**[ 3 ] Servlet 맛보기 ; Servlet은 java 파일**

목표 : Servlet 문서 작성하기, 서블릿 맵핑, doGet(), doPost()

1. Servlet 문서 작성하기
   1. Servlet의 특징
      1. 동적 웹어플리케이션 컴포넌트 ; JSP랑 같음
      2. .java 확장자 ; JAVA 언어를 사용하여 웹 프로그램을 제작하는 것

Servlet (Interface)

↑(구현)

GenericServlet (abstract)

↑(상속)

HttpServlet

↑(상속)

우리가 작성한 서블릿 클래스

* + 1. 클라이언트의 요청에 동적으로 작동하고, 응답은 html을 이용
    2. **java thread**이용하여 동작 ; 서블릿을 이용하는 가장 큰 장점. 웹컴포넌트를 구성하는 다른 언어에 비해, 요청이 많이 들어와도 서버의 부하가 많이 걸리지 않는 장점
  1. Servlet 파일 생성 ; 서블릿 생성 위치 확인해 보자(cf. 프로젝트 생성시 context root는 프로젝트명과 동일하게 이클립스가 만들어 준 대로 쓰자)
     1. Java packate : com.tj.ex / Class Name : HelloWorld /
     2. SupperClass : javax.servlet.http.HttpServlet ; 서블릿은 일반 클래스와 다르게 상속받아야 하는 클래스가 있다. 요 글래스를 상속 받아야 서블릿 클래스가 됨
     3. 매핑 단계 : 웹상에서 접근해야 하는 주소 (닉네임을 정함)

@WebServlet(name = "HelloWorld", description = "Ex1", urlPatterns = { "/HelloWorld" })

@WebServlet("/HelloWorld")

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

// **TODO** Auto-generated method stub

System.***out***.println("Hello World~~");

response.getWriter().append("Served at: ").append(request.getContextPath());

}

패키지명 나오고, 클래스명 체크, 상속받은 클래스도 체크, 이 클래스의 닉네임 정하고, 웹브라우저에서 접근할 때 닉네임으로 접근한다. 서블릿은 doGet과 doPost 메소드 존재. doGet과 doPost안에 있는 내용이 실제로 사용자 클라이언트 웹 브라우저에 출력되거나 실행되는 부분.

1. 어노테이션을 이용한 서블릿 맵핑

@WebServlet(name = "HelloWorld", description = "Ex1", urlPatterns = { "/HelloWorld" })

@WebServlet("/HelloWorld")

예상경로 : **http://localhost:8181/ch03\_semiServlet/com.lec.ex.Ex1\_HelloWorld**

URL 맵핑 경로 : **http://localhost:8181/ch03\_semiServlet/HelloWorld**

1. Web.xml에서 서블릿 맵핑

<servlet> 서블릿 이름을 정한다

<servlet-name>HelloWorld</servlet-name>

<servlet-class>com.lec.ex.Ex1\_HelloWorld</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>HelloWorld</servlet-name>

<url-pattern>/HelloWorld</url-pattern>

</servlet-mapping>

servlet-class : 매핑할 클래스 파일명을 패키지명을 포함하여 입력

url-pattern : ‘/’로 시작해야 함.

* 1. 요청처리객체 및 응답처리객체를 톰캣에서 받음

요청처리객체(**request**객체)

WAS

Client 요청

DB

응답처리객체(**response**객체)

클라이언트 측 → WAS ; 톰캣에서 request 객체 생성. 요청을 하는 객체

클라이언트 측 ← WAS ; 톰캣에서 response 객체 생성. 응답을 하는 객체

요청을 하는 객체는 클라이언트로부터 오는 정보를 갖고 올 수 있다. 예를 들어 로그인 페이지에 들어가서 아이디와 패스워드를 넣고 버튼을 누르면( 회원이면 로그온이 되고 비회원이면 회원가입페이지나 거절페이지로 가도록 한다.

응답하는 객체에서는 회원이 맞는지 아닌지 결과값을 response객체에 실어 보낸다.

**public** **class** Ex1\_HelloWorld **extends** HttpServlet {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

**protected** **void** doGet(HttpServletRequest request, **HttpServletResponse response**) **throws**

요청처리객체 응답처리객체

ServletException, IOException {

System.***out***.println("doGet 호출");

//한글이 포함된 HTML문서를 출력하려면 HTML을 출력하기위해 response.getWriter()메서드 호출전에

// 이 문서의 내용은 HTML문법으로 작성된 텍스트고 utf-8 문자셋(한글코드)로 인코딩되어 있다는 뜻으로

// response.setContentType("text/html;charset=utf-8") 을 추가

// 응답을 html로 하고 한글은 utf-8로 인코딩하여 응답한다는 뜻

response.setContentType("text/html;charset=UTF-8");

PrintWriter out = response.getWriter(); // **웹 브라우저에 출력하기 위한 스트림**

out.println("<html>");

out.println("<head>");

out.println("<style>h1{color:red; text-align:center;}</style>");

out.println("</head>");

out.println("<body>");

out.println("<h1>Hello, World</h1>");

out.println("</body>");

out.println("</html>");

out.close();

}

}

첫페이지에서 누적할 숫자를 입력받아 두번째 페이지에서 누적할 합계를 출력하도록 구현하시오.

ex2\_sumInput.html

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

</head>

<body>

<h1>얼마까지 누적할까?</h1>

<form action=*"../Ex2\_Sum"* id=*"frm"* method=*"get"*>

<p>숫자 <input type=*"text"* name=*"num"*></p>

<p><input type=*"submit"* value=*"누적된 수 확인"*></p>

</form>

</body>

</html>

Ex2\_Sum.java

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

// **TODO** Auto-generated method stub

/\*response.getWriter().append("Served at: ").append(request.getContextPath());\*/

String numStr = request.getParameter("num");

**int** num = Integer.parseInt(numStr);

**int** sum = 0;

**for** (**int** i=1 ; i<=num ; i++)

sum += i;

response.setContentType("text/html;charset=utf-8");

PrintWriter out = response.getWriter();

out.print("<html>");

out.print("<body>");

out.print("<h1>1부터 "+num+"까지 누적합은 "+sum+"</h2>");

out.print("</body>");

out.print("</html>");

out.close();

}