

# COMENZAMOS CON APLICACIONES WEB

Edgar Nuñez Arana

# TABLA DE CONTENIDO

<b>EXPLICACION SOBRE EL FUNCIONAMIENTO DEL PROGRAMA.....</b>	<b>2</b>
<b>CLASES DE NUESTRO PROGRAMA .....</b>	<b>2</b>
<b>EXPLICACION DE CODIGO .....</b>	<b>4</b>
 <b>INVESTIGA SOBRE LOS SIGUIENTES TEMAS Y PROPORCIONA UNA RESPUESTA CORTA (NO MÁS DE 50 PALABRAS CADA UNO):.....</b>	<b>5</b>
a.    ¿Cuál fue la parte más compleja de esta actividad? ¿Cómo lograste resolverla? .....	5
b.    ¿Puedes encontrar otro servidor de aplicaciones y compararlo con Tomcat? .....	6

## EXPLICACION SOBRE EL FUNCIONAMIENTO DEL PROGRAMA

### CLASES DE NUESTRO PROGRAMA

Primeramente crearemos la clase Activity\_jsp que será la clase la cual interactua con el usuario, en esta podremos ver el form con toda la información que deseemos que el usuario pueda ver e interactuar con él.

```
1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2   pageEncoding="ISO-8859-1"%>
3 <%@ page import="java.util.Date"%>
4 <!DOCTYPE html>
5 <html>
6 <head>
7   <meta charset="ISO-8859-1">
8   <title>Activity 3</title>
9 </head>
10 <body>
11   <p>CALCULA TU DNI!!</p>
12   <div></div>
13   <form method="post" action="MyServlet">
14     Introduzca la letra de su dni(Introduzca su dni sin letra): <br/>
15     <input type="text" name="dniIntroducido" id="dniIntroducido"/> <br/>
16     <input type="submit" value="Vamos allá"/>
17   </form>
18
19 </body>
20 </html>
```

Posteriormente tendremos nuestro servlet que será el encargado de desarrollar la aplicación web basadas en el protocolo HTTP es decir que nos abrirá una pagina en el navegador con la información de la clase jsp y podremos implementar las funciones creadas en este servlet.

```
import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

@WebServlet("/MyServlet")
public class MyServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    private static final String letrasDisposs = "TRWAGMYFPDXBNJZSQVHLCKE";

    public MyServlet() {
        super();
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<h1>Calculadora de DNI</h1>");
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        String dniValuee = request.getParameter("dniIntroducido");

        try {
            int dni = Integer.parseInt(dniValuee);
            int resto = dni % 23;
            char letra = letrasDisposs.charAt(resto);

            response.setContentType("text/html");
            PrintWriter out = response.getWriter();
            out.println("<html><body>");
            out.println("<h1>DNI CALCULADO</h1>");
            out.println("<p>El DNI completo es: " + dniValuee + letra + "</p>");
            out.println("<a href='Actividad_2.jsp'>Volver</a>");
            out.println("</body></html>");
        }
        catch (NumberFormatException e)
        {
            response.setContentType("text/html");
            PrintWriter out = response.getWriter();
            out.println("<html><body>");
            out.println("<h1>Error</h1>");
            out.println("<p>El número introducido no es válido. Por favor, introduce solo números.</p>");
            out.println("<a href='Actividad_2.jsp'>Volver</a>");
            out.println("</body></html>");
        }
    }
}
```

## EXPLICACION DE CODIGO

## CALCULA TU DNI!!

Introduzca la letra de su dni(Introduzca su dni sin letra):

72360631

Vamos allá

```

1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2 pageEncoding="ISO-8859-1"%>
3 <%@ page import="java.util.Date"%>
4 <!DOCTYPE html>
5 <html>
6 <head>
7 <meta charset="ISO-8859-1">
8 <title>Activity 3</title>
9 </head>
10 <body>
11 <p>CALCULA TU DNI!!</p>
12 <div></div>
13 <form method="post" action="MyServlet">
14     Introduzca la letra de su dni(Introduzca su dni sin letra): <br/>
15     <input type="text" name="dniIntroducido" id="dniIntroducido"/> <br/>
16     <input type="submit" value="Vamos allá"/>
17 </form>
18 </body>
19 </html>

```

Para crear este form usaremos la etiqueta form y las etiquetas input para el hueco para la respuesta, hay que tener en cuenta de asignarles id para su referencia en el servlet posterior

## DNI CALCULADO

El DNI completo es: 72360631D

[Volver](#)

```

@WebServlet("/")
public class MyServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    private static final String letrasDisposs = "TRWAGMYFPDXBNJZSQVHLCKE";

    public MyServlet() {
        super();
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<h1>Calculadora de DNI</h1>");
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        String dniValue = request.getParameter("dniIntroducido");

        try {
            int dni = Integer.parseInt(dniValue);
            int resto = dni % 23;
            char letra = letrasDisposs.charAt(resto);

            response.setContentType("text/html");
            PrintWriter out = response.getWriter();
            out.println("<html><body>");
            out.println("<h1>DNI CALCULADO</h1>");
            out.println("<p>El DNI completo es: " + dniValue + letra + "</p>");
            out.println("<a href='Actividad_2.jsp'>Volver</a>");
            out.println("</body></html>");
        }
        catch (NumberFormatException e) {
            response.setContentType("text/html");
            PrintWriter out = response.getWriter();
            out.println("<html><body>");
            out.println("<h1>Error</h1>");
            out.println("<p>El número introducido no es válido. Por favor, introduce solo números.</p>");
            out.println("<a href='Actividad_2.jsp'>Volver</a>");
            out.println("</body></html>");
        }
    }
}

```

Aquí la clase “WebServlet” “nos servirá para responder las solicitudes HTTP que lleguen a nuestro servlet

Ademas crearemos un atributo “letrasDisposs” que serán las únicas letras que puede tener un DNI

El “project void DoGet” es un método el cual se usa cuando el cliente hace una solicitud GET a la url del servlet

El “project doPost” es un método el cual se usa cuando se envía una solicitud POST desde el form HTML

```

try {
    int dni = Integer.parseInt(dniValue);
    int resto = dni % 23;
    char letra = LetrasDisposss.charAt(resto);

    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    out.println("<html><body>");
    out.println("<h1>DNI CALCULADO</h1>");
    out.println("<p>El DNI completo es: " + dniValue + letra + "</p>");
    out.println("<a href='Actividad_2.jsp'>Volver</a>");
    out.println("</body></html>");
}

```

En esta parte será como funciona el caluclo de la letra del dni y la generación de la pagina html una vez calculada la letra, es decir nos creara una nueva pagina web en el navegador con lo que le hayamos pedido

## DNI CALCULADO

El DNI completo es: 72360631D

[Volver](#)

Por ultimo necesitaremos el “catch” por si el usuario introduce letras se lanzara una excepción porque solo deberíamos introducir el DNi sin la letra

```

catch (NumberFormatException e)
{
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    out.println("<html><body>");
    out.println("<h1>Error</h1>");
    out.println("<p>El número introducido no es válido. Por favor, introduce solo números.</p>");
    out.println("<a href='Actividad_2.jsp'>Volver</a>");
    out.println("</body></html>");
}

```

Investiga sobre los siguientes temas y proporciona una respuesta corta (no más de 50 palabras cada uno):

a. ¿Cuál fue la parte más compleja de esta actividad? ¿Cómo lograste resolverla?

Al principio no pude abrir en el navegador la aplicación ya que me daban errores a la hora de ejecutar el jsp para que estuviera conectado al servlet

Logre resolver dicho problema ya que no había colocado de manera correcta los archivos que venia en la practica guiada

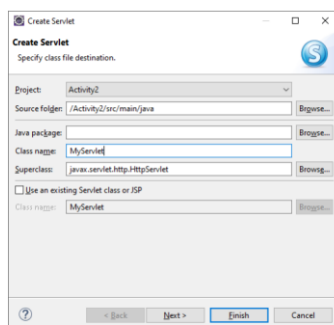


Figure 5. Creando un nuevo servlet

Estaba en un directorio incorrecto

**b. ¿Puedes encontrar otro servidor de aplicaciones y compararlo con Tomcat?**

Otro servidor de aplicaciones es WildFly de código java abierto y multiplataforma

Aunque se asemejan en muchas cosas se diferencian en:

- Tipo de servidor y características: Apache Tomcat se centra en JSP y Java Servlets en cambio Wildfly admite más especificaciones con Java EE además de Java Persistence API y Java Message Service(más conocido como JMS)
- Además Wildfly ofrece una consola de administración más avanzada la cual proporciona un amplio control sobre la configuración del servidor en relación a la Administración y gestión de configuraciones