입력 폼 #1

2018-03-07

이승진

**학습 목표**

입력 양식(form)을 웹브라우저에 표시하기 위한 HTML 태그를 JSP에서 출력한다.

입력 양식(form)에 값을 입력하고 웹 서버에 전송한다.

전송된 값을 웹서버의 JSP에서 받아서 처리한다.

**목차**

[1. 텍스트 입력 2](#_Toc508293028)

[1) text1.jsp 2](#_Toc508293029)

[2) text2.jsp 11](#_Toc508293030)

[3) text3a.jsp 19](#_Toc508293031)

[4) text3b.jsp 26](#_Toc508293032)

[5) text3c.jsp 32](#_Toc508293033)

[2. 목록 선택 34](#_Toc508293034)

[1) select1.jsp 34](#_Toc508293035)

[2) select2a.jsp 38](#_Toc508293036)

[3) select2b.jsp 41](#_Toc508293037)

[4) select2c.jsp 48](#_Toc508293038)

[3. 실습 과제 55](#_Toc508293039)

[1) 코드 분석 55](#_Toc508293040)

[2) 구현 실습 57](#_Toc508293041)

[3) 구현 실습 57](#_Toc508293042)

# 텍스트 입력

웹 폼에서 문자열을 입력하고 웹서버에 전달해 보자.

## text1.jsp

### WebContent/form01폴더 생성

지난시간에 생성했던 lecture1 프로젝트에 WebContent/form01폴더를 생성한다.

생성방법:

이클립스의 화면 왼쪽 Project Explorer창에서 WebContent 폴더를 마우스 오른쪽 버튼으로 클릭하고

메뉴 - New - Folder 클릭

Folder name: 칸에 form01를 입력하고 Finish 버튼 클릭

### WebContent/form01/text1.jsp 파일 생성.

생성방법:

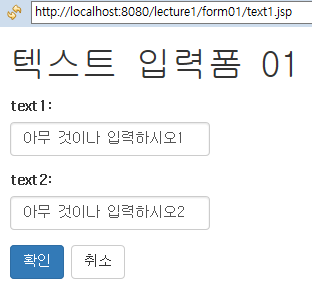
이클립스의 화면 왼쪽 Project Explorer창에서 form01 폴더를 마우스 오른쪽 버튼으로 클릭하고

메뉴 - New - JSP File 클릭

File name: 칸에 text1.jsp를 입력하고 Finish 버튼 클릭

WebContent/form01/text1.jsp

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36 | <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  body { font-family: 굴림체; }  input.form-control { width: 200px; }  </style>  </head>  <body>  <div class="container">  <form action="text1action.jsp" method="post">  <h1>텍스트 입력폼 01</h1>  <div class="form-group">  <label>text1:</label>  <input type="text" name="param1" class="form-control" value="아무 것이나 입력하시오1" />  </div>  <div class="form-group">  <label>text2:</label>  <input type="text" name="param2" class="form-control" value="아무 것이나 입력하시오2" />  </div>  <div class="form-group">  <input type="submit" class="btn btn-primary" name="cmd" value="확인">  <input type="reset" class="btn btn-default" value="취소">  </div>  </form>  </div>  </body>  </html> |



(1) 웹브라우저가 인터넷을 통하여 웹서버에 http://localhost:8080/lecture1/form01/text1.jsp URL의

실행을 요청한다.

(2) 웹서버는 요청 받은 text1.jsp를 실행한다.

text1.jsp 파일 내부에 Java 코드는 없고 html 태그들 뿐이다.  
 text1.jsp의 실행결과로 이 html 태그들이 그대로 출력된다.

(3) 실행 결과 출력인 html 태그들이 인터넷을 통해 웹브라우저에 전송된다.

(4) 웹브라우저는 전송받은 html 태그들을 웹브라우저창에 그린다. 그 결과는 위 그림과 같다.

웹브라우저 창에 위 그림과 같이 입력 폼이 그려져 있는 것은, 이미 웹서버에서 text1.jsp 실행은 완료되었고,

그 실행 결과 출력이 인터넷을 통해 웹브라우저에 전송되었음을 의미한다.

즉 웹서버에서 text1.jsp 실행은 이미 끝난 후에 웹브라우저에 그 결과가 보인다.

(줄17~33) form 태그는 웹브라우저에서 입력폼으로 그려진다.

(줄17) action="text1action.jsp"

form 태그의 action 애트리뷰트(attribute)는 입력폼에 입력된 데이터를 전송할 목적지 URL을 지정한다.

위 입력폼에서 '확인' 버튼을 클릭하면, 웹브라우저는 즉시 text1action.jsp URL의 실행을 웹서버에 요청한다.

입력폼에 입력된 데이터도 이 요청(request)에 같이 담아서 웹서버에 보낸다.

이 데이터를 **request parameter**라고 부른다.

C언어 함수를 호출할 때, 호출되는 함수 이름 뒤에 함수 파라미터도 전달하듯이,

웹서버의 JSP 파일의 실행을 요청할 때, 요청되는 JSP 파일의 URL과 같이 request parameter도 전달된다.

action="text1action.jsp"

action 애트리뷰트값은 URL이다. 그런데 "text1action.jsp" 파일명만 적었으므로 **상대경로 URL** 이다.

상대경로 URL은 현재 웹브라우저의 주소창에 표시된 현재 URL과 조합되어 **절대경로 URL**이 된다.

현재 웹브라우저의 주소창에 표시된 현재 URL이 http://localhost:8080/lecture1/form01/text1.jsp 이므로,

이것과 text1action.jsp 상대경로를 조합하면 http://localhost:8080/lecture1/form01/text1action.jsp 가 된다.

즉 위 입력폼에서 입력을 마치고 [확인] 버튼을 클릭하면,

상대경로 URL로 간단하게 적자면 text1action.jsp 이고,

생략하지 않고 절대경로 URL로 다 적으면 http://localhost:8080/lecture1/form01/text1action.jsp 인 URL을

웹서버에 요청하게 된다. 폼에 입력된 데이터도 이 요청에 같이 담겨져 웹서버에 전송된다.

요청된 http://localhost:8080/lecture1/form01/text1action.jsp가 웹서버에서 실행될 때, 전송받은 데이터(request parameter)를 꺼내서 처리해야 한다.

(줄17) method="post"

form 태그 내부의 submit 버튼이 클릭되어 웹브라우저가 웹서버에 URL을 요청할 때,

요청 방식은 method="post" 방식과 method="get" 방식이 있다.

"POST", "GET"에서 대소문자는 중요하지 않다.

form 태그 내부의 버튼이 클릭되어 서버를 요청하는 경우가 아닌, 다른 형태의 요청은 전부 GET 방식이다.

즉 웹브라우저 주소창에 URL을 직접 입력하여 요청할 때,

웹페이지의 링크를 클릭하여 요청할 때 등등은 **GET 방식 요청**이다.

form 태그 내부의 버튼이 클릭되어 요청하는 경우에 요청방식은

form 태그의 method 애트리뷰트값으로 결정된다.

method 애트리뷰트값을 생략하면 디폴트 요청 방식은 GET 방식 요청이다.

즉 method="POST" 인 form 태그 내부의 버튼이 클릭된 경우에만 **POST 방식 요청**이고,

그 외의 경우는 전부 **GET 방식 요청**이다.

웹브라우저가 웹서버에 URL을 요청할 때, 그 요청에 request parameter도 같이 담아 보내기도 한다.

GET 방식 요청은, request parameter를 query string 형태로 URL에 붙여서 전송한다.

데이터의 사이즈가 크다면, URL에 데이터를 붙여 보내는 GET 방식보다 POST 방식이 안전하다.

입력 폼은 대부분 POST 방식을 사용한다.

(줄22) <intput type="text" ... />

문자열을 입력하기 위한 텍스트 박스 태그이다.

submit 버튼이 클릭되어 웹서버에 요청이 전송될 때,

이 텍스트 박스에 입력된 문자열도 그 요청에 같이 담겨져 웹서버에 전송된다. (request parameter)

이 요청에 의해서 웹서버에서 실행되는 text1action.jsp는, 요청에 담겨 전송된 request parameter를 꺼내서 처리해야 한다. request parameter 각각을 구별해서 꺼낼 수 있으려면, 이름으로 구별할 수 있어야 한다.

(줄22) name="param1"

입력폼의 입력 태그들에 이름을 부여한다.

입력된 데이터가 요청에 담겨져 웹서버에 전송될 때 이 이름이 request parameter의 이름이 된다.

나중에 웹서버에서 이 이름으로 request parameter를 꺼내게 된다.

(줄22) value="아무 것이나 입력하시오1"

텍스트 박스가 웹브라우저에 그려질 때, 이 value 값이 미리 채워진 상태로 그려진다.



value 값이 주어지지 않으면 비어있는 상태로 그려진다.

(줄30) <input type="submit" ... />

입력폼에 데이터를 입력하고 마지막으로 웹서버에 전송(submit)하기 위해 클릭하는 버튼 태그이다.

이 버튼을 클릭하면, 이 버튼 태그를 둘러싸고 있는 (줄17~33) form 태그의

action 값인 text1action.jsp URL을 웹서버에 요청하게 된다. 이 때 폼에 입력된 데이터도 요청에 같이 담겨서 전송된다. (request parameter)

(줄30) class="btn btn-primary"

bootstrap에서 정의된 CSS클래스를 태그에 부여함.

<a> 태그, <button> 태그, <input type="submit"> 태그, <input type="reset"> 태그에

btn CSS클래스를 부여하면 버튼 형태로 그려진다.

btn CSS클래스와 동시에 btn-primary CSS클래스도 부여하면 파란색 버튼 형태로 그려진다.



(줄30) name="cmd" value="확인"

<input type="submit" > 태그의 value값은 버튼에 표시될 문자열이다.

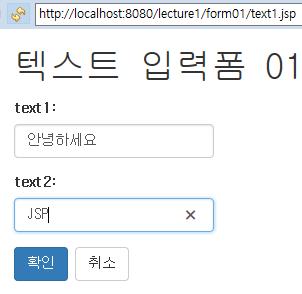
submit 버튼을 클릭하면, 입력 폼에 입력된 데이터가 웹서버에 전송된다. 이 때 클릭된 submit 버튼의 name과 value 값도 역시 서버에 전송된다. 클릭되지 않은 버튼의 name과 value 값은 전송되지 않는다.

(줄31) <input type="reset" ... />

이 버튼을 클릭하면 입력폼에 입력된 값들이 전부 초기화(reset) 된다.

즉 웹페이지가 처음 그려졌을 때의 상태로 돌아간다.

웹 서버에 무엇이 요청되지는 않고, 웹브라우저에서만 초기화 된다.



위와 같이 입력하고 확인 버튼을 클릭하자.

### WebContent/form01/text1action.jsp

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48 | <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <div class="container">  <% request.setCharacterEncoding("utf-8"); %>  <h1>request parameter</h1>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>param1</td>  <td><%= request.getParameter("param1") %></td>  </tr>  <tr>  <td>param2</td>  <td><%= request.getParameter("param2") %></td>  </tr>  <tr>  <td>cmd</td>  <td><%= request.getParameter("cmd") %></td>  </tr>  <tr>  <td>method</td>  <td><%= request.getMethod() %></td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

### request parameter

text1.jsp 웹페이지의 입력 폼에 데이터를 입력하고 submit 버튼을 클릭하면,

text1action.jsp URL 요청(request)이 웹브라우저에서 웹서버에 전송된다

입력 폼에 입력된 데이터도 이 요청에 같이 담겨서 전송된다. 이 데이터를 request parameter라고 부른다.

웹브라우저가 웹서버에 URL을 요청하여 JSP 파일이 실행될 때, 그 요청과 관련된 모든 것이 request 객체에 담겨져 전달된다. request 객체에서 request parameter를 꺼내야 한다.

(줄17) request.setCharacterEncoding("utf-8");

request parameter에 한글 데이터가 포함되어 있다면, 먼저 문자 인코딩부터 맞춰야 한다.

줄17은, 요청(request)에 들어있는 request parameter를 꺼내기 전에 문자 인코딩(character encoding)을 utf-8로 맞추기 위한 메소드 호출이다.

(줄30, 34, 38) request.getParameter("데이터\_이름")

요청에 담긴 request parameter를 꺼내는 메소드 호출이다.

이 메소드의 파라미터는 request parameter의 이름(name)이다.

text1.jsp 파일의 줄22의 텍스트 박스에 입력되어 웹서버에 전송된 문자열이 text1action.jsp 줄30의 메소드 호출로 꺼내진다.

text1.jsp 파일의 줄27의 텍스트 박스에 입력되어 웹서버에 전송된 문자열이 text1action.jsp 줄34의 메소드 호출로 꺼내진다.

text1.jsp 파일의 줄30의 버튼이 클릭되어 요청이 서버에 전송된 것이기 때문에, 이 버튼의 value인 "확인" 문자열이 text1action.jsp 줄38의 메소드 호출로 꺼내진다.

(줄30, 34, 38) <%= request.getParameter("데이터\_이름") %>

request parameter 값이 출력된다.

(줄42) request.getMethod()

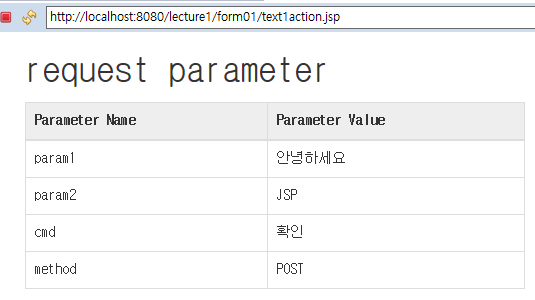
웹브라우저가 웹서버에 URL을 요청할 때 요청 방식은 method="post" 방식과, method="get" 방식이 있다.

request.getMethod() 메소드 호출은 그 요청 방식을 문자열로 리턴한다. 따라서 "GET" 아니면 "POST" 문자열이 리턴되는데, 이 경우에는 POST 방식으로 전송했기 때문에 "POST"가 리턴된다.

웹서버에 요청된 text1action.jsp 파일이 웹서버에서 실행되고,

실행결과 출력된 HTML 태그들이 인터넷을 통해 다시 웹브라우저에게 전송되고,

웹브라우저 창에 아래와 같이 그려진다.



위와 같이 text1action.jsp의 실행결과 출력이 웹브라우저에 그려진 것은 이미 웹서버에서 text1action.jsp의 실행은 완료되었고, 그 실행 결과 출력이 인터넷을 통해 웹브라우저에 전송되었음을 의미한다.

### 실행 과정 설명

(1) 웹브라우저는 text1.jsp URL을 웹서버에 요청한다.

(2) 웹서버는 요청받은 text1.jsp를 실행한다.

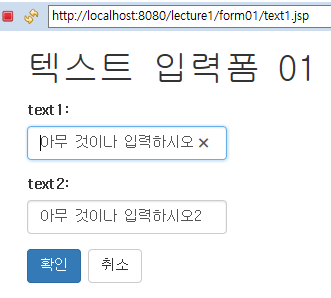
(3) text1.jsp의 실행이 종료된다.

(4) 웹서버는 text1.jsp의 실행결과 출력을 웹브라우저에 전달한다.

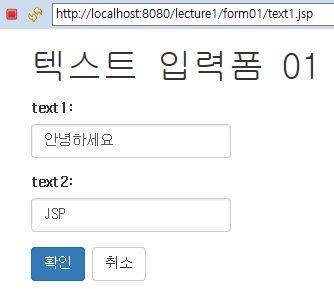
웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  body { font-family: 굴림체; }  input.form-control { width: 200px; }  </style>  </head>  <body>  <div class="container">  <form action="text1action.jsp" method="post">  <h1>텍스트 입력폼 01</h1>  <div class="form-group">  <label>text1:</label>  <input type="text" name="param1" class="form-control" value="아무 것이나 입력하시오1" />  </div>  <div class="form-group">  <label>text2:</label>  <input type="text" name="param2" class="form-control" value="아무 것이나 입력하시오2" />  </div>  <div class="form-group">  <input type="submit" class="btn btn-primary" name="cmd" value="확인">  <input type="reset" class="btn btn-default" value="취소">  </div>  </form>  </div>  </body>  </html> |

(5) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.



(6) 사용자는 웹브라우저 창의 텍스트 박스에 다음과 같이 입력하고 확인 버튼을 클릭한다.



(7) 웹브라우저는 text1action.jsp URL을 웹서버에 요청한다. 그리고 입력된 데이터도 요청에 담아 같이 보낸다.

요청에 담겨 보내지는 request parameter는 다음과 같다.

|  |  |
| --- | --- |
| name | value |
| param1 | 안녕하세요 |
| param2 | JSP |
| cmd | 확인 |

(8) 웹서버는 요청된 text1action.jsp를 실행한다.

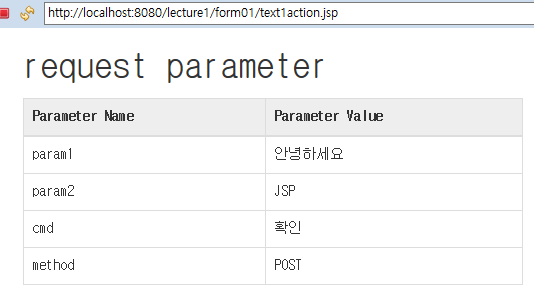
(9) text1action.jsp의 실행이 종료된다.

(10) 웹서버는 text1action.jsp의 실행결과 출력을 웹브라우저에 전달한다.

웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  body { font-family: 굴림체; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <div class="container">    <h1>request parameter</h1>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>param1</td>  <td>안녕하세요</td>  </tr>  <tr>  <td>param2</td>  <td>JSP</td>  </tr>  <tr>  <td>cmd</td>  <td>확인</td>  </tr>  <tr>  <td>method</td>  <td>POST</td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

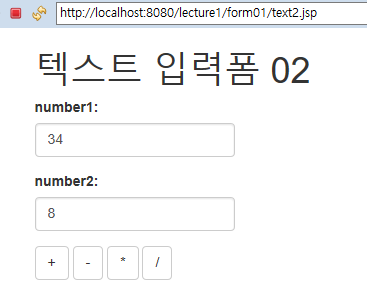
(11) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.



## text2.jsp

### WebContent/form01/text2.jsp

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38 | <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  </style>  </head>  <body>  <div class="container">  <form action="text2action.jsp" method="get">  <h1>텍스트 입력폼 02</h1>  <div class="form-group">  <label>number1:</label>  <input type="text" name="number1" class="form-control" value="34" />  </div>  <div class="form-group">  <label>number2:</label>  <input type="text" name="number2" class="form-control" value="8" />  </div>  <div class="form-group">  <input type="submit" class="btn btn-default" name="cmd" value="+">  <input type="submit" class="btn btn-default" name="cmd" value="-">  <input type="submit" class="btn btn-default" name="cmd" value="\*">  <input type="submit" class="btn btn-default" name="cmd" value="/">  </div>  </form>  </div>  </body>  </html> |



(줄16~35) 이 입력폼(form)에 데이터를 입력하고 submit 버튼을 클릭하면, text2action.jsp URL 실행 요청이 웹서버에 전달된다.

(줄21, 26) 데이터를 입력하기 위한 텍스트 박스 태그이다.

위 그림에서 볼 수 있듯이, 텍스트 박스가 웹브러우져 창에 처음 그려질 때,

value 애트리뷰트값 34", "8"이 미리 채워진 상태로 그려진다.

(줄30~33) submit 버튼은 4개이다.

4개의 submit 버튼 중 아무거나 클릭하면 text2action.jsp의 실행 요청이 웹서버에 전달된다.

이때 (줄21,26)의 텍스트 박스에 입력한 데이터도 요청에 담겨 서버에 같이 전송된다. (request parameter)

클릭된 submit 버튼의 name과 value 애트리뷰트값도 요청에 담겨 서버에 같이 전송된다.

(줄16) method="get"

요청은 GET 방식이다.

GET 방식에서는 데이터를 query string 형태로 URL에 붙여서 보낸다.

예: http://localhost:8080/lecture1/form01/text2action.jsp?number1=34&number2=8&cmd=\*

### WebContent/form01/text2action.jsp

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64 | <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  body { font-family: 굴림체; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <%  String s1 = request.getParameter("number1");  int number1 = Integer.parseInt(s1);  String s2 = request.getParameter("number2");  int number2 = Integer.parseInt(s2);  String cmd = request.getParameter("cmd");  int result = 0;  if (cmd.equals("+")) result = number1 + number2;  else if (cmd.equals("-")) result = number1 - number2;  else if (cmd.equals("\*")) result = number1 \* number2;  else if (cmd.equals("/")) result = number1 / number2;  %>  <div class="container">  <h1>request parameter</h1>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>number1</td>  <td><%= number1 %></td>  </tr>  <tr>  <td>number2</td>  <td><%= number2 %></td>  </tr>  <tr>  <td>cmd</td>  <td><%= cmd %></td>  </tr>  <tr>  <td>method</td>  <td><%= request.getMethod() %></td>  </tr>  <tr>  <td>계산결과</td>  <td><%= result %></td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

(줄19) String s1 = request.getParameter("number1");

요청(request)에 담겨져 같이 전송된 데이터를 꺼내는 request.getParameter(...) 메소드 호출이다.

꺼낼 데이터의 이름은 "number1"이다.

이 메소드의 리턴 타입은 String 이다.

전송된 데이터가 숫자나 날짜이더라도, 이 메소드는 그 데이터를 String 타입으로 리턴한다.

(줄20) String 타입의 데이터를 int 타입으로 변환하기 위한 Integer.parseInt(...) 메소드 호출이다.

이것은 Integer 클래스의 정적(static) 메소드이다.

public static int parseInt(String s);

Java 문법에서 정적(static) 메소드가 무엇인지, 인스턴스(instance) 메소드와 호출 방법이 어떻게 다른지 확인하라. 정적 메소드와 인스턴스 메소드의 구현 방법에 어떤 차이가 있는지 꼭 확인하라.

(줄24) cmd는 String 타입의 변수이다.

우리는 보통 간단하게 줄여서 말할 때 변수에 객체를 저장한다고 설명하지만,

정확하게 말하자면 Java 언어의 변수에 객체를 저장하는 것은 불가능하다.

변수에 저장되는 것은 객체가 아니라, 그 객체에 대한 참조(reference)이다.

Java 언어에서 객체에 대한 참조란 그 객체를 가르키는 주소 같은 것이다.

(줄26) Java에서 객체를 비교할 때는 equals 메소드를 사용해야 한다.

객체를 비교할 때 == 연산자로 비교하면, 두 객체의 내부 값이 같은지 비교하는 것이 아니고 참조가 같은지 비교한다. 참조가 같다는 말은 동일한 하나의 객체라는 말이다.

equals 메소드는 참조를 비교하는 것이 아니고, 참조가 가르키는 객체의 내부 값들을 비교한다. 즉 객체 내부 멤버 변수값들이 전부 같은지 비교한다.

두 객체의 참조가 같다는 말은, 사실 두 객체가 아니라 동일한 하나의 객체라는 말이다.

두 객체를 equals 메소드로 비교했더니 true라는 말은, 두 객체의 내부 데이터가 완전히 동일하다는 말이다.

(줄29) result = number1 / number2

number1, number2 둘 다 int 형 변수이다.

int 나누기 int의 계산 결과도 정수가 된다. 이때 계산 결과 소수점 아래 자리는 그냥 전부 버린다.

예를들어 9 / 10의 계산 결과는 0.9가 아니라 0이다.

### 실행 과정 설명

(1) 웹브라우저는 text2.jsp URL을 웹서버에 요청한다.

(2) 웹서버는 요청받은 text2.jsp를 실행한다.

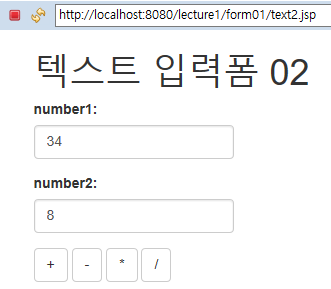
(3) text2.jsp의 실행이 종료된다.

(4) 웹서버는 text2.jsp의 실행결과 출력을 웹브라우저에 전달한다.

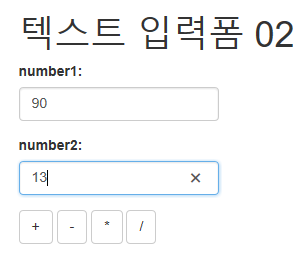
웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  </style>  </head>  <body>  <div class="container">  <form action="text2action.jsp" method="get">  <h1>텍스트 입력폼 02</h1>  <div class="form-group">  <label>number1:</label>  <input type="text" name="number1" class="form-control" value="34" />  </div>  <div class="form-group">  <label>number2:</label>  <input type="text" name="number2" class="form-control" value="8" />  </div>  <div class="form-group">  <input type="submit" class="btn btn-default" name="cmd" value="+">  <input type="submit" class="btn btn-default" name="cmd" value="-">  <input type="submit" class="btn btn-default" name="cmd" value="\*">  <input type="submit" class="btn btn-default" name="cmd" value="/">  </div>  </form>  </div>  </body>  </html> |

(5) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.



(6) 사용자는 웹브라우저 창의 텍스트 박스에 다음과 같이 입력한다.



(7) 위 그림에서 마지막 버튼인 "/" 버튼을 클릭하면 다음과 같은 요청이 웹서버에 전송된다.

|  |  |
| --- | --- |
| 요청되는 URL | text2action.jsp |
| 요청 방식 | GET |

요청에 담겨 보내지는 데이터는 다음과 같다.

|  |  |
| --- | --- |
| name | value |
| number1 | 90 |
| number2 | 13 |
| cmd | / |

"text2action.jsp"는 상대경로 URL 이다.

상대경로 URL은 현재 웹브라우저의 주소창에 표시된 현재 URL과 조합되어 절대경로 URL이 된다.

현재 웹브라우저의 주소창에 표시된 현재 URL이 http://localhost:8080/lecture1/form01/text2.jsp 이므로,

이것과 text2action.jsp 상대경로를 조합하면 http://localhost:8080/lecture1/form01/text2action.jsp 가 된다.

### url encoding

요청은 GET 방식이다. GET 방식에서는 데이터를 query string 형태로 URL에 붙여서 보낸다.

요청에 담겨 보내지는 데이터를 query string 형태로 URL에 추가하면 다음과 같다.

http://localhost:8080/lecture1/form01/text2action.jsp**?number1=90&number2=13&cmd=%2F**

위 query string을 살펴보면, 전송될 데이터의 이름(name)과 값(value)이 포함되어 있음을 알 수 있다.

number1 = 90

number2 = 13

cmd = %2F

query string의 이름과 값에 사용할 수 있는 문자는, 알파벳과 숫자와 몇몇 기호 문자로 제한되어 있다.

클릭된 submit 버튼의 name="cmd"와 value="/"도 query string에 포함되어 웹서버에 전송되어야 하는데,

/ 문자는 query string에 포함될 수 없다.

query string에 포함될 수 없는 문자는, 웹브라우저에 의해 그 문자 코드의 16진수 형태로 변환된다.

이 변환을 url encoding이라고 부른다.

즉 / 문자의 코드는 16진수로 2F 이므로, / 문자를 url encoding하면 %2F 이다.

url encoding은 웹브라우저에 의해 자동으로 실행된다.

정리하자면, 위 그림에서 마지막 버튼인 "/" 버튼을 클릭하면 다음과 URL 요청이 웹서버에 전송된다.

http://localhost:8080/lecture1/form01/text2action.jsp?number1=90&number2=13&cmd=%2F

### url decoding

url encoding의 반대 작업을 url decoding이라고 한다.

웹브라우저에서 자동으로 url encoding되어 전송된 데이터는 웹서버에서 자동으로 url decoding 된다.

즉 request.getParameter(...) 메소드를 호출하여 꺼낼 때, url encoding 되기 전의 원래 값으로 꺼내진다.

입력 폼이 전송될 때, url encoding과 url decoding은 이렇게 웹브라우저와 웹서버에 의해 자동으로 처리된다.

(8) 웹서버는 요청된 text2action.jsp를 실행한다.

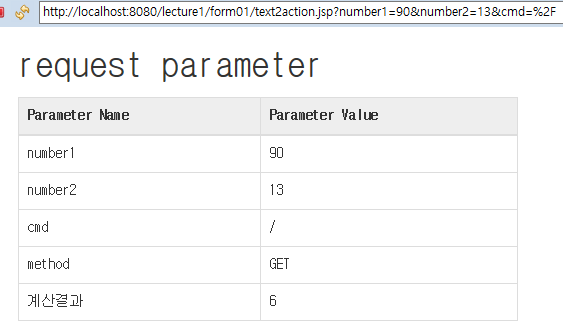
(9) text2action.jsp의 실행이 종료된다.

(10) 웹서버는 text2action.jsp의 실행결과 출력을 웹브라우저에 전달한다.

웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  body { font-family: 굴림체; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <div class="container">  <h1>request parameter</h1>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>number1</td>  <td>90</td>  </tr>  <tr>  <td>number2</td>  <td>13</td>  </tr>  <tr>  <td>cmd</td>  <td>/</td>  </tr>  <tr>  <td>method</td>  <td>GET</td>  </tr>  <tr>  <td>계산결과</td>  <td>6</td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

(11) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.



위 그림에서 주소창의 URL을 확인하자.

http://localhost:8080/lecture1/form01/text2action.jsp?number1=90&number2=13&cmd=%2F

현재 주소창에 표시된 URL을 웹브라우저가 웹서버에 요청하였고, 이 URL 요청 결과로 웹서버로부터 전송받은 HTML 태그들이 현재 창에 출력되었다는 것을 의미한다.

## text3a.jsp

### WebContent/form01/text3a.jsp

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64 | <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  <%@ page import="java.util.Date, java.text.SimpleDateFormat" %>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <% request.setCharacterEncoding("utf-8"); %>  <div class="container">  <form action="text3a.jsp">  <h3>텍스트 입력폼 03</h3>  <div class="form-group">  <label>text1:</label>  <input type="text" class="form-control" name="text1" value="hello" />  </div>  <div class="form-group">  <label>text2:</label>  <input type="text" class="form-control" name="text2" value="world" />  </div>  <div class="form-group">  <label>time:</label>  <input type="text" class="form-control" name="time"  value='<%= new SimpleDateFormat("HH:mm:ss").format(new Date()) %>' />  </div>  <div class="form-group">  <input type="submit" class="btn btn-primary" name="cmd" value="확인">  </div>  </form>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>text1</td>  <td><%= request.getParameter("text1") %></td>  </tr>  <tr>  <td>text2</td>  <td><%= request.getParameter("text2") %></td>  </tr>  <tr>  <td>time</td>  <td><%= request.getParameter("time") %></td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

(줄20) action="text3a.jsp"

text3a.jsp의 입력폼에 데이터를 입력한 후 submit 버튼을 누르면, 다시 text3a.jsp가 요청된다.

즉 text3a.jsp에서 다시 text3a.jsp를 요청하게 된다.

(줄33) new SimpleDateFormat("HH:mm:ss").format(new Date())

먼저 new SimpleDateFormat("HH:mm:ss") 부분이 실행되어, SimpleDateFormat 객체가 생성된다.

생성된 객체의 .format(new Date()) 메소드가 호출된다.

즉 SimpleDateFormat 객체의 format 메소드가 호출된다.

이 메소드는, 파라미터로 주어진 Date 값으로 "HH:mm:ss" 포멧의 문자열을 만들어 리턴한다.

new Date() 표현식은, 현재 시각의 Date 객체를 생성한다.

실행순서:

new SimpleDateFormat("HH:mm:ss") → SimpleDateFormat 클래스 객체가 생성된다.

new SimpleDateFormat("HH:mm:ss").format(new Date()) → 생성된 객체에 format(...) 메소드가 호출된다.

<%= new SimpleDateFormat("HH:mm:ss").format(new Date()) %> → format(...) 메소드의 리턴값이 출력된다.

구현#1

String s = new SimpleDateFormat("HH:mm:ss").format(new Date());

구현#2

SimpleDateFormat fmt = new SimpleDateFormat("HH:mm:ss");

String s = fmt.format(new Date());

구현#1과 구현#2은 서로 동일하다.

(줄33) <%= new SimpleDateFormat("HH:mm:ss").format(new Date()) %>

현재 시각이 HH:mm:ss 포멧으로 출력된다.

(줄33) value='<%= new SimpleDateFormat("HH:mm:ss").format(new Date()) %>'

예를 들어 현재 시각이 09시 32분 13초이라면, 위 코드는 다음과 같이 출력된다.

value='09:32:13'

웹브라우저 창에서 소스보기 메뉴를 클릭하여, 출력된 HTML 태그 내용을 확인해 보자.

(줄50) <%= request.getParameter("text1") %>

request 객체의 getParameter 메소드 리턴 값이 출력된다.

text3a.jsp의 실행을 웹서버에 요청한 요청에 담겨져 있는 request parameter 중에서

이름이 text1인 것이 출력된다. 그런데 만약 이름이 text1인 request parameter가 없다면

getParameter 메소드는 null을 리턴한다.

JSP 표현식 태그에서 출력하는 값이 null이면 "null" 문자열이 출력된다.

즉 <%= null > 에서 문자열 null 이 출력된다.

### 실행 과정

(1) 웹브라우저는 text3a.jsp URL을 웹서버에 요청한다.

입력폼에 데이터를 입력하고 submit 버튼을 누른 것이 아니고 그냥 처음에 요청하는 것이라서,  
 요청에 같이 담아 보내는 데이터는 아직 없다.

GET 방식 요청이므로 요청에 담아 보낼 데이터를 query string 형태로 URL에 붙여야 하는데,

보내는 데이터가 없으므로 요청하는 URL에 query string이 붙어있지 않다.

요청되는 절대경로 URL은 http://localhost:8080/lecture1/form01/text3a.jsp 이다.

(2) 웹서버는 요청 받은 text3a.jsp를 실행한다.

실행도중 request.getParameter(...) 메소드를 호출하여 요청에 담겨져 전송된 request parameter를 꺼내지만

전송된 데이터가 없으므로 null이 리턴된다.

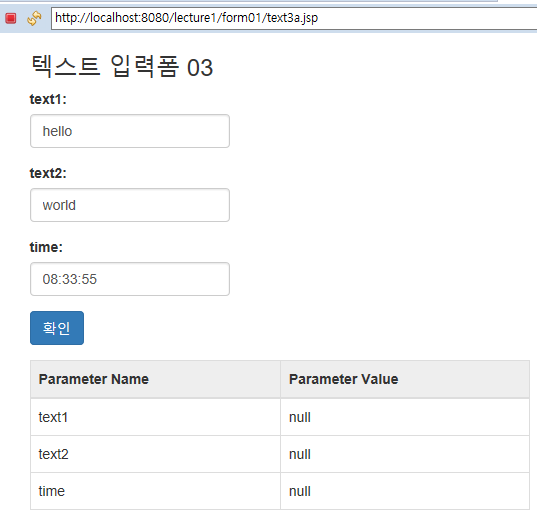
(3) text3a.jsp의 실행이 종료된다.

(4) 웹서버는 text3a.jsp의 실행결과 출력을 웹브라우저에 전달한다.

웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>    <div class="container">  <form action="text3a.jsp">  <h3>텍스트 입력폼 03</h3>  <div class="form-group">  <label>text1:</label>  <input type="text" class="form-control" name="text1" value="hello" />  </div>  <div class="form-group">  <label>text2:</label>  <input type="text" class="form-control" name="text2" value="world" />  </div>  <div class="form-group">  <label>time:</label>  <input type="text" class="form-control" name="time"  value='08:33:55' />  </div>  <div class="form-group">  <input type="submit" class="btn btn-primary" name="cmd" value="확인">  </div>  </form>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>text1</td>  <td>null</td>  </tr>  <tr>  <td>text2</td>  <td>null</td>  </tr>  <tr>  <td>time</td>  <td>null</td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

(5) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.



웹브라우저에서 URL 주소칸의 URL을 살펴보자. query string이 붙어있지 않다.

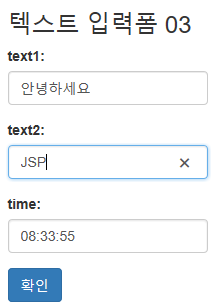
웹브라우저에서 URL 주소칸의 URL이 의미하는 것은 다음과 같다.

- 조금 전에 이 웹브라우저는 이 URL을 웹서버에 요청했다.

- 그 요청에 대해 웹서버에서 무엇인가가 실행되었고

- 그 실행 결과로 출력되어 전송된 html 태그들이 지금 웹브라우저 창에 표시되어 있다.

(6) 사용자는 웹브라우저 창의 텍스트 박스에 다음과 같이 입력하고 확인 버튼을 클릭한다.



(7) 웹브라우저는 text3b.jsp URL을 웹서버에 요청한다. 그리고 입력된 데이터도 요청에 담아 같이 보낸다.

요청에 담겨 보내지는 request parameter는 다음과 같다.

|  |  |
| --- | --- |
| name | value |
| text1 | 안녕하세요 |
| text2 | JSP |
| time | 08:33:55 |
| cmd | 확인 |

GET 방식 요청이므로, 요청에 담겨 전송되는 request parameter는 query string 형태로 URL에 추가된다.

그리고 한글 문자와 콜론(:) 문자는 query string에 허용되지 않기 때문에 url encoding되어야 한다.

url encoding은 웹브라우저에 의해 자동으로 수행된다.

그렇게 만들어진 절대 경로 URL은 다음과 같다.

http://localhost:8080/lecture1/form01/text3a.jsp?**text1**=**%EC%95%88%EB%85%95%ED%95%98%EC%84%B8%EC%9A%94**&**text2**=**JSP**&**time**=**08%3A33%3A55**&**cmd**=**%ED%99%95%EC%9D%B8**

위 URL의 query string에서 파란색으로 표시한 부분이 request parameter 이름, 빨간색으로 표시한 부분이 request parameter 값이다.

url encoding 결과는 다음과 같다

|  |  |
| --- | --- |
| 전송할 데이터 | url encoding 결과 |
| 안녕하세요 | %EC%95%88%EB%85%95%ED%95%98%EC%84%B8%EC%9A%94 |
| : | %3A |
| 확인 | %ED%99%95%EC%9D%B8 |

(8) 웹서버는 요청된 text3a.jsp 전체를 **다시 처음부터 새로 실행한다.**

JSP 페이지들은 요청될 때마다 웹서버에서 언제나 처음부터 새로 실행된다.

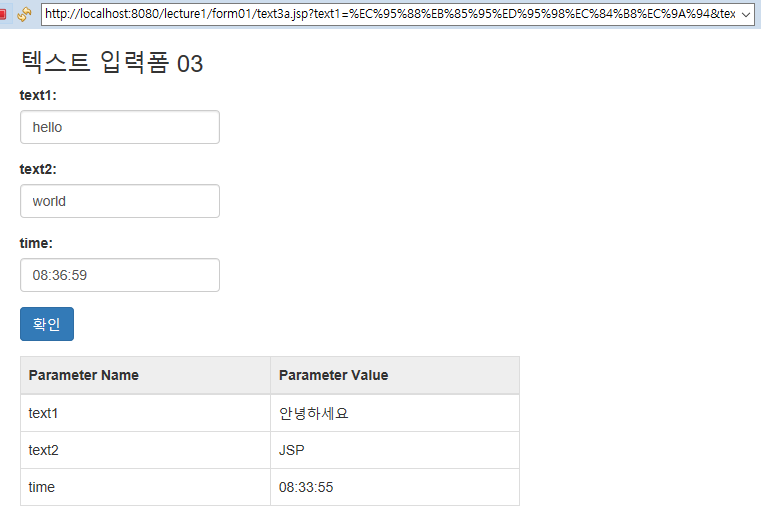
(9) text3a.jsp의 실행이 종료된다.

(10) 웹서버는 text3a.jsp의 실행 결과 출력을 웹브라우저에 전달한다.

웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>    <div class="container">  <form action="text3a.jsp">  <h3>텍스트 입력폼 03</h3>  <div class="form-group">  <label>text1:</label>  <input type="text" class="form-control" name="text1" value="hello" />  </div>  <div class="form-group">  <label>text2:</label>  <input type="text" class="form-control" name="text2" value="world" />  </div>  <div class="form-group">  <label>time:</label>  <input type="text" class="form-control" name="time"  value='08:36:59' />  </div>  <div class="form-group">  <input type="submit" class="btn btn-primary" name="cmd" value="확인">  </div>  </form>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>text1</td>  <td>안녕하세요</td>  </tr>  <tr>  <td>text2</td>  <td>JSP</td>  </tr>  <tr>  <td>time</td>  <td>08:33:55</td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

(11) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.



위 실행 결과 그림에서 아래 표부분의 두번째 열 즉 Parameter Value 열에 출력된 값들을 살펴보자.

이 값들은 text3a.jsp의 (줄52,56,60)에서 각각 출력한 것이다.

<td><%= request.getParameter("text1") %></td>

. . .

<td><%= request.getParameter("text2") %></td>

. . .

<td><%= request.getParameter("time") %></td>

url encoding되어 전송된 데이터도 다른 데이터와 마찬가지로 request.getParameter(...) 메소드를 호출하여 꺼내면 된다. 이미 웹서버에 의해서 자동으로 url decoding되어 있다.

위 실행 결과 그림에서 입력폼의 텍스트 박스에 채워진 값들을 살펴보자.

이 값들은 text3a.jsp의 (줄24,28,32)에서 각각 출력한 것이다.

<input type="text" name="text1" value="hello" />

. . .

<input type="text" name="text2" value="world" />

. . .

<input type="text" name="time" value='<%= new SimpleDateFormat("HH:mm:ss").format(new Date()) %>' />

value 애트리뷰트값으로 주어진 것들이 각각 "hello", "world", 현재시각 이므로,

언제나 "hello", "world", 현재시각이 텍스트 박스에 채워진 상태로 웹브라우저 창에 그려지는 것이다.

텍스트 박스의 value 애트리뷰트에 "hello"를 채웠으면 "hello"가 채워진 상태로 출력된다.

만약 텍스트 박스에, 아까 입력폼에 입력했고 웹서버에 전송한 값들이 채워진 상태로 출력되게 하려면,

웹서버에서 텍스트 박스 태그를 출력할 때 "hello"가 아니라 그 값을 value에 채워야 한다.

(줄33) 입력폼에 출력된 time 값은 서버에서 text3a.jsp가 실행될 당시의 현재 시각이다.

(줄58) 표에 출력된 time 값은 request.getParameter(...) 메소드를 호출하여 꺼낸 값이다.

## text3b.jsp

### WebContent/form01/text3b.jsp

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64 | <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  <%@ page import="java.util.Date, java.text.SimpleDateFormat" %>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <% request.setCharacterEncoding("utf-8"); %>  <div class="container">  <form>  <h3>텍스트 입력폼 03</h3>  <div class="form-group">  <label>text1:</label>  <input type="text" class="form-control" name="text1" value='<%= request.getParameter("text1") %>' />  </div>  <div class="form-group">  <label>text2:</label>  <input type="text" class="form-control" name="text2" value='<%= request.getParameter("text2") %>' />  </div>  <div class="form-group">  <label>time:</label>  <input type="text" class="form-control" name="time"  value='<%= new SimpleDateFormat("HH:mm:ss").format(new Date()) %>' />  </div>  <div class="form-group">  <input type="submit" class="btn btn-primary" name="cmd" value="확인">  </div>  </form>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>text1</td>  <td><%= request.getParameter("text1") %></td>  </tr>  <tr>  <td>text2</td>  <td><%= request.getParameter("text2") %></td>  </tr>  <tr>  <td>time</td>  <td><%= request.getParameter("time") %></td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

앞의 text3a.jsp의 코드에서 줄22,27,32 부분만 수정되었다.

|  |  |  |
| --- | --- | --- |
| 줄 | text3a.jsp | text3b.jsp |
| 20 | <form action="text3a.jsp"> | <form> |
| 24 | value="hello" | value="<%= request.getParameter("text1") %>" |
| 28 | value="world" | value="<%= request.getParameter("text2") %>" |

(줄20) form 태그에서 action값이 현재 페이지의 URL과 같은 경우에, action 애트리뷰트를 생략할 수 있다.

form 태그의 method 애트리뷰트의 디폴트 값이 "get"이다.

즉 text3b.jsp에서 줄22 <form> 태그는 <form action="text3b.jsp" method="get"> 과 동일하다.

(줄24,28) input 태그의 value 애트리뷰트는 그 input 태그가 웹브라우저창에 처음 그려질 때 채워질 값을 나타낸다.

앞에서 살펴본 text3a.jsp의 경우에는 페이지가 새로 그려질 때 언제나 "hello", "world"가 각각 채워진 상태로 텍스트 박스가 그려졌다.

text3b.jsp의 경우에는 request.getParameter("text1"), request.getParameter("text2") 메소드 호출로 꺼낸 값이 채워진 상태로 텍스트 박스가 그려질 것이다. 만약 꺼낸 값이 null이면 문자열 "null"이 채워진 상태로 텍스트 박스가 그려질 것이다.

### 실행 과정

(1) 웹브라우저는 text3b.jsp URL을 웹서버에 요청한다.

입력폼에 데이터를 입력하고 submit 버튼을 누른 것이 아니고, 그냥 처음 요청하는 것이라서,  
 요청에 같이 담아 보내는 데이터는 아직 없다.

즉 요청에 request parameter가 들어있지 않다.

GET 방식 요청이므로 요청에 담아 보낼 데이터를 query string 형태로 URL에 붙여야 하는데,

보내는 데이터가 없으므로 요청하는 URL에 query string이 붙어있지 않다.

요청되는 절대경로 URL은 http://localhost:8080/lecture1/form01/text3b.jsp 이다.

(2) 웹서버는 요청 받은 text3b.jsp를 실행한다.

실행도중 request.getParameter(...) 메소드를 호출하여 request parameter를 꺼내지만

전송된 데이터가 없으므로 null 이 꺼내진다.

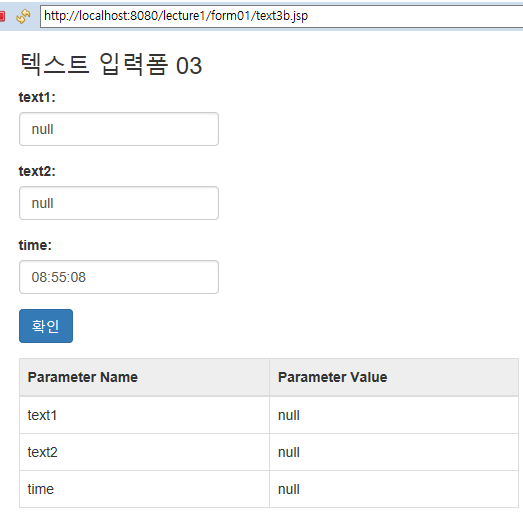
(3) text3b.jsp의 실행이 종료된다.

(4) 웹서버는 text3b.jsp의 실행결과 출력을 웹브라우저에 전달한다.

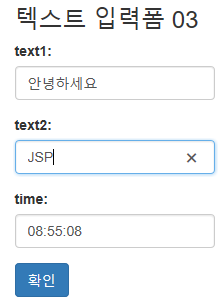
웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>    <div class="container">  <form>  <h3>텍스트 입력폼 03</h3>  <div class="form-group">  <label>text1:</label>  <input type="text" class="form-control" name="text1" value='null' />  </div>  <div class="form-group">  <label>text2:</label>  <input type="text" class="form-control" name="text2" value='null' />  </div>  <div class="form-group">  <label>time:</label>  <input type="text" class="form-control" name="time"  value='08:55:08' />  </div>  <div class="form-group">  <input type="submit" class="btn btn-primary" name="cmd" value="확인">  </div>  </form>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>text1</td>  <td>null</td>  </tr>  <tr>  <td>text2</td>  <td>null</td>  </tr>  <tr>  <td>time</td>  <td>null</td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

(5) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.



(6) 사용자는 웹브라우저 창의 텍스트 박스에 다음과 같이 입력하고 확인 버튼을 클릭한다.



(7) 웹브라우저는 text3a.jsp URL을 웹서버에 요청한다. 그리고 입력된 데이터도 요청에 담아 같이 보낸다.

요청에 담겨 보내지는 데이터는 다음과 같다.

|  |  |
| --- | --- |
| name | value |
| text1 | 안녕하세요 |
| text2 | JSP |
| time | 08:55:08 |
| cmd | 확인 |

GET 방식 요청이므로, 요청에 담겨 전송되는 데이터는 query string 형태로 URL에 추가된다.

그렇게 만들어진 절대 경로 URL은 다음과 같다.

|  |
| --- |
| http://localhost:8080/lecture1/form01/text3b.jsp?text1=%EC%95%88%EB%85%95%ED%95%98%EC%84%B8%EC%9A%94&text2=JSP&time=08%3A55%3A08&cmd=%ED%99%95%EC%9D%B8 |

(8) 웹서버는 요청된 text3b.jsp를 다시 처음부터 새로 실행한다.

JSP 페이지들은 요청될 때마다 웹서버에서 언제나 처음부터 새로 실행된다.

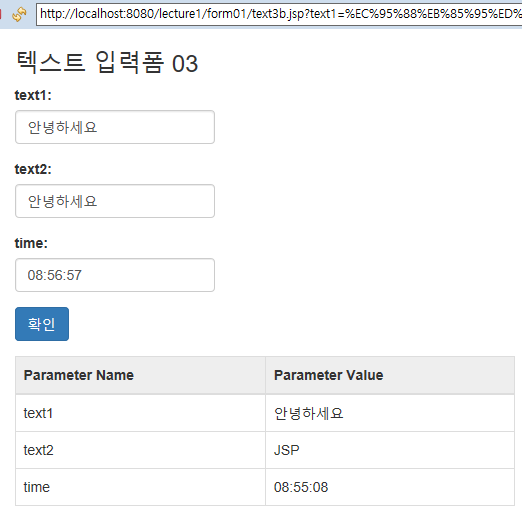
(9) text3b.jsp의 실행이 종료된다.

(10) 웹서버는 text3b.jsp의 실행결과 출력을 웹브라우저에 전달한다.

웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>    <div class="container">  <form>  <h3>텍스트 입력폼 03</h3>  <div class="form-group">  <label>text1:</label>  <input type="text" class="form-control" name="text1" value='안녕하세요' />  </div>  <div class="form-group">  <label>text2:</label>  <input type="text" class="form-control" name="text2" value='안녕하세요' />  </div>  <div class="form-group">  <label>time:</label>  <input type="text" class="form-control" name="time"  value='08:56:57' />  </div>  <div class="form-group">  <input type="submit" class="btn btn-primary" name="cmd" value="확인">  </div>  </form>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>text1</td>  <td>안녕하세요</td>  </tr>  <tr>  <td>text2</td>  <td>JSP</td>  </tr>  <tr>  <td>time</td>  <td>08:55:08</td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

(11) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.



## text3c.jsp

### WebContent/form01/text3c.jsp

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72 | <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  <%@ page import="java.util.Date, java.text.SimpleDateFormat" %>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <%  request.setCharacterEncoding("utf-8");  String text1 = request.getParameter("text1");  String text2 = request.getParameter("text2");  String time = request.getParameter("time");  if (text1 == null) text1 = "";  if (text2 == null) text2 = "";  if (time == null) time = "";  String currentTime = new SimpleDateFormat("HH:mm:ss").format(new Date());  %>  <div class="container">  <form>  <h3>텍스트 입력폼 03</h3>  <div class="form-group">  <label>text1:</label>  <input type="text" class="form-control" name="text1" value="<%= text1 %>" />  </div>  <div class="form-group">  <label>text2:</label>  <input type="text" class="form-control" name="text2" value="<%= text2 %>" />  </div>  <div class="form-group">  <label>time:</label>  <input type="text" class="form-control" name="time" value='<%= currentTime %>' />  </div>  <div class="form-group">  <input type="submit" class="btn btn-primary" name="cmd" value="확인">  </div>  </form>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>text1</td>  <td><%= text1 %></td>  </tr>  <tr>  <td>text2</td>  <td><%= text2 %></td>  </tr>  <tr>  <td>time</td>  <td><%= time %></td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

text3c.jsp는 text3b.jsp를 약간 개선하였다.

(줄20~26) 출력할 값들을 미리 Java 지역 변수에 저장하는 코드를 JSP 스크립릿 태그에서 구현하였다.

String text1 = request.getParameter("text1");

String text2 = request.getParameter("text2");

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

String currentTime = new SimpleDateFormat("HH:mm:ss").format(new Date());

(줄23~25) request.getParameter(...)로 꺼낸 값이 null인 경우에는 null 대신 빈 문자열을 변수에 대입한다.

if (text1 == null) text1 = "";

if (text2 == null) text2 = "";

if (time == null) time = "";

(줄33,37,41,57,62,66) JSP 표현식 태그에서는 변수값만 출력하면 되니 간편하다.

<%= text1 %>

<%= text2 %>

<%= time %>

<%= currentTime %>

구현 형태만 약간 다를뿐 실행 과정 및 결과는 text3b.jsp와 동일하다.

# 목록 선택

## select1.jsp

### WebContent/form01/select1.jsp

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46 | <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  select.form-control { width: 70px; }  </style>  </head>  <body>  <div class="container">  <form action="text2action.jsp" method="get">  <h1>select 01</h1>  <div class="form-group">  <label>number1:</label>  <input type="text" name="number1" class="form-control" value="34" />  </div>    <div class="form-group">  <label>operator:</label>  <select name="cmd" class="form-control">  <option value="+">+</option>  <option value="-">-</option>  <option value="\*">\*</option>  <option value="/">/</option>  </select>  </div>  <div class="form-group">  <label>number2:</label>  <input type="text" name="number2" class="form-control" value="8" />  </div>  <div class="form-group">  <button type="submit" class="btn btn-primary">확인</button>  </div>  </form>  </div>  </body>  </html> |

(줄27~32) select 태그는 목록에서 항목 하나를 선택하기 위한 입력 요소다.

select 태그 내부의 option 들이 선택 항목들에 해당한다.

option 태그 사이의 내용이 화면에 표시된다.

(줄41) submit 버튼이 클릭되어 웹서버에 "text2action.jsp"가 요청될 때,

select 태그의 선택된 항목(option)의 value 값도 request parameter에 포함되어 전송된다.

이 request parameter의 이름은 select 태그의 name 값이다.

### 실행 과정

(1) 웹브라우저는 select1.jsp URL을 웹서버에 요청한다.

(2) 웹서버는 요청받은 select1.jsp를 실행한다.

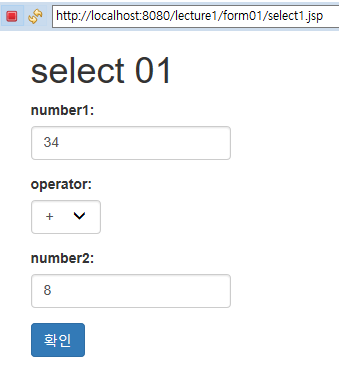
(3) select1.jsp의 실행이 종료된다.

(4) 웹서버는 select1.jsp의 실행결과 출력을 웹브라우저에 전달한다.

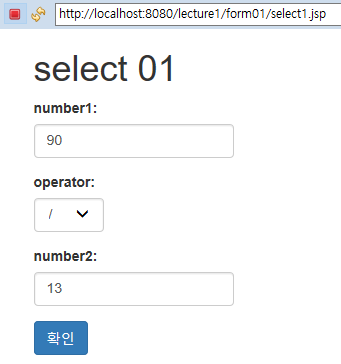
웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  select.form-control { width: 70px; }  </style>  </head>  <body>  <div class="container">  <form action="text2action.jsp" method="get">  <h1>select 01</h1>  <div class="form-group">  <label>number1:</label>  <input type="text" name="number1" class="form-control" value="34" />  </div>    <div class="form-group">  <label>operator:</label>  <select name="cmd" class="form-control">  <option value="+">+</option>  <option value="-">-</option>  <option value="\*">\*</option>  <option value="/">/</option>  </select>  </div>  <div class="form-group">  <label>number2:</label>  <input type="text" name="number2" class="form-control" value="8" />  </div>  <div class="form-group">  <button type="submit" class="btn btn-primary">확인</button>  </div>  </form>  </div>  </body>  </html> |

(5) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.



(6) 사용자는 웹브라우저 창의 텍스트 박스에 다음과 같이 입력한다.



(7) 위와 같이 입력하고 마지막 "확인" 버튼을 클릭하면 다음과 같은 요청이 웹서버에 전송된다.

|  |  |
| --- | --- |
| 요청되는 URL | text2action.jsp |
| 요청 방식 | GET |

요청에 담겨 보내지는 데이터는 다음과 같다.

|  |  |
| --- | --- |
| name | value |
| number1 | 90 |
| number2 | 13 |
| cmd | / |

GET 방식 요청이므로 요청되는 URL은 다음과 같다.

http://localhost:8080/lecture1/form01/text2action.jsp**?number1=90&number2=13&cmd=%2F**

(8) 웹서버는 요청된 text2action.jsp를 실행한다.

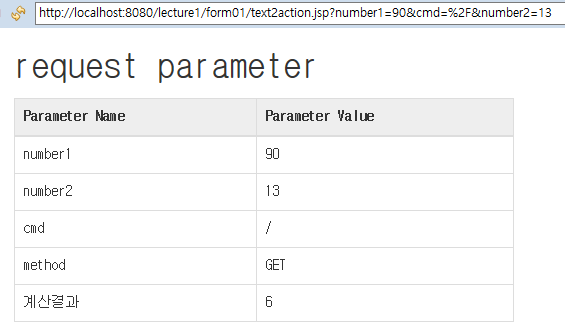
(9) text2action.jsp의 실행이 종료된다.

(10) 웹서버는 text2action.jsp의 실행결과 출력을 웹브라우저에 전달한다.

웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  body { font-family: 굴림체; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <div class="container">  <h1>request parameter</h1>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>number1</td>  <td>90</td>  </tr>  <tr>  <td>number2</td>  <td>13</td>  </tr>  <tr>  <td>cmd</td>  <td>/</td>  </tr>  <tr>  <td>method</td>  <td>GET</td>  </tr>  <tr>  <td>계산결과</td>  <td>6</td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

(11) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.



위 그림에서 주소창의 URL을 확인하자.

http://localhost:8080/lecture1/form01/text2action.jsp?number1=90&number2=13&cmd=%2F

현재 주소창에 표시된 URL을 웹브라우저가 웹서버에 요청하였고, 이 URL 요청 결과로 웹서버로부터 전송받은 HTML 태그들이 현재 창에 출력되었다는 것을 의미한다.

## select2a.jsp

### WebContent/form01/select2a.jsp

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96 | <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  <%@ page import="java.util.Date, java.text.SimpleDateFormat" %>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  select.form-control { width: 70px; }  </style>  </head>  <body>  <%  String s1 = request.getParameter("number1");  int number1 = Integer.parseInt(s1);  String s2 = request.getParameter("number2");  int number2 = Integer.parseInt(s2);  String cmd = request.getParameter("cmd");  int result = 0;  if (cmd.equals("+")) result = number1 + number2;  else if (cmd.equals("-")) result = number1 - number2;  else if (cmd.equals("\*")) result = number1 \* number2;  else if (cmd.equals("/")) result = number1 / number2;  %>  <div class="container">  <form method="get">  <h1>select 01</h1>  <div class="form-group">  <label>number1:</label>  <input type="text" name="number1" class="form-control" value="34" />  </div>    <div class="form-group">  <label>operator:</label>  <select name="cmd" class="form-control">  <option value="+">+</option>  <option value="-">-</option>  <option value="\*">\*</option>  <option value="/">/</option>  </select>  </div>  <div class="form-group">  <label>number2:</label>  <input type="text" name="number2" class="form-control" value="8" />  </div>  <div class="form-group">  <button type="submit" class="btn btn-primary">확인</button>  </div>  </form>    <h1>request parameter</h1>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>number1</td>  <td><%= number1 %></td>  </tr>  <tr>  <td>number2</td>  <td><%= number2 %></td>  </tr>  <tr>  <td>cmd</td>  <td><%= cmd %></td>  </tr>  <tr>  <td>method</td>  <td><%= request.getMethod() %></td>  </tr>  <tr>  <td>계산결과</td>  <td><%= result %></td>  </tr>  <tr>  <td>시각</td>  <td><%= new SimpleDateFormat("HH:mm:ss").format(new Date()) %></td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

이 select2a.jsp를 실행하면 에러가 발생한다.

(줄18) String s1 = request.getParameter("number1");

이름이 "number1"인 request parameter 값을 지역변수 s1에 대입한다.

이름이 " number1"인 request parameter 가 없다면, 지역변수 s1에 null 이 대입된다.

(줄19) int number1 = Integer.parseInt(s1);

지역변수 s1의 값 문자열이 숫자가 아니거나 null이라면,

Integer.parseInt(s1) 메소드 호출에서 NumberFormatException 에러가 발생한다.

### 실행 과정

(1) 웹브라우저는 웹서버에 http://localhost:8080/lecture1/form01/select2a.jsp URL의 실행을 요청한다.

그냥 단순히 URL만 요청한 것이고 URL에 query string이 포함되지 않았으므로,

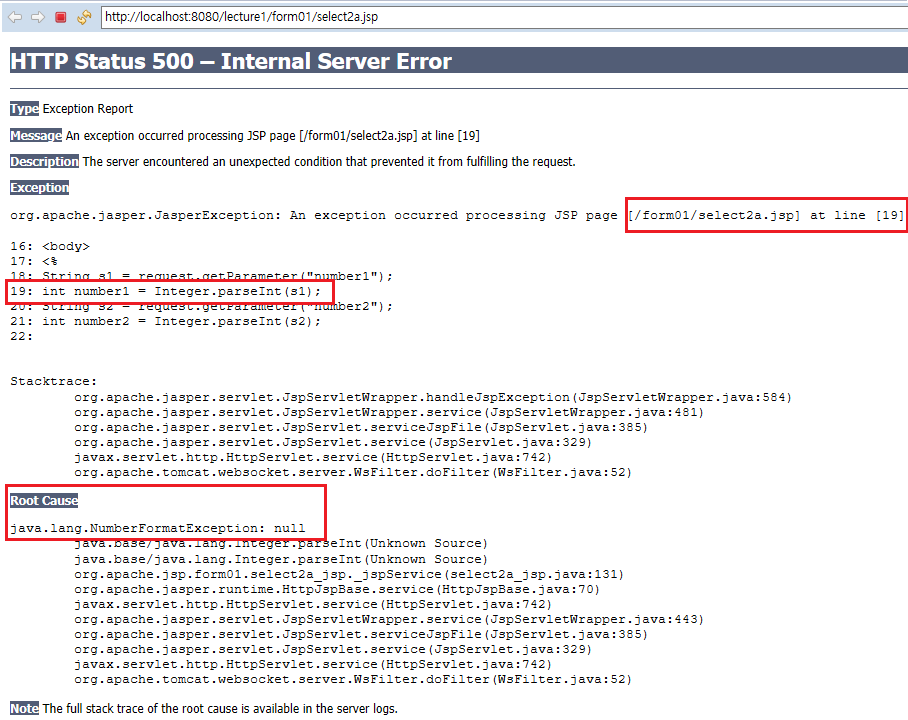
요청에 담겨져 전송되는 request parameter는 없다.

(2) 웹서버는 요청된 select2a.jsp를 실행한다.

(3) select2a.jsp 실행도중 에러가 발생한다.

(4) 웹서버는, 발생한 에러에 대한 정보를 다음 HTML 태그를 웹브라우저에 전송한다.

(5) 웹브라우저는 전송된 내용을 웹브라우저 창에 다음과 같이 그린다.



위 화면에서 발생한 에러가 NullPointerException이라는 것,

그리고 에러가 발생한 위치가 select2a.jsp의 줄19라는 것을 확인할 수 있다.

## select2b.jsp

### WebContent/form01/select2b.jsp

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99 | <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  <%@ page import="java.util.Date, java.text.SimpleDateFormat" %>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  select.form-control { width: 70px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <%  String s1 = request.getParameter("number1");  int number1 = (s1 == null) ? 0 : Integer.parseInt(s1);  String s2 = request.getParameter("number2");  int number2 = (s2 == null) ? 0 : Integer.parseInt(s2);  String cmd = request.getParameter("cmd");  int result = 0;  if ("+".equals(cmd)) result = number1 + number2;  else if ("-".equals(cmd)) result = number1 - number2;  else if ("\*".equals(cmd)) result = number1 \* number2;  else if ("/".equals(cmd)) result = number1 / number2;  %>  <div class="container">  <form method="get">  <h1>select 01</h1>  <div class="form-group">  <label>number1:</label>  <input type="text" name="number1" class="form-control" value="34" />  </div>    <div class="form-group">  <label>operator:</label>  <select name="cmd" class="form-control">  <option value="+">+</option>  <option value="-">-</option>  <option value="\*">\*</option>  <option value="/">/</option>  </select>  </div>  <div class="form-group">  <label>number2:</label>  <input type="text" name="number2" class="form-control" value="8" />  </div>  <div class="form-group">  <button type="submit" class="btn btn-primary">확인</button>  </div>  </form>    <h1>request parameter</h1>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>number1</td>  <td><%= number1 %></td>  </tr>  <tr>  <td>number2</td>  <td><%= number2 %></td>  </tr>  <tr>  <td>cmd</td>  <td><%= cmd %></td>  </tr>  <tr>  <td>method</td>  <td><%= request.getMethod() %></td>  </tr>  <tr>  <td>계산결과</td>  <td><%= result %></td>  </tr>  <tr>  <td>시각</td>  <td><%= new SimpleDateFormat("HH:mm:ss").format(new Date()) %></td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

s1, s2, cmd 지역 변수 값이 null 이어도 문제가 발생하지 않도록 수정하였다.

(줄21) int number1 = (s1 == null) ? 0 : Integer.parseInt(s1);

s1 값이 null 이면, number1 지역변수에 0 이 대입된다.

(줄27) "+".equals(cmd)

"+" 이것도 문자열 객체이다.

"+" 문자열 객체에 equals 메소드를 호출한다.

파라미터 cmd 값이 null 이어도 NullPointerException 이 발생하지 않는다.

### 실행 과정

(1) 웹브라우저는 웹서버에 http://localhost:8080/lecture1/form01/select2b.jsp URL의 실행을 요청한다.

그냥 단순히 URL만 요청한 것이고 URL에 query string이 포함되지 않았으므로,

요청에 담겨져 전송되는request parameter는 없다.

(2) 웹서버는 요청받은 select2b.jsp를 실행한다.

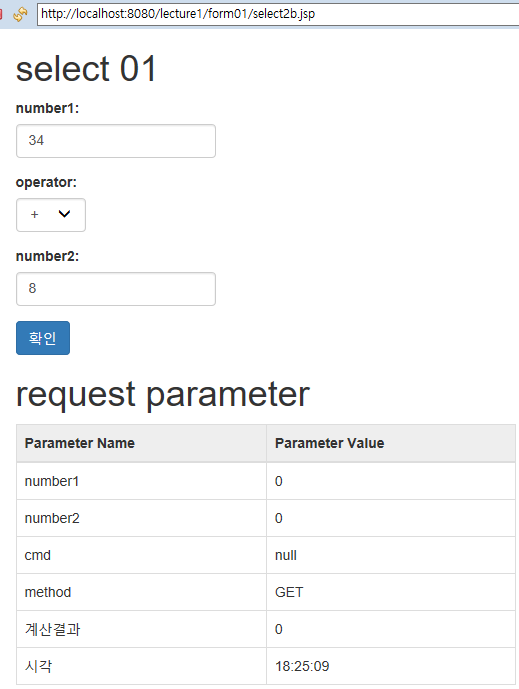
(3) select2b.jsp의 실행이 종료된다.

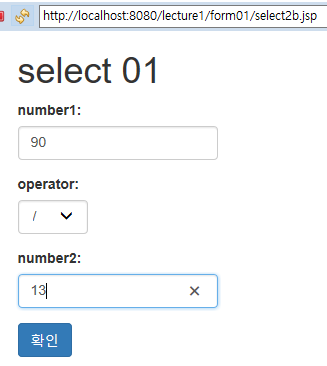
(4) 웹서버는 select2b.jsp의 실행결과 출력을 웹브라우저에 전달한다.

웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  select.form-control { width: 70px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <div class="container">  <form method="get">  <h1>select 01</h1>  <div class="form-group">  <label>number1:</label>  <input type="text" name="number1" class="form-control" value="34" />  </div>    <div class="form-group">  <label>operator:</label>  <select name="cmd" class="form-control">  <option value="+">+</option>  <option value="-">-</option>  <option value="\*">\*</option>  <option value="/">/</option>  </select>  </div>  <div class="form-group">  <label>number2:</label>  <input type="text" name="number2" class="form-control" value="8" />  </div>  <div class="form-group">  <button type="submit" class="btn btn-primary">확인</button>  </div>  </form>    <h1>request parameter</h1>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>number1</td>  <td>0</td>  </tr>  <tr>  <td>number2</td>  <td>0</td>  </tr>  <tr>  <td>cmd</td>  <td>null</td>  </tr>  <tr>  <td>method</td>  <td>GET</td>  </tr>  <tr>  <td>계산결과</td>  <td>0</td>  </tr>  <tr>  <td>시각</td>  <td>18:25:09</td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

(5) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.





(7) 위와 같이 입력하고 마지막 "계산" 버튼을 클릭하면 다음과 같은 요청이 웹서버에 전송된다.

|  |  |
| --- | --- |
| 요청되는 URL | select2b.jsp |
| 요청 방식 | GET |

요청에 담겨 보내지는 데이터는 다음과 같다.

|  |  |
| --- | --- |
| name | value |
| number1 | 90 |
| number2 | 13 |
| cmd | / |

GET 방식 요청이므로 요청되는 URL은 다음과 같다.

http://localhost:8080/lecture1/form01/select2b.jsp**?number1=90&cmd=%2F&number2=13**

(8) 웹서버는 요청된 select2b.jsp를 실행한다.

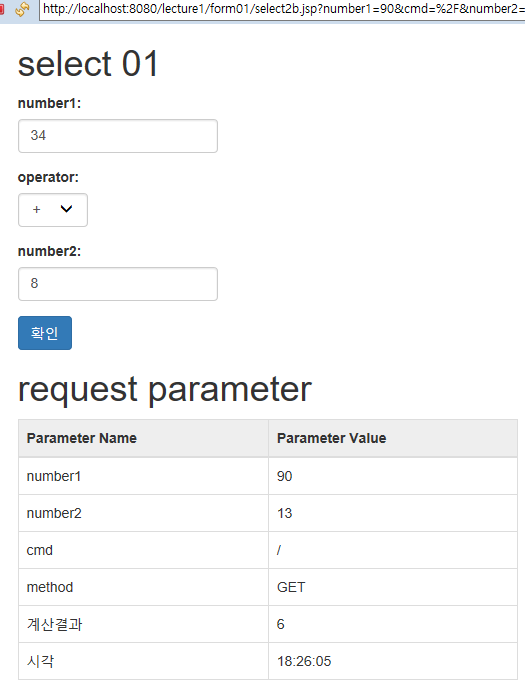
(9) select2b.jsp의 실행이 종료된다.

(10) 웹서버는 select2b.jsp의 실행결과 출력을 웹브라우저에 전달한다.

웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  select.form-control { width: 70px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <div class="container">  <form method="get">  <h1>select 01</h1>  <div class="form-group">  <label>number1:</label>  <input type="text" name="number1" class="form-control" value="34" />  </div>    <div class="form-group">  <label>operator:</label>  <select name="cmd" class="form-control">  <option value="+">+</option>  <option value="-">-</option>  <option value="\*">\*</option>  <option value="/">/</option>  </select>  </div>  <div class="form-group">  <label>number2:</label>  <input type="text" name="number2" class="form-control" value="8" />  </div>  <div class="form-group">  <button type="submit" class="btn btn-primary">확인</button>  </div>  </form>    <h1>request parameter</h1>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>number1</td>  <td>90</td>  </tr>  <tr>  <td>number2</td>  <td>13</td>  </tr>  <tr>  <td>cmd</td>  <td>/</td>  </tr>  <tr>  <td>method</td>  <td>GET</td>  </tr>  <tr>  <td>계산결과</td>  <td>6</td>  </tr>  <tr>  <td>시각</td>  <td>18:26:05</td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

(11) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.



위 그림에서 주소창의 URL을 확인하자.

http://localhost:8080/lecture1/form01/select2b.jsp?number1=90&cmd=%2F&number2=13

현재 주소창에 표시된 URL을 웹브라우저가 웹서버에 요청하였고, 이 URL 요청 결과로 웹서버로부터 전송받은 HTML 태그들이 현재 창에 출력되었다는 것을 의미한다.

아까 입력한 값들이(90, 13) 입력 폼에 남아있지 않은 것에 주목하자.

select2b.jsp 에서 출력된 HTML 태그 그대로 웹브라우저에 그려진다.

select2b.jsp의 입력 폼 태그는 언제나 34, 8 값이 채워진 형태로 출력된다.

## select2c.jsp

### WebContent/form01/select2c.jsp

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99 | <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  <%@ page import="java.util.Date, java.text.SimpleDateFormat" %>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  select.form-control { width: 70px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <%  String s1 = request.getParameter("number1");  int number1 = (s1 == null) ? 0 : Integer.parseInt(s1);  String s2 = request.getParameter("number2");  int number2 = (s2 == null) ? 0 : Integer.parseInt(s2);  String cmd = request.getParameter("cmd");  int result = 0;  if ("+".equals(cmd)) result = number1 + number2;  else if ("-".equals(cmd)) result = number1 - number2;  else if ("\*".equals(cmd)) result = number1 \* number2;  else if ("/".equals(cmd)) result = number1 / number2;  %>  <div class="container">  <form method="get">  <h1>select 01</h1>  <div class="form-group">  <label>number1:</label>  <input type="text" name="number1" class="form-control" value="<%= number1 %>" />  </div>    <div class="form-group">  <label>operator:</label>  <select name="cmd" class="form-control">  <option value="+" <%= "+".equals(cmd) ? "selected" : "" %>>+</option>  <option value="-" <%= "-".equals(cmd) ? "selected" : "" %>>-</option>  <option value="\*" <%= "\*".equals(cmd) ? "selected" : "" %>>\*</option>  <option value="/" <%= "/".equals(cmd) ? "selected" : "" %>>/</option>  </select>  </div>  <div class="form-group">  <label>number2:</label>  <input type="text" name="number2" class="form-control" value="<%= number2 %>" />  </div>  <div class="form-group">  <button type="submit" class="btn btn-primary">확인</button>  </div>  </form>    <h1>request parameter</h1>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>number1</td>  <td><%= number1 %></td>  </tr>  <tr>  <td>number2</td>  <td><%= number2 %></td>  </tr>  <tr>  <td>cmd</td>  <td><%= cmd %></td>  </tr>  <tr>  <td>method</td>  <td><%= request.getMethod() %></td>  </tr>  <tr>  <td>계산결과</td>  <td><%= result %></td>  </tr>  <tr>  <td>시각</td>  <td><%= new SimpleDateFormat("HH:mm:ss").format(new Date()) %></td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

(줄39) <input type="text" name="number1" class="form-control" value="<%= number1 %>" />

문자열을 입력할 input 태그가 그려질 때, number1 변수 값이 채워진 상태로 그려진다.

(줄45) <%= "+".equals(cmd) ? "selected" : "" %>

cmd 변수값이 "+" 이면 selected 문자열이 출력되고, 아니면 빈 문자열이 출력된다.

(줄45~48) 예를들어 cmd 변수값이 "-" 이라면, JSP 실행 결과 다음과 같은 HTML 태그가 출력된다.

<select name="cmd">

<option value="+">+</option>

<option value="-" selected>-</option>

<option value="\*">\*</option>

<option value="/">/</option>

</select>

위 태그가 웹브라우저창에 그려질 때 다음과 같이 그려진다.



selected 애트리뷰트가 있는 option이 선택된 상태로 그려진다.

### 실행 과정

(1) 웹브라우저는 웹서버에 http://localhost:8080/lecture1/form01/select2c.jsp URL의 실행을 요청한다.

그냥 단순히 URL만 요청한 것이고 URL에 query string이 포함되지 않았으므로,

요청에 담겨져 전송되는 request parameter는 없다.

(2) 웹서버는 요청받은 select2c.jsp를 실행한다.

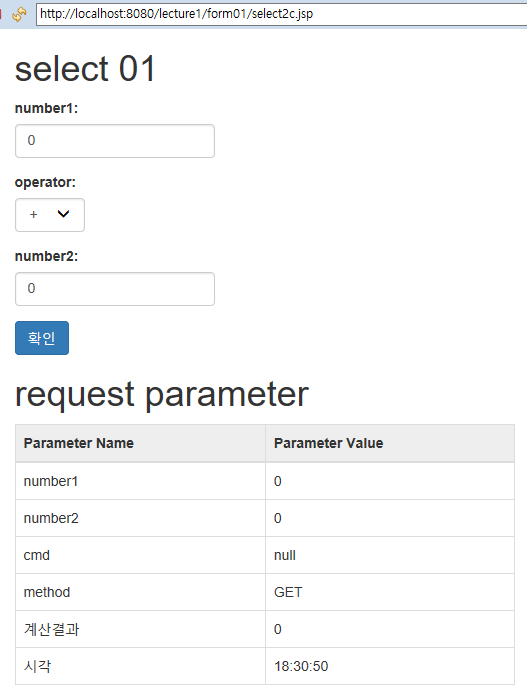
(3) select2c.jsp의 실행이 종료된다.

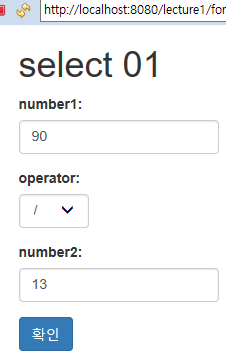
(4) 웹서버는 select2c.jsp의 실행결과 출력을 웹브라우저에 전달한다.

웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  select.form-control { width: 70px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <div class="container">  <form method="get">  <h1>select 01</h1>  <div class="form-group">  <label>number1:</label>  <input type="text" name="number1" class="form-control" value="0" />  </div>    <div class="form-group">  <label>operator:</label>  <select name="cmd" class="form-control">  <option value="+" >+</option>  <option value="-" >-</option>  <option value="\*" >\*</option>  <option value="/" >/</option>  </select>  </div>  <div class="form-group">  <label>number2:</label>  <input type="text" name="number2" class="form-control" value="0" />  </div>  <div class="form-group">  <button type="submit" class="btn btn-primary">확인</button>  </div>  </form>    <h1>request parameter</h1>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>number1</td>  <td>0</td>  </tr>  <tr>  <td>number2</td>  <td>0</td>  </tr>  <tr>  <td>cmd</td>  <td>null</td>  </tr>  <tr>  <td>method</td>  <td>GET</td>  </tr>  <tr>  <td>계산결과</td>  <td>0</td>  </tr>  <tr>  <td>시각</td>  <td>18:30:50</td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

(5) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.





(7) 위와 같이 입력하고 마지막 "계산" 버튼을 클릭하면 다음과 같은 요청이 웹서버에 전송된다.

|  |  |
| --- | --- |
| 요청되는 URL | select2c.jsp |
| 요청 방식 | GET |

요청에 담겨 보내지는 데이터는 다음과 같다.

|  |  |
| --- | --- |
| name | value |
| number1 | 90 |
| number2 | 13 |
| cmd | / |

GET 방식 요청이므로 요청되는 URL은 다음과 같다.

http://localhost:8080/lecture1/form01/select2c.jsp**?number1=90&cmd=%2F&number2=13**

(8) 웹서버는 요청된 select2c.jsp를 실행한다.

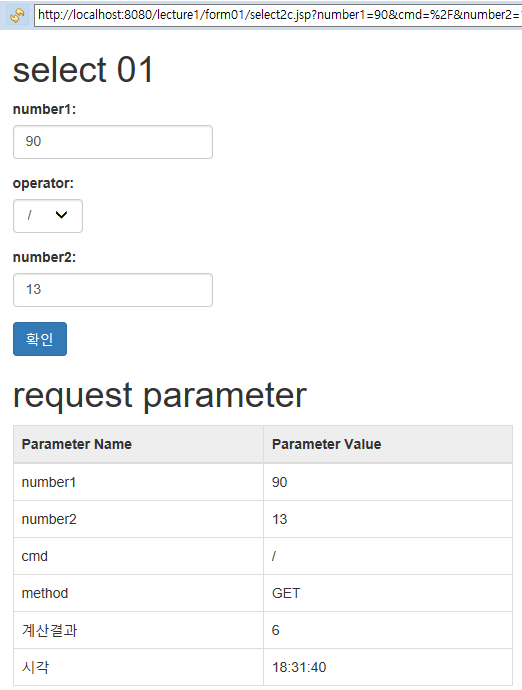
(9) select2c.jsp의 실행이 종료된다.

(10) 웹서버는 select2c.jsp의 실행결과 출력을 웹브라우저에 전달한다.

웹브라우저에 전달된 내용은 다음과 같다.

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  input.form-control { width: 200px; }  select.form-control { width: 70px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <div class="container">  <form method="get">  <h1>select 01</h1>  <div class="form-group">  <label>number1:</label>  <input type="text" name="number1" class="form-control" value="90" />  </div>    <div class="form-group">  <label>operator:</label>  <select name="cmd" class="form-control">  <option value="+" >+</option>  <option value="-" >-</option>  <option value="\*" >\*</option>  <option value="/" selected>/</option>  </select>  </div>  <div class="form-group">  <label>number2:</label>  <input type="text" name="number2" class="form-control" value="13" />  </div>  <div class="form-group">  <button type="submit" class="btn btn-primary">확인</button>  </div>  </form>    <h1>request parameter</h1>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>number1</td>  <td>90</td>  </tr>  <tr>  <td>number2</td>  <td>13</td>  </tr>  <tr>  <td>cmd</td>  <td>/</td>  </tr>  <tr>  <td>method</td>  <td>GET</td>  </tr>  <tr>  <td>계산결과</td>  <td>6</td>  </tr>  <tr>  <td>시각</td>  <td>18:31:40</td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

(11) 웹브라우저는 전달된 내용을 웹브라우저 창에 다음과 같이 그린다.



위 그림에서 주소창의 URL을 확인하자.

http://localhost:8080/lecture1/form01/select2c.jsp?number1=90&cmd=%2F&number2=13

현재 주소창에 표시된 URL을 웹브라우저가 웹서버에 요청하였고, 이 URL 요청 결과로 웹서버로부터 전송받은 HTML 태그들이 현재 창에 출력되었다는 것을 의미한다.

아까 입력한 값들이(90, 13) 입력 폼에 남아있는 것에 주목하자.

select2c.jsp 는 요청과 함계 전달 받은 값을 입력 폼 태그에 채워서 출력한다.

# 실습 과제

## 코드 분석

다음 코드의 실행 과정을 자세히 분석하라.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92 | <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  <%@ page import="java.util.Date, java.text.SimpleDateFormat" %>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  body form label { margin-right: 20px; }  table.table { width: 500px; }  thead tr { background-color: #eee; }  </style>  </head>  <body>  <%  String c1 = request.getParameter("checkbox1");  String 동의\_checked = "true".equals(c1) ? "checked" : "";  String r1 = request.getParameter("radio1");  String 남\_checked = "남".equals(r1) ? "checked" : "";  String 여\_checked = "여".equals(r1) ? "checked" : "";  String r2 = request.getParameter("radio2");  String A\_checked = "A".equals(r2) ? "checked" : "";  String B\_checked = "B".equals(r2) ? "checked" : "";  String AB\_checked = "AB".equals(r2) ? "checked" : "";  String O\_checked = "O".equals(r2) ? "checked" : "";  %>  <div class="container">  <form method="get">  <h3>checkbox</h3>  <div class="checkbox">  <label>  <input type="checkbox" name="checkbox1" value="true" <%= 동의\_checked %> />  동의하십니까?  </label>  </div>  <hr />  <h3>radio - 성별은?</h3>  <div class="radio">  <label><input type="radio" name="radio1" value="남" <%= 남\_checked %> /> 남자</label>  </div>  <div class="radio">  <label><input type="radio" name="radio1" value="여" <%= 여\_checked %> /> 여자</label>  </div>  <hr />  <h3>radio - 혈액형은?</h3>  <div class="radio">  <label><input type="radio" name="radio2" value="A" <%= A\_checked %> /> A형</label>  <label><input type="radio" name="radio2" value="B" <%= B\_checked %> /> B형</label>  <label><input type="radio" name="radio2" value="AB" <%= AB\_checked %> /> AB형</label>  <label><input type="radio" name="radio2" value="O" <%= O\_checked %> /> O형</label>  </div>  <div class="form-group">  <button type="submit" class="btn btn-primary">확인</button>  </div>  </form>  <hr />    <h1>request parameter</h1>  <table class="table table-bordered">  <thead>  <tr>  <th>Parameter Name</th>  <th>Parameter Value</th>  </tr>  </thead>  <tbody>  <tr>  <td>checkbox1</td>  <td><%= c1 %></td>  </tr>  <tr>  <td>radio1</td>  <td><%= r1 %></td>  </tr>  <tr>  <td>radio2</td>  <td><%= r2 %></td>  </tr>  </tbody>  </table>  </div>  </body>  </html> |

<input type="checkbox" name="checkbox1" value="true" />

체크박스 태그이다.

이 체크박스가 체크된 상태에서 입력폼의 submit 버튼이 클릭되면,

request parameter에 name="checkbox1" value="true" 값이 포함된다.

체크되지 않은 상태에서 입력폼의 submit 버튼이 클릭되면, request parameter에 데이터가 포함되지 않는다.

<input type="checkbox" name="checkbox1" value="true" checked />

체크박스가 처음 화면에 그려질 때, 체크된 상태로 그려진다.

<input type="radio" name="radio1" value="남" />

<input type="radio" name="radio1" value="여" />

radio 버튼 태그이다.

name 값이 같은 radio 버튼들 중에서 한 개만 체크될 수 있고, 동시에 여러 개 체크 될 수 없다.

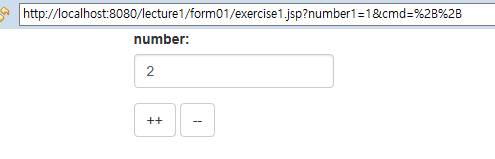
value="여" 버튼이 체크된 상태에서 입력폼의 submit 버튼이 클릭되면,

request parameter에 name="radio1" value="여" 값이 포함된다.

<input type="radio" name="radio1" value="남" checked />

radio 버튼이 처음 화면에 그려질 때, 체크된 상태로 그려진다.

## 구현 실습



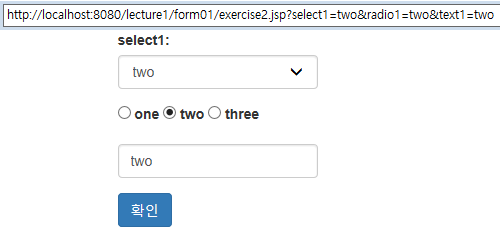
입력 폼에 텍스트박스 한 개와, submit 버튼 두 개를 구현하라.

처음 실행하면, 텍스트박스에 0이 채워져 있어야 한다.

++ 버튼을 클릭할 때 마다, 텍스트박스에 채워진 숫자가 1씩 증가하고,

-- 버튼을 클릭할 때 마다, 1씩 감소해야 한다.

## 구현 실습



입력 폼의 첫 항목은 select 태그이다. 이 태그에서 선택할 수 있는 값은 one, two, three 이다.

그 아래 항목은 radio 버튼 세 개이고,

그 아래 항목은 텍스트박스이다.

확인 버트은 submit 버튼이다.

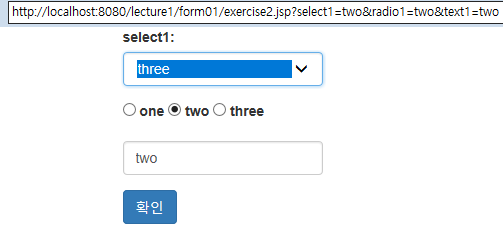
select 태그에서 one, two, three 중에 하나를 선택하고, 확인 버튼을 클릭하면,

그 값이 해당하는 radio 버튼이 자동으로 선택된 상태가 되고,

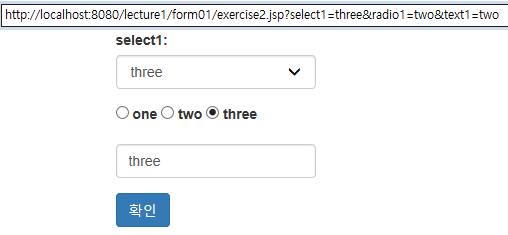
그 값이 텍스트박스에도 자동으로 채워져야 한다.

select 태그도, 그 값이 선택된 상태가 되어야 한다.

예를 들어 위 화면의 select 태그에서 three를 선택하고



확인 버튼을 클릭하면



화면이 위와 같이 변해야 한다.