

- 1 JavaBean
- 2 1. Component-based Development
- 3 1) Component?
- 4 -특별한 기능을 수행하기 위해 독립적으로 개발/보급되고,
- 5 -잘 정의된 interface를 가지며,
- 6 -다른 component들과 조립되어
- 7 -응용 program에 활용할 수 있는
- 8 -Software 단위(부품)이다.
- 9
- 10 2)지원특성
- 11 -식별 가능(Identifiable)
- 12 -추적 가능(Traceable)
- 13 -교체 가능(Replaceable)
- 14 -인터페이스를 통해 접근 가능(Accessible)
- 15 -독립적으로 개발/test/재사용 가능
- 16 -캡슐화(Encapsulated)
- 17
- 18 3)장점
- 19 -개발기간 단축
- 20 -개발비용감소
- 21 -생산성 증대
- 22 -Risk 감소
- 23 -일관성 확대
- 24 -최고 solution 제공
- 25
- 26 4)단점
- 27 -설계기간 연장
- 28 -개발자교육 필요
- 29
- 30 5)JSP 에서 component를 사용하면서 얻을 수 있는 장점
- 31 -Presentation과 business logic의 분리-역할분담
- 32 -Component 재사용을 통한 생산성 향상
- 33 -유지 보수의 용이
- 34
- 35
- 36 2. JavaBeans
- 37 1)개발 specification
- 38 -상속에 대한 제한 없다.
- 39 -반드시 class를 package해야 한다.
- 40 -필요에 따라 직렬화를 할 수 있다.
- 41 -필요에 따라 기본생성자를 생성할 수 있다.
- 42 -멤버변수를 property 라고 부른다.
- 43 -Bean 에서 사용할 property는 반드시 private 접근제한자를 가져야 한다.
- 44 -private 접근제한자를 갖고 있는 property에 접근/설정하기 위해서는 반드시 public method를 사용해야 한다.
- 45 -public method는 반드시 setXxx()/getXxx()의 형식을 취해야 한다.
- 46 -만일 property의 datatype이 boolean형이면 getXxx() 대신 isXxx() method type을 취한다.
- 47 -추가적으로 필요한 method를 정의할 수 있다.
- 48
- 49 2)Component의 저장 위치
- 50 -WEB-INF\classes\
- 51
- 52
- 53 3. <jsp:useBean />
- 54 1)JavaBean 객체를 생성하거나 이미 생성된 객체를 추출하는 tag

```
55 2)객체의 default 생성자만 호출
56 3)Syntax
57 <jsp:useBean id="beanInstanceName" class="package.class"
    scope="page|request|session|application" />
58
59 4)id
60 -Java 식별자 규칙 적용
61 -주어진 page에서 유일한 이름
62 -대소문자 구별
63 -문자로 시작
64 -영문자 대소문자, 숫자, _ 만 가능
65 5)class
66 -사용하고자 하는 class이름의 전체이름으로 지정
67 6)scope
68 -page | request | session | application
69 -Scope 생략하면 page 로 설정
70 -JavaBean class의 유효시간과 사용 범위를 결정
71
72
73 4. Lab
74 1)useBeanDemo.jsp
75 <body>
76 <jsp:useBean id="today" class="java.util.Date" />
77 <%=today %>
78 </body>
79
80
81 5. Lab
82 1)LoginBean.java
83 package com.example.libs;
84
85 public class LoginBean{
86     private String userid;
87     private String passwd;
88     public LoginBean() {}
89     public String getUserid() {
90         return userid;
91     }
92     public void setUserid(String userid) {
93         this.userid = userid;
94     }
95     public String getPasswd() {
96         return passwd;
97     }
98     public void setPasswd(String passwd) {
99         this.passwd = passwd;
100     }
101 }
102
103 2)usebean.jsp
104 <jsp:useBean id="myBean" class="com.example.libs.LoginBean" />
105
106 <body>
107 <h1>useBean 사용 실습</h1>
```

```

108     <%
109         myBean.setUserid("test");
110         myBean.setPasswd("1234");
111     %>
112     사용자 아이디: <%= myBean.getUserid() %><br>
113     사용자 비밀번호: <%= myBean.getPasswd() %><br>
114 </body>
115
116 3)useBean1.jsp
117 <jsp:useBean id="myBean" class="com.example.libs.LoginBean" scope="session"/>
118
119 <body>
120 <h1>useBean scope사용 실습</h1>
121 <%
122     myBean.setUserid("test");
123     myBean.setPasswd("1234");
124 %>
125 사용자 아이디: <%= myBean.getUserid() %><br>
126 사용자 비밀번호: <%= myBean.getPasswd() %><br>
127 </body>
128
129 4)useBean2.jsp
130 <jsp:useBean id="myBean" class="com.example.libs.LoginBean" scope="session"/>
131
132 <body>
133 <h1>useBean scope 사용 실습</h1>
134
135 사용자 아이디: <%= myBean.getUserid() %><br>
136 사용자 비밀번호: <%= myBean.getPasswd() %><br>
137 </body>
138
139
140 6. <jsp:setProperty />
141 1)JavaBean 객체의 property 값을 수정할 때 사용
142 2)JavaBean 객체의 모든 setXxx() method를 호출하여 값 설정
143 3)JavaBean 객체의 지정된 property의 setXxx() method를 호출하여 값 설정
144 4)설정할 값을 상수로 할 것인지 parameter로 할 것인지를 결정
145 5)Syntax
146 <jsp:setProperty name="beanInstanceName"
147     {property="*" | property="propertyName" }
148     [{value="propertyValue" | param="parameterName"}]
149 />
150
151 6)name
152 useBean 의 id 와 동일한 이름사용
153
154 7)property="propertyName" value="propertyValue"
155 -property가 propertyName과 동일한 setXxx() method 호출
156 -설정되는 값은 value 속성에 주어지는 값
157 8)property="propertyName" param="parameterName"
158 -property 가 propertyName과 동일한 setXxx() method 호출
159 -설정되는 값은 Query 문자열에서 param 속성에 할당된 값과 동일한 name의 값
160 9)property="*" param="paramName"
161 -모든 setXxx() method 호출.

```

162 -설정되는 값은 Query 문자열에서 param 속성에 할당된 값과 동일한 name의 값  
163 10)property="\*" value="propertyValue"  
164 -모든 setXxx() method 호출  
165 -설정되는 값은 value 속성에 주어진 상수 값  
166  
167

## 168 7. Lab

### 169 1)useBeanSet.jsp

```
170 <jsp:useBean id="myBean" class="com.example.libs.LoginBean" />
171 <jsp:setProperty name="myBean" property="userid" value="test"/>
172 <jsp:setProperty name="myBean" property="passwd" value="1234"/>
173
174 <body>
175 <h1>setProperty 실습</h1>
176 사용자 아이디: <%= myBean.getUserid() %><br>
177 사용자 비밀번호: <%= myBean.getPasswd() %><br>
178 </body>
```

### 180 2)login.html

```
181 <h1>param 실습</h1>
182 <form action="useBeanParam.jsp" method="get">
183 <fieldset>
184 <legend>로그인 폼</legend>
185 <ul>
186 <li>
187 <label for="userid">아이디</label>
188 <input type="text" name="myUserId" />
189 </li>
190 <li>
191 <label for="passwd">비밀번호</label>
192 <input type="password" name="myPasswd" />
193 </li>
194 <li><input type="submit" value="전송"></li>
195 </ul>
196 </fieldset>
197 </form>
198 </body>
```

### 200 3)useBeanParam.jsp

```
201 <jsp:useBean id="myBean" class="com.example.libs.LoginBean" />
202 <jsp:setProperty name="myBean" property="userid" param="myUserId"/>
203 <jsp:setProperty name="myBean" property="passwd" param="myPasswd"/>
204
205 <body>
206 <h1>param 사용 실습</h1>
207 사용자 아이디: <%= myBean.getUserid() %><br>
208 사용자 비밀번호: <%= myBean.getPasswd() %><br>
209 </body>
```

### 211 4)login.html

```
212 <h1>param 실습</h1>
213 <form action="useBeanParamAll.jsp" method="get">
214 <fieldset>
215 <legend>로그인 폼</legend>
```

```

216         <ul>
217             <li>
218                 <label for="userid">아이디</label>
219                 <input type="text" name="myUserId" >
220             </li>
221             <li>
222                 <label for="passwd">비밀번호</label>
223                 <input type="password" name="myPasswd" >
224             </li>
225             <li><input type="submit" value="전송"></li>
226         </ul>
227     </fieldset>
228 </form>
229 </body>

```

### 231 5)useBeanParamAll.jsp

```

232 <jsp:useBean id="myBean" class="com.example.libs.LoginBean" />
233 <jsp:setProperty name="myBean" property="*" />
234
235 <body>
236     <h1>* 실습</h1>
237     사용자 아이디: <%= myBean.getUserid() %><br>
238     사용자 비밀번호: <%= myBean.getPasswd() %><br>
239 </body>

```

### 242 8. <jsp:getProperty />

243 1)JavaBean 객체의 property 값을 추출할 경우

244 2)JavaBean 의 getXxx() 호출

245 3)Syntax

```
246 <jsp:getProperty name="beanInstanceName" property="propertyName" />
```

248 4)name

249 -useBean 의 id 와 동일한 이름사용

251 5)property

252 -호출하고자 하는 getXxx() method의 Xxx() 부분만을 지정하는데 첫 글자는 무조건 소문자로 처리

254 6)주의사항

255 -getXxx() 호출시 null 이 리턴되면 NullPointerException 발생

256 -getXxx() 리턴 값이 무슨 유형이던지 String 형으로 변환되어 처리

### 259 9. Lab

260 1)useBeanGet.jsp

```

261 <jsp:useBean id="myBean" class="com.example.libs.LoginBean" />
262 <jsp:setProperty name="myBean" property="userid" value="test"/>
263 <jsp:setProperty name="myBean" property="passwd" value="1234"/>
264
265 <body>
266     <h1>getProperty 실습</h1>
267
268     사용자 아이디: <jsp:getProperty name="myBean" property="userid" /><br>
269     사용자 비밀번호: <jsp:getProperty name="myBean" property="passwd" /><br>

```

```
270     </body>
271
272
273 10. Lab
274 1)CountBean.java
275     package com.example.libs;
276     public class CountBean {
277         private int count;
278         public int getCount() {
279             return ++count;
280         }
281         public void setCount(int count) {
282             this.count = count;
283         }
284     }
285
286 2)requestcount.jsp
287     <%@ page import="com.example.libs.CountBean" %>
288     <jsp:useBean id="ct" class="com.example.libs.CountBean" scope="request">
289         <jsp:setProperty name="ct" property="count" value="1" />
290     </jsp:useBean>
291     <html>
292     <body>
293         <h1>Scope : request or page</h1>
294         귀하는 <font color='red' size='7'>
295         <jsp:getProperty property="count" name="ct"/>
296         </font> 번째 방문자입니다.<br />
297         <a href="requestcount.jsp">다시 방문</a>
298     </body>
299
300 3)sessioncount.jsp
301     <%@ page import="com.example.libs.CountBean" %>
302     <jsp:useBean id="ct" class="com.example.libs.CountBean" scope="session">
303         <jsp:setProperty name="ct" property="count" value="1" />
304     </jsp:useBean>
305     <html>
306     <body>
307         <h1>Scope : session</h1>
308         귀하는 <font color='red' size='7'>
309         <jsp:getProperty property="count" name="ct"/>
310         </font> 번째 방문자입니다.<br />
311         <a href="sessioncount.jsp">다시 방문</a>
312     </body>
313
314 4)applicationcount.jsp
315     <%@ page import="com.example.libs.CountBean" %>
316     <jsp:useBean id="ct" class="com.example.libs.CountBean" scope="application">
317         <jsp:setProperty name="ct" property="count" value="1" />
318     </jsp:useBean>
319     <html>
320     <body>
321         <h1>Scope : application</h1>
322         귀하는 <font color='red' size='7'>
323         <jsp:getProperty property="count" name="ct"/>
```

```
324     </font> 번째 방문자입니다.<br />
325     <a href="applicationcount.jsp">다시 방문</a>
326 </body>
```

327

328 5)ColorBean.java

```
329 package com.example.libs;
330 public class ColorBean {
331     private String name; //property
332     private String color;
333     public ColorBean(){
334         this.color = "black";
335     }
336     public String getName() {
337         return name;
338     }
339     public void setName(String name) {
340         this.name = name;
341     }
342     public String getColor() {
343         return color;
344     }
345     public void setColor(String color) {
346         this.color = color;
347     }
348 }
```

349

350 6)color.html

[illegible]

363

364 7)color.jsp

```

365 <%@ page import="com.example.libs.*" %>
366 <% request.setCharacterEncoding("utf-8"); %>
367 <jsp:useBean id="cb" class="com.example.libs.ColorBean">
368     <jsp:setProperty name="cb" property="*" />
369 </jsp:useBean>
370 <jsp:getProperty property="name" name="cb"/> 님!<br />
371 귀하가 좋아하시는 색깔은
372 <font color="<jsp:getProperty property='color' name='cb' />">
373 <jsp:getProperty property="color" name="cb" /></font>
374 <%--
375     <%
376         String name = request.getParameter("name");
377         String color = request.getParameter("color");

```

```
378         ColorBean cb = new ColorBean();
379         cb.setName(name);  cb.setColor(color);
380     %>
381 --%>
382 <%--<%=cb.getName() 님! <br />
383 귀하가 좋아하시는 색깔은 <font color="<%=cb.getColor() %>">
384         <%=cb.getColor() %></font> 이군요.--%>
```