



Universidad Autónoma de Nuevo León  
FACULTAD DE CIENCIAS FISICO MATEMATICAS



## **Laboratorio 2 Servlet**

Alumno: Jose Alejandro Martinez Rivera

Matricula: 1680723 Grupo: 006

Maestro: Miguel Salazar

Monterrey, Nuevo León, Lunes 06 de Noviembre del 2017

## Practica laboratorio 2 Uso de servlet

En esta práctica crearemos y haremos uso de un servlet, para ver el funcionamiento y el comportamiento de este.

Iniciamos con la creación de un documento html para crear nuestra página, donde agregaremos un párrafo describiendo lo que vamos hacer en la página lo cual es el crear un servlet de manera sencilla, usamos la etiqueta “form” la cual su action la llamaremos “SaludoServlet” , dentro del “form” pediremos ingresar un “Nombre” y por su parte agregamos un botón del tipo “submit” para poder enviar la información ingresada en el “form”, en la parte inferior agregamos nuestro nombre completo, a continuación nuestro código :

```
11 <body>
12
13 <header>
14 <h1>Laboratorio 1</h1>
15 </header>
16
17 <p>En esta pagina crearemos un servlet de manera sencilla
18 </p>
19 <form action="SaludoServlet">
20
21 <input type="text" name="nombre"/>
22 <input type="submit" value="Enviar"/>
23
24 </form><br>
25
26 <h2>Jose Alejandro Martinez Rivera</h2>
27
28 </body>
```

Por parte del cliente este es aspecto de la pagina:

localhost:8080/Lab\_1/

### Laboratorio 1

En esta pagina crearemos un servlet de manera sencilla

Jose Alejandro Martinez Rivera

## Servlet

Creamos un servlet con el nombre de “SaludoServlet”, en el cual al momento de crearlo se crea un documento con extensión “.java” donde el cual se crea primeramente un clase con el nombre que asignamos que extiende de la librería “HttpServlet” :

```
18 public class SaludoServlet extends HttpServlet {
19
20     /**
21      * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
22      * methods.
23      *
24      * @param request servlet request
25      * @param response servlet response
26      * @throws ServletException if a servlet-specific error occurs
27      * @throws IOException if an I/O error occurs
28      */
```

Mas abajo en el código se crea un método llamado "processRequest" el cual cuenta con varios parámetros y demás código par el buen funcionamiento del servlet.

Establecemos una variable nombre donde le asignamos mediante el objeto request y obtener parámetro donde "cacha" el valor que se asigna al "input" que creamos en el html, mediante el "name" del input, que en este caso le dimos el valor de "nombre".

```
29     protected void processRequest(HttpServletRequest request, HttpServletResponse response)
30     {
31         throws ServletException, IOException {
32             response.setContentType("text/html;charset=UTF-8");
33             try (PrintWriter out = response.getWriter()) {
34                 /* TODO output your page here. You may use following sample code. */
35
36                 String nombre= request.getParameter("nombre");
```

Ya por ultimo solo agregamos la variable creada anteriormente en print correspondiente para poder mostrar un saludo seguido el nombre ingresado por el cliente en nuestra página html.

```
34
35
36         String nombre= request.getParameter("nombre");
37
38         out.println("<!DOCTYPE html>");
39         out.println("<html>");
40         out.println("<head>");
41         out.println("<title>Hola</title>");
42         out.println("</head>");
43         out.println("<body>");
44         out.println("<h1> hola " + nombre + "</h1>");
45         out.println("</body>");
46         out.println("</html>");
47     }
```

Ya ingresado un nombre y enviado el formulario la página contesta de la siguiente manera del lado del cliente:



## Reflexión

Al momento de ingresar código JS la página no tiene ningún cambio, pero el navegador en la consola marca un error donde está detectando el ingreso de un script y lo bloquea es por eso que no ocurre nada en la vista cliente. Desde el punto de vista de seguridad esto es muy riesgoso ya que personas con malas intenciones pueden ingresar estos scripts y podrían cambiar el funcionamiento de la página y el robo de información de los servidores. Para poder evitar esto creo yo que sería el crear condiciones para autenticar lo que se está ingresando.

Al final del URL aparece el texto ingresado (el nombre) si lo cambiamos se modifica el texto que mostramos por lo que ponemos en el URL. Esto es muy riesgoso ya que desde el URL se podría modificar los valores que mostraremos.

Para evitar esto cambiaremos el método del form por un "get" ya que por default viene el post.