Ex:4. Write programs in Java using Servlets:

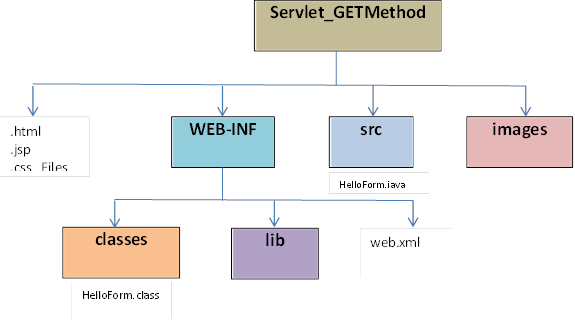
1. To invoke servlets from HTML forms
2. Session tracking using hidden form fields and Session tracking for a hit count

i.To invoke servlets from HTML forms

Aim:

To write programs in Java to invoke servlets from HTML forms

1. Create a directory structure under Tomcat and inside webapps for your application named directory **Servlet\_GETMethod**.



1. Design index.html file to submit the First and Last name of a user to the servlet Helloworld.java which was URL name mapped to /welcome (web.xml)by using HTTP GET method and place it in the directory **Servlet\_GETMethod**.
2. Write the servlet source code in this case **HelloForm.java** and place it in **src** folder. You need to import the javax.servlet package and the javax.servlet.http package in your source file. Include the code to get the input from index file by getParameter() method and generate a dynamic html file to print the input obtained from index file.
3. Compile your source code and place the **HelloForm.class** inside **classes** folder.
4. Create a deployment descriptor **web.xml** inside **WEB-INF.** Specify name of the servlet, class name of servlet and URL name mapping of the servlet as **“/welcome**”
5. Set the path variable to **“servlet-api.jar**” in **Tomcat lib**. If not there download and place it.
6. Run Tomcat.
7. Type localhost:8080/ **Servlet\_GETMethod/** in the browser to fetch the .index page.

i)

**Index.html**

<html>

<body>

<h1>EXAMPLE OF SERVLET</h1>

<BR/><BR/>

<form action = "./welcome" method = "GET">

First Name: <input type = "text" name = "first\_name">

<br />

Last Name: <input type = "text" name = "last\_name" />

<input type = "submit" value = "Submit" />

</form>

</body>

</html>

**HelloForm**

// Import required java libraries

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

// Extend HttpServlet class

public class HelloForm extends HttpServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

// Set response content type

response.setContentType("text/html");

PrintWriter out = response.getWriter();

String title = "Using GET Method to Read Form Data";

String docType =

"<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";

out.println(docType +

"<html>\n" +

"<head><title>" + title + "</title></head>\n" +

"<body bgcolor = \"#f0f0f0\">\n" +

"<h1 align = \"center\">" + title + "</h1>\n" +

"\n" +

" <b>First Name</b>: "

+ request.getParameter("first\_name") + "\n" +

" <b>Last Name</b>: "

+ request.getParameter("last\_name") + "\n" +

"\n" +

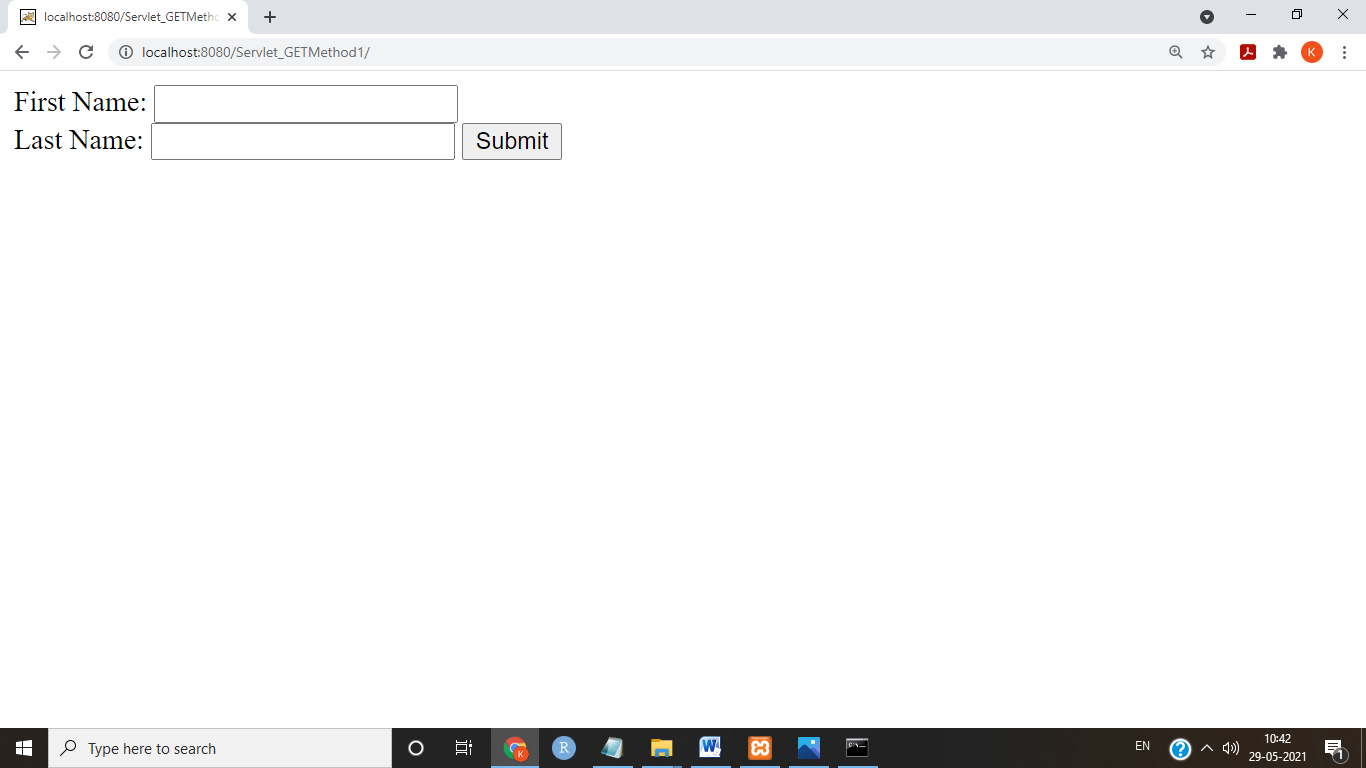
"</body>" +

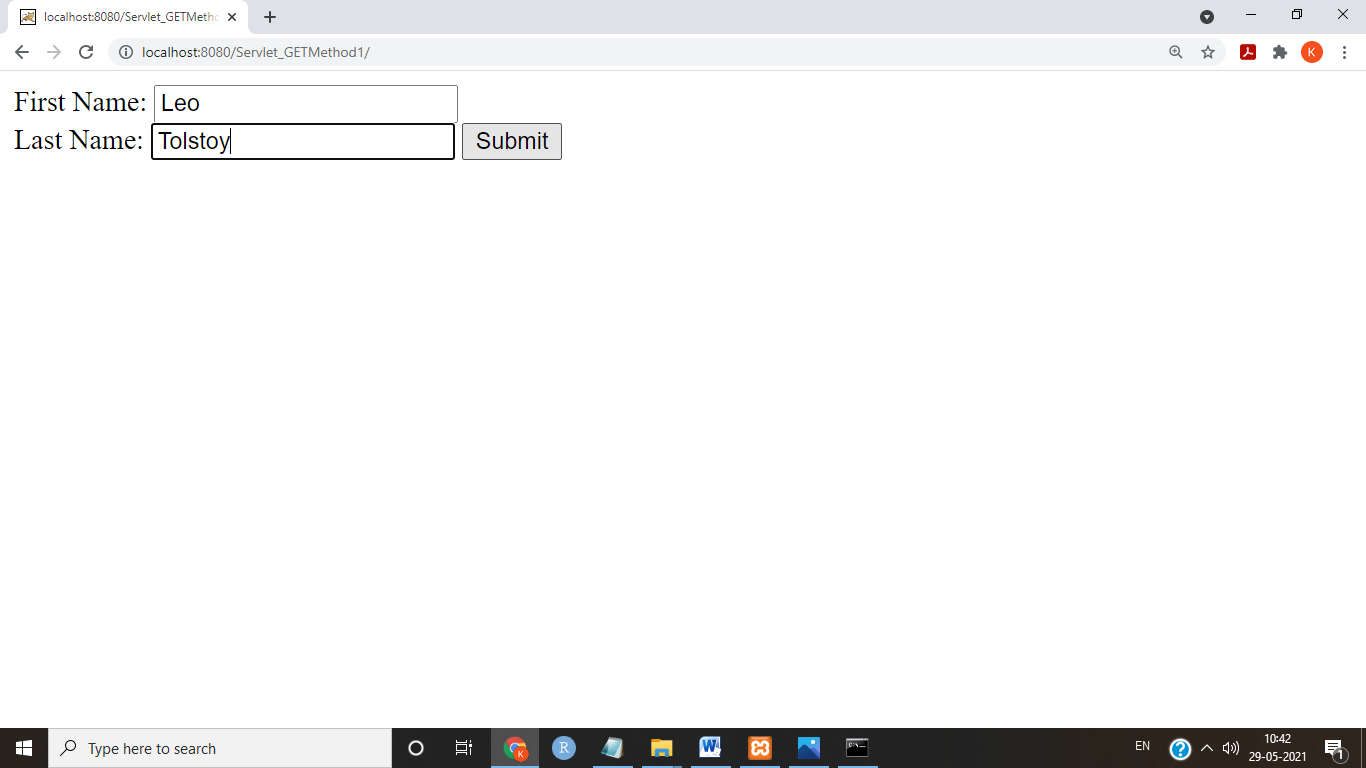
"</html>"

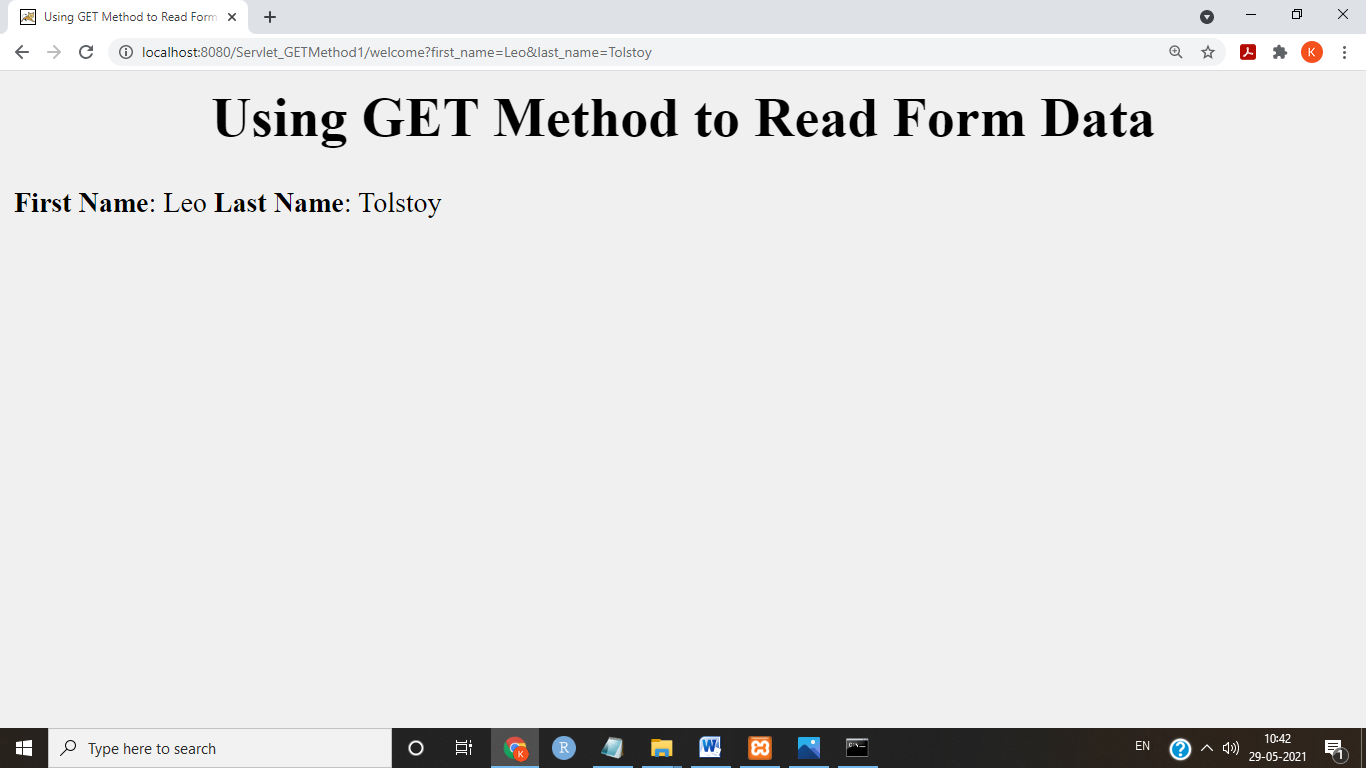
);

}

}

**Output: **

****

****

**Result:**

Thus servlet is invoked from HTML forms.

ii.Session tracking using hidden form fields and Session tracking for a hit count

a)Session tracking using hidden form fields

Aim:

To write programs in Java for Session tracking using hidden form fields

Algorithm

* + - 1. Create a directory structure under Tomcat and inside webapps for your application named directory **Servlet\_Hidden\_Field1.**
      2. Create index.html with a form when submitted takes to the first servlet FirstServlet.java.(url pattern - ="./servlet1) using HTTP GET Method
      3. Write the servlet source code in this case **FirstServlet.java** and place it in **src** folder.. Include the code to get the input from index file by getParameter() method. Store the name in a variable and set it as a default value to an input type-‘hidden’. Generate a dynamic html file to print the input obtained from index file
      4. Write the servlet source code in this case **SecondServlet.java** and place it in **src** folder.. Include the code to get the input from hidden field of **FirstServlet.java** by getParameter() method. Generate a dynamic html file to print the input obtained from **FirstServlet.java.**
      5. Compile your source code and place the class files of **FirstServlet.java and SecondServlet.java** and inside **classes** folder

6.Create a deployment descriptor **web.xml** inside **WEB-INF.** Specify name of the servlets, class name of servlets and URL name mapping of the servlet as **“/servlet1**” and **“/servlet2**”

7. Set the path variable to **“servlet-api.jar**” in **Tomcat lib**. If not there download and place it.

8. Run Tomcat.

9.Type localhost:8080/ **Servlet\_Hidden\_Field1/** in the browser to fetch the .index page.

**index.html**

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<body>

<center>

<h2>Click the Submit button to check the hidden field</h2>

<form action="./servlet1" method="GET">

Name:<input type="text" name="userName"/>

<br/><br/>

<input type="submit" value="Go"/>

</form>

</body>

</html>

**FirstServlet.java**

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

public class FirstServlet extends HttpServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response){

try{

response.setContentType("text/html");

PrintWriter out = response.getWriter();

String n=request.getParameter("userName");

out.print("Welcome "+n);

//creating form that have invisible textfield

out.print("<form action='servlet2'>");

out.print("<input type='hidden' name='hiddenfield' value='"+n+"'>");

out.print("<input type='submit' value='Submit'>");

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

out.close();

}catch(Exception e){System.out.println(e);}

}

}

**SecondServlet.java**

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

public class SecondServlet extends HttpServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response){

try{

response.setContentType("text/html");

PrintWriter out = response.getWriter();

//Getting the value from the hidden field

String n=request.getParameter("hiddenfield");

out.print("Hello "+n);

out.close();

}catch(Exception e){System.out.println(e);}

}

}

**web.xml**

<web-app>

<servlet>

<servlet-name>s1</servlet-name>

<servlet-class>FirstServlet</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>s1</servlet-name>

<servlet-class>FirstServlet</servlet-class>

<url-pattern>/servlet1</url-pattern>

</servlet-mapping>

<servlet>

<servlet-name>s2</servlet-name>

<servlet-class>SecondServlet</servlet-class>

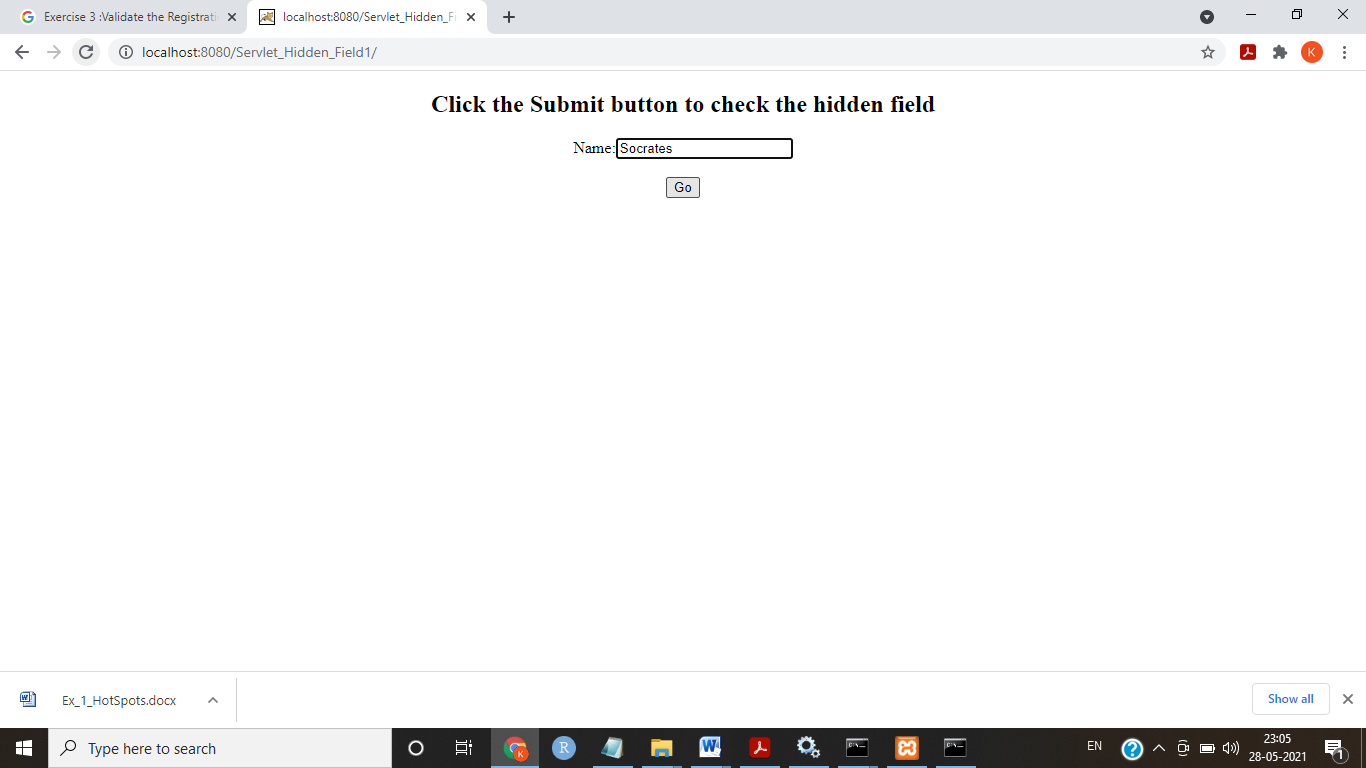
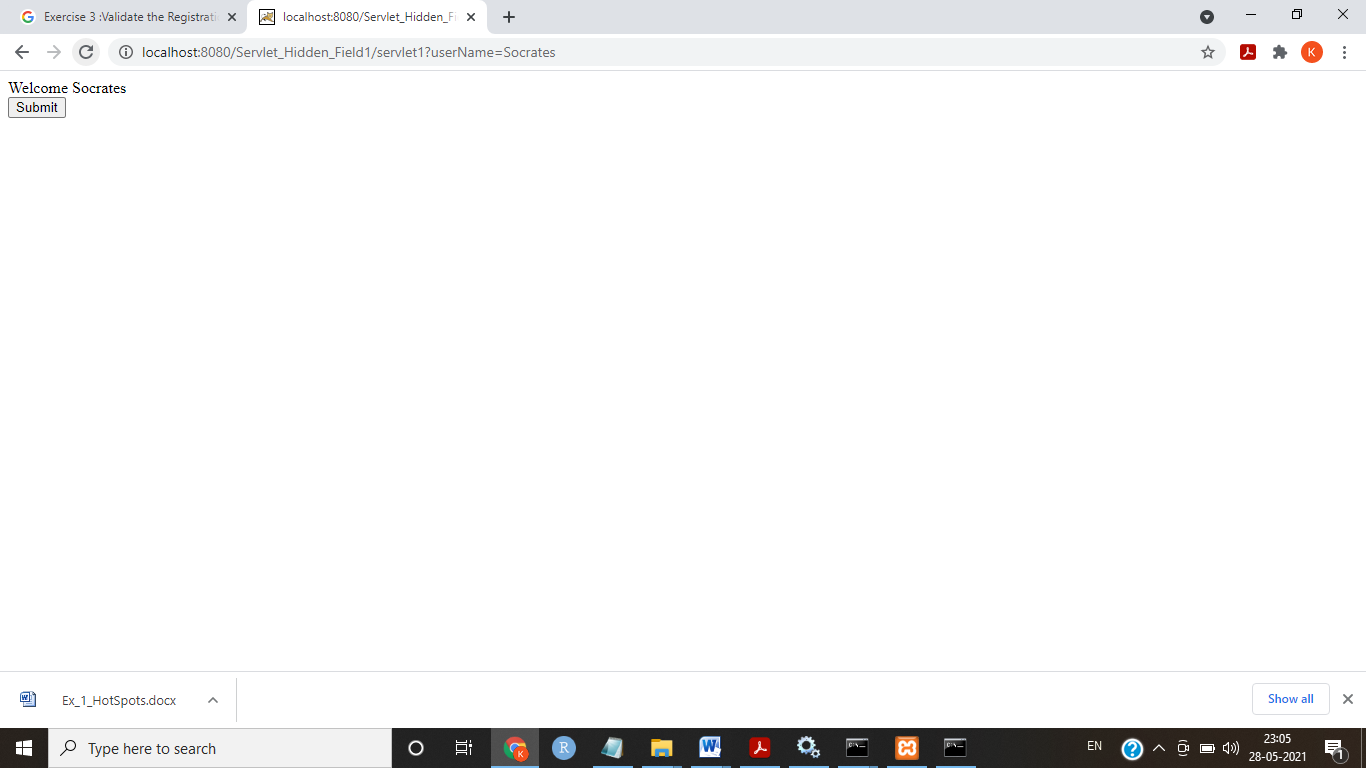
</servlet>

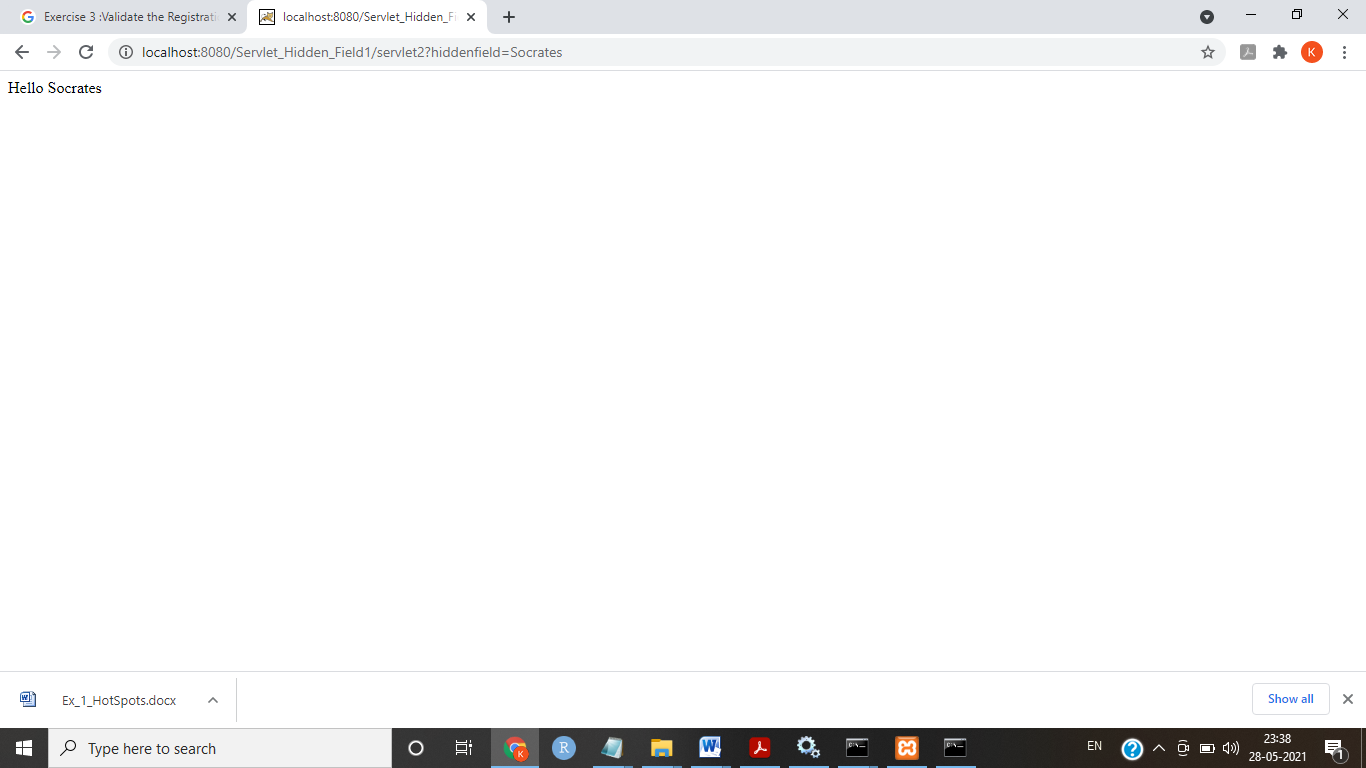
<servlet-mapping>

<servlet-name>s2</servlet-name>

<url-pattern>/servlet2</url-pattern>

</servlet-mapping> </web-app>

**Output:**  



**Result:**

Thus the servlet for session tracking using hidden fields is executed.

b) Session tracking for a hit count

Aim:

To write programs in Java for Session tracking using Hit Count.

Algorithm

1. Create a directory structure under Tomcat and inside webapps for your application named directory **Servlet\_Session\_Tracking1.**
2. Write the servlet source code in this case **PageHitCounter.java** and place it in **src** folder. Include the code to initialize a counter variable as zero in init()method. Increment the counter variable to 1 in doGet() method. Generate a dynamic html file to print the value of counter variable.
3. Compile your source code and place the class files of **PageHitCounter.java and SecondServlet.java** and inside **classes** folder

6.Create a deployment descriptor **web.xml** inside **WEB-INF.** Specify name of the servlets, class name of servlets and URL name mapping of the servlet as “/PageHitCounter”.

7. Set the path variable to **“servlet-api.jar**” in **Tomcat lib**. If not there download and place it.

8. Run Tomcat.

9.Type “localhost:8080/ **Servlet\_Session\_Tracking1/** **PageHitCounter”** in the

browser to run the Servlet.

10.Whenver the page gets refreshed the counter variable gets incremented by count 1.

**PageHitCounter.java**

import java.io.\*;

import java.sql.Date;

import java.util.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

public class PageHitCounter extends HttpServlet {

private int hitCount;

public void init() {

// Reset hit counter.

hitCount = 0;

}

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

// Set response content type

response.setContentType("text/html");

// This method executes whenever the servlet is hit

// increment hitCount

hitCount++;

PrintWriter out = response.getWriter();

String title = "Total Number of Hits";

String docType = "<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";

out.println(docType +

"<html>" +"\n"+

"<head><title>" + title + "</title></head>"+"\n" +

"<body bgcolor = \"#f0f0f0\">"+"\n" +

"<h1 align = \"center\">" + title + "</h1>"+"\n" +

"<h2 align = \"center\">" + hitCount + "</h2>"+"\n" +

"</body>"+"\n"+

"</html>"

);

}

public void destroy() {

// This is optional step but if you like you

// can write hitCount value in your database.

}

}

**web.xml**

<web-app>

<servlet>

<servlet-name>PageHitCounter</servlet-name>

<servlet-class>PageHitCounter</servlet-class>

</servlet>

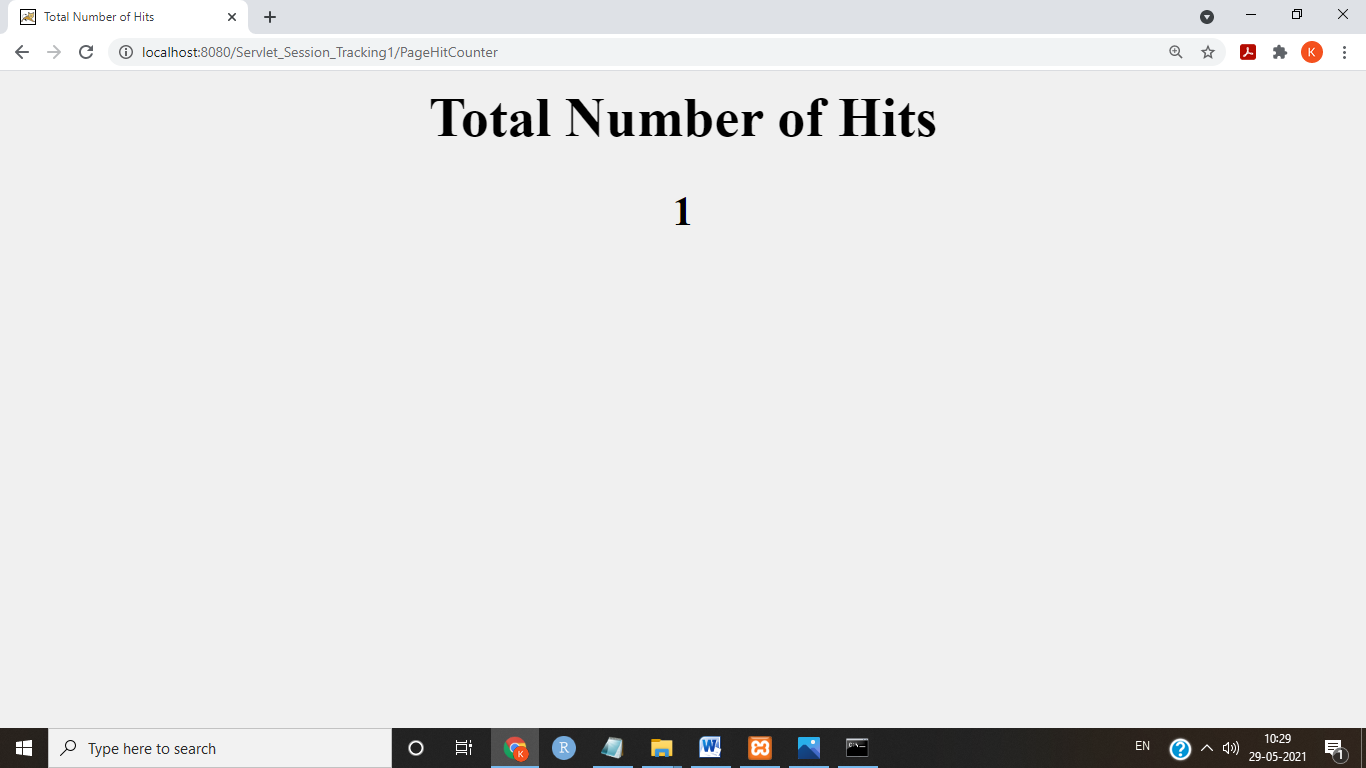
<servlet-mapping>

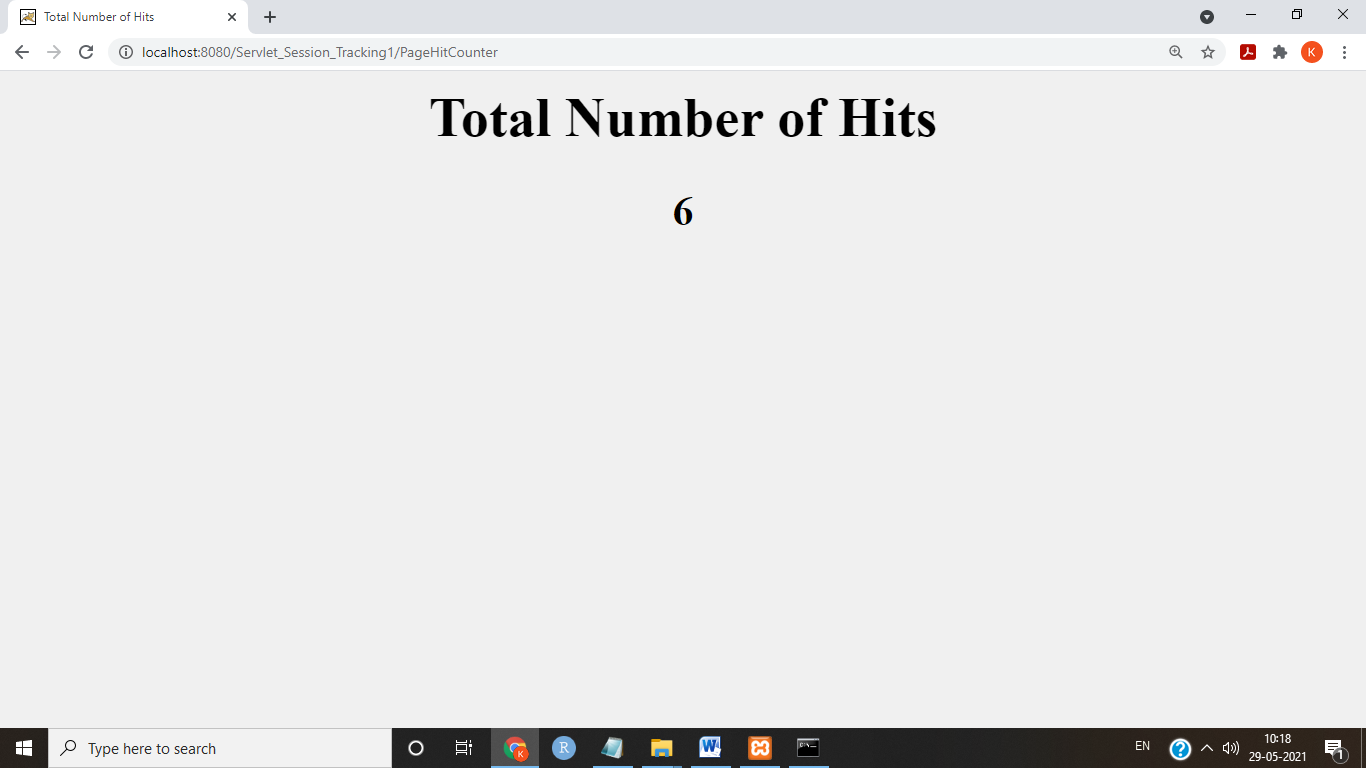
<servlet-name>PageHitCounter</servlet-name>

<url-pattern>/PageHitCounter</url-pattern>

</servlet-mapping>

</web-app>

Output: 



**Result:**

Thus the servlet for session tracking Page Hit Counter is executed.