[0630]SCOPE

scope: 객체 전달 범위

1. page: 현재 페이지에서 객체를 전달 (현재 페이지의 객체 유지)

2. request : 현재 페이지에서 요청한 다음 페이지까지만 객체 전달 (요청된 페이지까지 객체 유지)

3. session: 현재 Browser 에서 context 내에 있는 모든 페이지에서 객체 유지

4. application : 현재 context 내에서 객체 유지. Web.xml 에서도 사용 가능

공통메소드: setAttribute, getAttribute, removeAttribute, getAttribute

ScopController.java

```
package com.scope.controller;
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletContext;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import\ javax.servlet.http. Http Servlet Response;
import javax.servlet.http.HttpSession;
@WebServlet("/ScopeController")
public class ScopeController extends HttpServlet {
 private static final long serialVersionUID = 1L;
 doPost(request, response);
 request.setCharacterEncoding("UTF-8");
   response.setContentType("text/html; charset=UTF-8");
   String requestId = (String) request.getAttribute("requestId");
   System.out.println("requestId: "+requestId);
   HttpSession session = request.getSession();
   String sessionId = (String) session.getAttribute("sessionId");
   System.out.println("sessionId : "+sessionId);
   ServletContext application = getServletContext();
   String applicationId = (String) application.getAttribute("applicationId");
   System.out.println("applicationId : "+applicationId);
   PrintWriter out = response.getWriter()
       .append("<h1>Controller에서 만들어준 table</h1>"
          + ""
           + ""
           + "requestId"
           + ""+requestId+""
           + ""
           + ""
           + "sessionId"
           + ""+sessionId+""
           + ""
           + ""
           + "applicationId"
           + ""+sessionId+"
           + ""
           + "");
```

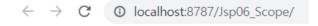
index.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"</pre>
   pageEncoding="UTF-8"%>
<%
 request.setCharacterEncoding("UTF-8");
    response.setContentType("text/html; charset=UTF-8");
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<%
  pageContext.setAttribute("pageId", "my pageId value!!");
request.setAttribute("requestId", "my requestId value!!");
session.setAttribute("sessionId", "my sessionId value!!"); // session은 invalidate전까지
  application.set Attribute ("applicationId", "my applicationId value!!");\\
%>
  <h1>index</h1>
  pageId : <%=pageContext.getAttribute("pageId") %> <br/>
  requestId : <%=request.getAttribute("requestId") %> <br/>sessionId : <%=session.getAttribute("sessionId") %><br/>
  <a href="result.jsp?param=myrequestparamvalue">result...</a>
 <br>
  <a href="controller.do">controller@</a>
   <% pageContext.forward("controller.do"); %>
</body>
</html>
```

result.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"</pre>
   pageEncoding="UTF-8"%>
<%
 request.setCharacterEncoding("UTF-8");
   response.setContentType("text/html; charset=UTF-8");
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
  <h1>result <a></h1></h1>
  pageId : <%=pageContext.getAttribute("pageId") %> <br/>
  requestId : <%=request.getAttribute("requestId") %> <br/>sessionId : <%=session.getAttribute("sessionId") %><br/>
  applicationId : <%=application.getAttribute("applicationId") %><br/>
  requestParam : <%=request.getParameter("param") %> <br/>
</body>
</html>
```

index페이지에 마지막 pageContext.forward("controller.do") 를 주석처리 하고 실행시키면



index

pageId: my pageId value!! requestId: my requestId value!! sessionId: my sessionId value!!

applicationId: my applicationId value!!

result...
controller

모든 값이 유지가 잘 되는 것을 볼 수 있다.

result 를 클릭했을때 과연 모든 값들이 다 유지가 될까?

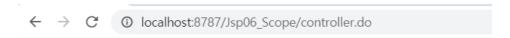
처음 생각으로선 pageld : null 을 제외한 나머지는 다 값이 있을 거라고 생각했다. 하지만



param값은 location.href 뒷부분 url에 name : value 로 값을 넘겨줬기 때문에 잘 나오고 session, application 은 페이지가 얼마가 이동이 되든 값(객체)를 가지고 다니기 때문에 잘 나오고

requestId는 왜 안나올까?

request는 요청한 페이지의 다음 페이지까지만 객체가 유지가 되는데 index.jsp → web.xml → result.jsp 로 가기 때문에 객체가 휘발이 되어 버린다. 마찬가지로 index.jsp로 돌아와서 controller를 클릭해보면



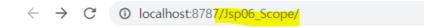
Controller에서 만들어준 table

requestId	null
sessionId	my sessionId value!!
applicationId	my sessionId value!!

index.jsp→ web.xml → <u>ScopController.java</u> 로 와서 PrintWriter로 response를 해주는데 테이블을 같이 만들어 값을 보여준다. requestId이 null 인것을 볼 수가 있다.

그렇다면 requestId가 나오게 request의 객체를 어떻게 유지를 시킬까? → forward를 써야한다.

index.jsp에서 주석처리해준 pageContext.forward("controller.do") 를 주석해제 하고 다시 실행



Controller에서 만들어준 table

requestId	my requestld value!!
sessionId	my sessionId value!!
applicationId	my sessionId value!!

 $index.jsp \rightarrow web.xml \rightarrow \underline{ScopController.java}$

많은 페이지 이동이 있어도 request를 포함한 모든 객체가 유지가 된다. forward만세 (같은 serlvet을 사용한다고 해도 href 처럼 바로 url-pattern을 사용해 보내주느냐 forward에 url-pattern을 사용해 사용해 보내주느냐의 차이는 주소창에 보내준 곳의 이름이 나오느냐.안나오느냐의 차이다)