>
```

Cho Field Validator:

```
<validators>
  <!-- Ví dụ cho Field Validator -->
  <field name="data">
    <field-validator type="regex">
      <param name="expression">[A-Z,a-z,0-9]{5}</param>
      <message>Du lieu phai gom 5 ky so dang chu-so</message>
    </field-validator>
  </field>
</validators>
```

```
</validators>
```

## Ví dụ đầy đủ regex validation trong Struts 2

### Tạo index.jsp

Cho input từ người dùng. Nó nhận name, password, và email id từ người dùng.

```
<%@ taglib uri="/struts-tags" prefix="s" %>
<html>
<head>
<STYLE type="text/css">
.errorMessage{color:red;}
</STYLE>
</head>
<body>
<marquee>validation.....</marquee>
<s:form action="register">
<s:textfield name="data" label="Data"></s:textfield>
<s:submit value="register"></s:submit>
</s:form>

</body>
</html>
```

## Tạo lớp action

:

Lớp này kế thừa lớp ActionSupport và ghi đè phương thức validate.

### RegisterAction.java

```
package com.vietjack;

import com.opensymphony.xwork2.ActionSupport;

public class Register extends ActionSupport{
```

```
private String data;

public String getData() {
    return data;
}

public void setData(String data) {
    this.data = data;
}

public String execute(){
    return "success";
}

}
```

## Tạo validation file

Tại đây chúng ta đang sử dụng bundled validator để thực hiện trình validation.

### Register-validation.xml

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

<!DOCTYPE validators PUBLIC
    "-//OpenSymphony Group//XWork Validator 1.0.2//EN"
    "http://www.opensymphony.com/xwork/xwork-validator-1.0.2.dtd">

<validators>

    <!--<field name="data">
        <field-validator type="regex">
            <param name="expression">[A-Z,a-z,0-9]{5}</param>
            <message>Du lieu phai gom 5 ky so dang chu-so</message>
        </field-validator>
```

```
        </field>

        -->
        <field name="data">
        <field-validator type="regex">
        <param name="expression">[A,a][A-Z,a-z,0-9]{5}</param>
        <message>Du lieu phai gom 6 ky so dang chu-so va bat dau voi a hoac
A</message>

        </field-validator>
        </field>

    </validators>
```

## Tạo struts.xml

xml file định nghĩa một result bởi tên đã nhập, và một interceptor là jsonValidatorWorkflowStack.

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

<!DOCTYPE struts PUBLIC "-//Apache Software Foundation//DTD Struts Configuration 2.1//EN"
"http://struts.apache.org/dtds/struts-2.1.dtd">

<struts>

<package name="default" extends="struts-default">
<action name="register" class="com.vietjack.Register">
<result name="input">index.jsp</result>
<result>welcome.jsp</result>
</action>

</package>

</struts>
```

## Tạo các thành phần view

JSP file đơn giản này hiển thị thông tin về người dùng.

## welcome.jsp

```
<%@ taglib uri="/struts-tags" prefix="s" %>
```

```
Du lieu la,<s:property value="data"/>
```