Attributes in Servlets

Place holder to transfer the data between resources

Three different types of attributes:

1. Request
   1. Request attributes are available in the memory till the response to that request is sent to the client or another resource.
2. Session
   1. Session attributes are available till the session of the user is destroyed.
3. Context
   1. Context attributes are available till the web server is shutdown.

**package** com.sssit.attrd.servlets;

**import** java.io.IOException;

**import** javax.servlet.RequestDispatcher;

**import** javax.servlet.ServletContext;

**import** javax.servlet.ServletException;

**import** javax.servlet.http.HttpServlet;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

**import** javax.servlet.http.HttpSession;

/\*\*

\* Servlet implementation class First

\*/

**public** **class** First **extends** HttpServlet {

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

/\*\*

\* **@see** HttpServlet#HttpServlet()

\*/

**public** First() {

**super**();

// **TODO** Auto-generated constructor stub

}

/\*\*

\* **@see** HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

/\* Setting the value to request attribute \*/

request.setAttribute("reqA", "reqV");

/\* Setting the value to session Attribute \*/

HttpSession userSession = request.getSession();

userSession.setAttribute("sessA", "sessV");

/\* Setting the value to Context Attribute \*/

ServletContext context = getServletContext();

context.setAttribute("ctxtA", "ctxtV");

RequestDispatcher rd = request.getRequestDispatcher("second");

rd.include(request, response);

}

}

**package** com.sssit.attrd.servlets;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** javax.servlet.ServletContext;

**import** javax.servlet.ServletException;

**import** javax.servlet.http.HttpServlet;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

**import** javax.servlet.http.HttpSession;

/\*\*

\* Servlet implementation class Second

\*/

**public** **class** Second **extends** HttpServlet {

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

/\*\*

\* **@see** HttpServlet#HttpServlet()

\*/

**public** Second() {

**super**();

// **TODO** Auto-generated constructor stub

}

/\*\*

\* **@see** HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

/\* Read the value from Request \*/

String reqAttributeValue = (String)request.getAttribute("reqA");

/\* Read the value from Session \*/

HttpSession userSession = request.getSession();

String sessAttributeValue = (String)userSession.getAttribute("sessA");

/\* Read the value from Context \*/

ServletContext context = getServletContext();

String ctxtAttributeValue = (String)context.getAttribute("ctxtA");

PrintWriter pw = response.getWriter();

pw.println("Request Attribute Value = " + reqAttributeValue);

pw.println("Session Attribute Value = " + sessAttributeValue);

pw.println("Context Attribute Value = " + ctxtAttributeValue);

}

}

Web.xml

<servlet>

<description></description>

<display-name>First</display-name>

<servlet-name>First</servlet-name>

<servlet-class>com.sssit.attrd.servlets.First</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>First</servlet-name>

<url-pattern>/first</url-pattern>

</servlet-mapping>

<servlet>

<description></description>

<display-name>Second</display-name>

<servlet-name>Second</servlet-name>

<servlet-class>com.sssit.attrd.servlets.Second</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>Second</servlet-name>

<url-pattern>/second</url-pattern>

</servlet-mapping>

Different CURD operations on any web application:

C – Create a data in DB

U – Update the data in DB

R – Retrieve the data from DB

D – Delete the data from DB

Required Dependencies:

1. Servlet-api.jar 🡪 for running the servlets
2. Ojdbc6.jar 🡪 to connect with the Oracle DB

Place both the dependencies in src🡪 main🡪webapp🡪WEB-INF🡪lib folder



