[ 9 ] 포워딩(Forwarding)

1. HttpServletResponse 클래스
   * RequestDispatcher 클래스와 동일하게 요청을 위임하는 클래스.
   * RequestDispatcher 클래스와 차이점은 요청 받은 요청객체를 위임 받은 컴포넌트에 전달 하는 것이 아닌, 새로운 요청객체를 생성

클라이언트

(웹브라우저)

요청받은

컴포넌트

위임받은

컴포넌트

요청 요청위임

(request객체) (request객체)

**suInput.html**

<body>

<!-- <form action="ex1\_suPro.jsp"> -->

<!-- <form action="../ex1\_suPro"> -->

<form action=*"*<%=conPath %>*/ex1\_suPro"*>

누적을 원하는 수 <input type=*"number"* min=*"1"* name=*"su"* required=*"required"*>

<input type=*"submit"* value=*"누적"*>

</form>

</body>

**Ex1\_suPro.java**

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

**int** su = Integer.*parseInt*(request.getParameter("su"));

**int** sum = 0; // 누적변수

**for**(**int** i=1 ; i<=su ; i++){

sum += i;

}

// 1. redirect는 request객체가 새롭게 생성되면서 요청을 위임. 전달할 페이지에 파라미터 다시 넣어줌

response.sendRedirect("1\_redirect/ex1\_suView.jsp?su="+su+"&sum="+sum);

}

**suView.jsp**

<body>

<h2>1\_redirect폴더 파일</h2>

<h2>1부터 ${param.su }까지 누적합은 ${param.sum }입니다</h2>

</body>

**memberInput.jsp**

<body>

<form action=*"*<%=conPath %>*/ex2\_servlet"* method=*"post"*>

<P>ID <input type=*"text"* name=*"id"*></P>

<P>PW <input type=*"password"* name=*"pw"*></P>

<P>NAME <input type=*"text"* name=*"name"*></P>

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

</form>

</body>

**Ex2\_Servlet.java**

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

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

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

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

// 1. redirect는 request객체가 새롭게 생성되면서 요청을 위임. 전달할 페이지에 파라미터 다시 넣어줌

// url에 한글을 전달시 encodeing 처리해 줘야 함

response.sendRedirect("1\_redirect/ex2\_memberView.jsp?id="+id+"&pw="+pw

+"&name="+URLEncoder.*encode*(name,"utf-8") );

}

**memberView.jsp**

<body>

<h1>회원정보(ex2\_memberView.jsp)</h1>

<h3>ID : ${param.id }</h3>

<h3>PW : ${param.pw }</h3>

<h3>name : ${param.name }</h3>

</body>

1. RequestDispatcher 클래스 ; 내가 받은 요청을 내가 처리하지 않고 다른 쪽으로 위임해 버릴 때 그것을 포워드 한다고 한다. 그것을 JSP 서블릿에서는 두가지 클래스를 많이 이용한다.
   * 서블릿 또는 JSP에서 요청을 받은 후 다른 콤포넌트로 요청을 위임 할 수 있다. 그리고 이러한 위임 방법에는 2개의 클래스를 이용 합니다.
   * 하나는 RequestDispatcher 클래스 이고, 또하나는 HttpServletResponse 클래스
   * RequestDispatcher클래스의 경우 요청 받은 **요청객체(request)를 위임하는 컴표넌트에 동일하게 전달**할 수 있다.

클라이언트

(웹브라우저)

요청받은

컴포넌트

위임받은

컴포넌트

요청 요청위임

(request객체) (request객체)

**suInput.html**

<body>

<!-- <form action="ex1\_suPro.jsp"> -->

<!-- <form action="../ex1\_suPro"> -->

<form action=*"*<%=conPath %>*/ex1\_suPro"*>

누적을 원하는 수 <input type=*"number"* min=*"1"* name=*"su"* required=*"required"*>

<input type=*"submit"* value=*"누적"*>

</form>

</body>

**Ex1\_suPro.java**

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

**int** su = Integer.*parseInt*(request.getParameter("su"));

**int** sum = 0; // 누적변수

**for**(**int** i=1 ; i<=su ; i++){

sum += i;

}

// 1. redirect는 request객체가 새롭게 생성되면서 요청을 위임. 전달할 페이지에 파라미터 다시 넣어줌

// response.sendRedirect("1\_redirect/ex1\_suView.jsp?su="+su+"&sum="+sum);

// 2. dispatcher에서 view 단으로 보여질 객체를 request 객체에 담는다

request.setAttribute("sum", sum);

RequestDispatcher dispatcher = request.getRequestDispatcher("2\_dispatcher/ex1\_suView.jsp");

dispatcher.forward(request, response);

}

**suView.jsp**

<body>

<h2>2\_dispatcher 폴더 파일</h2>

<h2>1부터 ${su }까지 누적합은 ${sum }입니다</h2>

</body>

**memberInput.jsp**

<body>

<form action=*"*<%=conPath %>*/ex2\_servlet"* method=*"post"*>

<P>ID <input type=*"text"* name=*"id"*></P>

<P>PW <input type=*"password"* name=*"pw"*></P>

<P>NAME <input type=*"text"* name=*"name"*></P>

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

</form>

</body>

**Ex2\_Servlet.java**

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

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

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

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

// 1. redirect는 request객체가 새롭게 생성되면서 요청을 위임. 전달할 페이지에 파라미터 다시 넣어줌

// url에 한글을 전달시 encodeing 처리해 줘야 함

// response.sendRedirect("1\_redirect/ex2\_memberView.jsp?id="+id+"&pw="+pw

// +"&name="+URLEncoder.encode(name,"utf-8") );

// 2. dispatcher에서 view 단으로 보여질 객체를 request 객체에 담는다

Member member = **new** Member(id, pw, name);

request.setAttribute("member", member);

RequestDispatcher dispatcher = request.getRequestDispatcher("2\_dispatcher/ex2\_memberView.jsp");

dispatcher.forward(request, response);

}

**memberView.jsp**

<body>

<h1>회원정보(ex2\_memberView.jsp)</h1>

<h3>ID : ${member.id }</h3>

<h3>PW : ${member.pw }</h3>

<h3>name : ${member.name }</h3>

</body>

* Request 객체는 파라미터에 값을 넣어 전달하기도 하고 attribute에 set하여 전달하기도 한다 ※