1. **Two servlets share data in Session scope ………..**
2. In this application, there are two servlets, by name “SourceServlet” and “TargetServlet”.
3. Create a html form “user.html” with one text field which takes user name. action value should be the <url-pattern> name of the “SourceServlet”.
4. In  **“**SourceServlet **”,** get the user name from html form and set the username into session scope using setAttribute() method of session object. Create a hyper link in this page using anchor tag as below:

out.println(“<a href=./target>click here to get the user name</a>”);

where out is the reference pointing to writer object.

1. In “TargetServlet”, get the HttpSession object using getSession() method of request object, and using this session object, get the attribute value using getAttribute() method and store in some String variable.
2. And finally print the user name on the browser using writer object.
3. Handle the exceptions properly and deploy the application.
4. **URL rewriting implementation………..**
5. In the above example, modify the “SourceServlet” as follows:

Instead of the following code,

out.println(“<a href=./target>click here to get the user name</a>”);

write as follows:

String newURL=response.encodeURL(“./target”);

out.println(“<a href=”+newURL+” >click here to get the user name</a>”);

1. **Using Cookies to save session Data..**

Write a program to create a servlet ‘AddCookieServlet’ which extends HttpServlet .

Inside the overridden doGet() method get the session using getSession() method of HttpServletRequest class. Now get the session id of this session using getId() method of HttpSession and store it in a String ‘sessionId’. Now create a object ‘cookie ’ of type Cookie ,passing appropriate messages into its constructor (say new Cookie(“sessionid”,sessionId)) . Now add this Cookie info into response using addCookie() method of HttpSession by passing the cookie object as a parameter to it. Display a message cookie added successfully using PrintWriter.

Now write a servlet program to create a servlet class ‘RetrieveCookieServlet’ which extends HttpServlet . Inside the Overridden doGet() method retrieve the session Id and print it in the console as following …

Cookie[] cookies = request.getCookies();

**for**(Cookie cookie : cookies){

**if**(cookie.getName().equals("sessionid")){

String sessionId = cookie.getValue();

System.*out*.println(sessionId);

}

}

1. **Invalidate Users Session..**

Write a program to create a Servlet class ‘InvalidateSessionServlet’ which extends HttpServlet .

Inside the Overridden doGet() method , get the session of this servlet using getSession method of request method. Now invalidate the session using invalidate() method of HttpSession.

Get an HttpSession related to this request, if no session exist don't

create a new one(To do this pass ‘false’ as parameter to the getSession() method). Then just check after invalidation the session is null or not by printing the session object .