

JSF och AJAX

JSF validators

- JSF valideringar används då man vill ha lite bättre valideringar i sin kod
- Det finns en del fördefinierade
 - `f:validateDoubleRange` – value maximum, minimum
 - `f:validateLength` – length maximum, minimum
 - `f:validateLongRange` – value maximum, minimum
 - `f:validator` – Används för att binda en component till en klas som implementerar `javax.faces.validator.Validator` interface.

- Finns två sätt att göra valideringar
 - Implementera en Valideringsklass
 - Implementera en metod som anropas

```
<h:inputText id="tjo" size="30" value="#{bean.text}"  
  validator="#{bean.validateText}" />
```

I bönan

```
public void validateText(FacesContext ctx, UIComponent toVal, Object val) {  
    String txt = (String) val;  
  
    if (txt.length() > 10) {  
        ((UIInput)toVal).setValid(false);  
  
        ctx.addMessage(toVal.getClientId(ctx), new FacesMessage("Hej och  
hå!"));  
    }  
}
```

JSF Events

- Det finns tre typer av events
- action event som triggar för en komponent blir aktiv, t.e.x. knappar och länkar.
- ValueChange event som triggar när ett värde förändras i en UIInput. Typ inputtext, checkbox...
- data-model event som triggas när en UIData komponent blir vald.

- Två sätt att reagera på events.
 - Implementera en event lyssnar class
 - Implementera en metod som anropas i en backing-bean.
-

- `<h:inputText id="name" size="50" value="#{bean.name}">`
- `<f:valueChangeListener type="listeners.valueChanged" />`
- `</h:inputText>`
- `public void processValueChange(ValueChangeEvent event)`
- `throws AbortProcessingException {`
- `if (null != event.getNewValue()) {`
- `FacesContext.getCurrentInstance().`
- `getExternalContext().getSessionMap().`
- `put("name", event.getNewValue());`
- `}`

faces-config.xml

- `<?xml version='1.0' encoding='UTF-8'?>`
- `<faces-config>`
- `<managed-bean>`
- `<managed-bean-name>bean</managed-bean-name>`
- `<managed-bean-class>test.TestBean</managed-bean-class>`
- `<managed-bean-scope>session</managed-bean-scope>`
- `<managed-property>`
- `<property-name>text</property-name>`
- `<value>bajen!</value>`
- `</managed-property>`
- `</managed-bean>`


```
<f:view>
    <h:messages />
    <h1><h:outputText value="Logga in här" /></h1>
    <h:form>
        <table>
            <tr>
                <td> Namn </td>
                <td> <h:inputText size="20" id="name"
value="#{userBean.name}"
validator="#{userBean.validateText}"/></td>
```

```
</tr>
```

```
<tr>
```

```
<td>Passord</td>
```

```
<td> <h:inputSecret size="20" id="pass"  
value="#{userBean.password}" /></td>
```

```
</tr>
```

```
<tr>
```

```
<td>
```

```
<h:commandButton type="submit"  
value="Login" action="#{userBean.login}" />
```

```
</td>
```

```
</tr>
```

```
</table>
```

```
</h:form>
```

```
</f:view>
```

- package beans;
- public class UserBean {
- String name;
- String password;
- public String getName() {
- return name;
- }
- public void setName(String name) {
- this.name = name;
- }

```
public String getPassword() {
```

```
    return password;
```

```
}
```

```
public void setPassword(String password) {
```

```
    this.password = password;
```

```
}
```

```
public HttpSession getSession()
{
    return ((HttpServletRequest) FacesContext.getCurrentInstance().
        getExternalContext().getRequest()).getSession();
}
```

```
public String login() {  
    System.out.println(name+":"+password);  
    if (LoginHandler.validateLogin(name, password)) {  
        HttpSession session = getSession();  
        session.setAttribute("user", name);  
        return "success";  
    } else  
        return "failed";  
}
```

```
public void validateText(FacesContext ctx, UIComponent toVal, Object val) {  
  
    String txt = (String) val;  
  
    if (txt.length() < 4) {  
  
        ((UIInput)toVal).setValid(false);  
  
        ctx.addMessage(toVal.getClientId(ctx), new FacesMessage("Du måste  
ange ett användarnamn med minst längd 4."));  
  
    }  
  
    System.out.println("Tjoho"+txt);  
  
}
```



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

```
<faces-config
```

```
  xmlns="http://java.sun.com/xml/ns/javaee"
```

```
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
```

```
    http://java.sun.com/xml/ns/javaee/web-facesconfig_1_2.xsd"
```

```
  version="1.2">
```

```
<managed-bean>
```

```
  <managed-bean-name>userBean</managed-bean-name>
```

```
  <managed-bean-class>beans.UserBean</managed-bean-  
class>
```

```
  <managed-bean-scope>session</managed-bean-scope>
```

```
</managed-bean>
```

<navigation-rule>

<from-view-id>/login.jsp</from-view-id>

<navigation-case>

<from-outcome>success</from-outcome>

<to-view-id>/login-success.jsp</to-view-id>

</navigation-case>

<navigation-case>

<from-outcome>failed</from-outcome>

<to-view-id>/login-failed.jsp</to-view-id>

</navigation-case>

</navigation-rule>