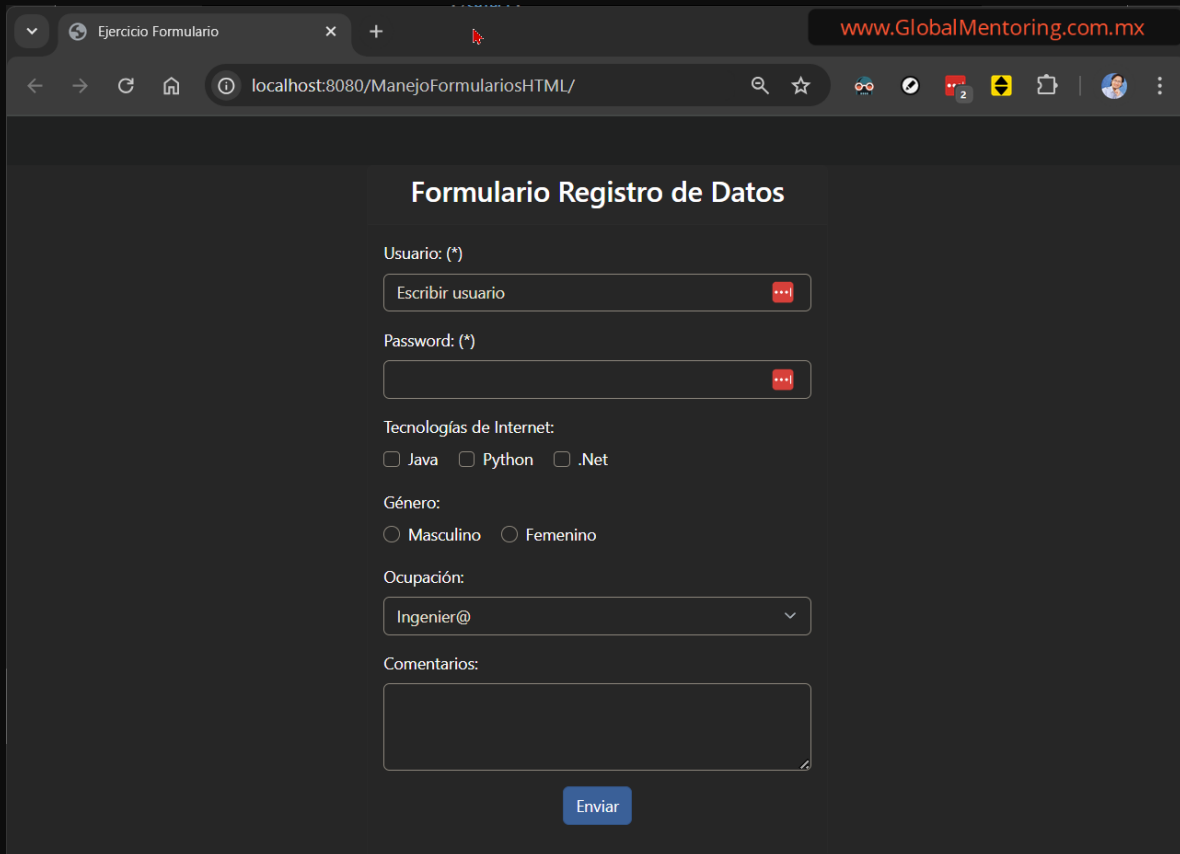


# Manejo de Formularios con Servlets



## Guía paso a paso para crear un proyecto Java con NetBeans y GlassFish para el Manejo de Formularios

Al finalizar veremos el siguiente resultado:



The screenshot shows a web browser window with the address bar displaying 'localhost:8080/ManejoFormulariosHTML/'. The page title is 'Ejercicio Formulario'. The main content is a registration form titled 'Formulario Registro de Datos'. The form includes the following fields and controls:

- Usuario: (\*)**: A text input field with the placeholder 'Escribir usuario' and a red eye icon for toggling password visibility.
- Password: (\*)**: A password input field with a red eye icon for toggling password visibility.
- Tecnologías de Internet:**: Three checkboxes labeled 'Java', 'Python', and '.Net'.
- Género:**: Two radio buttons labeled 'Masculino' and 'Femenino'.
- Ocupación:**: A dropdown menu with 'Ingenier@' selected.
- Comentarios:**: A large text area for comments.
- Enviar**: A blue button to submit the form.



The screenshot shows a web browser window with the address bar displaying 'localhost:8080/ManejoFormulariosHTML/Servlet'. The page title is 'Resultado Servlet'. The main content is a table titled 'Parámetros procesados por el Servlet'.

Parámetro	Valor
Usuario	Ubaldo
Password	123
Tecnologías	Java / Python
Género	masculino
Ocupación	Ingenier@
Comentarios	Nuevo comentario
Valor oculto	nuevo valor oculto

### Requisitos previos:

1. **NetBeans IDE** (versión compatible con Jakarta EE 11).
2. **GlassFish Server** (GlassFish 8 o superior).
3. **JDK 21 o superior** (Aunque se recomienda la versión estable JDK 21).

### Paso 1: Crear un nuevo proyecto en NetBeans

<https://www.globalmentoring.com.mx>

1. Abre **NetBeans** y selecciona **File > New Project**.
2. En la ventana emergente, selecciona **Java with Maven > Web Application** y haz clic en **Next**.
3. Asigna un nombre al proyecto, por ejemplo: `ManejoFormulariosHTML`.
4. Selecciona la versión de **Jakarta EE 10** o superior y haz clic en **Next**.
5. Configura **GlassFish Server** como el servidor de ejecución.
6. Haz clic en **Finish** para crear el proyecto.

## Paso 2: Configurar el proyecto en NetBeans

1. Ve al archivo `pom.xml` y verifica que las dependencias de **Jakarta Servlet API** y **Jakarta Web** estén incluidas:

```
<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <jakartaee>10.0.0</jakartaee>
</properties>
```

```
  <dependency>
    <groupId>jakarta.platform</groupId>
    <artifactId>jakarta.jakartaee-api</artifactId>
    <version>${jakartaee}</version>
    <scope>provided</scope>
  </dependency>
```

## Paso 3: Estructura básica del proyecto

### Crear la página HTML

1. Dentro de la carpeta `src/main/webapp`, crea un archivo HTML llamado `index.html`.
2. Código HTML en este archivo:

```
<!DOCTYPE html>
<html lang="es" data-bs-theme="dark">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Ejercicio Formulario</title>
  <!-- Bootstrap CSS -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>
```

```

<div class="container mt-5">
  <div class="row">
    <div class="col-md-6 offset-md-3">
      <div class="card">
        <div class="card-header text-center">
          <h3>Formulario Registro de Datos</h3>
        </div>
        <div class="card-body">
          <form name="form1" action="/ManejoFormulariosHTML/Servlet" method="post">
            <input type="hidden" name="oculto" value="nuevo valor oculto"/>

            <!-- Usuario -->
            <div class="mb-3">
              <label for="usuario" class="form-label">Usuario: (*)</label>
              <input type="text" class="form-control" name="usuario" id="usuario"
value="Escribir usuario" required/>
            </div>

            <!-- Password -->
            <div class="mb-3">
              <label for="password" class="form-label">Password: (*)</label>
              <input type="password" class="form-control" name="password"
id="password" required/>
            </div>

            <!-- Tecnologías de Internet -->
            <div class="mb-3">
              <label class="form-label">Tecnologías de Internet:</label><br>
              <div class="form-check form-check-inline">
                <input class="form-check-input" type="checkbox"
name="tecnologia" id="java" value="Java">
                <label class="form-check-label" for="java">Java</label>
              </div>
              <div class="form-check form-check-inline">
                <input class="form-check-input" type="checkbox"
name="tecnologia" id="python" value="Python">
                <label class="form-check-label" for="python">Python</label>
              </div>
              <div class="form-check form-check-inline">
                <input class="form-check-input" type="checkbox"
name="tecnologia" id="net" value=".Net">
                <label class="form-check-label" for="net">.Net</label>
              </div>
            </div>
          </form>
        </div>
      </div>
    </div>
  </div>
</div>

```

```

        <!-- Género -->
        <div class="mb-3">
            <label class="form-label">Género:</label><br>
            <div class="form-check form-check-inline">
                <input class="form-check-input" type="radio" name="genero"
id="masculino" value="masculino">
                <label class="form-check-label"
for="masculino">Masculino</label>
            </div>
            <div class="form-check form-check-inline">
                <input class="form-check-input" type="radio" name="genero"
id="femenino" value="femenino">
                <label class="form-check-label" for="femenino">Femenino</label>
            </div>
        </div>

        <!-- Ocupación -->
        <div class="mb-3">
            <label for="ocupacion" class="form-label">Ocupación:</label>
            <select name="ocupacion" id="ocupacion" class="form-select">
                <option value="Ingenier@">Ingenier@</option>
                <option value="Contador@">Contador@</option>
                <option value="Abogad@">Abogad@</option>
                <option value="Otro">Otro</option>
            </select>
        </div>

        <!-- Comentarios -->
        <div class="mb-3">
            <label for="comentarios" class="form-label">Comentarios:</label>
            <textarea class="form-control" name="comentarios" id="comentarios"
rows="3"></textarea>
        </div>

        <!-- Botón Enviar -->
        <div class="mb-3 text-center">
            <button type="submit" class="btn btn-primary">Enviar</button>
        </div>

    </form>
</div>
</div>
</div>
</div>
</div>

```

```
<!-- Bootstrap JS and Popper.js -->
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

Aquí tienes una descripción de las clases de Bootstrap utilizadas:

1. **container**: Contenedor centralizado con márgenes laterales automáticos.
2. **mt-5**: **margin-top** (margen superior) grande. El número 5 indica un tamaño mayor.
3. **text-center**: Centra el texto horizontalmente.
4. **text-warning**: Aplica color de advertencia (amarillo) al texto.
5. **mb-4**: **margin-bottom** (margen inferior) moderado. El número 4 es un tamaño medio.
6. **form-label**: Estiliza las etiquetas de los formularios.
7. **form-control**: Estiliza los campos de entrada como `input` y `textarea`.
8. **form-select**: Estiliza los menús desplegables.
9. **form-check**: Agrupa y estiliza casillas de verificación (`checkbox`) o radios.
10. **form-check-input**: Estiliza los `inputs` tipo `checkbox` o `radio`.
11. **form-check-label**: Estiliza las etiquetas asociadas a `checkbox` o `radio`.
12. **btn**: Clase base para todos los botones.
13. **btn-primary**: Botón con estilo primario (azul).
14. **table**: Aplica estilos básicos a tablas HTML.
15. **table-bordered**: Añade bordes a las celdas de la tabla.
16. **table-striped**: Aplica colores alternos a las filas de la tabla.

## Paso 4: Crear el Servlet

1. Dentro de la carpeta `src/main/java`, crea un paquete llamado `web`.
2. Dentro del paquete `web`, crea una clase Java llamada `Servlet.java`.

Código en el archivo `Servlet.java`:

```
package web;

import java.io.IOException;
import java.io.PrintWriter;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.*;

@WebServlet("/Servlet")
public class Servlet extends HttpServlet {
```

```
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();

    var usuario = request.getParameter("usuario");
    var password = request.getParameter("password");
    String tecnologias[] = request.getParameterValues("tecnologias");
    var genero = request.getParameter("genero");
    var ocupacion = request.getParameter("ocupacion");
    var comentarios = request.getParameter("comentarios");
    var valorOculto = request.getParameter("oculto");

    out.print("<html data-bs-theme='dark'>");
    out.print("<head>");
    out.print("<title>Resultado Servlet</title>");
    out.print("<link
href='https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css' rel='stylesheet'>");
    out.print("</head>");
    out.print("<body class='container mt-5'>");
    out.print("<h1 class='mb-4 text-center text-warning'>Parámetros procesados por el
Servlet</h1>");
    out.print("<table class='table table-bordered table-striped'>");

    out.print("<tr><th>Usuario</th><td>" + usuario + "</td></tr>");
    out.print("<tr><th>Password</th><td>" + password + "</td></tr>");

    out.print("<tr><th>Tecnologías</th><td>");

    if (tecnologias != null) {
        out.print(String.join(" / ", tecnologias));
    }
    else{
        out.print("Tecnologías no proporcionadas");
    }
    out.print("</td></tr>");

    out.print("<tr><th>Género</th><td>" + genero + "</td></tr>");
    out.print("<tr><th>Ocupación</th><td>" + ocupacion + "</td></tr>");

    out.print("</td></tr>");

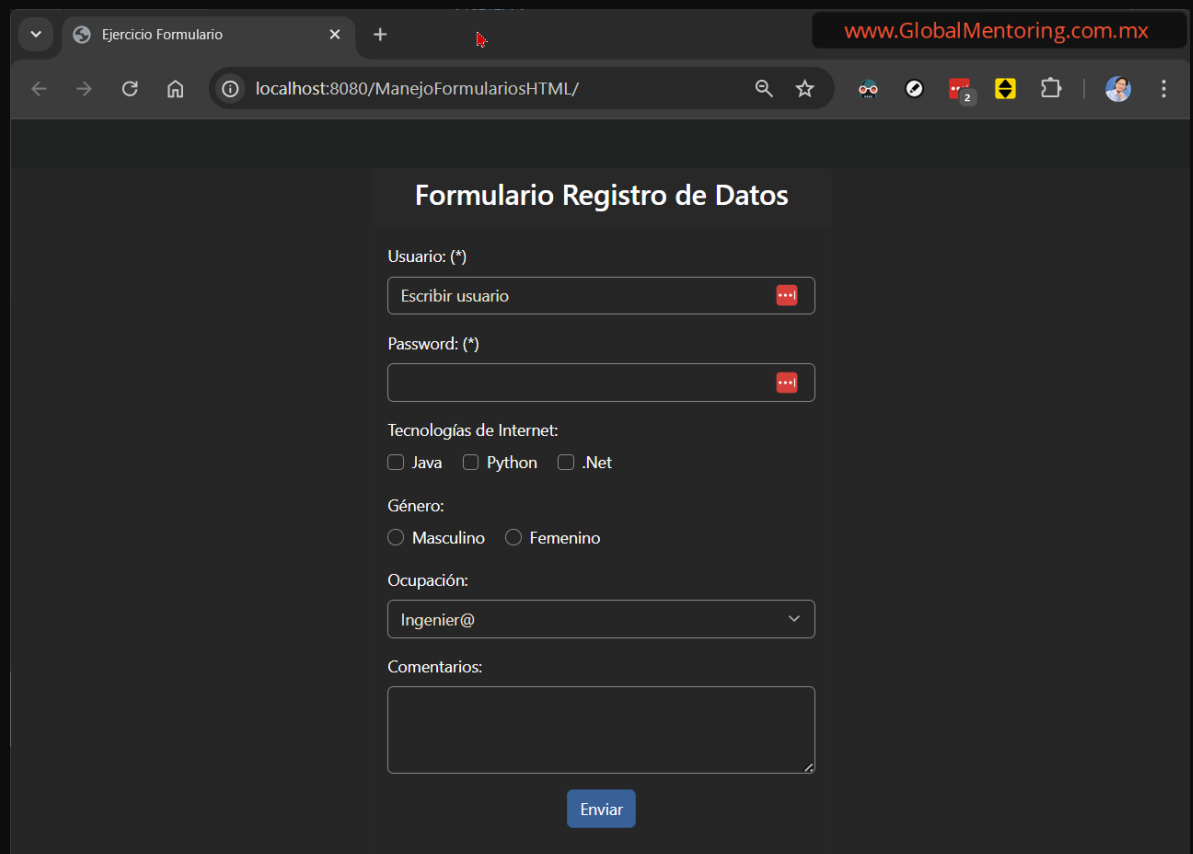
    out.print("<tr><th>Comentarios</th><td>" + comentarios + "</td></tr>");
```

```
out.print("<tr><th>Valor oculto</th><td>" + valorOculto + "</td></tr>");

out.print("</table>");
out.print("</body>");
out.print("</html>");
}
}
```

## Paso 7: Probar la aplicación

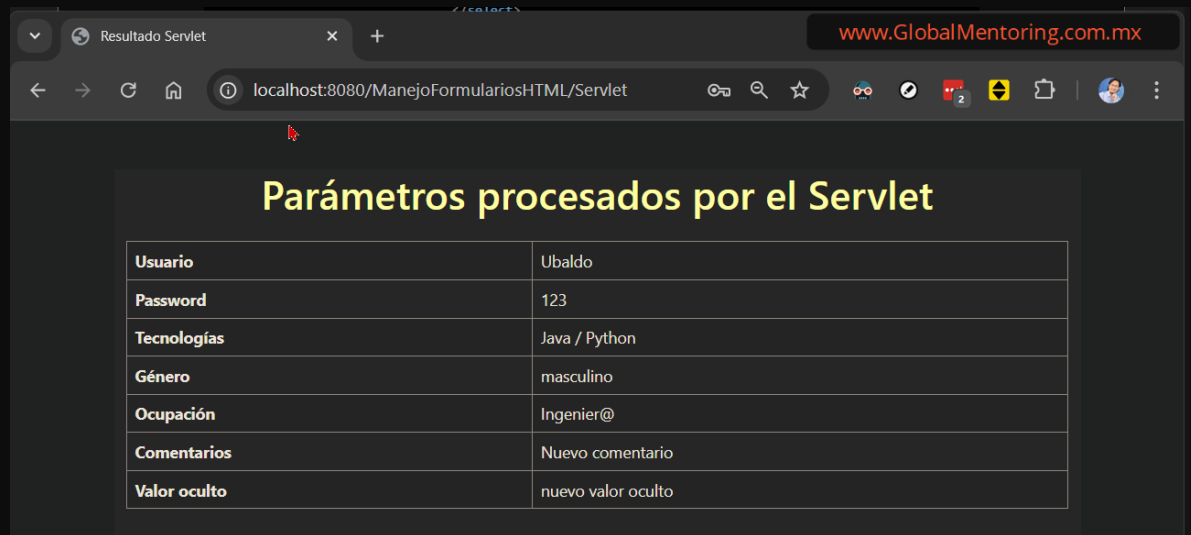
1. Al ejecutar el proyecto, se abrirá la página `index.html`. Rellena los campos y envía el formulario.
2. El Servlet procesará los datos y mostrará los resultados en una tabla.



The screenshot shows a web browser window with the address bar displaying `localhost:8080/ManejoFormulariosHTML/`. The page title is "Ejercicio Formulario". The main content is a registration form titled "Formulario Registro de Datos". The form includes the following fields and options:

- Usuario: (\*)**: A text input field with the placeholder "Escribir usuario".
- Password: (\*)**: A password input field.
- Tecnologías de Internet:**: Three radio buttons labeled "Java", "Python", and ".Net".
- Género:**: Two radio buttons labeled "Masculino" and "Femenino".
- Ocupación:**: A dropdown menu with "Ingenier@" selected.
- Comentarios:**: A large text area for comments.
- Enviar**: A blue button to submit the form.





Con esto hemos puesto en práctica el uso de formularios HTML y el procesamiento con Servlets.

Saludos!

**Ing. Ubaldo Acosta**

Fundador de [GlobalMentoring.com.mx](https://www.globalmentoring.com.mx)