

# int Validation trong Struts 2

int Validator kiểm tra xem số đã cung cấp có nằm trong dãy giá trị đã cho không. Nó có thể được sử dụng trong productId, employeeld, ... Có ba tham số được định nghĩa cho int validator, đó là:

- **fieldName**: xác định tên trường mà để được validate. Nó chỉ bắt buộc trong Plain-Validator.
- **min**: xác định giá trị tối thiểu. Theo mặc định, nó được bỏ qua.
- **max**: xác định giá trị tối đa. Theo mặc định, nó được bỏ qua.

## Ví dụ int validation

Cho Plain Validator:

```
<validators>
  <!-- Vi du cho Plain Validator -->
  <validator type="int">
    <param name="fieldName">age</param>
    <param name="min">16</param>
    <param name="max">50</param>
    <message>Age phai nam trong khoang tu toi </message>
  </validator>
</validators>
```

Cho Field Validator:

```
<validators>
  <!-- Vi du cho Field Validator -->
  <field name="age">
    <field-validator type="int">
      <param name="min">16</param>
      <param name="max">50</param>
      <message>Age phai nam trong khoang tu toi </message>
    </field-validator>
  </field>
</validators>
```

```
</validators>
```

## Ví dụ đầy đủ int 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>

<s:form action="register">
<s:textfield name="id" label="Product Id"></s:textfield>
<s:textfield name="price" label="Product Price"></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 int id;
    private double price;

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public double getPrice() {
        return price;
    }

    public void setPrice(double price) {
        this.price = price;
    }

    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="id">
            <field-validator type="int">
                <param name="min">1</param>
                <param name="max">999</param>

                <message>Id phải nằm trong khoảng từ 1 tới 999 </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" %>
```

```
Product Id:<s:property value="id"/><br/>
```

```
Product price:<s:property value="price"/>
```