

# 8\_19\_2022\_\_Web\_Scope

Step03\_Scope

Step04\_Final

## Step03\_Scope

- Step03\_Scope 의 index.jsp에서 시작 servlet 으로 이동해 사용한다.
  - servlet은 새 서블릿파일을 만들고 빈공간에 ctrl+space해서 service 한다음 super 지우면 된다.
  - req.setAttribute(String name, Object o)
    - key값(특정java객체)과 value값(저장할 data)으로 전달된다.
    - 마치 hashmap 에 put을 사용해서 값을 담듯 하면 된다.
- ▼ FortuneServlet.java (분업화된 구조를 볼 수 있다)

```
req.setAttribute("fortuneToday", str);
```

- 요청 전달자 객체(RequestDispatcher)

HttpServletRequest 객체의 .getRequestDispatcher("요청을 전달받을 webapp 폴더 안에 있는 jsp 페이지의 위치") 메소드를 사용해 얻어낼 수 있다.

```
RequestDispatcher rd=req.getRequestDispatcher("/test/fortune.jsp");
```

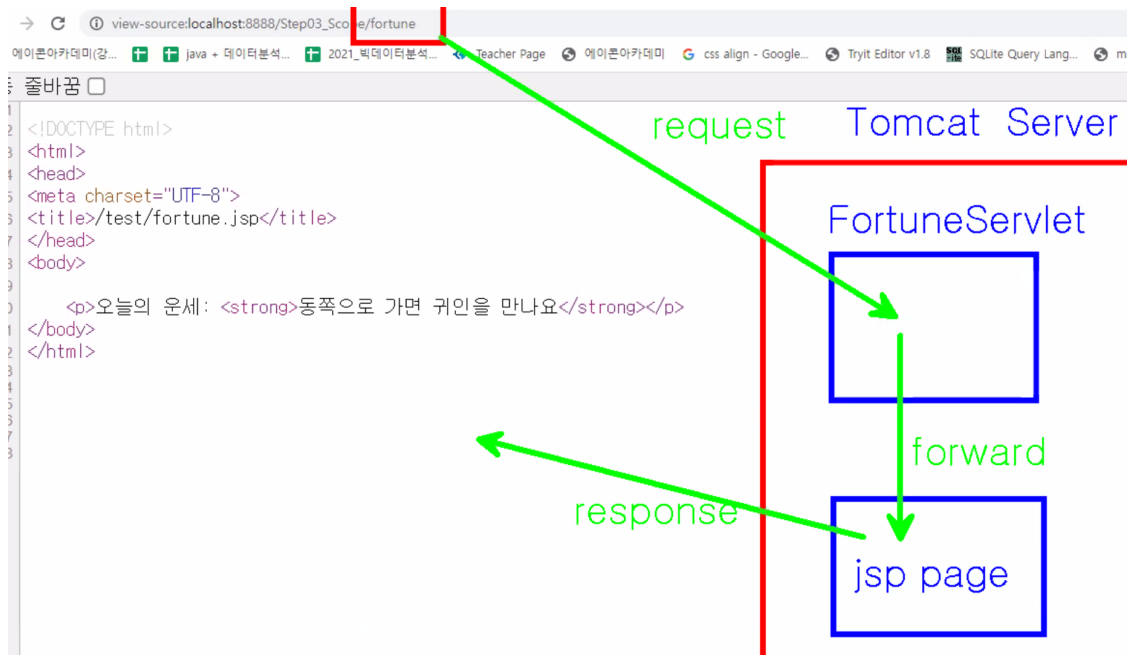
```
rd.forward(req, resp);
```

▼ fortune.jsp

```
String fortune=(String)request.getAttribute("fortuneToday");
```

- Servlet을 거쳐 가지 않으면 getAttribute로 FortuneServlet과 연결 되어 있기 때문에 값이 null이 나온다.

▼ 사진



◦ 이렇게 분리해서 사용하는 이유

- 개발자 입장에선 더 많은 일을 해야 할 지도 모르지만 기업입장에서 유지 보수와 훈련이 더 쉬워진다.
- 분업을 안하게 되면 jsp에서 logic 처리와 response를 같은 페이지에서 하게 되는데 이렇게 되면 유지 보수가 쉽지 않다.
- 나중에 Framework를 사용하게 되면 jsp 모든 java 코드를 제거(분리)할 수 있다. 결국 가독성이 좋아지게 된다.

• MemberServlet.java

- 어떤 key값을 사용했는지 기억을 잘 할 필요가 있다.

```
req.setAttribute("dto", dto);
```

- member.jsp

```
MemberDto dto=(MemberDto)request.getAttribute("dto");
```

• Scope(session 영역)

- request scope

▪

- session scope

▪

- 웹 서버는 동시에 불특정 다수의 요청을 받고 응답을 하는 application이다.
- 응답을 할 때 쿠키를 같이 요청하면 다음에 불러올 때 쿠키도 같이 불러와 식별한다.

▼ save.jsp

```
session.setAttribute("nick", nick);
```

```
session.setMaxInactiveInterval(60);
```

#### ▼ play.jsp

```
<%
//HttpSession 객체에 "nick" 이라는 키값으로 저장된 값이 있는지 읽어와 본다.
String nick=(String)session.getAttribute("nick");
%>
<h1>play.jsp 페이지 입니다.</h1>
<%if(nick==null){ %>
  <p>session 영역(scope) 에 "nick" 이라는 키값으로 저장된 값이 없습니다.</p>
<%}else{ %>
  <p>
    <strong><%=nick %></strong> 님 놀러 오셨네요? 신나게 놀다 가세요!
    <a href="logout.jsp">세션에 저장된 nick 삭제(로그아웃)</a>
  </p>
<%} %>
```

#### ▼ logout.jsp

```
String nick=(String)session.getAttribute("nick");
session.removeAttribute("nick");
```

#### ▼ saveServlet

```
package test.servlet;

import java.io.IOException;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

@WebServlet("/test/save")
public class SaveServlet extends HttpServlet{

    @Override
    protected void service(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
        req.setCharacterEncoding("utf-8");
        String nick=req.getParameter("nick");

        //HttpSession 객체에 "nick" 이라는 키값으로 입력한 닉네임을 저장하려고 하는데...
        HttpSession session=req.getSession();
        session.setAttribute("nick", nick);
        session.setMaxInactiveInterval(60);

        //jsp 페이지한테 대신 응답하라고 떠넘기기

        //forward 이동은 보안되는 폴더인 /WEB-INF/ 안에 있는 jsp 페이지로도 가능하다
        RequestDispatcher rd=req.getRequestDispatcher("/WEB-INF/views/monkey.jsp");
        rd.forward(req, resp);
    }
}
```



#### ▼ HttpServletRequest

.getRequestDispatcher()

.setAttribute()  
.getAttribute()  
.setCharacterEncoding()  
.getParameter()  
.getParameterValues()

▼ HttpSession

.setAttribute()  
.getAttribute()  
.removeAttribute()  
.invalidate()  
.setMaxInactiveInterval()

## Step04\_Final

- Expression Language \${ }

▼ 연산자

webapp/testel/test01.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>test_el/test01.jsp</title>
</head>
<body>
  <h1>Expression Language (EL)</h1>
  <p>jsp 페이지에서 특별하게 해석되는 코드 블록</p>
  <p>EL 영역은 &#36;{ } 로 만들수 있습니다.</p>

  <h2>산술연산</h2>
  <p>1+1 = ${1+1 }</p>
  <p>10-1 = ${10-1 }</p>
  <p>10*10 = ${10*10 }</p>
  <p>10/3 = ${10/3 }</p>

  <h2>비교연산</h2>
  <p>10 &gt; 2 : ${10 > 2 }</p>
  <p>10 &gt; 2 : ${10 gt 2 }</p>
  <p>10 &ge; 2 : ${10 >= 2 }</p>
  <p>10 &ge; 2 : ${10 ge 2 }</p>

  <p>10 &lt; 2 : ${10 < 2 }</p>
  <p>10 &lt; 2 : ${10 lt 2 }</p>
  <p>10 &le; 2 : ${10 <= 2 }</p>
  <p>10 &le; 2 : ${10 le 2 }</p>

  <p>10 == 10 : ${10 == 10 }</p>
  <p>10 == 10 : ${10 eq 10 }</p>
  <p>10 != 10 : ${10 != 10 }</p>
  <p>10 != 10 : ${10 ne 10 }</p>

  <h2>논리연산</h2>
  <p>true || false : ${true || false }</p>
  <p>true or false : ${true or false }</p>
  <p>true && false : ${true && false }</p>
  <p>true and false : ${true and false }</p>
  <p>!true : ${!true }</p>
  <p>not true: ${not true }</p>
```

```

<h2>empty 연산자 (비어 있는지 여부)</h2>
<p> null 혹은 빈 문자열("") 는 true 로 판정된다.</p>
<p> empty null : ${empty null }</p>
<p> empty "" : ${empty "" }</p>
<p> not empty null : ${ not empty null }</p>
<p> not empty "" : ${ not empty "" }</p>

<h3>3 항 연산</h3>
<p> ${ true ? 'coffee' : 'water' }</p>
<p> ${ false ? 'coffee' : 'water' }</p>
</body>
</html>

```

## ▼ 키값, 밸류값

webapp/testel/test02.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
    String myName="김구라";
    /*
        page scope 에 "myName" 이라는 키값으로 myName 담기
        해당 jsp 페이지 에서만 사용할수 있다.
    */
    pageContext.setAttribute("myName", myName);

    String yourName="해골";
    /*
        request scope 에 "yourName" 이라는 키값으로 yourName 담기
        request scope 에 담은 값은 응답하기 전까지 사용할수 있다.
        (다른 페이지로 forward 이동해도 사용할수 있다)
        (다른 페이지로 redirect 이동하면 사용할수 없다)
    */
    request.setAttribute("yourName", yourName);

    String ourName="원숭이";
    /*
        session scope 에 "ourName" 이라는 키값으로 ourName 담기
        session scope 에 담은 값은 세션이 유지 되는 동안 사용할수 있다.
        (다른 페이지로 forward, redirect 이동 해도 사용할수 있다)
        (웹브라우저를 닫거나 세션을 초기화 하기 전까지 사용할수 있다)
    */
    session.setAttribute("ourName", ourName);

    String companyName="에이콘";
    /*
        application scope 에 "companyName" 이라는 키값으로 companyName 담기
        application scope 에 담은 내용은 서버를 끄기 전까지 사용할수 있다.
        (웹브라우저를 닫아도 지워지지 않는다)
    */
    application.setAttribute("companyName", companyName);

%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>/test_el/test02.jsp</title>
</head>
<body>
    <h1> EL 로 page scope 에 저장된 값 추출</h1>
    <p>내이름은 <strong>${pageScope.myName }</strong></p>
    <p>내이름은 <strong>${myName }</strong></p>

    <h1>EL 로 request scope 에 저장된 값 추출</h1>
    <p>너의 이름은 <strong>${requestScope.yourName }</strong></p>
    <p>너의 이름은 <strong>${yourName }</strong></p>
    <!-- 위의 EL 은 아래의 코드를 대신할수 있다. -->
    <%
        String result = (String)request.getAttribute("yourName");
    %>
    <p>너의 이름은 <strong><%=result %></strong></p>

    <h1>EL 로 session scope 에 저장된 값 추출</h1>
    <p>우리 이름은 <strong>${sessionScope.ourName }</strong></p>

```

```

<p>우리 이름은 <strong>${ourName }</strong></p>
<!-- 위의 EL 은 아래의 코드를 대신할수 있다. -->
<%
    String result2 = (String)session.getAttribute("ourName");
    %>
<p>우리 이름은 <strong><%=result2 %></strong></p>

<h1>EL 로 application scope 에 저장된 값 추출</h1>
<p>회원 이름은 <strong>${applicationScope.companyName }</strong></p>
<p>회원 이름은 <strong>${companyName }</strong></p>
</body>
</html>

```

## ▼ dto 호출

### webapp/testel/test03.jsp

```

<%@page import="test.member.dto.MemberDto"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
    // MemberDto 객체를 생성해서 회원 한명의 정보를 담고
    MemberDto dto=new MemberDto();
    dto.setNum(1);
    dto.setName("김구라");
    dto.setAddr("노랑진");

    // "dto" 라는 키값으로 request scope 에 담기
    request.setAttribute("dto", dto);
    %>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>/test_el/test03.jsp</title>
</head>
<body>
    <%
        MemberDto result = (MemberDto)request.getAttribute("dto");
    %>
    <p> 번호 : <strong><%=result.getNum() %></strong></p>
    <p> 이름 : <strong><%=result.getName() %></strong></p>
    <p> 주소 : <strong><%=result.getAddr() %></strong></p>

    <!--
        dto 의 getter 메소드를 호출하는 표현식 대신에
        dto 의 필드명만 적어도 알아서 getter 메소드가 호출된다.
    -->
    <p> 번호 : <strong>${dto.num }</strong></p>
    <p> 이름 : <strong>${dto.name }</strong></p>
    <p> 주소 : <strong>${dto.addr }</strong></p>
</body>
</html>

```

## ▼ 파라미터 추출

### ▼ webapp/testel/test04.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>/test_el/test04.jsp</title>
</head>
<body>
    <h1>요청 파라미터도 el 로 추출할수 있다.</h1>
    <form action="test05.jsp" method="post">
        <input type="text" name="msg"/>
        <button type="submit">전송</button>
    </form>

```

```

<br>

<a href="test05.jsp?msg=hello">test05.jsp</a>
</body>
</html>

```

#### ▼ webapp/testel/test05.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>/test_el/test05.jsp</title>
</head>
<body>
<h1>param.파라미터명 형식으로 추출할수 있다</h1>
<p> 전달된 파라미터 : <strong>${param.msg }</strong></p>
<p> 전달될 파라미터 : <strong><%=request.getParameter("msg") %></strong></p>
</body>
</html>

```

#### ▼ 쿠키값 추출

#### ▼ webapp/testel/test06.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
    Cookie cook=new Cookie("savedId", "kimgura");
    cook.setMaxAge(60);
    response.addCookie(cook);
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>test06.jsp</title>
</head>
<body>
<h1>쿠키에 저장된 내용도 EL 로 추출할수 있다.</h1>
<p>savedId 라는 키값으로 60초 동안 유지되는 쿠키를 응답했습니다.</p>
<a href="test07.jsp">쿠키값 확인하기</a>
</body>
</html>

```

#### ▼ webapp/testel/test07.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>/test_el/test07.jsp</title>
</head>
<body>
<p> savedId 라는 키값으로 쿠키가 저장되었는지 여부 : ${not empty cookie.savedId }</p>
<p> savedId 라는 키값으로 저장된 값 : ${cookie.savedId.value }</p>
</body>
</html>

```

## ▼ 사용

```
<%
//HttpServletRequest 객체를 이용해서 요청과 함께 전달된 쿠키 읽어내기
Cookie[] cooks=request.getCookies();
//"savedMsg" 라는 키값으로 저장된 문자열을 저장할 변수
String savedMsg=null;
if(cooks!=null){
    //반복문 돌면서 쿠키 객체를 하나씩 참조
    for(Cookie tmp:cooks){
        //쿠키의 키값을 읽어온다.
        String key=tmp.getName();
        if(key.equals("savedMsg")){//만일 우리가 찾는 키값이라면
            //해당 키값으로 저장된 value 를 읽어낸다.
            savedMsg=tmp.getValue();
        }
    }
}
%>
```



### • <c:XXX>

#### ◦ 먼저 등록 해야함

```
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
```

```
<c:forEach var="i" begin="0" end="9" step="1">
    <p>안녕 ${i }</p>
</c:forEach>

<%for(int i=0; i<=9 ; i++){ %>
    <p>안녕 <%=i %></p>
<%} %>
```

## ▼ test02.jsp

```
<%@page import="java.util.ArrayList"%>
<%@page import="java.util.List"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%
//테스트를 위해 sample 데이터를 request scope 에 담는다.
List<String> names=new ArrayList<String>();
names.add("김구라");
names.add("해골");
names.add("원숭이");

request.setAttribute("list", names);
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>test_jstl/test02.jsp</title>
</head>
<body>
    <%
        List<String> list=(List<String>)request.getAttribute("list");
    %>
    <h1>친구 목록</h1>
    <ul>
        <%for(String tmp:list){ %>
            <li><%=tmp %></li>
        <%} %>
    </ul>
```



```

<h1>친구 목록</h1>
<ul>
  <c:forEach var="tmp" items="${list }">
    <li>${tmp }</li>
  </c:forEach>
</ul>

<h1>친구 목록 입니다. 인덱스 표시</h1>
<ul>
  <c:forEach var="tmp" items="${list }" varStatus="status">
    <li>${tmp } <strong>${status.index }</strong></li>
  </c:forEach>
</ul>

<h1>친구 목록 입니다. 순서표시</h1>
<ul>
  <c:forEach var="tmp" items="${list }" varStatus="status">
    <li>${tmp } <strong>${status.count }</strong></li>
  </c:forEach>
</ul>

<h1>친구 목록 입니다. 첫번째 인지 여부</h1>
<ul>
  <c:forEach var="tmp" items="${list }" varStatus="status">
    <li>${tmp } <strong>첫번째 : ${status.first}</strong></li>
  </c:forEach>
</ul>

<h1>친구 목록 입니다. 마지막 인지 여부</h1>
<ul>
  <c:forEach var="tmp" items="${list }" varStatus="status">
    <li>${tmp } <strong>마지막 : ${status.last}</strong></li>
  </c:forEach>
</ul>
</body>
</html>

```

```

<%
  List<String> list=(List<String>)request.getAttribute("list");
%>
<h1>친구 목록</h1>
<ul>
  <for (String tmp:list){ %>
    <li><%=tmp %></li>
  <%> %>
</ul>

<h1>친구 목록</h1>
<ul>
  <c:forEach var="tmp" items="${list }" >
    <li>${tmp }</li>
  </c:forEach>
</ul>

```

```

3 </head>
1 <body>
2 <h1>친구 목록</h1>
3 <ul>
4 <for (String tmp:list){ %>
5 <li><%=tmp %></li>
6 <%> %>
7 </ul>
8 <h1>친구 목록</h1>
9 <ul>
10 <c:forEach var="tmp" items="${list }" >
11 <li>${tmp }</li>
12 </c:forEach>
13 </ul>
14 <h1>친구 목록 입니다. 인덱스 표시</h1>

```

```

9 <meta charset="UTF-8">
10 <title>test_jst1/test02.jsp</title>
11 </head>
12 <body>
13 <h1>친구 목록</h1>
14 <ul>
15 <li>김구라</li>
16 <li>해골</li>
17 <li>원숭이</li>
18 </ul>
19 <h1>친구 목록</h1>
20 <ul>
21 <li>김구라</li>
22 <li>해골</li>
23 <li>원숭이</li>
24 </ul>
25 <h1>친구 목록 입니다. 인덱스 표시</h1>
26 <ul>
27 <li>0 김구라</li>
28 <li>1 해골</li>
29 <li>2 원숭이</li>
30 </ul>
31 <h1>친구 목록 입니다. 순서표시</h1>
32 <ul>
33 <li>0 김구라</li>
34 <li>1 해골</li>
35 <li>2 원숭이</li>
36 </ul>
37 <h1>친구 목록 입니다. 첫번째 인지 여부</h1>
38 <ul>
39 <li>0 김구라 첫번째 : true</li>
40 <li>1 해골 첫번째 : false</li>
41 <li>2 원숭이 첫번째 : false</li>
42 </ul>
43 <h1>친구 목록 입니다. 마지막 인지 여부</h1>
44 <ul>
45 <li>0 김구라 마지막 : false</li>
46 <li>1 해골 마지막 : false</li>
47 <li>2 원숭이 마지막 : true</li>
48 </ul>

```

```

status.index // 0,1,2,3
status.count // 1,2,3,4
status.first // 첫번째인지 여부
status.last // 마지막인지 여부

```

▼ test03.jsp

```

<%@page import="java.util.ArrayList"%>
<%@page import="test.file.dto.FileDto"%>
<%@page import="java.util.List"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%
    //Servlet 단에서 담은 sample 데이터라고 가정하자
    List<FileDto> list=new ArrayList<>();
    FileDto dto1=new FileDto();
    dto1.setNum(1);
    dto1.setWriter("김구라");
    dto1.setTitle("제목1");

    FileDto dto2=new FileDto();
    dto2.setNum(2);
    dto2.setWriter("해골");
    dto2.setTitle("제목2");

    FileDto dto3=new FileDto();
    dto3.setNum(3);
    dto3.setWriter("원숭이");
    dto3.setTitle("제목3");

    list.add(dto1);
    list.add(dto2);
    list.add(dto3);
    // EL 로 추출할수 있도록 request 영역에 담는다.
    request.setAttribute("list", list);
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>/test_jstl/test03.jsp</title>
</head>
<body>
    <h1>List 에 들어 있는 Dto 활용</h1>
    <table>
        <thead>
            <tr>
                <th>번호</th>
                <th>작성자</th>
                <th>제목</th>
            </tr>
        </thead>
        <tbody>
            <c:forEach var="tmp" items="${list }">
                <tr>
                    <td>${tmp.num }</td>
                    <td>${tmp.writer }</td>
                    <td>${tmp.title }</td>
                </tr>
            </c:forEach>
        </tbody>
    </table>
</body>
</html>

```

#### ▼ test04.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%
    // test 를 위해 sample data 를 request 영역에 담기
    request.setAttribute("jumsu", 75);

```

```

%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>/test_jstl/test04.jsp</title>
</head>
<body>
  <c:if test="${ 10%2 eq 0 }">
    <p>10 은 짝수 입니다.</p>
  </c:if>
  <c:if test="${ 11%2 eq 0 }">
    <p>11 은 짝수 입니다.</p>
  </c:if>
  <c:if test="${jumsu%2 eq 0 }">
    <p>${jumsu } 는 짝수 입니다.</p>
  </c:if>

  <p> 획득한 점수는 <strong>${jumsu }</strong> 입니다.</p>
  <p>
    따라서 이번 학기 학점은
    <c:choose>
      <c:when test="${jumsu ge 90 }">A</c:when>
      <c:when test="${jumsu ge 80 }">B</c:when>
      <c:when test="${jumsu ge 70 }">C</c:when>
      <c:when test="${jumsu ge 60 }">D</c:when>
      <c:otherwise>F</c:otherwise>
    </c:choose>
    입니다
  </p>
  <p>
    <strong>${jumsu }</strong> 는
    <c:choose>
      <c:when test="${jumsu%2 eq 0 }">
        짝수
      </c:when>
      <c:otherwise>
        홀수
      </c:otherwise>
    </c:choose>
    입니다.
  </p>

  <%
    String id=(String)session.getAttribute("id");
  %>

  <%if(id == null){ %>
    <p> 로그인이 필요 합니다.</p>
  <%}else{ %>
    <p> <strong><%=id %></strong> 님 로그인중...</p>
  <%} %>

  <c:choose>
    <c:when test="${ empty id }">
      <p>로그인이 필요 합니다.</p>
    </c:when>
    <c:otherwise>
      <p> <strong>${id }</strong> 님 로그인중...</p>
    </c:otherwise>
  </c:choose>
</body>
</html>

```

#### ▼ test05.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%
  //sample 데이터
  request.setAttribute("size", 123456789);
  request.setAttribute("price", 2500000);
%>
<!DOCTYPE html>
<html>

```

```

<head>
<meta charset="UTF-8">
<title>test05.jsp</title>
</head>
<body>
  <p>
    파일의 크기는
    <fmt:formatNumber value="123456789" pattern="#,###"/>
    byte 입니다.
  </p>
  <p>
    파일의 크기는
    <fmt:formatNumber value="${size }" pattern="#,###"/>
    byte 입니다.
  </p>
  <p>
    가격은
    <fmt:formatNumber value="2500000" type="currency"/>
    입니다.
  </p>
  <p>
    가격은
    <fmt:formatNumber value="${price }" type="currency"/>
    입니다.
  </p>
  <p>
    가격은
    <fmt:formatNumber value="${price }" type="currency" currencySymbol="$"/>
    입니다.
  </p>
</body>
</html>

```

#### ▼ test06.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<%
    //sample 데이터
    request.setAttribute("hobby", "game/programming/piano");
    request.setAttribute("friends", "김구라,해골,원숭이");
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>/test_jstl/test06.jsp</title>
</head>
<body>
  <h1>forTokens 활용</h1>
  <h2>취미 목록</h2>
  <ul>
    <c:forTokens var="tmp" items="${hobby }" delims="/">
      <li>${tmp }</li>
    </c:forTokens>
  </ul>
  <h2>친구 목록</h2>
  <ul>
    <c:forTokens var="tmp" items="${friends }" delims=",">
      <li>${tmp }</li>
    </c:forTokens>
  </ul>
</body>
</html>

```

#### ▼ test07.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%

```

```

//sample 데이터
request.setAttribute("msg", "abcde구라1234");
request.setAttribute("str", "<div></div> 는 문단을 나타내는 요소 입니다.");
request.setAttribute("content", "하나\n두울\n세엿");
request.setAttribute("newLine", "\n");
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>/test_jstl/test07.jsp</title>
</head>
<body>
  <h1>JSTL function (EL 코드 블록 안에서 사용하는 함수) 활용</h1>
  <p> "구라" 가 포함되어 있는지 여부 : ${fn:contains(msg, '구라')}</p>
  <c:if test="${fn:contains(msg, '구라')} ">
    <p>메세지에 "구라" 가 포함되어 있군요!</p>
  </c:if>
  <p>${str }</p>
  <p>${fn:escapeXml(str) }</p>
  <p>${content }</p>
  <!-- content 문자열에서 개행 기호를 <br> 로 바꾸서 출력 -->
  <p>${fn:replace(content, newLine, '<br>')}</p>
</body>
</html>

```

