

## 04. 품 관련 태그들

04-1 품 만들기

04-2 <input> 태그

04-3 <input> 태그의 다양한 속성

04-4 여러 데이터 나열해서 보여주기

04-5 기타 다양한 품 요소들





# 폼 만들기

## 폼의 동작 방식

LOGIN

로그인을 하시면 더욱 편리하게 이용하실 수 있습니다.

아이디를 입력하세요.

비밀번호를 입력하세요.

로그인

회원가입

아이디 찾기

비밀번호 찾기

- 1) 사용자가 [아이디]와 [비밀번호]에 정보 입력 → [로그인] 클릭
- 2) 사용자가 입력한 아이디와 비밀번호가 웹 서버로 보내짐
- 3) 서버는 자신이 가지고 있는 사용자 데이터베이스를 뒤져서 사용자가 보내온 아이디와 비밀번호가 서로 일치하는 정보인지 확인하고 그 결과를 브라우저에 보냄.
- 4) 회원이라면 로그인한 후의 결과 화면이 나타나고, 아니라면 로그인 실패 화면이 나타남.

# 폼 만들기

## <form> 태그

폼을 만드는 기본 태그. <form>과 </form> 사이에 여러 폼 요소 삽입

기본형 <form [속성="속성 값"] > 여러 폼 요소 </form>

## <form> 태그에서 사용 하는 속성들

속성	설명			
method	사용자가 입력한 내용들을 서버 쪽 프로그램으로 어떻게 넘겨줄지 지정합니다.			
	<table><tr><td>속성 값</td><td>get - 주소 표시줄에 사용자가 입력한 내용이 그대로 드러납니다. 256byte~4,096byte까지의 데이터만 서버로 넘길 수 있습니다.</td></tr><tr><td></td><td>post - 대부분 이 방식을 사용합니다. 사용자의 입력을 표준 입력(standard input)으로 넘겨주기 때문에 입력 내용의 길이에 제한을 받지 않고 사용자가 입력한 내용이 드러나지 않습니다.</td></tr></table>	속성 값	get - 주소 표시줄에 사용자가 입력한 내용이 그대로 드러납니다. 256byte~4,096byte까지의 데이터만 서버로 넘길 수 있습니다.	
속성 값	get - 주소 표시줄에 사용자가 입력한 내용이 그대로 드러납니다. 256byte~4,096byte까지의 데이터만 서버로 넘길 수 있습니다.			
	post - 대부분 이 방식을 사용합니다. 사용자의 입력을 표준 입력(standard input)으로 넘겨주기 때문에 입력 내용의 길이에 제한을 받지 않고 사용자가 입력한 내용이 드러나지 않습니다.			
name	폼의 이름을 지정합니다. 한 문서 안에 여러 개의 <form> 태그가 있을 경우, 폼들을 구분하기 위해 사용합니다.			
action	<form> 태그 안의 내용들을 처리해 줄 서버 상의 프로그램을 지정합니다.			
target	<action> 태그에서 지정한 스크립트 파일을 현재 창이 아닌 다른 위치에 열도록 지정합니다.			

<> form.html X

04 > <> form.html > html

```
<!DOCTYPE html>
```

```
<html lang="ko">
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<title>HTML5 폼</title>
```

```
</head>
```

```
<body>
```

```
<form action="search.php" method="post">
```

```
<input type="text" title="검색">
```

```
<input type="submit" value="검색">
```

```
</form>
```

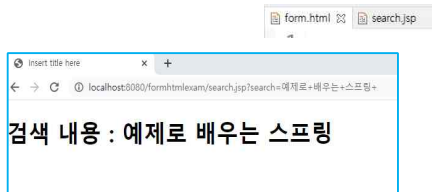
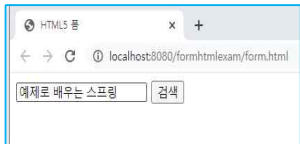
```
</body>
```

```
</html>
```

1. 검색어를 입력하고 [검색] 버튼을 클릭
2. 입력한 내용이 웹 서버에 있는 search.jsp 파일로 전송
3. 서버에서 search.jsp 파일 실행 후
4. 그 결과가 다시 웹 브라우저로 전달되어 화면에 표시됨

# <form> 태그 활용



# <form> 태그 활용



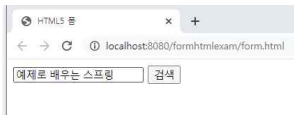
```
<!DOCTYPE html>

<html lang="ko">
<head>
  <meta charset="utf-8">
  <title>HTML5 품</title>
</head>
<body>
<form action="search.jsp" method="get">
<input type="text" title="검색" name = "search">
<input type="submit" value="검색">
</form>
</body>
</html>
```



```
<%@ page language="java" contentType="text/html; charset=utf-8"
    pageEncoding="utf-8"%>
<!DOCTYPE html>
<html>
<head>
<title>Insert title here</title>
</head>
<body>
<%
    request.setCharacterEncoding("utf-8");
    String search=request.getParameter("search");

    %>
    <h1>검색 내용 : <%= search %></h1>
</body>
</html>
```



검색 내용 : 예제로 배우는 스프링

# 폼 만들기

## <label> 태그

- 폼 요소에 레이블(텍스트)을 붙이는 태그
- 라디오 버튼이나 체크 박스에서 텍스트 부분을 클릭해도 라디오 버튼과 체크 박스 버튼이 선택된다.

### [방법1]

기본형    <label [속성="속성 값"] > 레이블 <input ...> </label>

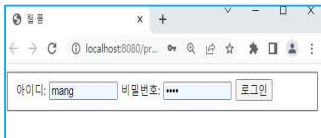
<label>아이디(6자 이상)<input type="text" .....></label>

### [방법2]

기본형    <label for="아이름">레이블</label>  
          <input id="아이름" [속성 = "속성 값"]>

<label for="user-id">아이디(6자 이상)</label>  
<input type="text" id="user-id">

## <label> 태그 활용





# <label> 태그 활용

```
<!doctype html>
```

```
<html lang="ko">
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<title> 웹 폼 </title>
```

```
<style>
```

```
label {
```

```
font-size:14px; /* 글자 크기 */
```

```
line-height:25px; /* 줄 간격 */
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<form method="get" action="/login" encType="UTF-8">
```

```
<fieldset>
```

```
<label>아이디: <input type="text" size="10" name="user_id" > </label>
```

```
<label>비밀번호: <input type="password" size="10" name="user_pw" > </label>
```

```
<input type="submit" value="로그인">
```

```
</fieldset>
```

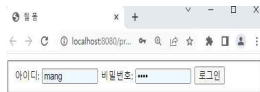
```
</form>
```

```
</body>
```

```
</html>
```

```
package sec01.ex01;
import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet("/login")
public class LoginServlet extends HttpServlet{
    public void init() throws ServletException {
        System.out.println("init 메서드 호출");
    }
    protected void doGet(HttpServletRequest request , HttpServletResponse response)
        throws ServletException, IOException {
        request.setCharacterEncoding("utf-8");
        String user_id=request.getParameter("user_id");
        String user_pw=request.getParameter("user_pw");
        System.out.println("아이디:"+user_id);
        System.out.println("비밀번호:"+user_pw);
    }
    public void destroy() {
        System.out.println("destroy 메서드 호출");
    }
}
```

login.html \*LoginServlet.java



Problems Servers Property  
Tomcat v9.0 Server at localhost [Apache

아이디:mang  
비밀번호:1234

# 폼 만들기

## <fieldset> 태그

폼 요소를 그룹으로 묶는 태그

기본형 <fieldset [속성="속성 값"] > ... </fieldset>

## <legend> 태그 <fieldset> 태그

그룹으로 묶는 구역에 제목을 붙이는 태그

개인 정보

이름	<input type="text"/>
메일 주소	<input type="text"/>

로그인 정보

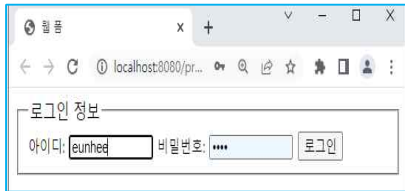
아이디	<input type="text"/>
비밀번호	<input type="text"/>

```
04 > < fieldset.html > html > head > style >
<!DOCTYPE html>

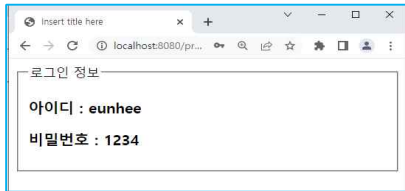
<html lang="ko">
<head>
  <meta charset="utf-8">
  <title>fieldset과 legend 태그</title>
  <style>
    ul {
      list-style:none;
    }
    li {
      margin:20px;
    }
    li label {
      width:80px;
      float:left;
    }
    fieldset {
      margin:15px;
    }
  </style>
</head>
```

```
<body>
  <form>
    <fieldset>
      <legend>개인 정보</legend>
      <ul>
        <li>
          <label for="name">이름</label>
          <input type="text" id="name">
        </li>
        <li>
          <label for="mail">메일 주소</label>
          <input type="text" id="mail">
        </li>
      </ul>
    </fieldset>
    <fieldset>
      <legend>로그인 정보</legend>
      <ul>
        <li>
          <label for="id">아이디</label>
          <input type="text" id="id">
        </li>
        <li>
          <label for="pwd">비밀번호</label>
          <input type="text" id="pwd">
        </li>
      </ul>
    </fieldset>
  </form>
</body>
</html>
```

# <legend> 태그 <fieldset> 태그 활용



A screenshot of a web browser window with the title "웹폼" and address bar "localhost:8080/pr...". The page contains a login form titled "로그인 정보" enclosed in a `<fieldset>` tag. Inside the form, there is a label "아이디:" followed by a text input field containing "eunhee", a label "비밀번호:" followed by a password input field with masked characters "....", and a "로그인" button.



A screenshot of a web browser window with the title "Insert title here" and address bar "localhost:8080/pr...". The page shows the same login form titled "로그인 정보" enclosed in a `<fieldset>` tag. The form now displays the filled-in values: "아이디 : eunhee" and "비밀번호 : 1234".

# 로그인 폼 만들기 활용

```
<!doctype html>

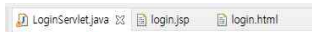
<html lang="ko">
  <head>
    <meta charset="utf-8">
    <title> 웹 폼 </title>
    <style>
      label {
        font-size:14px; /* 글자 크기 */
        line-height:25px; /* 줄 간격 */
      }
      label em{
        color:red; /* 글자색 */
        font-weight:800; /* 글자 굵기 */
      }
    </style>
  </head>
  <body>
    <form method="get" action="Login" encType="UTF-8">
      <fieldset>
        <legend>로그인 정보</legend>
        <label>아이디: <input type="text" size="10" name="user_id" ></label>
        <label>비밀번호: <input type="password" size="10" name="user_pw" ></label>
        <input type="submit" value="로그인">
      </fieldset>
    </form>
  </body>
</html>
```



# 로그인 폼 만들기 활용

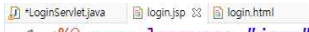
```
package sec01.ex01;
import java.io.IOException;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet("/login")
public class LoginServlet extends HttpServlet{

    protected void doGet(HttpServletRequest request , HttpServletResponse response)
        throws ServletException, IOException {
        request.setCharacterEncoding("utf-8");
        String user_id=request.getParameter("user_id");
        String user_pw=request.getParameter("user_pw");
        request.setAttribute("user_id", user_id);
        request.setAttribute("user_pw", user_pw);
        RequestDispatcher dispatch = request.getRequestDispatcher("/test01/login.jsp");
        dispatch.forward(request, response);
    }
    public void destroy() {
        System.out.println("destroy 메서드 호출");
    }
}
```



# 로그인 폼 만들기 활용

```
<%@ page language="java" contentType="text/html;
charset=utf-8"
    pageEncoding="utf-8"%>
<!DOCTYPE html>
<html>
<head>
<title>Insert title here</title>
</head>
<body>
<%
    request.setCharacterEncoding("utf-8");
    String user_id=request.getParameter("user_id");
    String user_pw=request.getParameter("user_pw");
%>
    <fieldset>
        <legend>로그인 정보</legend>
        <h3> 아이디   : <%= user_id %></h3>
        <h3> 비밀번호 : <%= user_pw %></h3>
    </fieldset>
</body>
</html>
```



# <input> 태그

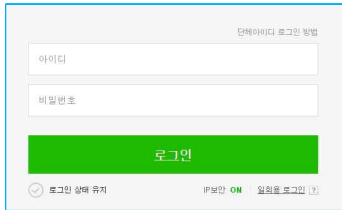
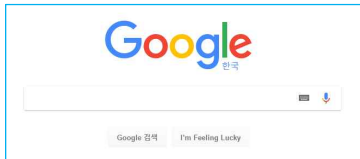
- 사용자가 입력하는 부분은 거의 <input> 태그를 이용해 처리
- 입력하는 내용의 종류는 <input> 태그의 type 속성을 통해 지정
- type 속성 값에 따라 함께 사용할 수 있는 속성들도 달라진다

기본형 <input type="유형" [속성="속성 값"] >

## <input> 태그의 id 속성

- 여러 번 사용된 폼 요소를 구분하기 위해 사용
- <label> 태그를 이용해 캡션을 붙일 수 있다.
- CSS를 이용해 각 요소마다 다른 형태로 꾸밀 수 있다.

```
<input type="text" id="user-name" size="10">  
<input type="text" id="addr" size="60">
```



# <input> 태그

유형	설명
hidden	사용자에게는 보이지 않지만 서버로 넘겨지는 값을 가집니다.
text	한 줄짜리 텍스트를 입력할 수 있는 텍스트 상자를 넣습니다.
search	검색 상자를 넣습니다.
tel	전화번호 입력 필드를 넣습니다.
url	URL 주소를 입력할 수 있는 필드를 넣습니다.
email	메일 주소를 입력할 수 있는 필드를 넣습니다.
password	비밀번호를 입력할 수 있는 필드를 넣습니다.
datetime	국제 표준시(UTC)로 설정된 날짜와 시간(연, 월, 일, 시, 분, 초, 분할 초)을 넣습니다.
datetime-local	사용자가 있는 지역을 기준으로 날짜와 시간(연, 월, 일, 시, 분, 초, 분할 초)을 넣습니다.
date	사용자 지역을 기준으로 날짜(연, 월, 일)를 넣습니다.
month	사용자 지역을 기준으로 날짜(연, 월)를 넣습니다.
week	사용자 지역을 기준으로 날짜(연, 주)를 넣습니다.

유형	설명
time	사용자 지역을 기준으로 시간(시, 분, 초, 분할 초)을 넣습니다.
number	숫자를 조절할 수 있는 화살표를 넣습니다.
range	숫자를 조절할 수 있는 슬라이드 막대를 넣습니다.
color	색상 표를 넣습니다.
checkbox	주어진 항목에서 2개 이상 선택 가능한 체크박스를 넣습니다.
radio	주어진 항목에서 1개만 선택할 수 있는 라디오 버튼을 넣습니다.
file	파일을 첨부할 수 있는 버튼을 넣습니다.
submit	서버 전송 버튼을 넣습니다.
image	submit 버튼 대신 사용할 이미지를 넣습니다.
reset	리셋 버튼을 넣습니다.
button	버튼을 넣습니다.



# <input> 태그

## type = "hidden"

- 화면 상의 폼에는 보이지 않는다.
- 폼을 서버로 전송할 때 서버로 함께 전송되는 요소

기본형 <input type="hidden" name="이름" value="서버로 넘길 값">

## type = "password"

- 비밀번호 입력란
- 사용자가 입력하는 내용이 ' \* '나 '•'로 표시된다

기본형 <input type="password" [속성="속성 값"]>

## type = "text"

- 한 줄짜리 텍스트 입력 필드
- 주로 아이디나 이름, 주소 등 텍스트 입력

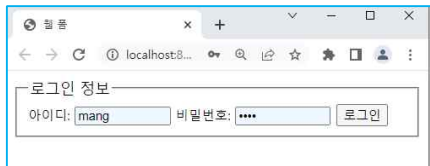
기본형 <input type="text" [속성="속성 값"]>

## 텍스트 필드와 패스워드 필드의 속성

(단, 패스워드 필드에는 value 속성이 없음)

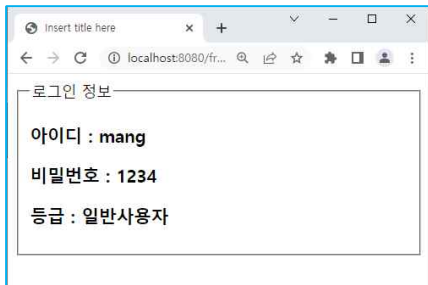
속성	설명
name	텍스트 필드를 구별할 수 있도록 이름을 붙입니다.
size	텍스트 필드의 길이를 지정합니다. 즉 화면에 몇 글자가 보이도록 할 것인지를 지정합니다. 예를 들어 최대 입력 가능한 글자 수가 10개인데 size를 5로 지정하면 텍스트 필드 크기가 5개 글자 크기에 맞추어져 나머지 5개 글자는 화면에 보이지 않습니다.
value	텍스트 필드 요소가 화면에 표시될 때 텍스트 필드 부분에 표시될 내용입니다. 이 속성을 사용하지 않으면 빈 텍스트 필드가 표시됩니다.
maxlength	텍스트 필드에 입력할 수 있는 최대 문자 개수를 지정합니다.

# <input> 태그 활용 1



로 로그인 정보

아이디:  비밀번호:



Insert title here

로 로그인 정보

아이디 : mang

비밀번호 : 1234

등급 : 일반사용자

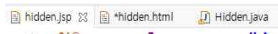
# <input> 태그 활용 1

```
<!doctype html>
<html lang="ko">
  <head>
    <meta charset="utf-8">
    <title> 웹 폼</title>
    <style>
      label {
        font-size:14px; /* 글자 크기 */
        line-height:25px; /* 줄 간격 */
      }
      label em{
        color:red; /* 글자색 */
        font-weight:800; /* 글자 굵기 */
      }
    </style>
  </head>
  <body>
    <form method="post" action="hidden" encType="UTF-8">
      <fieldset>
        <legend>로그인 정보</legend>
        <label>아이디: <input type="text" size="10" name="user_id" ></Label>
        <label>비밀번호: <input type="password" size="10" name="user_pw" ></Label>
        <input type="hidden" size="10" name="grade" value="일반사용자"></Label>
        <input type="submit" value="로그인">
      </fieldset>
    </form>
  </body>
</html>
```



# <input> 태그 활용 1

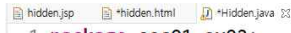
```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"
    isELIgnored="false" %>
<!DOCTYPE html>
<html>
<head>
<title>Insert title here</title>
</head>
<body>
<%
    request.setCharacterEncoding("utf-8");
    String user_id=request.getParameter("user_id");
    String user_pw=request.getParameter("user_pw");
%>
    <fieldset>
        <legend>로그인 정보</legend>
        <h3> 아이디   : <%= user_id %></h3>
        <h3> 비밀번호 : <%= user_pw %></h3>
        <h3> 등급    : ${grade}</h3>
    </fieldset>
</body>
</html>
```



# <input> 태그 활용 1

```
package sec01.ex02;
```

```
@WebServlet("/hidden")
public class Hidden extends HttpServlet {
    protected void doPost(HttpServletRequest request,
        HttpServletResponse response) throws ServletException, IOException {
        request.setCharacterEncoding("utf-8");
        String user_id=request.getParameter("user_id");
        String user_pw=request.getParameter("user_pw");
        String grade=request.getParameter("grade");
        request.setAttribute("user_id", user_id);
        request.setAttribute("user_pw", user_pw);
        request.setAttribute("grade", grade);
        RequestDispatcher dispatch =
        request.getRequestDispatcher("/test01/hidden.jsp");
        dispatch.forward(request, response);
    }
}
```



# <input> 태그

## type = "search"

- 검색 필드
- 검색 창에 x 표시가 되어 검색어 삭제 쉬움

기본형 <input type="search" [속성="속성 값"]>

## type = "url"

- 웹 주소 필드
- http://로 시작하는 사이트 주소 입력

기본형 <input type="url" [속성="속성 값"]>

## type = "email"

- 메일 주소 입력 필드
- 메일 주소 형식 자동 체크

기본형 <input type="email" [속성="속성 값"]>

## type = "tel"

- 전화번호 입력 필드
- 사용자 입력을 체크하지는 않음.

기본형 <input type="tel" [속성="속성 값"]>

# <input> 태그

```
1 <!doctype html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title> 웹 폼</title>
6     <style>
7       ul {
8         list-style:none;
9       }
10      li {
11        margin:10px;
12      }
13      li label {
14        width:120px;
15        float:left;
16        text-align:right;
17        padding-right:8px;
18      }
19      input[type="submit"] {
20        text-align:center;
21        width:100%;
22        height:30px;
23        margin-top:15px;
24      }
25    </style>
26  </head>
```

```
26 <body>
27   <h1>회원 가입</h1>
28   <form>
29     <fieldset>
30       <legend>로그인 정보</legend>
31       <ul>
32         <li>
33           <label for="user-id">아이디 </label>
34           <input type="text" id="user-id">
35         </li>
36         <li>
37           <label for="pwd1">비밀번호 </label>
38           <input type="password" id="pwd1">
39         </li>
40         <li>
41           <label for="pwd2">비밀번호 확인 </label>
42           <input type="password" id="pwd2">
43         </li>
44       </ul>
45     </fieldset>
```

```
46   <fieldset>
47     <legend>개인 정보</legend>
48     <ul>
49       <li>
50         <label for="user-name">이름 </label>
51         <input type="text" id="user-name">
52       </li>
53       <li>
54         <label for="mail">메일 주소</label>
55         <input type="email" id="mail">
56       </li>
57       <li>
58         <label for="phone">연락처</label>
59         <input type="tel" id="phone">
60       </li>
61       <li>
62         <label for="homep">블로그/홈페이지</label>
63         <input type="url" id="homep">
64       </li>
65     </ul>
66   </fieldset>
67   <input type="submit" value="가입하기">
68 </form>
69 </body>
70 </html>
```

회원 가입

로그인 정보

아이디

비밀번호

비밀번호 확인

개인 정보

이름

메일 주소

연락처

블로그/홈페이지

가입하기

# <input> 태그

## type = "number"

숫자 입력 필드. 브라우저에 따라 스펀 박스로 표시됨

기본형 `<input type="number" [속성="속성 값"]>`

`<b>주문 개수 : </b> <input type="number" min="1" max="5" value="1"> 개`



## type = "range"

숫자 입력 필드. 슬라이드 막대를 이용해 숫자 입력

기본형 `<input type="range" [속성="속성 값"]>`

`<b>주문 개수 : </b> <input type="range" min="1" max="5" value="1"> 개`



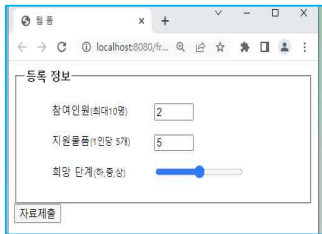


# <input> 태그

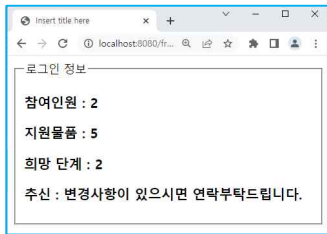
```
1 <!doctype html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title> 웹 폼</title>
6     <style>
7       body {
8         background-color: #fff;
9       }
10      form legend{
11        font-size:15px;
12        font-weight:600;
13      }
14      form label.reg {
15        font-size:14px;
16        width:150px;
17        float:left;
18      }
19      form ul li{
20        list-style:none;
21        margin: 15px 0;
22      }
23      #member, #stuffs {
24        width:50px;
25      }
26    </style>
27  </head>
```

```
28   <body>
29     <form>
30       <fieldset>
31         <legend> 등록 정보</legend>
32         <ul>
33           <li>
34             <label class="reg" for="member">참여인원<small>(최대10명)</small></label>
35             <input type="number" id="member" value="1" min="0" max="10" step="1">
36           </li>
37           <li>
38             <label class="reg" for="stuffs">지원물품<small>(1인당 5개)</small></label>
39             <input type="number" id="stuffs" value="1" min="0" max="50" step="5">
40           </li>
41           <li>
42             <label class="reg" for="satis">희망 단계<small>(하,중,상)</small></label>
43             <input type="range" id="satis" value="1" min="1" max="3">
44           </li>
45         </ul>
46       </fieldset>
47     </form>
48   </body>
49 </html>
```

## <input> 태그 활용 2



A screenshot of a web browser window titled '회원' (Member) at the URL 'localhost:8080/fr...'. The page contains a form titled '등록 정보' (Registration Information). The form has three input fields: '참여인원 (최대 10명)' (Number of participants, max 10) with the value '2', '지원물품 (1인당 5개)' (Support items, 5 per person) with the value '5', and '희망 단계 (하,중,상)' (Desired stage: low, medium, high) with a slider set to the middle. Below the form is a button labeled '자료제출' (Submit).



A screenshot of a web browser window titled 'Insert title here' at the URL 'localhost:8080/fr...'. The page contains a form titled '로그인 정보' (Login Information). The form has three pre-filled fields: '참여인원 : 2' (Number of participants: 2), '지원물품 : 5' (Support items: 5), and '희망 단계 : 2' (Desired stage: 2). Below these fields is a message: '추신 : 변경사항이 있으시면 연락부탁드립니다.' (Note: Please contact us if there are any changes). The browser's address bar shows 'localhost:8080/fr...'.

## <input> 태그 활용 2

```
<body>
<form method="post" action="number" enctype="UTF-8">
  <fieldset>
    <legend> 등록 정보</legend>
    <ul>
      <li>
        <label class="reg" for="member">참여인원<small>(최대10명)</small></label>
        <input type="number" id="member" value="1" min="0" max="10" step="1" name="number1">
      </li>
      <li>
        <label class="reg" for="stuffs">지원물품<small>(1인당 5개)</small></label>
        <input type="number" id="stuffs" value="1" min="0" max="50" step="5" name="number2">
      </li>
      <li>
        <label class="reg" for="satis">희망 단계<small>(하, 중, 상)</small></label>
        <input type="range" id="satis" value="1" min="1" max="3" name="range">
      </li>
    </ul>
  </fieldset>
  <input type="submit" value="자료제출"/>
</form>
</body>
</html>
```



```
<!doctype html>
<html lang="ko">
  <head>
    <meta charset="utf-8">
    <title> 웹 폼</title>
    <style>
      body {
        background-color:#fff;
      }
      form legend{
        font-size:15px;
        font-weight:600;
      }
      form label.reg {
        font-size:14px;
        width:150px;
        float:left;
      }
      form ul li{
        list-style:none;
        margin: 15px 0;
      }
      #member, #stuffs {
        width:50px;
      }
    </style>
  </head>
```

## <input> 태그 활용 2

```
package sec01.ex03;
```

```
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("/number")
```

```
public class Number extends HttpServlet {
```

```
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
```

```
        request.setCharacterEncoding("utf-8");
```

```
        String number1=request.getParameter("number1");
```

```
        String number2=request.getParameter("number2");
```

```
        String range=request.getParameter("grade");
```

```
        request.setAttribute("number1", number1);
```

```
        request.setAttribute("number2", number2);
```

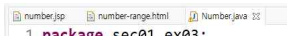
```
        request.setAttribute("range", range);
```

```
        RequestDispatcher dispatch = request.getRequestDispatcher("/test01/number.jsp?command=변경사항이 있으면 연락부탁드립니다.");
```

```
        dispatch.forward(request, response);
```

```
    }
```

```
}
```



## <input> 태그 활용 2

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"
    isELIgnored="false" %>
<!DOCTYPE html>
<html>
<head>
<title>Insert title here</title>
</head>
<%
    request.setCharacterEncoding("utf-8");
    String number1=request.getParameter("number1");
    String number2=request.getParameter("number2");
    String range=request.getParameter("range");
    String command=request.getParameter("command");
%>
<body>
    <fieldset>
        <legend>로그인 정보</legend>
        <h3> 참여인원 : <%= number1 %></h3>
        <h3> 지원물품 : <%= number2 %></h3>
        <h3> 희망 단계 : <%= range %> </h3>
        <h3> 추신 : <%= command %> </h3>
    </fieldset>
</body>
</html>
```

number.jsp number-range.html Number.java

# <input> 태그

## type = "radio"

여러 항목 중 하나만 선택할 때

기본형 <input type="radio" [속성="속성 값"]>

## type = "color"

색상 표에서 색상 선택

기본형 <input type="color" [value="기본 색" [속성="속성 값"]>

## type = "checkbox"

여러 항목 중 둘 이상을 선택할 때

기본형 <input type="checkbox" [속성="속성 값"]>

속성	설명
name	라디오 버튼이나 체크박스가 여러 개 있을 경우, 서버의 폼 프로그램에서 라디오 버튼이나 체크박스를 구분하기 위해 이름을 지정합니다. 라디오 버튼은 여러 개 중에서 한 가지만 선택하는 것이기 때문에 관련 있는 그룹끼리는 name 속성 값을 똑같이 만듭니다.
value	선택한 라디오 버튼이나 체크박스를 서버로 알려 줄 때 넘길 값을 지정합니다. 이 값은 영문이거나 숫자여야 하며 필수 속성입니다.
checked	라디오 버튼의 항목들은 처음에 아무 것도 선택되지 않은 상태로 화면에 표시되는데 기본으로 선택해 놓을 항목이 있다면 checked 속성을 사용합니다.

# <input> 태그

```
<!doctype html>
<html lang="ko">
```

```
<head>
  <meta charset="utf-8">
  <title> 웹 폼</title>
  <style>
    body {
      background-color: #ffff;
    }
    form fieldset{
      margin-bottom: 25px;
    }
    form legend{
      font-size: 15px;
      font-weight: 600;
    }
  </style>
</head>
```

```
form label.reg {
  font-size: 14px;
  color: #390;
  font-weight: bold;
  width: 110px;
  float: left;
  text-align: right;
  margin-right: 10px;
}
```

```
form ul li{
  list-style: none;
  margin: 15px 0;
  font-size: 14px;
}
```

```
#member, #stuffs {
  width: 50px;
}
```

```
#pre {
  margin-left: 15px;
}
</style>
```

```
</head>
```

```
<body>
```

```
<form>
```

```
<fieldset>
```

```
<legend>신청 과목</legend>
```

```
<p>이 달에 신청할 과목을 선택하세요 (1과목만 가능)</p>
```

```
<label><input type="radio" name="subject" value="speaking">회화</label>
```

```
<label><input type="radio" name="subject" value="grammar">문법</label>
```

```
<label><input type="radio" name="subject" value="writing">작문</label>
```

```
</fieldset>
```

```
<fieldset>
```

```
<legend>메일링</legend>
```

```
<p>메일로 받고 싶은 뉴스 주제를 선택해 주세요 (복수 선택 가능)</p>
```

```
<label><input type="checkbox" name="mailing1" value="news">해외 단신</label>
```

```
<label><input type="checkbox" name="mailing2" value="dialog">5분 회화</label>
```

```
<label><input type="checkbox" name="mailing3" value="pops">모닝팝스</label>
```

```
</fieldset>
```

```
</form>
```

## <input> 태그 활용 3

신청 과목

이 달에 신청할 과목을 선택하세요 (1과목만 가능)

☐ 회화 ☐ 문법 ☐ 작문

메일링

메일로 받고 싶은 뉴스 주제를 선택해 주세요 (복수 선택 가능)

☐ 해외 단신 ☐ 5분 회화 ☐ 모닝팝스

신청하기



```
Servers Properties Snippets Console Terminal  
omcat v9.0 Server at localhost [Apache Tomcat] C:\Program Files\Java\jdk-17.0.4\bin  
작문      5분 회화 모닝팝스
```



# <input> 태그 활용 3

```
<!doctype html>
<html lang="ko">
<head>
<meta charset="utf-8">
<title> 웹 폼</title>
<style>
body {
background-color:#fff;
}
form fieldset{
margin-bottom:25px;
}
form legend{
font-size:15px;
font-weight:600;
}
form label.reg {
font-size:14px;
color:#390;
font-weight:bold;
width:110px;
float:left;
text-align:right;
margin-right:10px;
}
form ul li{
list-style:none;
margin: 15px 0;
font-size:14px;
}
#member, #stuffs {
width:50px;
}
#pre {
margin-left:15px;
}
</style>
</head>
```

```
<body>
<form method="get" action="Radio_checkbox" enctype="UTF-8">
<fieldset>
<legend>신청 과목</legend>
<p>이 달에 신청할 과목을 선택하세요 (1과목만 가능)</p>
<label><input type="radio" name="subject" value="회화">회화</label>
<label><input type="radio" name="subject" value="문법">문법</label>
<label><input type="radio" name="subject" value="작문">작문</label>
</fieldset>
<fieldset>
<legend>메일링</legend>
<p>메일로 받고 싶은 뉴스 주제를 선택해 주세요 (복수 선택 가능)</p>
<label><input type="checkbox" name="mailing" value="해외 단신">해외 단신</label>
<label><input type="checkbox" name="mailing" value="5분 회화">5분 회화</label>
<label><input type="checkbox" name="mailing" value="모닝팝스">모닝팝스</label>
</fieldset>
<input type="submit" value="신청하기">
</form>
</body>
</html>
```



## <input> 태그 활용 3

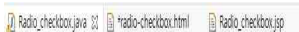
```
package sec01.ex04;

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;

@WebServlet("/Radio_checkbox")
public class Radio_checkbox extends HttpServlet {

    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        request.setCharacterEncoding("utf-8");
        String subject=request.getParameter("subject");
        String[] mailings=request.getParameterValues("mailing");
        System.out.print(subject+"\t");
        for(int i=0;i< mailings.length;i++) {
            if (mailings[i]==null) continue;
            else {
                System.out.print(mailings[i]+" \t");
            }
        }
    }
}
```



# <input> 태그

## type = "color"

색상 표에서 색상 선택

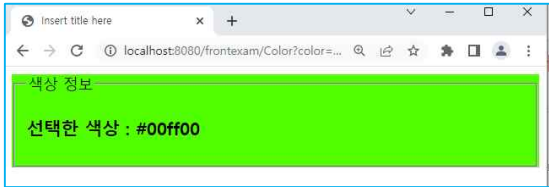
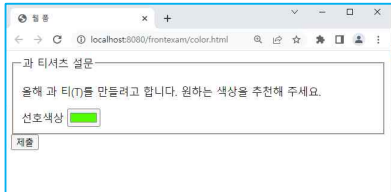
기본형 <input type="color" [value="기본 색" [속성="속성 값"]>

04 > <> color.html > html

```
1 <!doctype html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title> 웹 폼</title>
6   </head>
7   <body>
8     <form>
9       <fieldset>
10        <legend>과 티셔츠 설문</legend>
11        <p>올해 과 티(T)를 만들려고 합니다. 원하는 색상을 추천해 주세요.</p>
12        <label>선호색상 <input type="color" value="#00ff00"> </label>
13      </fieldset>
14    </form>
15  </body>
16 </html>
```



## <input> 태그 활용 4



## <input> 태그 활용 4

```
<!doctype html>
<html lang="ko">
  <head>
    <meta charset="utf-8">
    <title> 웹 폼</title>
  </head>
  <body>
<form method="get" action="Color" encType="UTF-8">
<fieldset>
<legend>과 티셔츠 설문</legend>
<p>올해 과 티(T)를 만들려고 합니다. 원하는 색상을 추천해 주세요.</p>
<label>선호색상 <input type="color" value="#00ff00" name="color"> </label>
</fieldset>
    <input type="submit" value="제출"/>
  </form>
</body>
</html>
```



## <input> 태그 활용 4

```
package sec01.ex05;

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;
```

```
@WebServlet("/Color")
public class Color extends HttpServlet {
    private static final long serialVersionUID = 1L;

    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        request.setCharacterEncoding("utf-8");
        String color=request.getParameter("color");
        request.setAttribute("color", color);
        RequestDispatcher dispatch = request.getRequestDispatcher("/test01/color.jsp");
        dispatch.forward(request, response);
    }
}
```

color.html Color.java color.jsp

## <input> 태그 활용 4

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"
    isELIgnored="false" %>
<!DOCTYPE html>
<html>
<head>
<title>Insert title here</title>
<style>
<%
    request.setCharacterEncoding("utf-8");
    String color=request.getParameter("color");
%>
div
{
    background-color:${color} ;
}
</style>

</head>
<body>
<div>
    <fieldset >
    <legend >색상 정보</legend>
    <h3> 선택한 색상 : <%= color %></h3>
    </fieldset>
    </div>
</body>
</html>
```



# <input> 태그

**type = "date", type="month", type="week"**

달력 이용해 날짜 입력

기본형 <input type="date | month | week" [value="기본 값" 속성="속성 값"]>

유형	설명
date	날짜를 선택합니다.
month	월(month)과 연도(year)를 선택합니다.
week	주(week)와 연도(year)를 선택합니다.



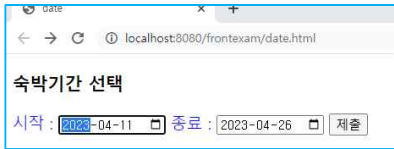
<> date.html X

04 > <> date.html > html

```
1  <!DOCTYPE html>
2
3  <html lang="ko">
4  <head>
5      <meta charset="utf-8">
6      <title>date</title>
7  </head>
8  <body>
9      <form>
10         <h3>조회기간 선택</h3>
11         <label><input type="date" id="start"></label>
12         <label><input type="date" id="end"></label>
13     </form>
14 </body>
15 </html>
```



## <input> 태그 활용 5

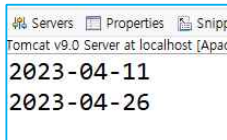


date

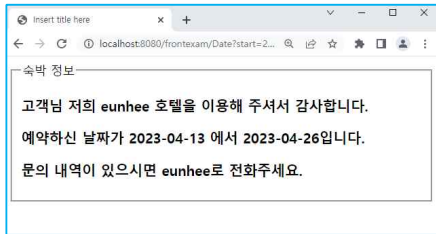
localhost:8080/frontexam/date.html

숙박기간 선택

시작 : 2023-04-11 종료 : 2023-04-26 제출



Servers Properties Snippets  
Tomcat v9.0 Server at localhost [Apache Tomcat v9.0.53]  
2023-04-11  
2023-04-26



Insert title here

localhost:8080/frontexam/Date?start=2023-04-11&end=2023-04-26

숙박 정보

고객님 저희 eunhee 호텔을 이용해 주셔서 감사합니다.  
예약하신 날짜가 2023-04-13 에서 2023-04-26입니다.  
문의 내역이 있으시면 eunhee로 전화주세요.

# <input> 태그 활용 5

```
<!DOCTYPE html>

<html lang="ko">
<head>
  <meta charset="utf-8">
  <title>date</title>
  <style>
    label {
      font-size:17px; /* 글자 크기 */
      color : blue;
      line-height:25px; /* 줄 간격 */
    }
  </style>
</head>
<body>
<form method="get" action="Date" enctype="UTF-8" >
<h3>숙박기간 선택</h3>
<label>시작 : <input type="date" id="start" name ="start"></label>
<label>종료 : <input type="date" id="end" name ="end"></label>
<input type="submit" value="제출">
</form>
</body>
</html>
```



# <input> 태그 활용 5

```
package sec01.ex06;
```

```
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;
```

```
@WebServlet("/Date")
```

```
public class Date extends HttpServlet {
```

```
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
        request.setCharacterEncoding("utf-8");  
        String start=request.getParameter("start");  
        String end=request.getParameter("end");  
        System.out.println(start);  
        System.out.println(end);  
        request.setAttribute("start", start);  
        request.setAttribute("end", end);  
        RequestDispatcher dispatch = request.getRequestDispatcher("/test01/date.jsp");  
        dispatch.forward(request, response);  
    }  
}
```

date.html

\*Date.java

date.jsp

# <input> 태그 활용 5

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"
    isELIgnored="false" %>
<!DOCTYPE html>
<html>
<head>
<title>Insert title here</title>
</head>
<body>
    <fieldset>
        <legend>숙박 정보</legend>
        <h3> 고객님 저희 eunhee 호텔을 이용해 주셔서 감사합니다.</h3>
        <h3> 예약하신 날짜가 ${start } 에서 ${end }입니다. </h3>
        <h3> 문의 내역이 있으시면 eunhee로 전화주세요. </h3>
    </fieldset>
</body>
</html>
```



# <input> 태그

## type = "time", type="datetime-local"

### 시간 입력

기본형 <input type="time | datetime | datetime-local" [value="기본 값" 속성="속성 값"]>

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="utf-8">
5   <title>time 속성</title>
6 </head>
7 <body>
8   <form>
9     <h3>대관시간을 선택하세요(오늘)</h3>
10    <label>시작 시간<input type="time" value="09:00" id="start1"></label>,
11    <label>종료 시간<input type="time" value="18:00" id="end1"></label>
12
13    <h3>대관시간을 선택하세요(다른날짜)</h3>
14    <label>시작 시간<input type="datetime-local" value="2016-03-02T09:00" id="start2"></label>,
15    <label>종료 시간<input type="datetime-local" value="2016-03-02T18:00" id="end2"></label>
16  </form>
17 </body>
18 </html>
```

### 사용하는 속성

속성	설명
min	날짜나 시간의 최소값을 지정합니다.
max	날짜나 시간의 최대값을 지정합니다.
step	스핀 박스의 화살표를 누를 때마다 날짜나 시간을 얼마나 조절할지를 지정합니다.
value	화면에 표시할 초기값을 지정합니다. type="time"일 경우, 시간은 00:00부터 23:59까지 입력하고 type="datetime"이나 type="datetime-local" 유형일 경우, 날짜 다음에 키워드 T를 쓰고 24시간제로 시간을 지정합니다. 예를 들어 오후 6시를 나타내려면 T18:00라고 하면 됩니다.

대관시간을 선택하세요(오늘)

시작 시간 2016-03-02 오전 09:00 × 종료 시간 2016-03-02 오후 06:00

대관시간을 선택하세요(다른날짜)

시작 시간 2016-03-02 오전 09:00 × 종료 시간 2016-03-02 오후 06:00

# <input> 태그

## type = "submit", type="reset"

- 폼 전송/리셋 버튼
- 전송(submit) 버튼 : 사용자 입력 내용을 서버로 전송
- 리셋(reset) 버튼 : 사용자 입력 내용 전부 삭제
- value 속성을 이용해 버튼 표시 내용 지정

기본형 <input type="submit | reset" [value="버튼 내용"] [속성="속성 값"]>

```
<form action="register.jsp" method="get">
  <label> 메일 주소 <input type="text"></label>
  <input type="submit" value="제출">
  <input type="reset" value="다시입력">
</form>
```

메일링 리스트 등록

메일 주소

<> submit-reset.html ●

04 > <> submit-reset.html > html > head

```
1  <!DOCTYPE html>
2
3  <html lang="ko">
4  <head>
5      <meta charset="utf-8">
6      <title>전송 - 리셋 </title>
7      <style>
8          body {
9              padding:20px;
10         }
11     </style>
12 </head>
13 <body>
14     <h3>메일링 리스트 등록</h3>
15     <form action="register.jsp" method="post">
16         <label> 메일 주소 <input type="text"></label>
17         <input type="submit" value="제출">
18         <input type="reset" value="다시입력">
19     </form>
20 </body>
21 </html>
```


# <input> 태그

## type = "image"

submit 버튼 대신 이미지 삽입

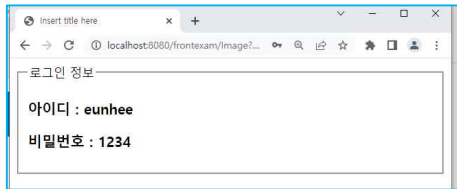
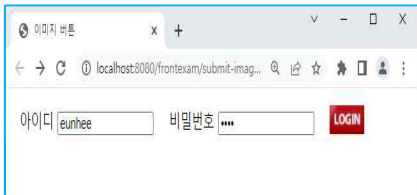
기본형 <input type="image" src="경로" alt="대체 텍스트" [속성="속성 값"]>

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="utf-8">
5   <title>이미지 버튼 </title>
6   <style>
7     table {
8       border:0;
9     }
10    td {
11      padding:5px 10px;
12      border:0;
13      vertical-align: middle;
14    }
15  </style>
16 </head>
```

아이디  비밀번호  

```
17 <body>
18   <form>
19     <table>
20       <tr>
21         <td><label>아이디 <input type="text" size="15"></label></td>
22         <td><label>비밀번호 <input type="password" size="15"></label></td>
23         <td><input type="image" id="butt" src="images/login.jpg" alt="login"></td>
24       </tr>
25     </table>
26   </form>
27 </body>
28 </html>
```

## <input> 태그 활용 6





## <input> 태그 활용 6

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="utf-8">
  <title>이미지 버튼 </title>
<style>
table {
border:0;
}
td {
padding:5px 10px;
border:0;
vertical-align: middle;
}
</style>
</head>
<body>
<form method="get" action="Image" enctype="UTF-8">
<table>
<tr>
<td><label>아이디 <input type="text" size="15" name="user_id"></label></td>
<td><label>비밀번호 <input type="password" size="15" name="user_pw"></label></td>
<td><input type="image" id="butt" src="images/login.jpg" alt="Login" onclick="submit()"></td>
</tr>
</table>
</form>
</body>
</html>
```



## <input> 태그 활용 6

```
package sec01.ex07;

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;

@WebServlet("/Image")
public class Image extends HttpServlet {

    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        request.setCharacterEncoding("utf-8");
        String user_id=request.getParameter("user_id");
        String user_pw=request.getParameter("user_pw");
        request.setAttribute("user_id", user_id);
        request.setAttribute("user_pw", user_pw);
        RequestDispatcher dispatch = request.getRequestDispatcher("/test01/login.jsp");
        dispatch.forward(request, response);
    }

}
```



## <input> 태그 활용 6

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"
    isELIgnored="false" %>
<!DOCTYPE html>
<html>
<head>
<title>Insert title here</title>
</head>
<body>
    <fieldset>
        <legend>로그인 정보</legend>
        <h3> 아이디   : ${user_id }</h3>
        <h3> 비밀번호 : ${user_pw }</h3>
    </fieldset>
</body>
</html>
```



# <input> 태그의 다양한 속성

## autofocus

페이지를 불러오자마자 원하는 폼 요소에 마우스 커서 표시

```
<li>  
  <label class="reg" for="id">학번</label>  
  <input type="text" id="id" autofocus>  
</li>
```

## placeholder

입력란에 표시하는 힌트로, 필드를 클릭하면 사라짐

```
<label class="reg" for="uname">이름</label>  
<input type="text" id="uname" autofocus>
```

```
<label class="reg" for="uid">학번</label>  
<input type="text" id="uid" placeholder="하이픈없이 입력">
```

A screenshot of a web form titled '신청자' (Applicant). It contains three input fields: '이름' (Name) with autofocus, '학번' (Student ID) with a placeholder '하이픈없이 입력' (Enter without hyphen), and '학과' (Department) with a dropdown menu showing '컴퓨터공학과' (Computer Science Department).

## readonly

- 내용을 보기만 하고 입력하지 못하게 함.
- 속성 값 없이 readonly 라고만 쓰면 됨.  
(readonly="readonly", readonly="true"로 표시하기도 함)

```
<label class="reg" for="subj">영어회화(초급)</label>  
<input type="text" id="subj" value="오전 9:00~11:00" readonly>
```

A screenshot of a form showing a '수강과목' (Course) section. It has a label '영어회화(초급)' (English Conversation (Beginner)) and a text input field containing '오전 9:00~11:00'. The input field is disabled (readonly). A speech bubble points to the field with the text '사용자가 내용을 입력할 수 없습니다' (User cannot enter content).

# <input> 태그 활용 7

```
<!doctype html>
<html lang="ko">
  <head>
    <title> 웹 폼</title>
    <meta charset="utf-8">
    <style>
  body {
background-color:#fff;
}
form fieldset{
margin-bottom:25px;
}
form legend{
font-size:15px;
font-weight:600;
}
form label.reg {
font-size:14px;
width:110px;
color:#390;
font-weight:bold;
float:left;
text-align:right;
margin-right:10px;
}
form ul li{
list-style:none;
margin: 15px 0;
font-size:14px;
}

#member, #stuffs {
width:50px;
}
</style>
</head>
```

```
<body>
  <h1> 여름방학 특강 신청</h1>
  <form method="get" action="Readonly"  enctype="UTF-8">
    <fieldset id="subject">
      <legend> 수강과목</legend>
      <ul>
        <li>
          <label class="reg" for="subj">영어회화(초급)</label>
          <input type="text" id="subj" value="오전 9:00~11:00" name="time" readonly>
        </li>
        <li>
          <label class="reg" for="subj">이름</label>
          <input type="text" id="subj" name="user_id" >
        </li>
      </ul>
    </fieldset>
    <fieldset>
      <button type="submit" value="submit">신청하기</button>
      <button type="reset" value="reset">다시쓰기</button>
    </fieldset>
  </form>
</body>
</html>
```

imagelogin.jsp

readonly.html

Readonly.java

# <input> 태그 활용 7

```
package sec01.ex08;
```

```
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;
```

```
/**
```

```
 * Servlet implementation class Readonly
```

```
 */
```

```
@WebServlet("/Readonly")
```

```
public class Readonly extends HttpServlet {
```

```
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
```

```
        request.setCharacterEncoding("utf-8");
```

```
        String user_id=request.getParameter("user_id");
```

```
        String time=request.getParameter("time");
```

```
        request.setAttribute("user_id", user_id);
```

```
        request.setAttribute("time", time);
```

```
        RequestDispatcher dispatch = request.getRequestDispatcher("/test01/readonly.jsp");
```

```
        dispatch.forward(request, response);
```

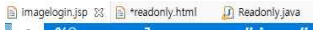
```
    }
```

```
}
```



## <input> 태그 활용 7

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"
    isELIgnored="false" %>
<!DOCTYPE html>
<html>
<head>
<title>Insert title here</title>
</head>
<body>
    <fieldset>
        <legend>로그인 정보</legend>
        <h3>아이디 : ${user_id }</h3>
        <h3>비밀번호 : ${user_pw } </h3>
    </fieldset>
</body>
</html>
```



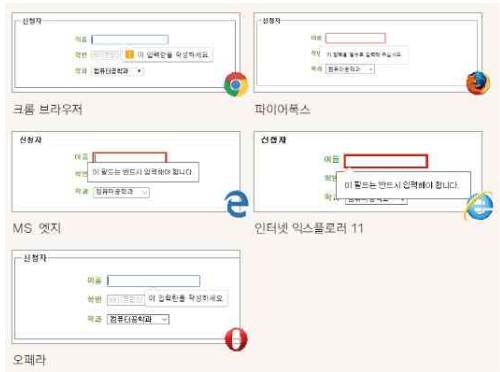
# <input> 태그의 다양한 속성

## required

- 필수 필드 체크
- 속성 값 없이 required 라고만 입력  
(required="required"라고 해도 됨)
- 필수 필드는 브라우저에서 직접 체크하는 것이므로 오류  
메시지 내용은 브라우저들마다 다르게 나타남

```
<label class="reg" for="uname">이름</label>
```

```
<input type="text" id="uname" autofocus required>
```





# <input> 태그의 다양한 속성

## min, max, step

- min, max : 해당 필드의 최소값, 최대값
- step : 허용된 범위 내의 숫자 간격
- type이 date, datetime, datetime-local, month, week, time, range, number 일 경우에만 사용

```
<label class="reg" for="group">단체주문</label>  
<input type="number" id="group" value="10" min="10"  
max="100" step="10">
```

# <input> 태그의 다양한 속성

```
1 <!doctype html>
2 <html lang="ko">
3   <head>
4     <title> 웹 폼</title>
5     <meta charset="utf-8">
6     <style>
7       body {
8         background-color: #ffff;
9       }
10      form fieldset{
11        margin-bottom: 25px;
12      }
13      form legend{
14        font-size: 15px;
15        font-weight: bold;
16      }
17      form label.reg {
18        font-size: 14px;
19        width: 110px;
20        color: #390;
21        font-weight: bold;
22        float: left;
23        text-align: right;
24        margin-right: 10px;
25      }
26      form ul li{
27        list-style: none;
28        margin: 15px 0;
29        font-size: 14px;
30      }
31      #member, #stuffs {
32        width: 50px;
33      }
34    </style>
35  </head>
```

```
37 <body>
38   <h1> 여름방학 특강 신청</h1>
39   <form>
40     <fieldset id="subject">
41       <legend>수강과목</legend>
42       <ul>
43         <li>
44           <label class="reg" for="subj">영어회화(초급)</label>
45           <input type="text" id="subj" value="오전 9:00~11:00" readonly>
46         </li>
47       </ul>
48     </fieldset>
49     <fieldset id="register">
50       <legend>신청자</legend>
51       <ul>
52         <li>
53           <label class="reg" for="uname">이름</label>
54           <input type="text" id="uname" autofocus required>
55         </li>
56         <li>
57           <label class="reg" for="uid">학번</label>
58           <input type="text" id="uid" placeholder="하이픈없이 입력" maxlength="8" required>
59         </li>
60         <li>
61           <label class="reg" for="uclass">학과</label>
62           <select id="uclass">
63             <option value="archi">건축공학과</option>
64             <option value="mechanic">기계공학과</option>
65             <option value="indust">산업공학과</option>
66             <option value="elec">전기전자공학과</option>
67             <option value="computer" selected>컴퓨터공학과</option>
68             <option value="chemical">화학공학과</option>
69           </select>
70         </li>
71       </ul>
72     </fieldset>
```

## 여름방학 특강 신청

수강과목

영어회화(초급) 오전 9:00~11:00

신청자

이름

학번

학과

교재 주문

교재

워크시트

당재주문

# <input> 태그의 다양한 속성

```
73 <fieldset>
74   <legend>교재 주문</legend>
75   <ul>
76     <li>
77       <label class="reg" for="book">교재</label>
78       <input type="number" id="book" value="1" min="1" max="3">
79     </li>
80     <li>
81       <label class="reg" for="wsheet">워크시트</label>
82       <input type="number" id="wsheet" value="1" min="1" max="3">
83     </li>
84     <li>
85       <label class="reg" for="group">단체주문</label>
86       <input type="number" id="group" value="10" min="10" max="100" step="10">
87     </li>
88   </ul>
89 </fieldset>
90 <fieldset>
91   <button type="submit" value="submit">신청하기</button>
92   <button type="reset" value="reset">다시쓰기</button>
93 </fieldset>
94
95 </form>
96 </body>
97 </html>
```



여름방학 특강 신청

수강과목

영어회화(초급) 오전 9:00~11:00

신청자

이름

학번

학과

교재 주문

교재

워크시트

단체주문

신청하기 다시쓰기

# <input> 태그의 다양한 속성

## size, minlength, maxlength

- size : 텍스트 관련 필드에서 화면에 몇 글자까지 보이게 할지 결정
- maxlength : 입력 가능한 최대 글자
- minlength : 입력해야 할 최소 글자.  
(크롬과 안드로이드 브라우저에서만 지원)

<label class="reg" for="uid">학번</label>

<input type="text" id="uid" placeholder="하이픈없이 입력" maxlength="8" required>

```
join.html
04 > < join.html > html > body > form > ul > li > label
1 <!doctype html>
2
3 <html lang="ko">
4 <head>
5   <meta charset="utf-8">
6   <title> 웹 폼</title>
7   <style>
8     ul {
9       list-style:none;
10    }
11    li {
12      margin:10px;
13    }
14  </style>
15 </head>
16 <body>
17   <h1>회원 가입을 환영합니다</h1>
18   <form>
19     <ul>
20       <li><label>아이디: <input type="text" id="user_id" size="10" minlength="4" maxlength="15">
21       </label><small style="color: red;"> 4~15자리 이내의 영문과 숫자</small></li>
22       <li><label>이메일: <input type="email" id="user_email"></label></li>
23       <li><label>비밀번호: <input type="password" id="user_pwd"></label></li>
24       <li><label>비밀번호 확인: <input type="password" id="check_pwd"></label></li>
25       <li><input type="submit" value="회원가입"></li>
26     </ul>
27   </form>
28 </body>
29 </html>
```



# 여러 데이터 나열해 보여주기

## <select>, <optgroup>, <option>

- 여러 옵션 중에서 선택 - 드롭다운 목록
- 공간을 최소한으로 사용하면서 여러 옵션 표시 가능

```
기본형 <select 속성="속성 값">
  <option value="값" [속성="속성 값"]> 내용1 </option>
  <option value="값" [속성="속성 값"]> 내용2 </option>
  <option value="값" [속성="속성 값"]> 내용3 </option>
  ...
</select>
```

```
<select id="class">
  <option value="archi">건축공학과</option>
  <option value="mechanic">기계공학과</option>
  <option value="indust">산업공학과</option>
  <option value="elec">전기전자공학과</option>
  <option value="computer" selected>컴퓨터공학과</option>
  <option value="chemical">화학공학과</option>
</select>
```



## <select> 태그의 속성

속성	설명
size	화면에 표시될 드롭다운 메뉴의 항목 개수를 지정합니다. ▶ 크롬 브라우저의 경우, size에서 지정한 개수보다 하나 더 많은 옵션이 표시됩니다. size="3" 으로 지정하면 4개의 옵션이 표시됩니다.
multiple	브라우저 화면에 여러 개의 옵션이 함께 표시되면서 (Ctrl) 키를 누른 상태로 드롭다운 메뉴에 있는 여러 항목을 선택할 수 있습니다.

## <option> 태그의 속성

속성	설명
value	옵션을 선택했을 때 서버로 넘겨질 값을 지정합니다.
selected	화면에 표시될 때 기본으로 선택되어 있는 옵션을 지정합니다.

# 여러 데이터 나열해 보여주기

< select1.html >

04 > select1.html > html > head > style

```
1 <!doctype html>
2 <html lang="ko">
3 <head>
4 <meta charset="utf-8">
5 <title> 웹 폼</title>
6 <style>
7 body {
8     background-color: #fff;
9 }
10 form fieldset{
11     margin-bottom: 25px;
12 }
13 form legend{
14     font-size: 15px;
15     font-weight: 600;
16 }
17 .reg {
18     font-size: 14px;
19     width: 110px;
20     color: #390;
21     font-weight: bold;
22     float: left;
23     text-align: right;
24     margin-right: 10px;
25 }
26 form ul li{
27     list-style: none;
28     margin: 15px 0;
29     font-size: 14px;
30 }
31 </style>
32 </head>
```

33

<body>

<h1> 여름방학 특강 신청</h1>

34

<form>

35

<fieldset>

36

<legend>수강 신청인</legend>

37

<ul>

38

<li>

39

<label class="reg" for="id">학번</label>

40

<input type="text" id="id" autofocus>

41

</li>

42

<li>

43

<label class="reg" for="name">이름</label>

44

<input type="text" id="name">

45

</li>

46

<li>

47

<label class="reg" for="class">학과</label>

48

<select size="5" id="class" multiple>

49

<option value="archi">건축공학과</option>

50

<option value="mechanic">기계공학과</option>

51

<option value="indust">산업공학과</option>

52

<option value="elec">전기전자공학과</option>

53

<option value="computer" selected>컴퓨터공학과</option>

54

<option value="chemical">화학공학과</option>

55

</select>

56

</li>

57

</ul>

58

</fieldset>

59

</form>

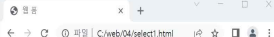
60

</body>

61

</html>

62



## 여름방학 특강 신청

수강 신청인

학번

이름

학과  
건축공학과  
기계공학과  
산업공학과  
전기전자공학과  
컴퓨터공학과

# 여러 데이터 나열해 보여주기

## <select>, <optgroup>, <option>

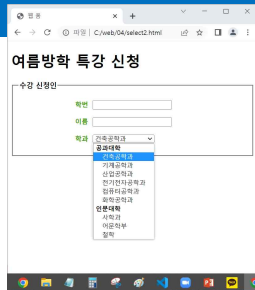
### <optgroup> 태그

- 여러 항목을 그룹을 묶을 때 사용
- label 속성을 사용해 그룹 제목을 붙임

34 > <> select2.html > html > head > style > form

```
1 <!doctype html>
2 <html lang="ko">
3 <head>
4 <meta charset="utf-8">
5 <title> 웹 폼</title>
6 <style>
7     body {
8         background-color: #fff;
9     }
10     form fieldset{
11         margin-bottom: 25px;
12     }
13     form legend{
14         font-size: 15px;
15         font-weight: 600;
16     }
17     .reg {
18         font-size: 14px;
19         width: 110px;
20         color: #390;
21         font-weight: bold;
22         float: left;
23         text-align: right;
24         margin-right: 10px;
25     }
26     form ul li{
27         list-style: none;
28         margin: 15px 0;
29         font-size: 14px;
30     }
31 </style>
32 </head>
```

```
33 <body>
34 <h1> 여름방학 특강 신청</h1>
35 <form>
36 <fieldset>
37 <legend>수강 신청인</legend>
38 <ul>
39 <li>
40 <label class="reg" for="id">학번</label>
41 <input type="text" id="id" autofocus>
42 </li>
43 <li>
44 <label class="reg" for="name">이름</label>
45 <input type="text" id="name">
46 </li>
47 <li>
48 <label class="reg" for="class">학과</label>
49 <select id="class">
50 <optgroup label="공과대학">
51 <option value="archi">건축공학과</option>
52 <option value="mechanic">기계공학과</option>
53 <option value="indust">산업공학과</option>
54 <option value="elec" selected>전기전자공학과</option>
55 <option value="computer">컴퓨터공학과</option>
56 <option value="chemical">화학공학과</option>
57 </optgroup>
58 <optgroup label="인문대학">
59 <option value="history">사학과</option>
60 <option value="lang">어문학부</option>
61 <option value="philo">철학</option>
62 </optgroup>
63 </select>
64 </li>
65 </ul>
66 </fieldset>
67 </form>
68 </body>
69 </html>
```



# 여러 데이터 나열해 보여주기

## <textarea>

- 텍스트 영역 – 여러 줄의 텍스트 입력
- 게시판의 게시물 입력 창, 회원 가입 양식의 약관 등

기본형 <textarea [속성="속성 값"]> 내용 </textarea>

## <textarea> 태그의 속성

속성	설명
name	다른 폼 요소와 구분하기 위해 텍스트 영역의 이름을 지정합니다.
cols	텍스트 영역의 가로 너비를 문자 단위로 지정합니다.
rows	텍스트 영역의 세로 길이를 줄 단위로 지정합니다. 지정한 숫자보다 줄 개수가 많아지면 스크롤 막대가 생깁니다.



# 여러 데이터 나열해 보여주기

```
1 <!doctype html>
2 <html lang="ko">
3 <head>
4   <meta charset="utf-8">
5   <title> 웹 폼</title>
6   <style>
7     form fieldset {
8       margin: 10px 0;
9     }
10    form legend {
11      font-size: 18px;
12      color: #0066FF;
13      font-weight: 600;
14    }
15    form label.reg {
16      font-size: 14px;
17      width: 120px;
18      float: left;
19    }
20    form label em {
21      font-size: 15px;
22      color: red;
23      font-weight: 800;
24    }
25    form ul li {
26      list-style: none;
27      margin: 10px 0;
28    }
29    form .easys {
30      text-align: left;
31      font-size: 12px;
32      color: blue;
33    }
34    form fieldset.sendform {
35      text-align: center;
36    }
37  </style>
38 </head>
39 <body>
40   <form>
41     <fieldset class="login">
42       <legend>로그인</legend>
43       <label for="user_id">아이디 </label>
44       <input type="text" id="user_id" size="18" autofocus>
45       <label for="user_pw">비밀번호 </label>
46       <input type="password" id="user_pw" size="18">
47       <input type="submit" value="로그인">
48     </fieldset>
49     <fieldset class="register">
50       <legend>가입하기</legend>
51       <ul>
52         <li>
53           <label class="reg" for="user_name">아이디 <em>*</em></label>
54           <input type="text" id="new_id" size="20" autocomplete="on" required>
55         </li>
56         <li>
57           <label class="reg" for="new_pw1">비밀번호 <em>*</em></label>
58           <input type="password" id="new_pw1" size="20" required>
59         </li>
60         <li>
61           <label class="reg" for="new_pw2">비밀번호 확인 <em>*</em></label>
62           <input type="password" id="new_pw2" size="20" required>
63         </li>
64         <li>
65           <label class="reg" for="user_name">이름 <em>*</em></label>
66           <input type="text" id="user_name" size="20" required>
67         </li>
68         <li>
69           <label class="reg" for="user_mail">메일 주소 <em>*</em></label>
70           <input type="email" id="user_mail" size="20" required>
71         </li>
72         <li>
73           <label class="reg" for="user_tel">전화번호 </label>
74           <input type="tel" id="user_tel" size="20">
75         </li>
76       </ul>
77     </fieldset>
```

```
78   <fieldset class="easys">
79     <legend>이시스퍼블리싱</legend>
80     <textarea name="intro" cols="60" rows="5">
81     열심히 사는 사람들의 손을 잡아주는 곳 - 이시스 퍼블리싱
82
83     우리는 책을 내기 전에 다시 한번 물어봅니다
84     "이 책이 사람들에게 도움이 되는가?"
85
86     더 쉽게, 더 빠르게 지식을 전달하고 싶습니다.
87     이시스퍼블리싱의 책과 앱을 만나보세요.
88   </fieldset>
89   <fieldset class="sendform">
90     <input type="submit" value="가입하기">
91     <input type="reset" value="다시쓰기">
92   </fieldset>
93 </form>
94 </body>
95 </html>
```



# 기타 다양한 폼 요소들

## <output>

- 계산 결과를 브라우저에 표시
- 브라우저 화면에 다르게 표시되는 것은 아니지만  
<output> 태그로 묶인 부분이 일반 텍스트가 아니라 계산의 결과값이라는 점을 웹 브라우저가 정확히 인식할 수 있습니다.

```
<form
oninput="result.value=parseInt(num1.value)+parseInt(num2.
value)">
  <input type="number" name="num1" value="0">
  +<input type="number" name="num2" value="0">
  =<output name="result" for="num"></output>
</form>
```



10 + 10 \* = 20

output.html X

04 > output.html > ...

```
1 <!doctype html>
2 <html lang="ko">
3 <head>
4   <meta charset="utf-8">
5   <title>웹 폼</title>
6 </head>
7
8 <body>
9   <form oninput="result.value=parseInt(num1.value)+parseInt(num2.value)">
10     <input type="number" name="num1" value="0">
11     +<input type="number" name="num2" value="0">
12     =<output name="result" for="num"></output>
13   </form>
14 </body>
15 </html>
16
```

# 기타 다양한 폼 요소들

## <meter>

- 전체 크기 중에서 얼마나 차지하는지를 표현할 때 사용  
예) 하드 디스크 사용량, 유권자 투표율 등

### 사용할 수 있는 속성

속성	설명
min, max	범위의 최솟값과 최댓값을 나타냅니다. 값을 정하지 않으면 0과 1로 간주합니다.
value	범위 내에서 차지하는 값을 나타냅니다.
low	"이 정도면 낮다."라고 할 정도의 값을 지정합니다.
high	"이 정도면 높다."라고 할 정도의 값을 지정합니다.
optimum	"이 정도면 적당하다."라고 할 정도의 범위를 지정합니다. optimum값이 high 값보다 크다면 value 값이 클수록 좋고 optimum 값이 low 값보다 작다면 value 값이 작을수록 좋습니다.

meter.html X

04 > meter.html > ...

```
1 <!doctype html>
2 <html lang="ko">
3   <head>
4     <meta charset="utf-8">
5     <title> 웹 폼</title>
6     <style>
7     body {
8       background-color: #fff;
9     }
10    ul li{
11      list-style:none;
12      margin: 15px 0;
13      font-size:14px;
14    }
15  </style>
16 </head>
```

```
17 <body>
18   <ul>
19     <li>
20       <label>점유율 0.8 </label>
21       <!-- 전체 크기 1을 기준으로 0.8만큼 차지합니다 -->
22       <meter value="0.8"></meter>
23     </li>
24     <li>
25       <label>사용량 64%</label>
26       <!-- 전체 100 중에서 64만큼 차지합니다 -->
27       <meter min="0" max="100" value="64"></meter>
28     </li>
29     <li>
30       <label>트래픽 초과</label>
31       <!-- 전체 크기는 1024~10240까지인데 높다고 설정한 8192보다 현재 값 9216이 더 큼니다 -->
32       <meter min="1024" max="10240" low="2048" high="8192" value="9216"></meter>
33     </li>
34     <li>
35       <label>적절한 트래픽</label>
36       <!-- 전체 1 중에서 현재 0.5를 차지하고 있으며 적정도를 0.8로 설정했습니다 -->
37       <meter value="0.5" optimum="0.8"></meter>
38     </li>
39   </ul>
40 </body>
41 </html>
```



점유율 0.8

사용량 64%

트래픽 초과

적절한 트래픽

# <input> 태그 활용

meter.html meter.jsp Meter.java

로그인 정보

상의 판매율:  하의 판매율:  신발 판매율:



# <input> 태그 활용

```
<!doctype html>

<html lang="ko">
  <head>
<meta charset="utf-8">
<title> 웹 폼</title>
    <style>
      label {
font-size:14px; /* 글자 크기 */
line-height:25px; /* 줄 간격 */
      }
      label em{
color:red; /* 글자색 */
font-weight:800; /* 글자 굵기 */
      }
    </style>
  </head>
  <body>
<form method="get" action="Meter" enctype="UTF-8">
  <fieldset>
    <legend>로그인 정보</legend>
    <label>상의 판매율: <input type="text" size="10" name="top" ></label>
    <label>하의 판매율: <input type="text" size="10" name="pants" ></label>
    <label>신발 판매율: <input type="text" size="10" name="shoes" ></label>
    <input type="submit" value="제출">
  </fieldset>
</form>
</body>
</html>
```

 meter.html meter.jsp Meter.java

# <input> 태그 활용

```
@WebServlet("/Meter")
public class Meter extends HttpServlet {
```

```
    /**
     * @see HttpServlet#HttpServlet()
     */
    public Meter() {
        super();
        // TODO Auto-generated constructor stub
    }
```

```
    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        String user_top=request.getParameter("top");
        String user_pants=request.getParameter("pants");
        String user_shoes=request.getParameter("shoes");
        System.out.println(user_top);
        request.setAttribute("top", user_top);
        request.setAttribute("pants",user_pants);
        request.setAttribute("shoes",user_shoes);
        RequestDispatcher dispatch = request.getRequestDispatcher("/test01/meter.jsp");
        dispatch.forward(request, response);
    }
```

```
}
```



# <input> 태그 활용

```
<%
request.setCharacterEncoding("utf-8");
String user_top=request.getParameter("top");
String user_pants=request.getParameter("pants") ;
String user_shoes=request.getParameter("shoes");
```

```
%>
<body>
<form>
<fieldset id="subject">
<legend> 쇼핑물 판매량</legend>
<ul>
<li>
<label>상의 </label>
<!-- 전체 크기 1을 기준으로 0.8만큼 판매 -->
<meter value=<%= user_top %>></meter>
</li>
<li>
<label>하의 </label>
<!-- 전체 100 중에서 64만큼 판매 -->
<meter min="0" max="100" value=<%= user_pants %>></meter>
</li>
<li>
<label>신발</label>
<!-- 전체 600개에서 504개 판매-->
<meter min="0" max="600" low="0" high="500" value=<%= user_shoes %>></meter>
</li>
</ul>
</fieldset>
</form>

</body>
</html>
```

```
<%@ page language="java"
contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"
isELIgnored="false" %>
```

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title> 웹 폼</title>
<style>
@import url('https://fonts.googleapis.com/css2?family=Noto+Serif+KR&display=swap');
body {
background-color:#fff;
}
ul li{
font-family: 'Noto Serif KR', serif;
list-style:none;
margin: 15px 0;
font-size:14px;
}
</style>
</head>
```

