

Java technológiák – 6. előadás

JSP elemkönyvtárak

ANTAL Margit

Sapientia - EMTE

2010

- JSP és JSTL
- A JSP elemkönyvtárak anatómiája
- JSTL

Kód újrafelhasználásának fejlődése

- 1 Copy - paste
- 2 Függvénykönyvtárak - Libraries
- 3 Osztálykönyvtárak - Class libraries
- 4 Keretrendszerek – Frameworks

JSTL = JavaServer Pages Standard Tag Library

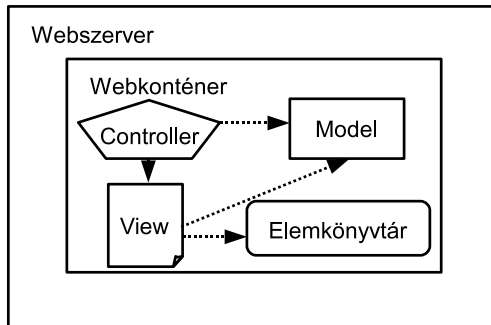


Figure: Az elemkönyvtárak helye az MVC architektúrában [SL314]

Mit használtunk eddig?

- JSP szkriptek `<% ... %>`
- JSP standard elemek `<jsp:useBean ... />`
- EL (Expression Language) `${course.name}`

Melyek a különbségek a fenti három kódbeágyazási technika között?

Listázás szkriptlettel

```
<%@page import = "java.util.*" %>
<%@page import = "model.*" %>
<ul>
<%
    List<Course> courses =
        request.getAttribute("courses");
    if( courses != null && !courses.isEmpty()){
        Iterator<Course> it = courses.iterator();
        while( it.hasNext() ){
            Course course = it.next();
%>
<li>
    <%= course.getName()%>
</li>
<%    }//endwhile
    }//endif %>
</ul>
```

```
<%@ taglib prefix="c"
    uri="http://java.sun.com/jsp/jstl/core" %>
<ul>
<c:forEach var="course" items="${courses}">
    <li><c:out value="${course.name}"/> </li>
</c:forEach>
</ul>
```

- mag (core) – `<c:out.../>`
- XML kezelő elemek – `<x:out.../>`
- SQL parancsokat támogató elemek – `<sql:query.../>`
- nemzetköziesítés és formázás – `<fmt:out.../>`
- szövegkezelő függvények – `<fn:split.../>`

Tagkezelők

JAR fájl – Java osztályok gyűjteménye

Tagleíró

TLD fájl – XML állomány, a tagelemek leírását tartalmazza

JSP lap

```
<%@ taglib prefix="c"
    uri="http://java.sun.com/jsp/jstl/core" %>
<html>
    ...
    <c:out value="${param}"/>
    ...
</html>
```

Konfigurálás: Logikai név (URL) \Rightarrow Fizikai fájl

```
<taglib>
  <taglib-uri>
    http://www.ms.sapientia.ro/sajatelemek.tld
  </taglib-uri>
  <taglib-location>
    /WEB-INF/sajatelemek.tld
  </taglib-location>
</taglib>
```

Használat

```
<%@taglib prefix="sajat"
  uri="http://www.ms.sapientia.ro/sajatelemek.tld"%>
```

- mag – <http://java.sun.com/jsp/jstl/core> –
`<c:out.../>`
- XML kezelő elemek – <http://java.sun.com/jsp/jstl/x> –
`<x:out.../>`
- SQL parancselemek - <http://java.sun.com/jsp/jstl/sql>
– `<sql:query.../>`
- nemzetköziesítés és formázás -
<http://java.sun.com/jsp/jstl/fmt> – `<fmt:out.../>`
- szövegkezelő függvények -
<http://java.sun.com/jsp/jstl/fn> – `<fn:split.../>`

Törzs részt tartalmazó elem

```
<c:forEach var="course" items="${courses}">  
    //torzs  
</c:forEach>
```

Törzs nélküli elem

```
<c:out value="${course.name}"/>
```

```
<c:out  
  value      = "value"  
  [escapeXml= {true\false}]  
  [default = "defaultValue"]  
>
```

```
<c:out  
  value      = "value"  
  [escapeXml= {true\false}]  
>  
default value  
</c:out>
```

```
<c:out value="{param.name}">  
  hiányzik a nev parameter  
</c:out>
```

escapeXml

```
<c:out value="<b>escapeXml=true </b>" escapeXml=true />  
<c:out value="<b>escapeXml=false</b>" escapeXml=false/>
```

Eredmény:

```
<b>escapeXml=true </b>
```

```
escapeXml=false
```

A set elem

```
<c:set  
  var="varName"  
  value="value"  
  [scope="{page|request|session|application}"]  
>
```

```
<c:set  
  var="varName"  
  [scope="{page|request|session|application}"]  
>  
  value  
</c:set>
```


A set használata Java babok esetében

- `<c:set`
 `target = "targetName"`
 `property = "propertyName"`
 `value = "value"`
 `/>`
- `<c:set`
 `target = "targetName"`
 `property = "propertyName"`
 `>`
 `value`
 `</c:set>`
- `target` : létező JavaBean vagy Map objektum
- `property` : tulajdonság neve
- `value` : tulajdonság értéke

Szintaxis

```
<c:if
  test      = "expression"
  [var      = "varName"]
  [scope    = "{page|request|session|application}"]
>
  body
</c:if>
```

Példa

```
<%@ taglib prefix="c"
      uri="http://java.sun.com/jsp/jstl/core" %>
<%
  int number =(int) (Math.random()*10)+1;
  pageContext.setAttribute("number", number);
%>

<c:if test="{number}<5">
  <c:out value="{number}, nem atmeno jegy"/>
</c:if>

<c:if test="{number}>=5">
  <c:out value="{number}, atmeno jegy"/>
</c:if>
```

A choose, when és az otherwise elem

Példa

```
<c:choose>
  <c:when test="\${number >0 && number<5 }">
    <c:out value="\${number}, nem atmeno jegy"/>
  </c:when>

  <c:when test="\${number>4 && number<=10 }">
    <c:out value="\${number}, atmeno jegy"/>
  </c:when>

  <c:otherwise>
    <c:out value="\$Ilyen jegy nem letezik"/>
  </c:otherwise>
</c:choose>
```

Szintaxis

```
<c:forEach  
  items="collection"  
  [var="varName"]  
  [varStatus="varStatusName"]  
  [begin="begin"]  
  [end="end"]  
  //body  
</forEach>
```

A forEach elem - példák

1. példa

```
<c:forEach items="{courses}">  
  <p>Hello<p>  
</c:forEach>
```

2. példa

```
<c:forEach var = "course" items="{it}">  
  <p>  
    <c:out value="{course.name}"/>,  
    <c:out value="{course.price}"/>  
  </p>  
</c:forEach>
```

A varStatus attribútum

Példa

```
<ul>
  <c:forEach var ="course" varStatus="status"
            items="{it}">
    <li>
      <c:out value="{status.count}"/>. &nbsp;
      <c:out value="{course.name}"/>,
      <c:out value="{course.price}"/>
    </li>
  </c:forEach>
</ul>
```

Példa

```
<c:forTokens
  items="alma,szilva,eper,barack"
  delims=","
  var="token"  >
  <p><c:out value="{token}"  /></p>
</c:forTokens>
```


Szintaxis

```
<c:remove  
  var    = "varName"  
  scope = "scope"  
>
```

A catch elem

Szintaxis

```
<c:catch var="name">  
  body content  
</c:catch>
```

Példa

```
<c:catch var="e">  
  <%  
    int nr = 0;  
    out.print(100/nr);  
  %>  
</c:catch>  
  
<c:out value="{e.message}"/>
```

Helyi erőforrás

```
<c:import url="header.jsp">  
  <c:param name="pageTitle" value="Distance Education"/>  
</c:import>
```

Távoli erőforrás

```
<c:import url="http://www.sapientia.ro" />
```

Szintaxis

```
<c:url
  value    = "expression"
  context  = "expression"
  var      = "name"
  scope    = "scope"
>
  <c:param name="expression" value="expression"/>
  ...
</c:url>
```

Példa

```
<a href="<c:url value=' /add_question.jsp' />">
  Add a question
</a>
```

A redirect elem

Szintaxis

```
<c:redirect url="expression" context="expression">  
  <c:param name="expression" value="expression"/>  
  ...  
</c:redirect>
```

Példa

```
<c:catch var="exception">  
  <c:import url="http://www.ms.sapientia.ro"/>  
</c:catch>  
<c:if test="${not empty exception}">  
  <c:redirect url="/errors/error.jsp"/>  
</c:if>
```

- 1 A tagkezelő osztály vagy függvény elkészítése. (jar, class)
- 2 A tagleíró elkészítése. (tld)
- 3 A tagkezelő használata. (jsp)

1. A tagkezelő osztály elkészítése

```
package model;
import javax.servlet.jsp.*;
public class AfaTagHandler extends SimpleTagSupport{
    private float osszeg;
    public void doTag() throws JspException {
        JspWriter out=getJspContext().getOut();
        try {
            out.print(" Osszeg   : "+ osszeg);
            out.print(" Afa       : "+(osszeg*0.19));
            out.print(" Osszesen: "+(osszeg * 1.19));
        }catch (java.io.IOException ex) {
            throw new JspException(ex.getMessage());
        }
    }
    public void setOsszeg(float osszeg) {
        this.osszeg = osszeg;
    }
}
```

2. A tagleíró elkészítése

```
/WEB-INF/taghandlers/mytags.tld
```

```
<taglib ...>  
  <tlib-version>1.0</tlib-version>  
  <short-name>mytags</short-name>  
  <uri>/WEB-INF/taghandlers/mytags</uri>  
  <tag>  
    <name>afa</name>  
    <tag-class>model.AfaTagHandler</tag-class>  
    <body-content>scriptless</body-content>  
    <attribute>  
      <name>osszeg</name>  
      <required>true</required>  
      <rtexprvalue>true</rtexprvalue>  
      <type>float</type>  
    </attribute>  
  </tag>  
</taglib>
```


3. A tagkezelő használata

JSP lap

```
<%@ taglib prefix="sajat"  
    uri="/WEB-INF/taghandlers/mytags.tld"%>  
<html>  
    ...  
    <h2><sajat:afa osszeg="300"/></h2>  
    ...  
</html>
```