

# Üst Düzey Programlama

## Struts Framework

# Form Alanlarının Geçerlilik Kontrolü (Form Field Validation)

1. Geçerlilik kontrolünü action da yapmak:

- En güçlü olandır, her tür nesneye buradan erişebilirsiniz(DB,Service,Security,...)
- Aynı işlerin birden fazla yapılması gerekebilir.
- Girdi sayfasına durum bilgilerinin geri döndürülmesi işlemi siz yapmalısınız.
- Geçerlilik kurallarını sizin yazmanız gereklidir.

# Form Alanlarının Geçerlilik Kontrolü (Form Field Validation)

2. Geçerlilik kontrolünü FormBean de yapmak:

- setter metodlarında yapılabilir ve yanlış değer atanması engellenebilir.

- validate()** metodu kullanılabilir.

- \*Birden fazla actionda tekrar yazılmak zorunda değildir.

- \*Otomatik olarak girdi sayfasını tekrar gösterir.

- \*Hala kendi kurallarınızı yazmalısınız.

# Form Alanlarının Geçerlilik Kontrolü (Form Field Validation)

## 3. Otomatik geçerlilik kontrolü yapmak:

- Birçok durum için hazır kontrolcüler vardır.
- Bu kontroller için Javascriptler de bulunmaktadır.

# Action da geçerlilik kontrolü yapmak

- ActionForm kendi tipine dönüştürülür.
- getter metotları ile değerler alınır ve kontrol yapılır.
- Oluşan her hata için bir hata nesnesini bir bean e ekle
  - \*genelde ActionForm nesnesine eklenir.
  - \*mapping.findForward() ile hata sayfasına git.
- Hataları oluşan sayfada <bean:write> etiketi ile yazdır.

```
<@page contentType="text/html"%>
<@page pageEncoding="UTF-8"%>

<@ taglib uri="/WEB-INF/struts-html.tld" prefix="html" %>

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <html:form action="form01Action">
      ADINIZ:<html:text property="ad"/><br>
      SOYADINIZ:<html:text property="soyad"/><br>
      <html:submit value="GÖNDER"/>
    </html:form>
  </body>
</html>
```

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

```
<!DOCTYPE struts-config PUBLIC
```

```
"-//Apache Software Foundation//DTD Struts Configuration 1.2//EN"
```

```
"http://jakarta.apache.org/struts/dtds/struts-config_1_2.dtd">
```

```
<struts-config>
```

```
  <form-beans>
```

```
    <form-bean name="Form01ActionForm"
```

```
      type="ce.ustduzey.struts.forms.Form01ActionForm" />
```

```
  </form-beans>
```

```
  <global-exceptions>
```

```
  <global-forwards>
```

```
  <action-mappings>
```

```
    <action path="/form01Action"
```

```
      type="ce.ustduzey.struts.actions.Form01Action"
```

```
      name="Form01ActionForm"
```

```
      scope="request">
```

```
        <forward name="basarili" path="/sonuclar/basarili.jsp" />
```

```
        <forward name="eksik-alan" path="/sonuclar/eksikalan.jsp" />
```

```
      </action>>
```

```
  </action-mappings>
```

```
  <message-resources parameter="ce/ustduzey/struts/ApplicationResource" />
```

```

package ce.ustduzey.struts.forms;

import ...

public class Form01ActionForm extends org.apache.struts.action.ActionForm {
    private String ad;
    private String soyad;
    private String uyari="";

    public Form01ActionForm() {...}
    public String getAd() {...}
    public void setAd(String ad) {...}
    public String getSoyad() {...}
    public void setSoyad(String soyad) {...}

    public String getUyari() {...}

    public void addUyari(String uyari) {
        this.uyari = this.uyari +
            "<b><font color=red>" + uyari + "</font></b><br>" ;
    }

    public void reset(ActionMapping mapping, ServletRequest request) {
        uyariTemizle();
    }

    private void uyariTemizle(){
        this.uyari = "";
    }

    public boolean eksikMi(String alan){
        if (alan==null || alan.trim().equals("")){
            return true;
        }
        return false;
    }
}

```



```
package ce.ustduzey.struts.actions;
```

```
+ import ...
```

```
public class Form01Action extends Action {
```

```
    public ActionForward execute(ActionMapping mapping, ActionForm form,  
        HttpServletRequest request, HttpServletResponse response)  
        throws Exception {
```

```
        Form01ActionForm form01 = (Form01ActionForm)form;  
        ActionForward forward = mapping.findForward("basarili");
```

```
        if (form01.eksikMi(form01.getAd())){  
            form01.addUyari("Adı Alanı Eksik");  
            forward=mapping.findForward("eksik-alan");  
        }
```

```
        if (form01.eksikMi(form01.getSoyad())){  
            form01.addUyari("Soyadı Alanı Eksik");  
            forward=mapping.findForward("eksik-alan");  
        }
```

```
        return forward;
```

```
    }
```

```
}
```

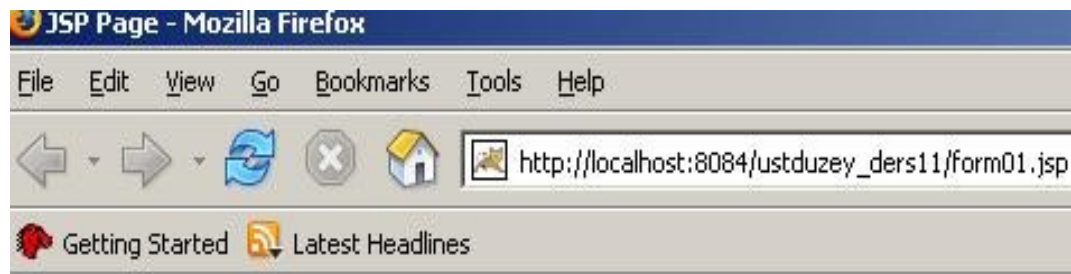
```
<%@page contentType="text/html"%>
<%@page pageEncoding="UTF-8"%>

<%@ taglib uri="/WEB-INF/struts-html.tld" prefix="html" %>
<%@ taglib uri="/WEB-INF/struts-bean.tld" prefix="bean" %>

<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>JSP Page</title>
</head>
<body>
  <bean:write name="Form01ActionForm" property="uyari" filter="false" />

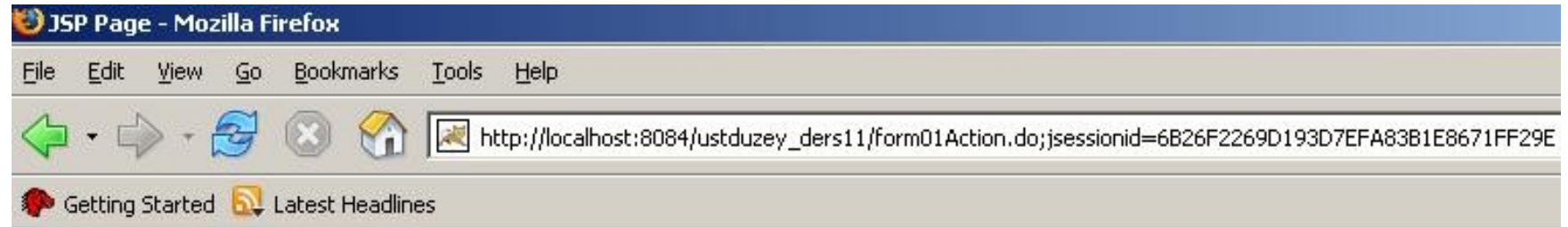
  <html:form action="form01Action">
    ADINIZ:<html:text property="ad" /><br>
    SOYADINIZ:<html:text property="soyad" /><br>
    <html:submit value="GÖNDER" />
  </html:form>

</body>
</html>
```



ADINIZ:

SOYADINIZ:



**Adı Alanı Eksik**

**Soyadı Alanı Eksik**

ADINIZ:

SOYADINIZ:

# ActionForm validate() metodunda geçerlilik kontrolü yapmak

- validate() metodu override edilir.
- eğer hata yoksa geriye **null** yada boş bir **ActionErrors** nesnesi döndürülür.
- Her hata için ActionErrors nesnesine bir **ActionMessage** eklenir.
- Eğer null yada boş ActionErrors nesnesi göndermezseniz sistem otomatik olarak kullanıcıyı struts-config.xml de tanımlı olan **input** alanına yönlendirir.

input: bu sayfada hataları **<html:errors/>** etiketi ile gösterebiliriz.

form02.jsp x

```
<%@page contentType="text/html"%>
<%@page pageEncoding="UTF-8"%>
<%@ taglib uri="/WEB-INF/struts-html.tld" prefix="html" %>

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <html:errors />
    <html:form action="form02Action">
      ADINIZ:<html:text property="ad" /><br>
      SOYADINIZ:<html:text property="soyad" /><br>
      <html:submit value="GÖNDER" />
    </html:form>
  </body>
</html>
```

struts-config.xml x

```
"-//Apache Software Foundation//DTD Struts Configuration 1.2//EN"
"http://jakarta.apache.org/struts/dtds/struts-config_1_2.dtd">

<struts-config>
  <form-beans>
    <form-bean name="Form02ActionForm"
      type="ce.ustduzey.struts.forms.Form02ActionForm" />
    <form-bean name="Form01ActionForm"
      type="ce.ustduzey.struts.forms.Form01ActionForm" />
  </form-beans>
  <action-mappings>
    <action>
      <action
        path="/form02Action"
        type="ce.ustduzey.struts.actions.Form02Action"
        input="/form02.jsp"
        name="Form02ActionForm"
        scope="request">
        <forward name="basarili" path="/sonuclar/basarili.jsp" />
      </action>
    </action-mappings>
    <message-resources parameter="ce/ustduzey/struts/ApplicationResource" />
  </struts-config>
```

```
Form02Action.java x
package ce.ustduzey.struts.actions;

import ...

public class Form02Action extends Action {

    public ActionForward execute(ActionMapping mapping, ActionForm form,
        HttpServletRequest request, HttpServletResponse response)
        throws Exception {

        return mapping.findForward("basarili");
    }
}
```



```
package ce.ustduzey.struts.forms;
```

```
+ import ...
```

```
public class Form02ActionForm extends org.apache.struts.action.ActionForm {  
    private String ad;  
    private String soyad;
```

```
+ public Form02ActionForm() {...}
```

```
+ public String getAd() {...}
```

```
+ public void setAd(String ad) {...}
```

```
+ public String getSoyad() {...}
```

```
+ public void setSoyad(String soyad) {...}
```

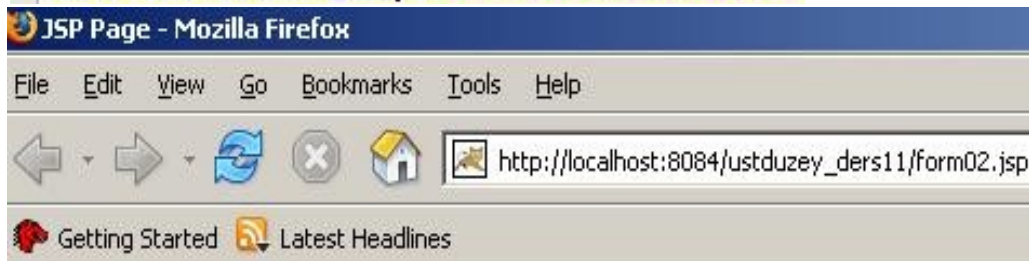
```
↑ [ public ActionErrors validate(ActionMapping mapping, HttpServletRequest request) {  
    ActionErrors errors = new ActionErrors();  
    if (eksikMi(getAd())) {  
        errors.add("ad", new ActionMessage("error.name.required"));  
    }  
    if (eksikMi(getSoyad())) {  
        errors.add("soyad", new ActionMessage("error.lastname.required"));  
    }  
    return errors;  
}  
[ private boolean eksikMi(String alan){  
    return (alan == null || alan.length() < 1);  
}  
}
```



ApplicationResource(default language) x

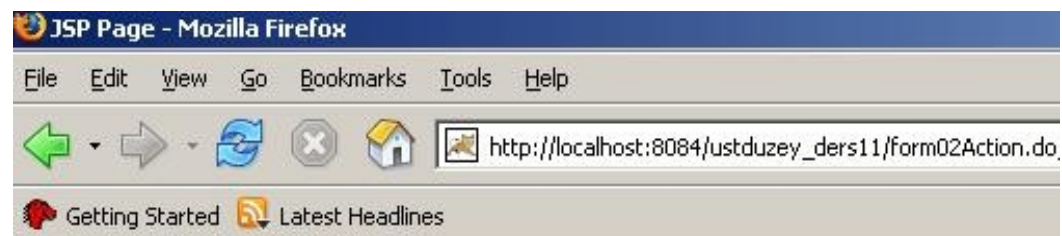
```
errors.header=<UL>
errors.prefix=<LI>
errors.suffix=</LI>
errors.footer=</UL>

error.name.required=Name is required
error.lastname.required=Last Name is required
```



ADINIZ:

SOYADINIZ:



- ◆ Name is required
- ◆ Last Name is required

ADINIZ:

SOYADINIZ:

# Otomatik geçerlilik kontrolü yapmak

- Struts tarafında size sunulan geçerlilik kontrolü eklentisini kullanırsınız.
- Geçerlilik kontrolü sunu tarafında olabileceği gibi, istemci tarafında da olabilir.
- XML dosyaları ile tanımlanırlar.

# Otomatik geçerlilik kontrolü yapmak

- İstemci tarafında geçerlilikte Javascript çalışır.
- Eğer alan değeri istenilen geçerliliğe uymaz ise bir mesaj kutusunda hata söylenir.
- değer geçersiz ise formun gönderilmesi(submit) yapılmaz.
- hızlıdır
- geçerlilik atlatılabilir
- daha fazla uygulama ile ilgili bir geçerlilik yapılamaz, girilen veri üzerinde yazımsal hatalarda kullanılabilir.
- Sunucuda geçerliliği sınıflar kontrol eder.
- Form hata mesajları ile yeniden gösterilir.

# Otomatik geçerlilik kontrolü yapmak

- 1.struts-config.xml dosyasında otomatik geçerlilik kontrollerini kullanmak için gerekli etiketleri girmelisiniz.
- 2.properties dosyasında struts'ın kullandığı etiketleri tanımlamalısınız.
- 3.geçerlilik kurallarını validation.xml içerisinde tanımlamalısınız.
4. Form bean nesnesi ValidatorForm dan türetilmelidir (eskiden ActionForm dan türetilmekteydi).
- 5.<html:errors> yine kullanılır.

```
<%@page contentType="text/html"%>
<%@page pageEncoding="UTF-8"%>
<%@ taglib uri="/WEB-INF/struts-html.tld" prefix="html" %>

<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title> Siparis Girisi</title>
</head>
<body>
  <html:errors />
  <html:form action="siparisAction"
    onsubmit="return validateSiparisActionForm(this);">
    <table>
      <tr><td>Adınız</td><td><html:text property="ad" /></td></tr>
      <tr><td>Soyadınız</td><td><html:text property="soyad" /></td></tr>
      <tr><td>Adres</td><td><html:textarea property="adres" /></td></tr>
      <tr><td>Posta kodu</td><td><html:text property="postaKodu" /></td></tr>
      <tr><td>Kredi kartı no</td><td><html:text property="krediKartiNo" /></td></tr>
      <tr><td>Email</td><td><html:text property="email" /></td></tr>
      <tr><td colspan="2"><html:submit value="Sipariş Ver" /></td></tr>
    </table>
  </html:form>
  <html:javascript formName="siparisActionForm" />
</body>
</html>
```

```
<struts-config>
  <form-beans>
    <form-bean name="siparisActionForm"
               type="ce.ustduzey.struts.forms.siparisActionForm"/>
    <form-bean>
    <form-bean>
  </form-beans>
  <action-mappings>
    <action>
    <action>
    <action
      path="/siparisAction"
      type="ce.ustduzey.struts.actions.siparisAction"
      input="/siparis.jsp"
      name="siparisActionForm"
      scope="request"
      validate="true">
      <forward name="basarili" path="/sonuclar/basarili.jsp" />
    </action>

  </action-mappings>
  <message-resources parameter="ce/ustduzey/struts/ApplicationResource"/>

  <!-- ===== Validator plugin =====>
  <plug-in className="org.apache.struts.validator.ValidatorPlugIn">
    <set-property
      property="pathnames"
      value="/WEB-INF/validator-rules.xml,/WEB-INF/validation.xml"/>
  </plug-in>
</struts-config>
```



```
SiparisAction.java x
package ce.ustduzey.struts.actions;

import ...

public class SiparisAction extends Action {
    public ActionForward execute(ActionMapping mapping, ActionForm form,
        HttpServletRequest request, HttpServletResponse response)
        throws Exception {

        return mapping.findForward("basarili");
    }
}
```



```
siparisActionForm.java x |
package ce.ustduzey.struts.forms;

import ...

public class siparisActionForm extends ValidatorForm {
    private String ad="";
    private String soyad="";
    private String adres="";
    private String postaKodu="";
    private String krediKartiNo="";
    private String email="";

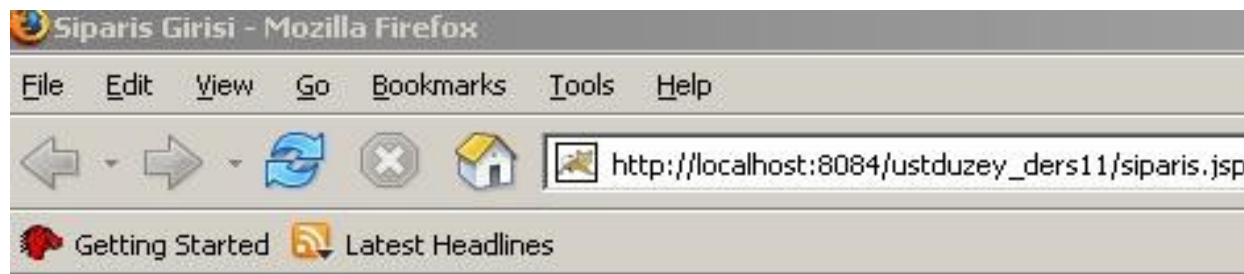
    public siparisActionForm() {...}
    public String getAd() {...}
    public void setAd(String ad) {...}
    public String getSoyad() {...}
    public void setSoyad(String soyad) {...}
    public String getAdres() {...}
    public void setAdres(String adres) {...}
    public String getPostaKodu() {...}
    public void setPostaKodu(String postaKodu) {...}
    public String getKrediKartiNo() {...}
    public void setKrediKartiNo(String krediKartiNo) {...}
    public String getEmail() {...}
    public void setEmail(String email) {...}
}
```



validation.xml x

```
<!DOCTYPE form-validation PUBLIC
"-//Apache Software Foundation//DTD Commons Validator Rules Configuration
"http://jakarta.apache.org/commons/dtds/validator_1_1_3.dtd">

<form-validation>
  <global>
    <formset>
      <form name="siparisActionForm">
        <field property="ad" depends="required">
          <arg key="siparisForm.name.required"/>
        </field>
        <field property="soyad" depends="required">
          <arg key="siparisForm.lastname.required"/>
        </field>
        <field property="adres" depends="required">
          <arg key="siparisForm.address.required"/>
        </field>
        <field property="postaKodu" depends="required,mask">
          <arg key="siparisForm.postakodu"/>
          <var>
            <var-name>mask</var-name>
            <var-value>^\d{5}\d*$</var-value>
          </var>
        </field>
        <field property="krediKartiNo" depends="required,creditCard">
          <arg key="siparisForm.krediKarti"/>
        </field>
        <field property="email" depends="required,email">
          <arg0 key="siparisForm.email"/>
        </field>
      </form>
    </formset>
  </form-validation>
```



Adınız

Soyadınız

Adres

Posta kodu

Kredi kartı no

Email

Sipariş Ver



# Üst Düzey Programlama

## Struts Framework