

D3-UD2. Encadenando formularios

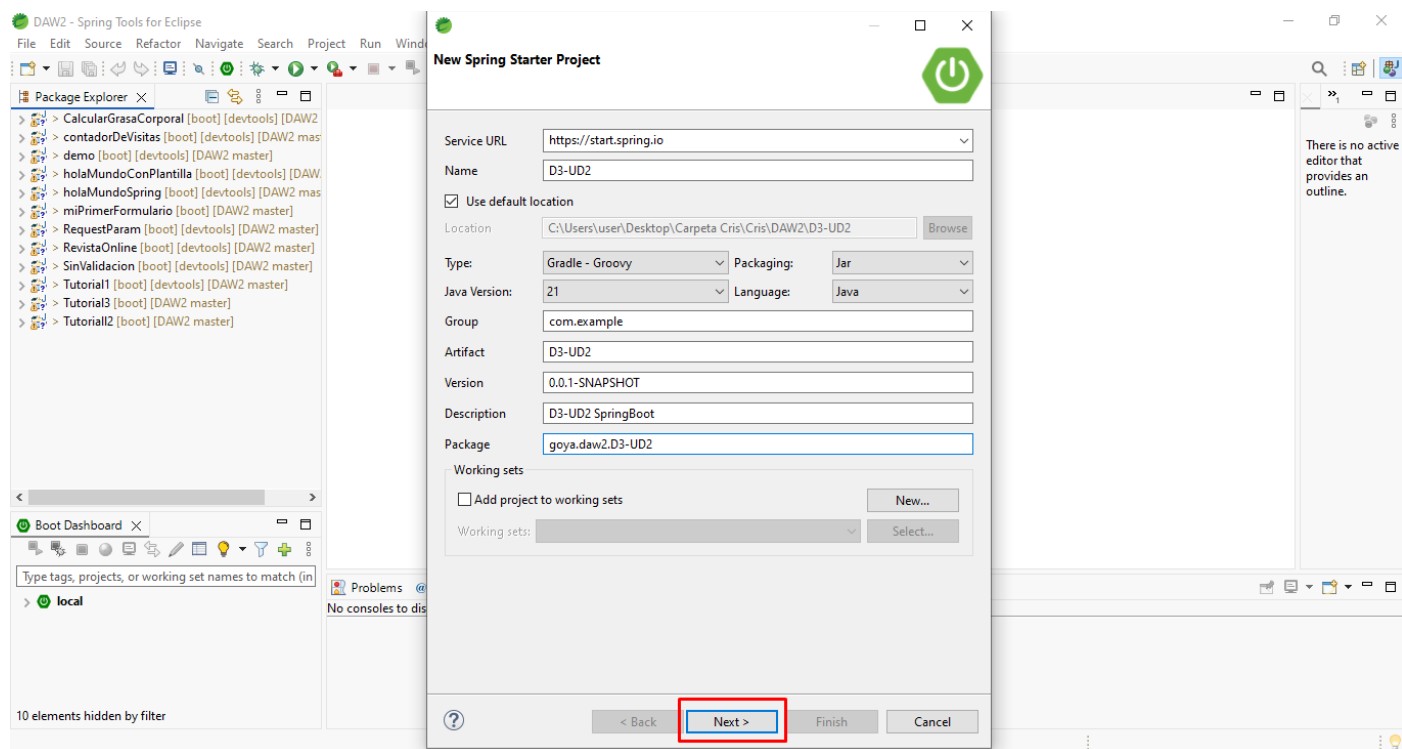
Tiempo: 5 horas de clase.

En este desafío se te pide realizar una web con temática de tu elección en la que se le pidan al usuario varios datos en pantallas distintas y al final se muestre en resumen de todos los datos introducidos.

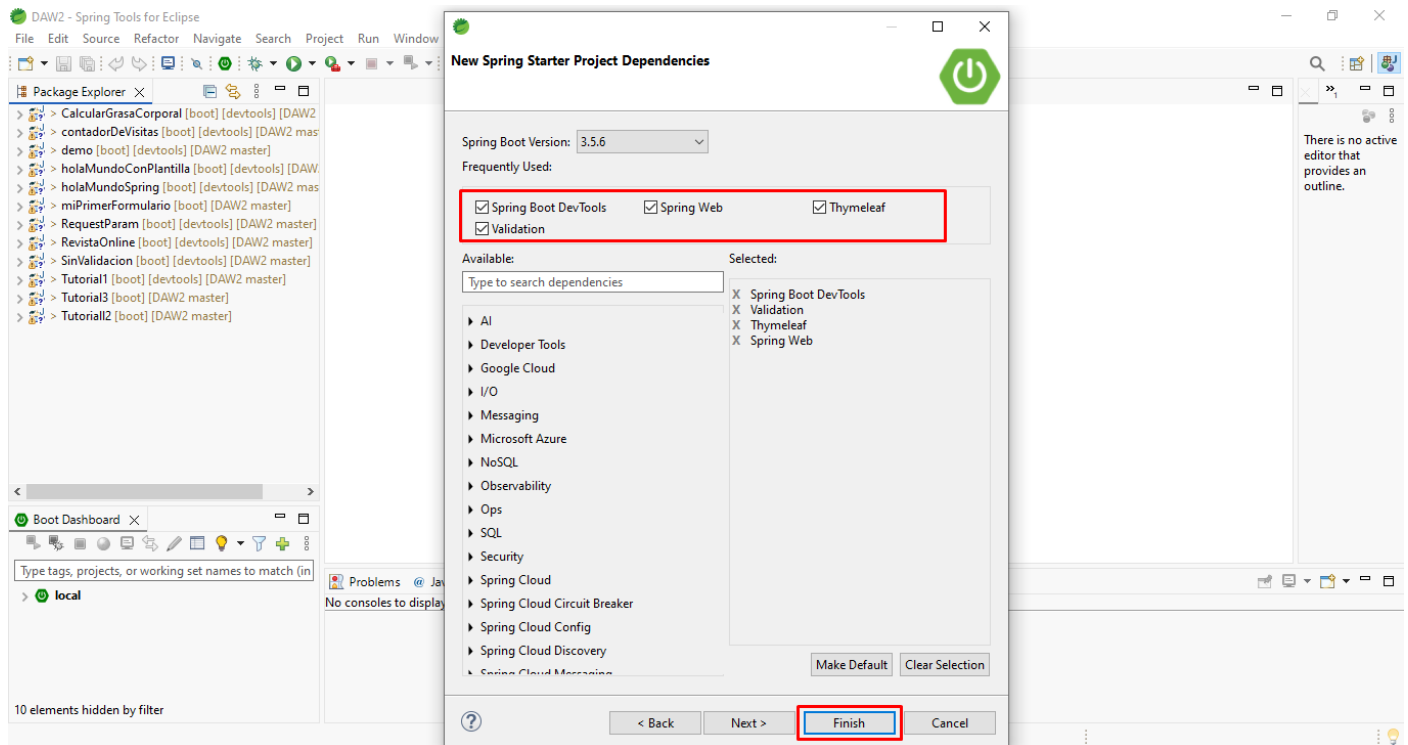
Te sugiero hacerlo en los siguientes pasos evaluables:

1. Genera una plantilla para cada pantalla, mínimo tres de captura de datos y una de resultados. En la captura de datos debe haber al menos dos que sean de tipo radio, checkbox o desplegable (select)
2. Decide si vas a hacer al paso de una pantalla a otra mediante parámetros sobre la misma url o con urls distintas. Crea un controlador y la lógica para ir pasando de una a otra.
3. Procesado de los datos. Al ser http un protocolo sin estado, verás que en la última etapa no tienes forma de acceder a los datos introducidos en la primera. Una solución posible es ir tratando en cada etapa lo recibido y añadirlo a la siguiente plantilla como un campo oculto (hidden)

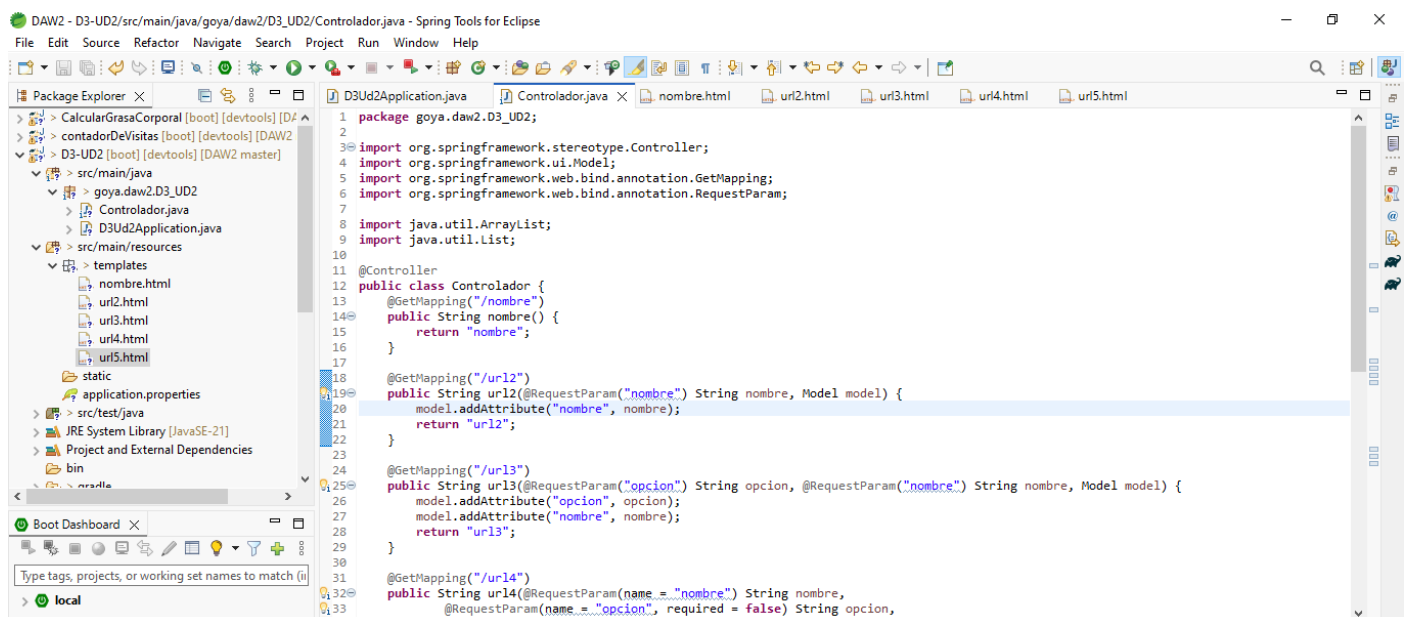
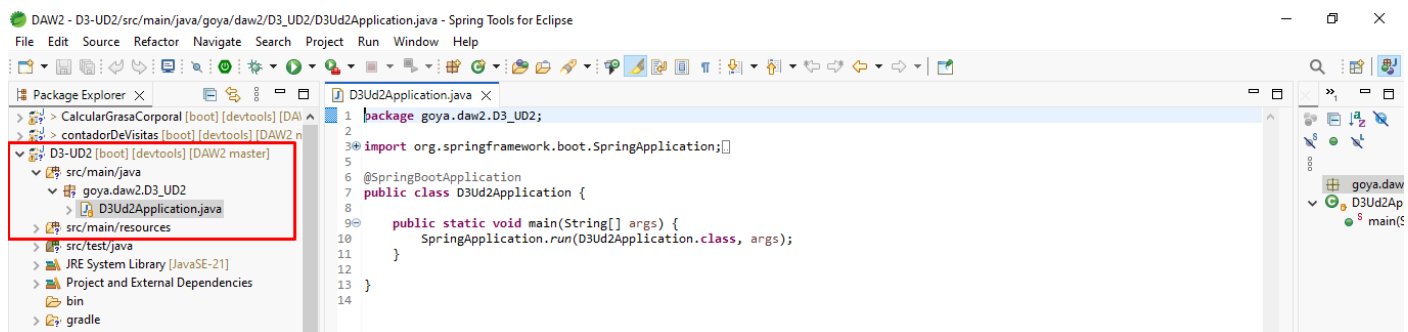
Creamos un nuevo proyecto llamado D3-UD2

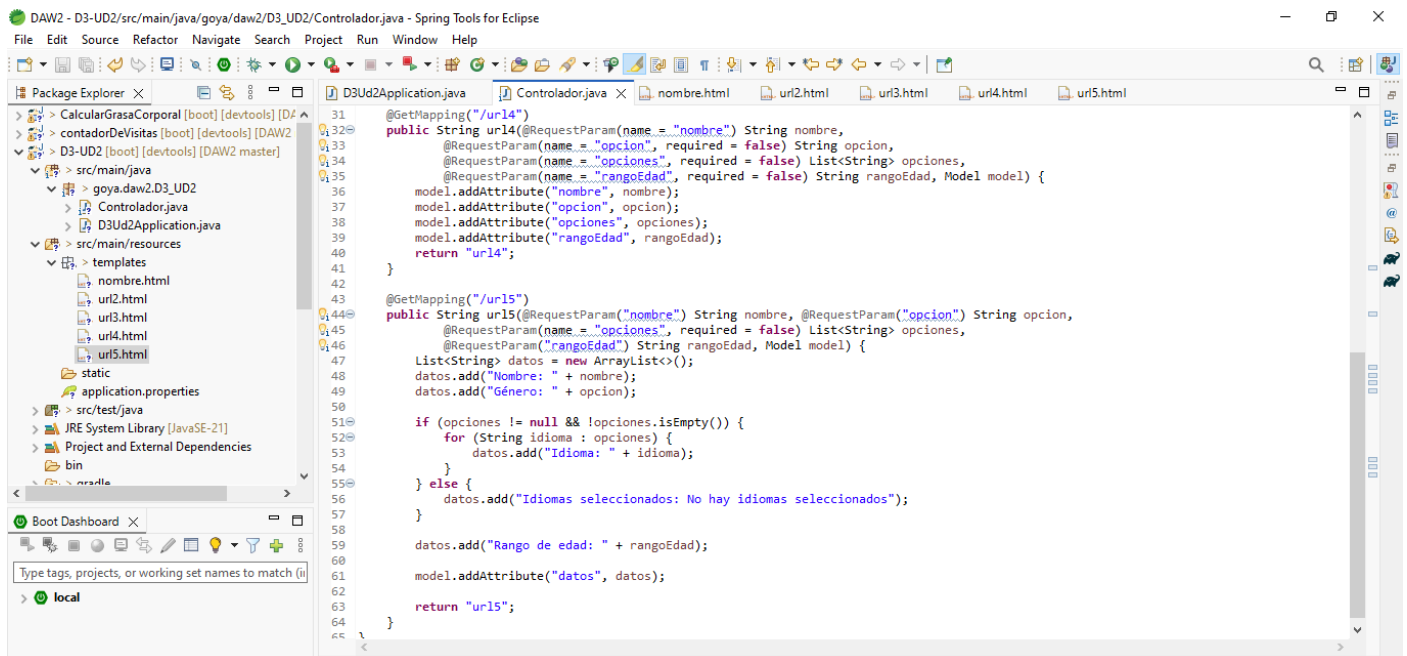


Añadimos las dependencias spring boot devtools, validation, thymeleaf y spring web

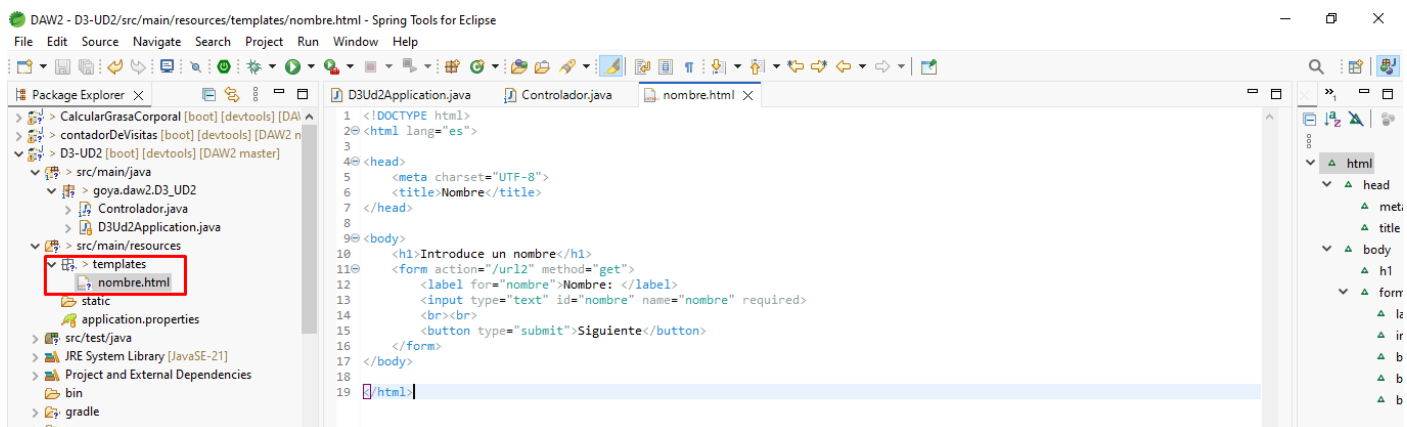
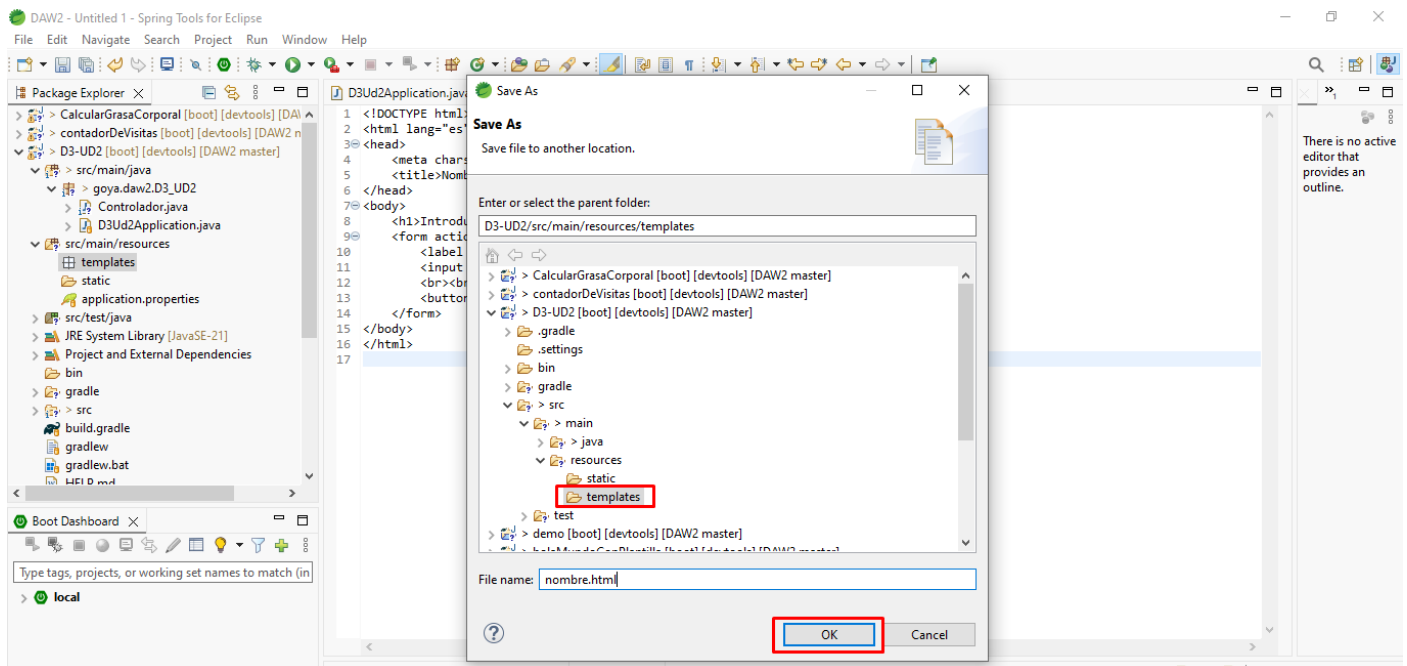


Nos crea el proyecto y la clase por defecto

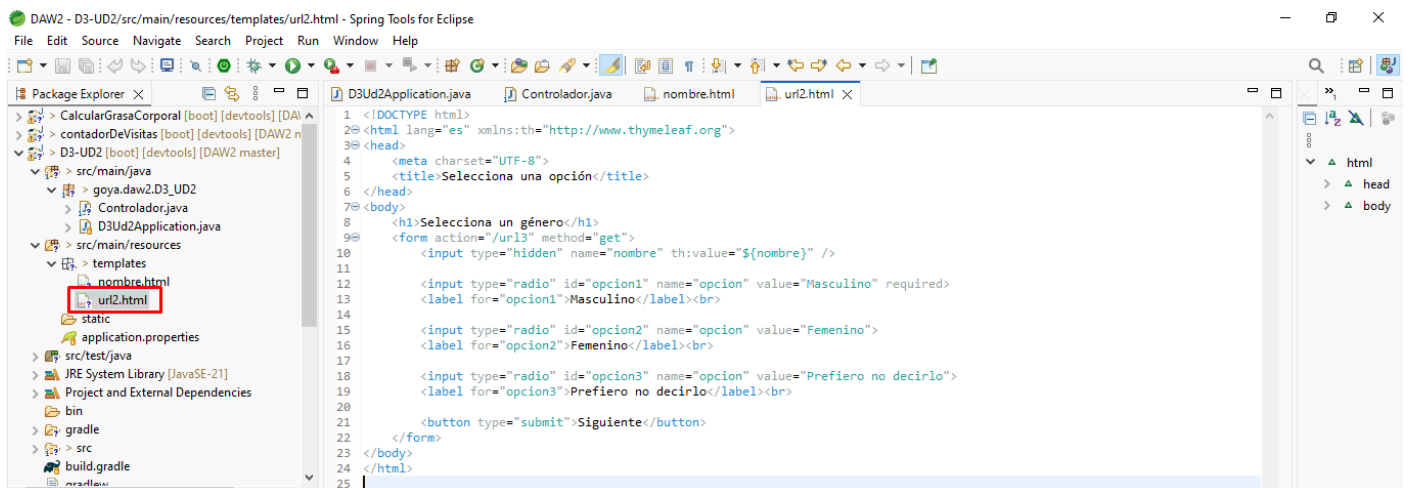
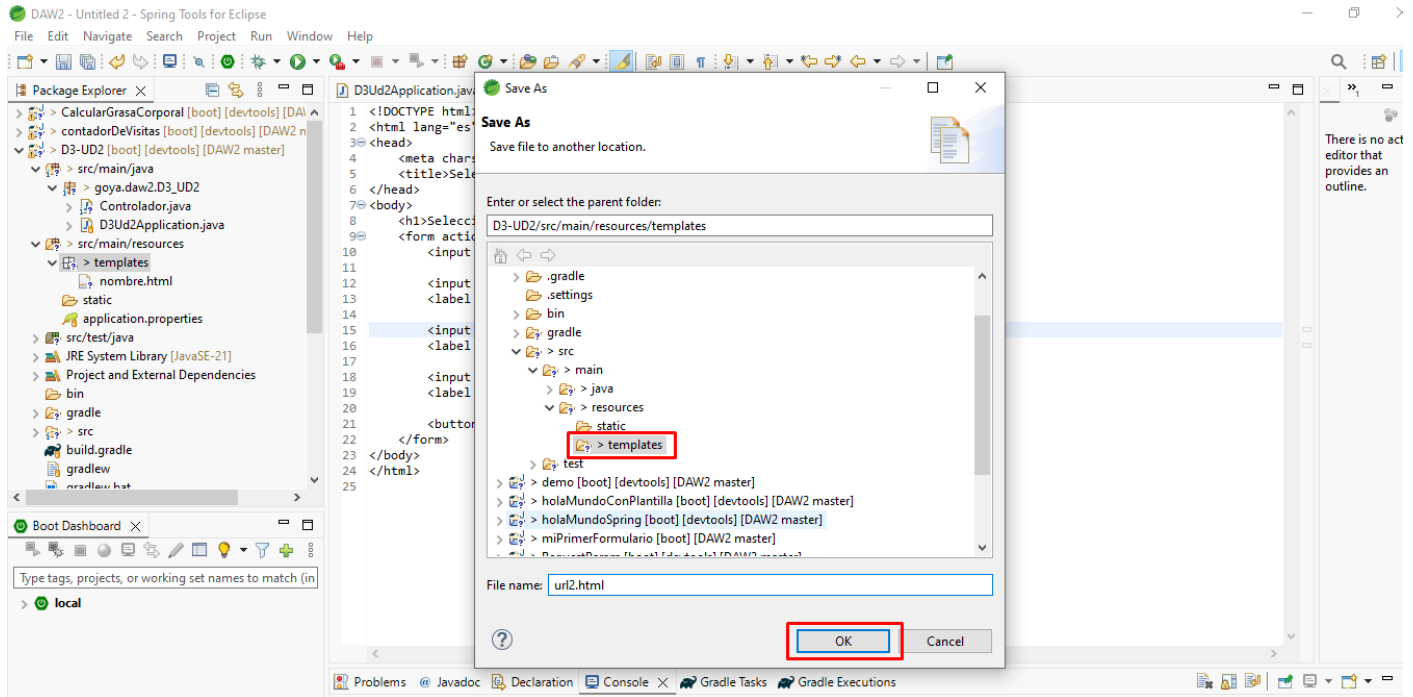




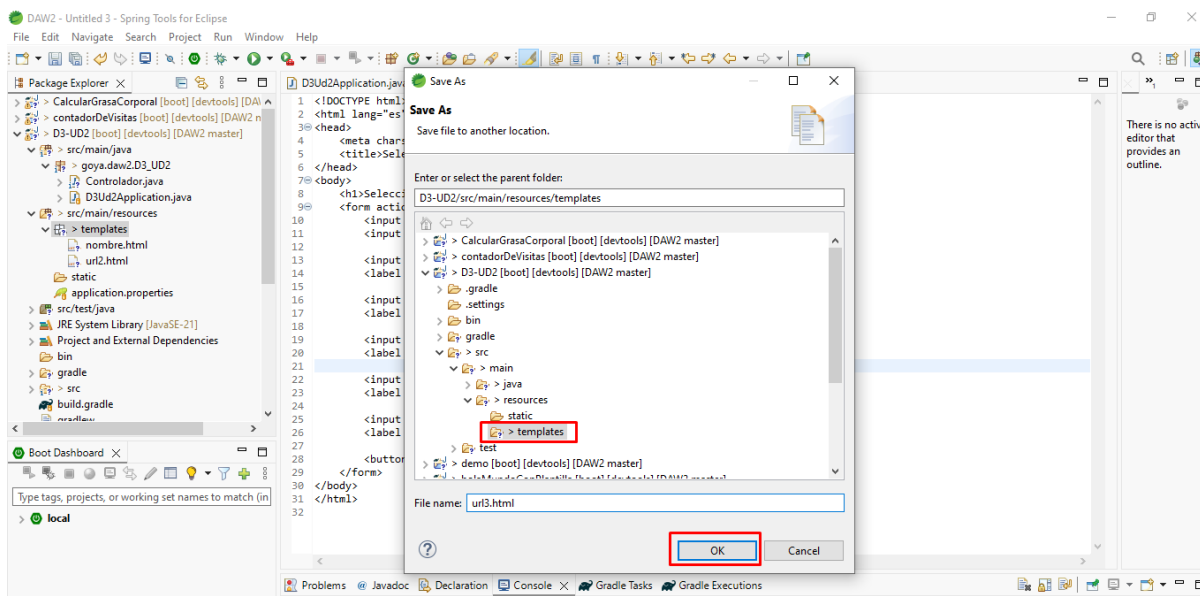
Hacemos un HTML llamado nombre.html, para poner el nombre

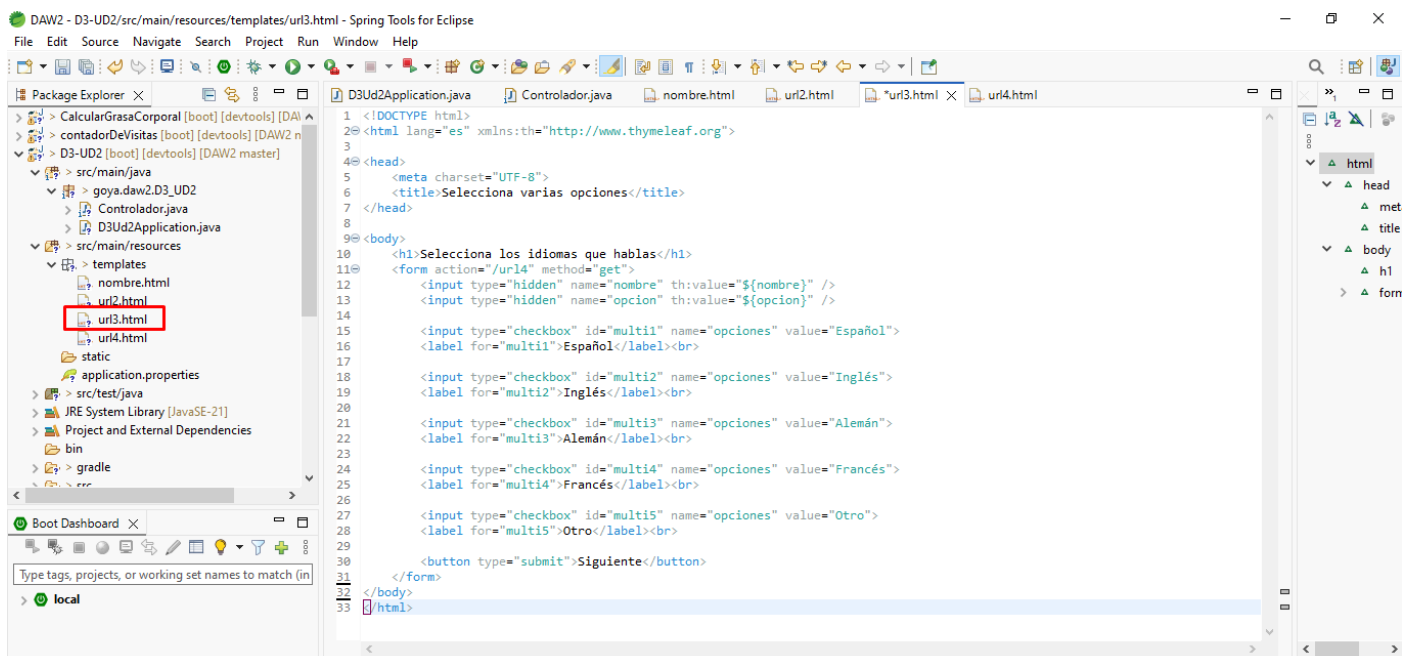


Creamos otro HTML llamado url2.html, para elegir el género hecho con radio

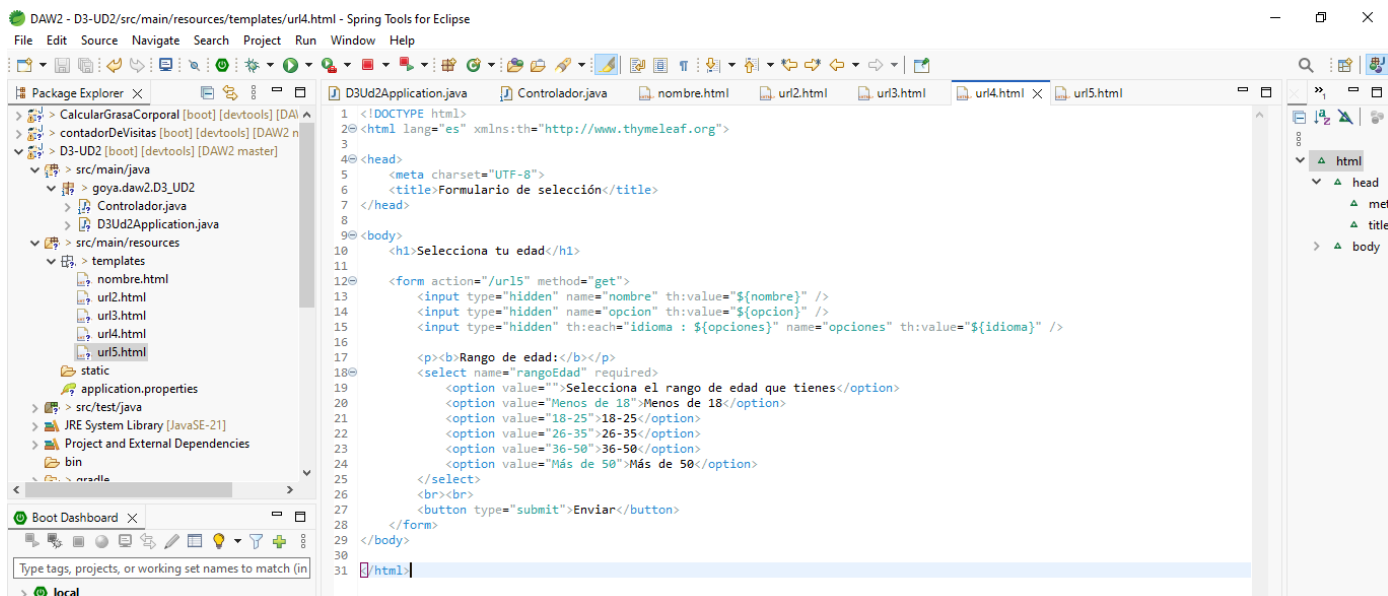
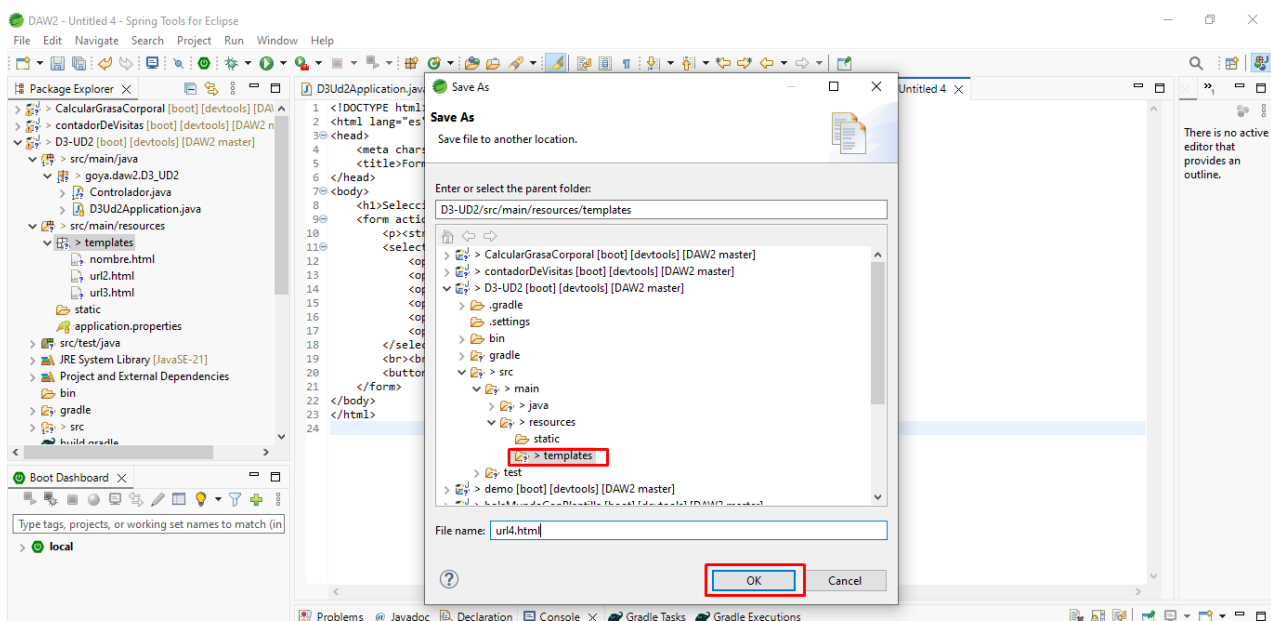


Creamos otro HTML llamado url3.html, para elegir varias opciones del idioma que hablas hecho con checkbox



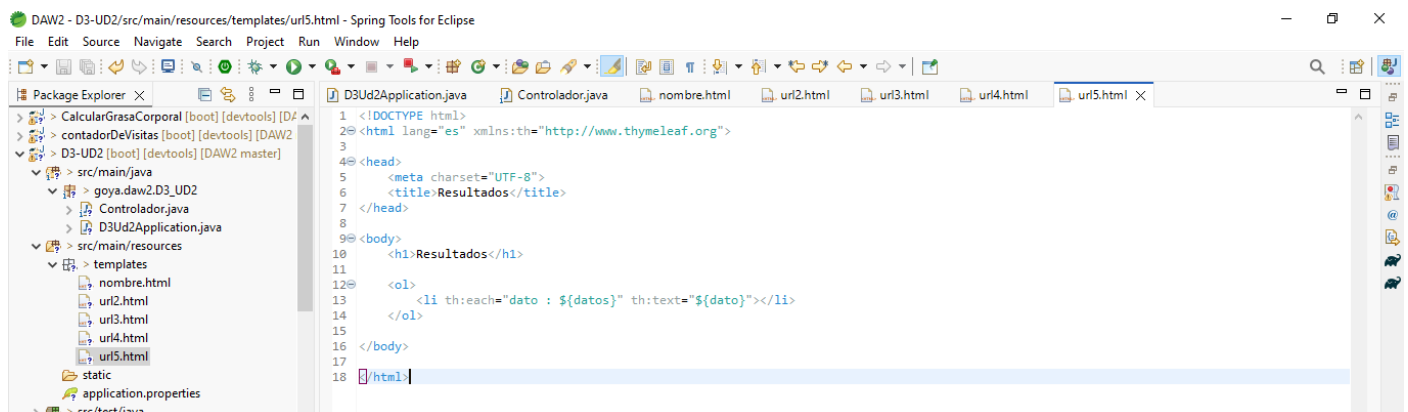
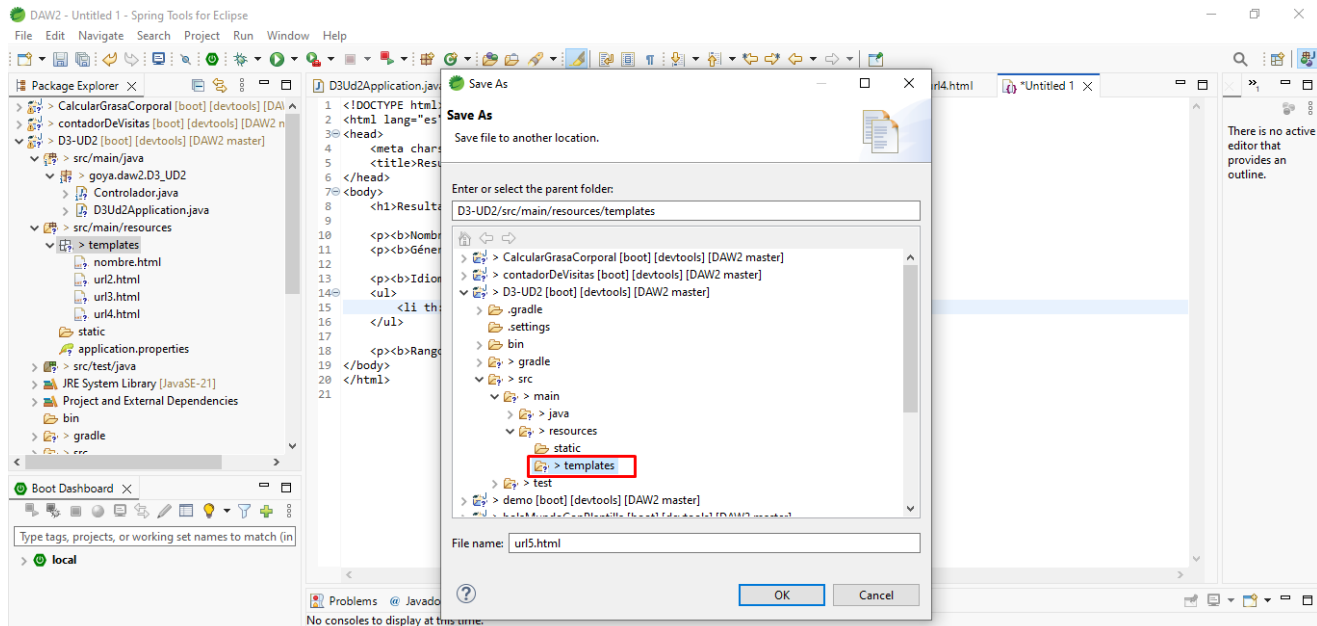


Creamos otro HTML llamado url4.html, para hacer un desplegable con el rango de edad que tienes con un select

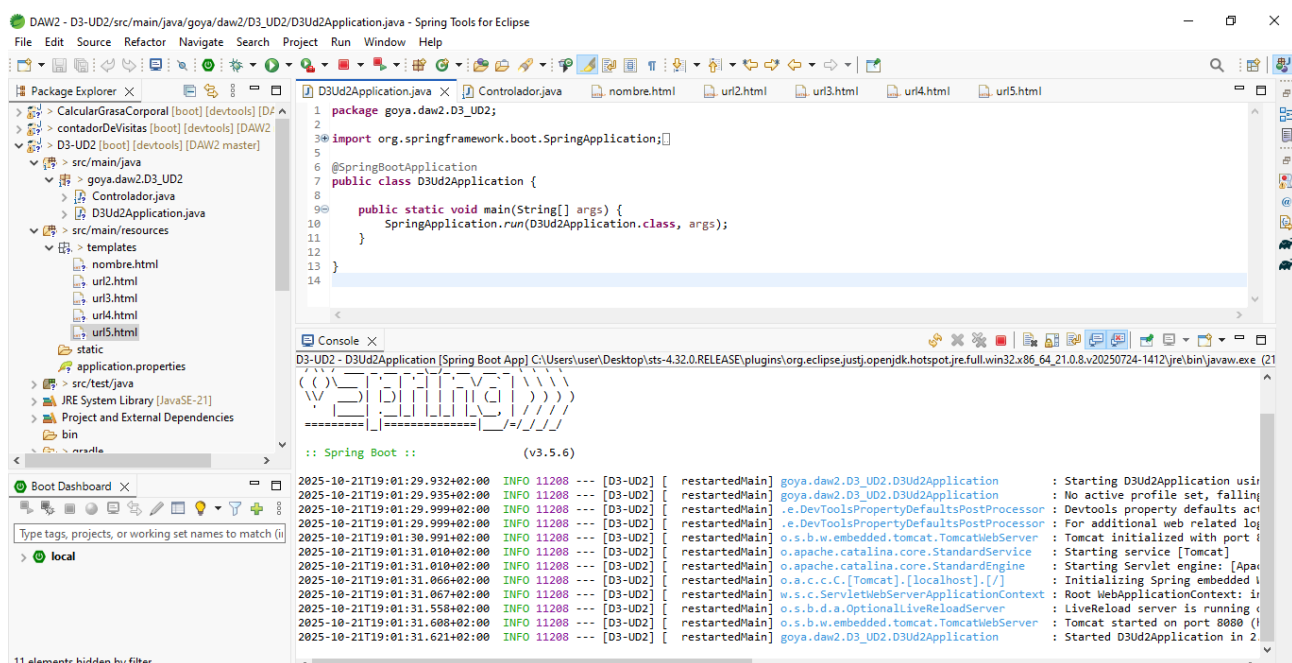


4. En la última etapa pasarás a la plantilla una lista de String con los datos de cada etapa y lo mostrarás en la plantilla con un `` con `th:each`. De esta manera en un futuro será fácil añadir o quitar etapas.

Creamos otro HTML llamado `url5.html`, que muestra los resultados



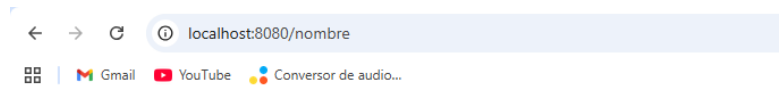
Iniciamos



Comprobamos que funciona en el navegador:

http://localhost:8080/nombre

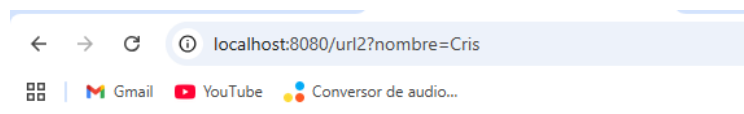
Primero nos sale para poner el nombre



Introduce un nombre

Nombre:

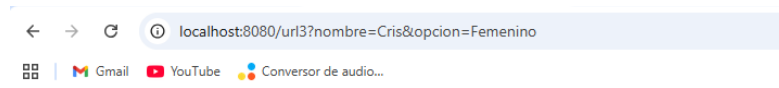
Seleccionamos el género



Selecciona un género

☐ Masculino
☒ Femenino
☐ Prefiero no decirlo

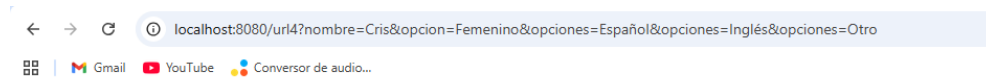
Seleccionamos varios idiomas



Selecciona los idiomas que hablas

☒ Español
☒ Inglés
☐ Alemán
☐ Francés
☒ Otro

Seleccionamos el rango de edad



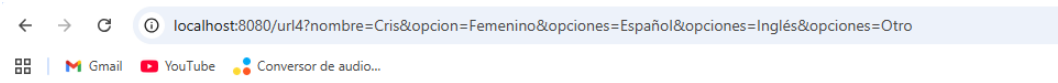
Selecciona tu edad

Rango de edad:

Selecciona el rango de edad que tienes ▼

Selecciona el rango de edad que tienes

Menos de 18
18-25
26-35
36-50
Más de 50

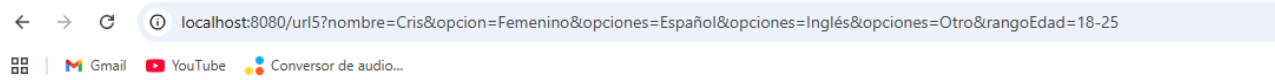


Selecciona tu edad

Rango de edad:

Enviar

Por último, nos muestra los resultados



Resultados

1. Nombre: Cris
2. Género: Femenino
3. Idioma: Español
4. Idioma: Inglés
5. Idioma: Otro
6. Rango de edad: 18-25

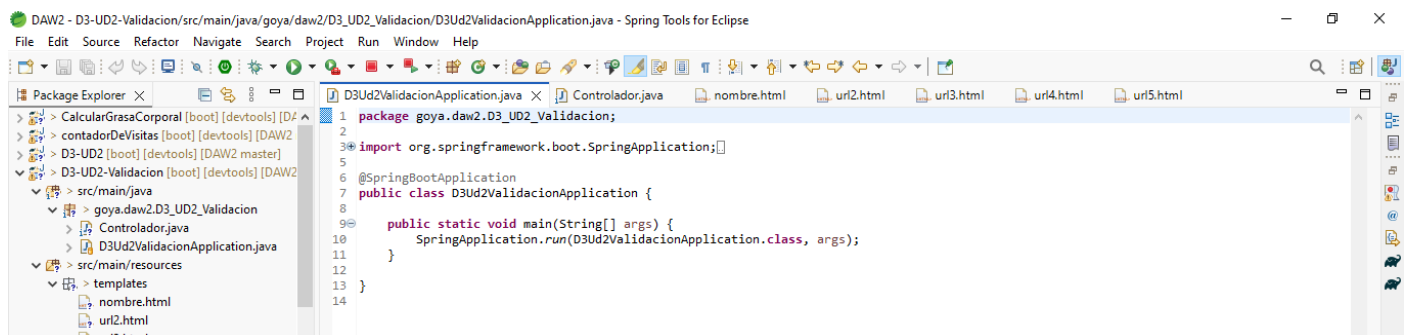
Nota: en este desafío no está permitido el uso de Cookies ni sesiones. Se trata de darnos cuenta de las limitaciones del http sin estado y en el siguiente tema ya lo subsanaremos con las sesiones.

Entrega: documento D3-UD2.PDF con la descripción de los pasos realizados, problemas encontrados, soluciones aportadas y conclusiones sobre la consecución de objetivos en cada apartado. El código se entregará en un enlace a cloud o git, ojo al .gitignore.

Calificación:

- Suficiente: quizz correctamente realizado y documentado.
- Bien: lo anterior más algunos controles de validación y más campos de entrada. Sin ninguna validación en el html, todo en el backend.

Creamos un proyecto con validaciones



Creamos la clase Controlador

DAW2 - D3-UD2-Validacion/src/main/java/goya/daw2/D3_UD2_Validacion/Controlador.java - Spring Tools for Eclipse

```
File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer X
  > CalculadoraCorporal [boot] [devtools] [DAW2]
  > contadorDeVisitas [boot] [devtools] [DAW2]
  > D3-UD2 [boot] [devtools] [DAW2 master]
  > D3-UD2-Validacion [boot] [devtools] [DAW2]
    > src/main/java
      > goya.daw2.D3_UD2_Validacion
        > Controlador.java
        > D3Ud2ValidacionApplication.java
    > src/main/resources
      > templates
        > nombre.html
        > url2.html
        > url3.html
        > url4.html
        > url5.html
      > static
        > application.properties
    > src/test/java
    > JRE System Library [JavaSE-21]
    > Project and External Dependencies

Boot Dashboard X
Type tags, projects, or working set names to match (i)
```

```
1 package goya.daw2.D3_UD2_Validacion;
2
3 import org.springframework.stereotype.Controller;
4 import org.springframework.ui.Model;
5 import org.springframework.web.bind.annotation.GetMapping;
6 import org.springframework.web.bind.annotation.RequestParam;
7
8 import java.util.ArrayList;
9 import java.util.List;
10
11 @Controller
12 public class Controlador {
13     @GetMapping("/nombre")
14     public String nombre() {
15         return "nombre";
16     }
17
18     @GetMapping("/url2")
19     public String url2(@RequestParam("nombre") String nombre, Model model) {
20         List<String> errores = new ArrayList<>();
21         if (nombre == null || nombre.trim().equals("")) {
22             errores.add("El nombre es obligatorio");
23         }
24         if (!errores.isEmpty()) {
25             model.addAttribute("errores", errores);
26             return "nombre";
27         }
28         model.addAttribute("nombre", nombre);
29         return "url2";
30     }
31 }
32
```

DAW2 - D3-UD2-Validacion/src/main/java/goya/daw2/D3_UD2_Validacion/Controlador.java - Spring Tools for Eclipse

```
File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer X
  > CalculadoraCorporal [boot] [devtools] [DAW2]
  > contadorDeVisitas [boot] [devtools] [DAW2]
  > D3-UD2 [boot] [devtools] [DAW2 master]
  > D3-UD2-Validacion [boot] [devtools] [DAW2]
    > src/main/java
      > goya.daw2.D3_UD2_Validacion
        > Controlador.java
        > D3Ud2ValidacionApplication.java
    > src/main/resources
      > templates
        > nombre.html
        > url2.html
        > url3.html
        > url4.html
        > url5.html
      > static
        > application.properties
    > src/test/java
    > JRE System Library [JavaSE-21]
    > Project and External Dependencies

Boot Dashboard X
Type tags, projects, or working set names to match (i)
```

```
32
33
34 @GetMapping("/url3")
35 public String url3(@RequestParam("nombre") String nombre, @RequestParam("opcion") String opcion, Model model) {
36
37     List<String> errores = new ArrayList<>();
38     if (opcion == null || opcion.trim().equals("")) {
39         errores.add("El género es obligatorio");
40     }
41
42     if (!errores.isEmpty()) {
43         model.addAttribute("errores", errores);
44         model.addAttribute("nombre", nombre);
45         return "url2";
46     }
47
48     model.addAttribute("nombre", nombre);
49     model.addAttribute("opcion", opcion);
50     return "url4";
51 }
52
53 @GetMapping("/url4")
54 public String url4(@RequestParam("nombre") String nombre, @RequestParam("opcion") String opcion,
55 @RequestParam(value = "opciones", required = false) List<String> opciones,
56 @RequestParam("rangoEdad") String rangoEdad, Model model) {
57
58     List<String> errores = new ArrayList<>();
59     if (nombre == null || nombre.trim().equals("")) {
60         errores.add("El nombre es obligatorio");
61     }
62     if (opcion == null || opcion.trim().equals("")) {
63         errores.add("El género es obligatorio");
64     }
65     if (rangoEdad == null || rangoEdad.trim().equals("")) {
66         errores.add("El rango de edad es obligatorio");
67     }
68 }
69
```

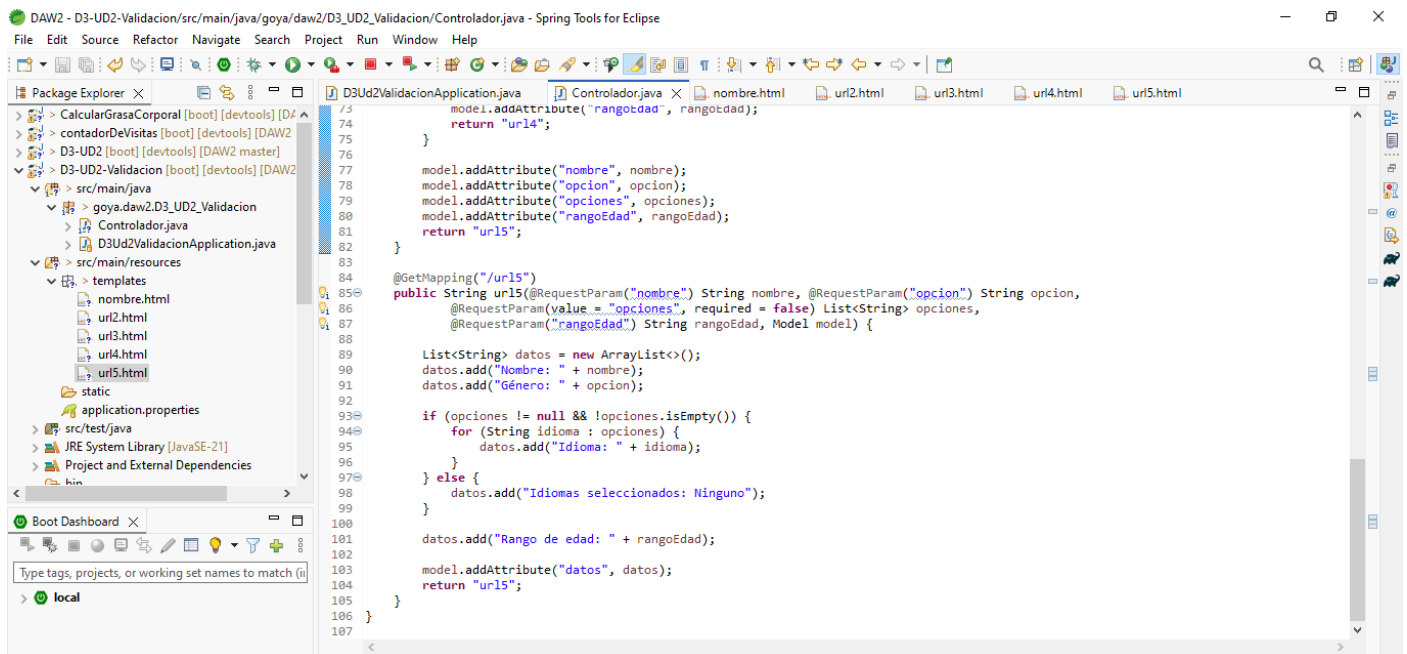
DAW2 - D3-UD2-Validacion/src/main/java/goya/daw2/D3_UD2_Validacion/Controlador.java - Spring Tools for Eclipse

```
File Edit Source Refactor Navigate Search Project Run Window Help

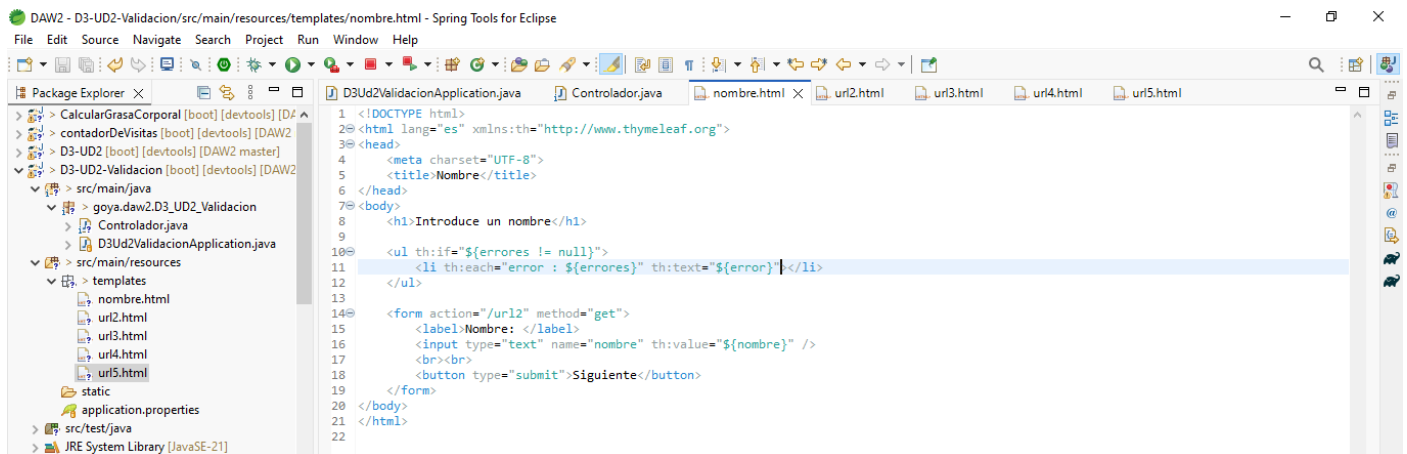
Package Explorer X
  > CalculadoraCorporal [boot] [devtools] [DAW2]
  > contadorDeVisitas [boot] [devtools] [DAW2]
  > D3-UD2 [boot] [devtools] [DAW2 master]
  > D3-UD2-Validacion [boot] [devtools] [DAW2]
    > src/main/java
      > goya.daw2.D3_UD2_Validacion
        > Controlador.java
        > D3Ud2ValidacionApplication.java
    > src/main/resources
      > templates
        > nombre.html
        > url2.html
        > url3.html
        > url4.html
        > url5.html
      > static
        > application.properties
    > src/test/java
    > JRE System Library [JavaSE-21]
    > Project and External Dependencies

Boot Dashboard X
Type tags, projects, or working set names to match (i)
```

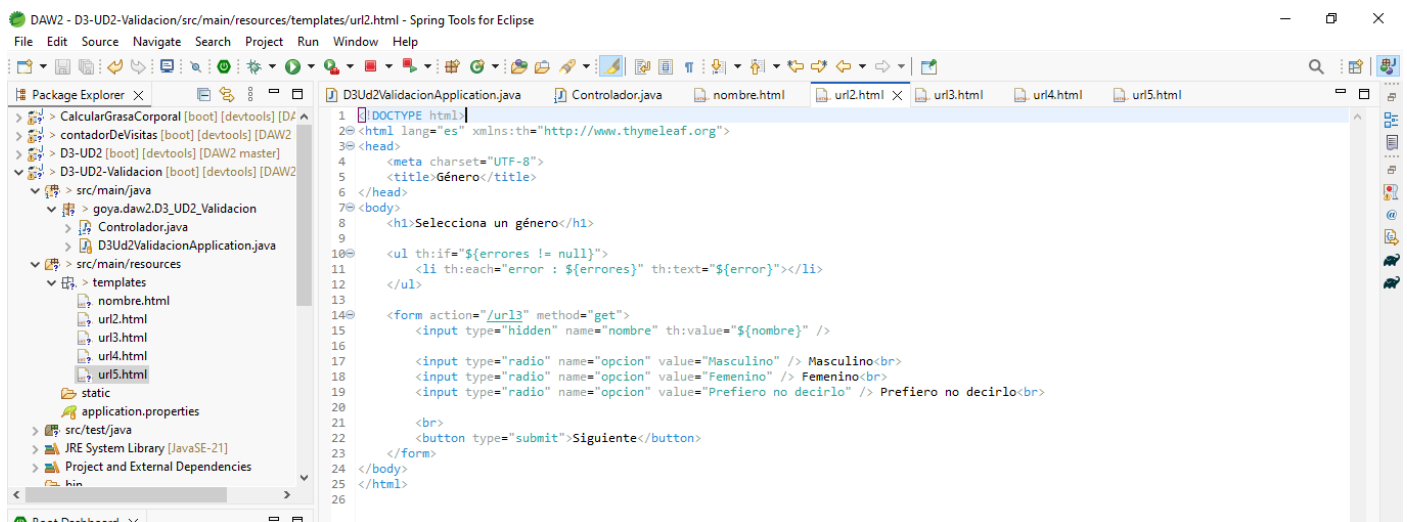
```
68
69
70 if (!errores.isEmpty()) {
71     model.addAttribute("errores", errores);
72     model.addAttribute("nombre", nombre);
73     model.addAttribute("opcion", opcion);
74     model.addAttribute("opciones", opciones);
75     model.addAttribute("rangoEdad", rangoEdad);
76     return "url4";
77 }
78
79 model.addAttribute("nombre", nombre);
80 model.addAttribute("opcion", opcion);
81 model.addAttribute("opciones", opciones);
82 model.addAttribute("rangoEdad", rangoEdad);
83 return "url5";
84
85 @GetMapping("/url5")
86 public String url5(@RequestParam("nombre") String nombre, @RequestParam("opcion") String opcion,
87 @RequestParam(value = "opciones", required = false) List<String> opciones,
88 @RequestParam("rangoEdad") String rangoEdad, Model model) {
89
90     List<String> datos = new ArrayList<>();
91     datos.add("Nombre: " + nombre);
92     datos.add("Género: " + opcion);
93
94     if (opciones != null && !opciones.isEmpty()) {
95         for (String idioma : opciones) {
96             datos.add("Idioma: " + idioma);
97         }
98     } else {
99         datos.add("Idiomas seleccionados: Ninguno");
100     }
101     datos.add("Rango de edad: " + rangoEdad);
102 }
103
```



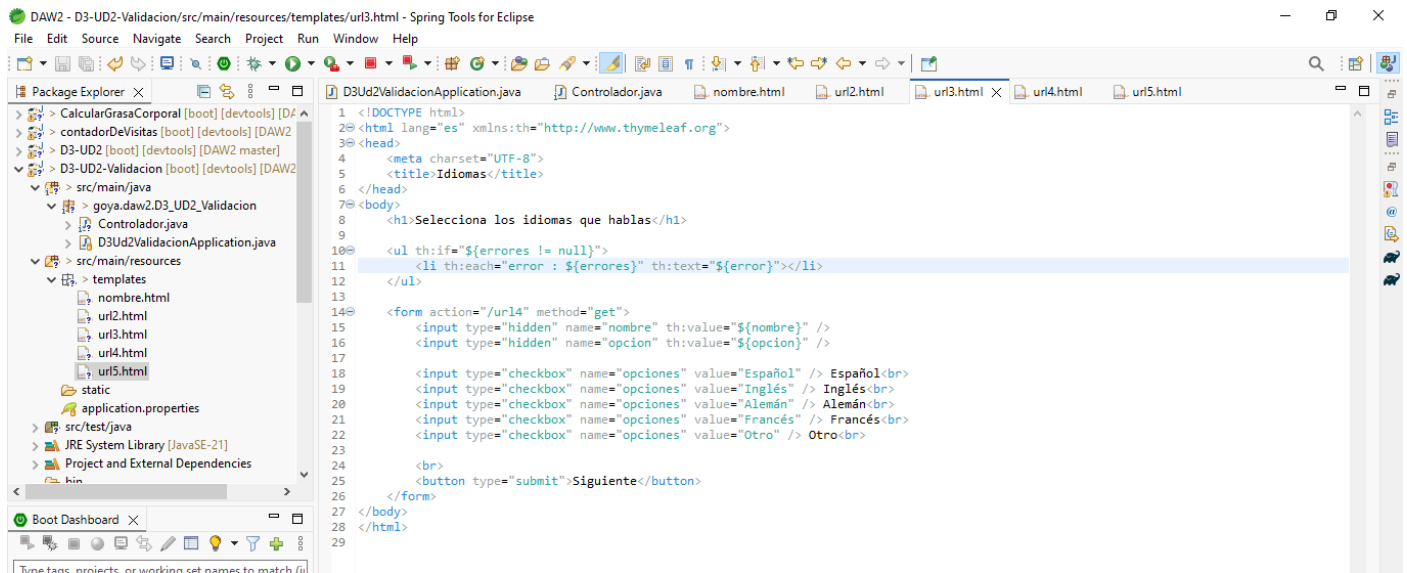
Creamos un HTML llamado nombre.html, para el nombre



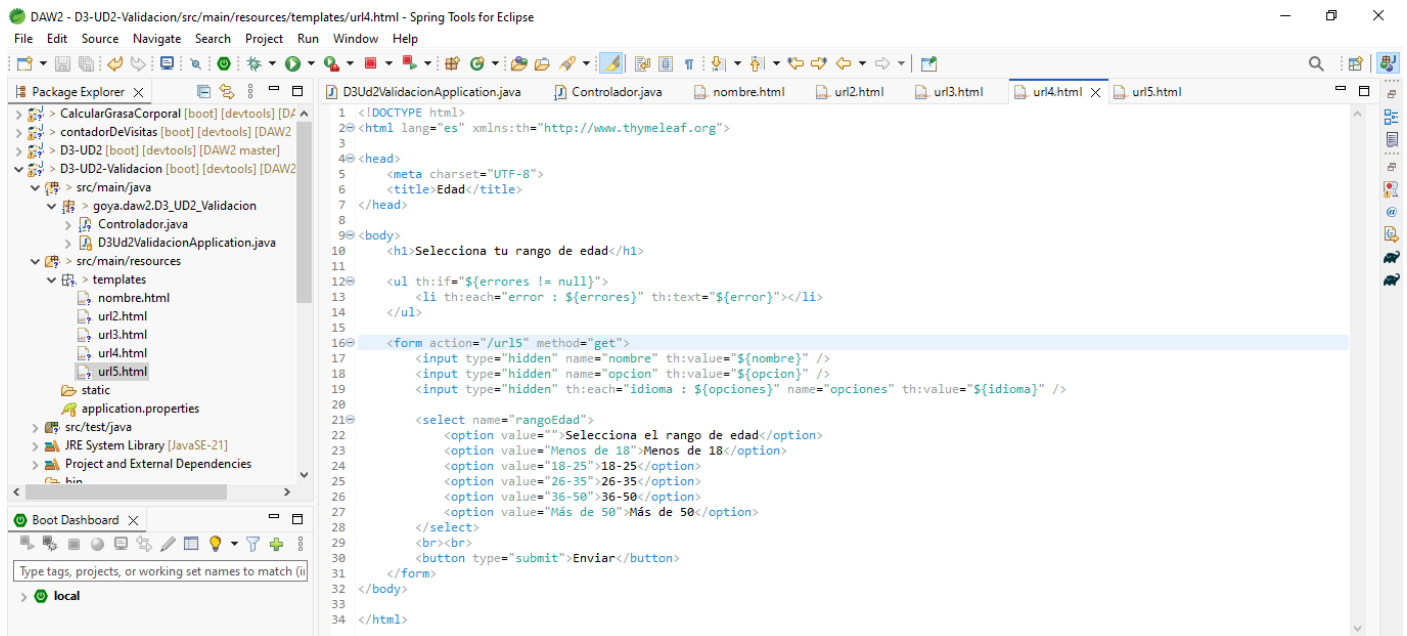
Creamos un HTML llamado url2.html, para seleccionar el género



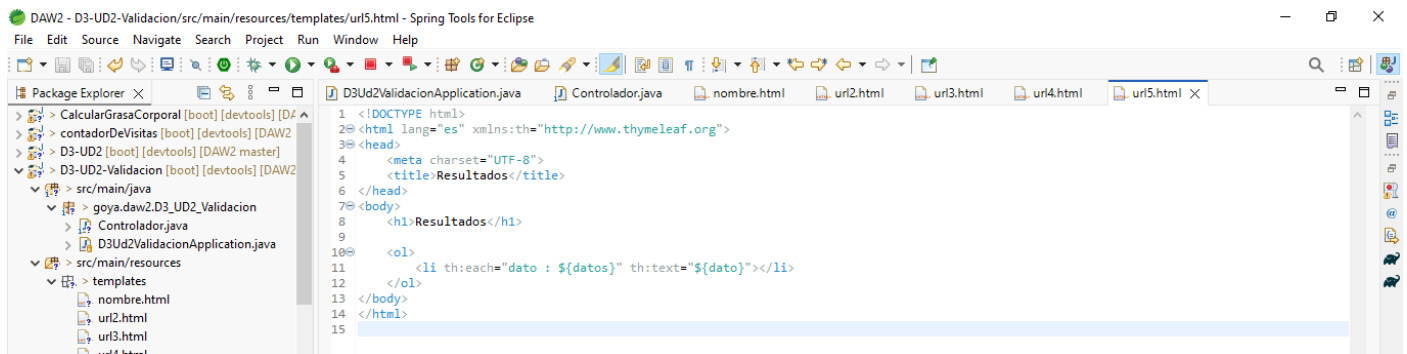
Creamos un HTML llamado url3.html, para seleccionar el idioma



Creamos un HTML llamado url4.html, para seleccionar el rango de edad



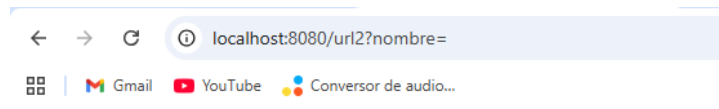
Creamos un HTML llamado url5.html, para ver los resultados



Probamos en el navegador:

http://localhost:8080/nombre

Primero tenemos que poner el nombre, si no lo ponemos nos sale el mensaje de es obligatorio

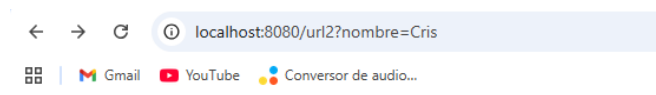


Introduce un nombre

- El nombre es obligatorio

Nombre:

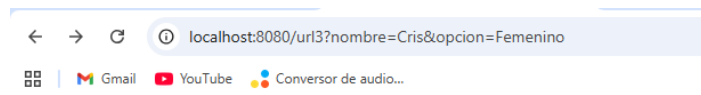
Nos pide el género



Selecciona un género

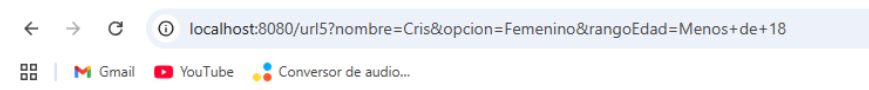
- ☐ Masculino
☒ Femenino
☐ Prefiero no decirlo

Nos pide la edad



Selecciona tu rango de edad

Nos imprime los resultados



Resultados

1. Nombre: Cris
2. Género: Femenino
3. Idiomas seleccionados: Ninguno
4. Rango de edad: Menos de 18

- Excelente: lo anterior y botón de ir atrás en cada etapa recodando valores.

Creas la clase Controlador

```

1 package goya.daw2.D3_UD2_Boton;
2
3 import org.springframework.stereotype.Controller;
4
5 @Controller
6 public class Controlador {
7
8     @GetMapping("/nombre")
9     public String nombre() {
10         return "nombre";
11     }
12
13     @GetMapping("/url2")
14     public String url2(@RequestParam("nombre") String nombre, Model model) {
15         List<String> errores = new ArrayList<>();
16         if (nombre == null || nombre.trim().equals("")) {
17             errores.add("El nombre es obligatorio");
18             model.addAttribute("errores", errores);
19             return "nombre";
20         }
21         model.addAttribute("nombre", nombre);
22         return "url2";
23     }
24
25     @GetMapping("/url3")
26     public String url3(@RequestParam("nombre") String nombre,
27                       @RequestParam("opcion") String opcion,
28                       Model model) {
29
30         List<String> errores = new ArrayList<>();
31         if (opcion == null || opcion.trim().equals("")) {
32             errores.add("El género es obligatorio");
33         }
34         if (!errores.isEmpty()) {
35             model.addAttribute("errores", errores);
36         }
37     }
38 }

```

```

39     @GetMapping("/url4")
40     public String url4(@RequestParam("nombre") String nombre,
41                       @RequestParam("opcion") String opcion,
42                       @RequestParam(value = "opciones", required = false) List<String> opciones,
43                       @RequestParam(value = "rangoEdad", required = false) String rangoEdad,
44                       Model model) {
45
46         List<String> errores = new ArrayList<>();
47         if (rangoEdad == null || rangoEdad.trim().equals("")) {
48             errores.add("El rango de edad es obligatorio");
49         }
50         if (!errores.isEmpty()) {
51             model.addAttribute("errores", errores);
52             model.addAttribute("nombre", nombre);
53             model.addAttribute("opcion", opcion);
54             model.addAttribute("opciones", opciones);
55             model.addAttribute("rangoEdad", rangoEdad);
56             return "url4";
57         }
58         model.addAttribute("nombre", nombre);
59     }
60
61     @GetMapping("/url5")
62     public String url5(@RequestParam("nombre") String nombre,
63                       @RequestParam("opcion") String opcion,
64                       @RequestParam(value = "opciones", required = false) List<String> opciones,
65                       @RequestParam(value = "rangoEdad", required = false) String rangoEdad,
66                       Model model) {
67
68         List<String> errores = new ArrayList<>();
69         if (opcion == null || opcion.trim().equals("")) {
70             errores.add("El género es obligatorio");
71         }
72         if (!errores.isEmpty()) {
73             model.addAttribute("errores", errores);
74             model.addAttribute("nombre", nombre);
75             model.addAttribute("opcion", opcion);
76             model.addAttribute("opciones", opciones);
77             model.addAttribute("rangoEdad", rangoEdad);
78             return "url5";
79         }
80         model.addAttribute("nombre", nombre);
81     }
82 }

```

DAW2 - D3-UD2-Boton/src/main/java/goya/daw2/D3_UD2_Boton/Controlador.java - Spring Tools for Eclipse

```
File Edit Source Refactor Navigate Search Project Run Window Help

Controlador.java X nombre.html url2.html url3.html url4.html url5.html D3Ud2BotonApplication.java

71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105

model.addAttribute("nombre", nombre);
model.addAttribute("opcion", opcion);
model.addAttribute("opciones", opciones);
model.addAttribute("rangoEdad", rangoEdad);

return "url4"; // mostrar el formulario de selección de idiomas y rango de edad
}

@GetMapping("/url5")
public String url5(@RequestParam("nombre") String nombre,
                  @RequestParam("opcion") String opcion,
                  @RequestParam(value="opciones", required=false) List<String> opciones,
                  @RequestParam("rangoEdad") String rangoEdad,
                  Model model) {

    List<String> datos = new ArrayList<>();
    datos.add("Nombre: " + nombre);
    datos.add("Género: " + opcion);

    if (opciones != null && !opciones.isEmpty()) {
        for (String idioma : opciones) {
            datos.add("Idioma: " + idioma);
        }
    } else {
        datos.add("Idiomas seleccionados: Ninguno");
    }

    datos.add("Rango de edad: " + rangoEdad);

    model.addAttribute("datos", datos);
    model.addAttribute("nombre", nombre);
    model.addAttribute("opcion", opcion);
    model.addAttribute("opciones", opciones);
    model.addAttribute("rangoEdad", rangoEdad);
}
```

DAW2 - D3-UD2-Boton/src/main/java/goya/daw2/D3_UD2_Boton/Controlador.java - Spring Tools for Eclipse

```
File Edit Source Refactor Navigate Search Project Run Window Help

Controlador.java X nombre.html url2.html url3.html url4.html url5.html D3Ud2BotonApplication.java

77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110

return "url4"; // mostrar el formulario de selección de idiomas y rango de edad
}

@GetMapping("/url5")
public String url5(@RequestParam("nombre") String nombre,
                  @RequestParam("opcion") String opcion,
                  @RequestParam(value="opciones", required=false) List<String> opciones,
                  @RequestParam("rangoEdad") String rangoEdad,
                  Model model) {

    List<String> datos = new ArrayList<>();
    datos.add("Nombre: " + nombre);
    datos.add("Género: " + opcion);

    if (opciones != null && !opciones.isEmpty()) {
        for (String idioma : opciones) {
            datos.add("Idioma: " + idioma);
        }
    } else {
        datos.add("Idiomas seleccionados: Ninguno");
    }

    datos.add("Rango de edad: " + rangoEdad);

    model.addAttribute("datos", datos);
    model.addAttribute("nombre", nombre);
    model.addAttribute("opcion", opcion);
    model.addAttribute("opciones", opciones);
    model.addAttribute("rangoEdad", rangoEdad);

    return "url5";
}
```

Creamos el HTML llamado nombre.html

DAW2 - D3-UD2-Boton/src/main/resources/templates/nombre.html - Spring Tools for Eclipse

```
File Edit Source Refactor Navigate Search Project Run Window Help

Controlador.java X nombre.html url2.html url3.html url4.html url5.html D3Ud2BotonApplication.java

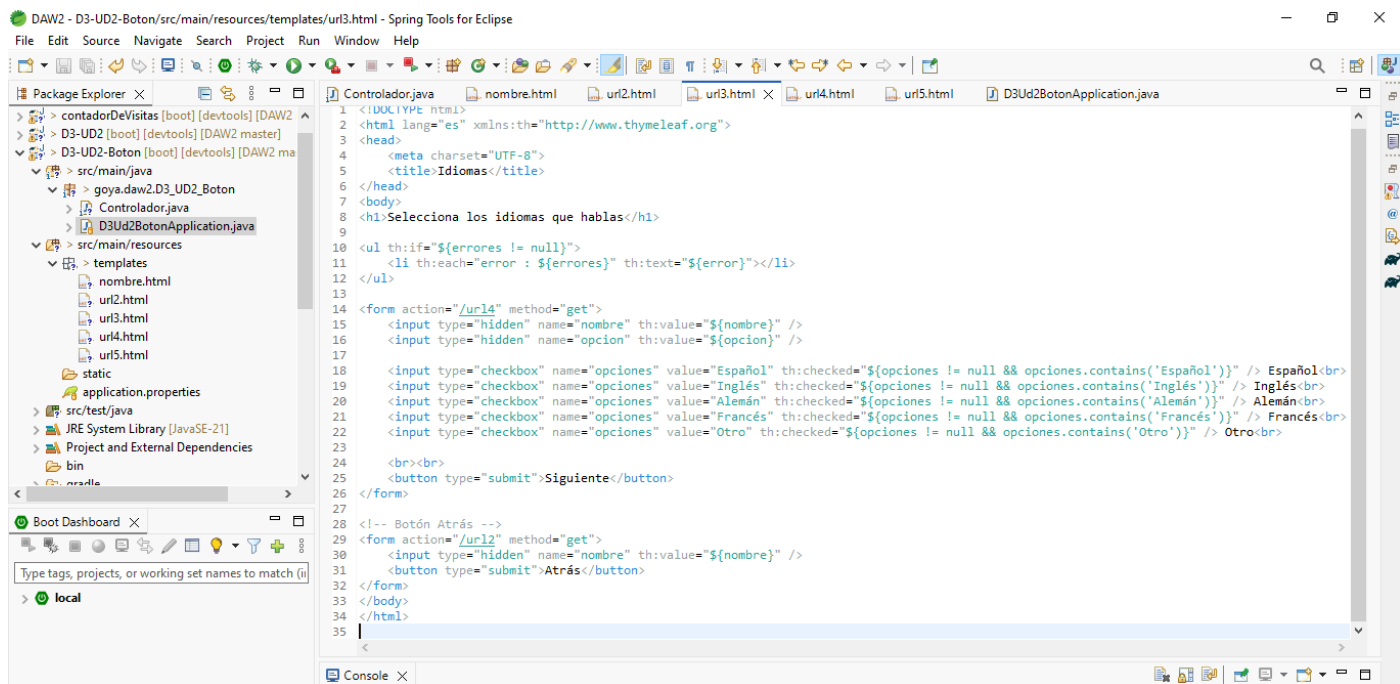
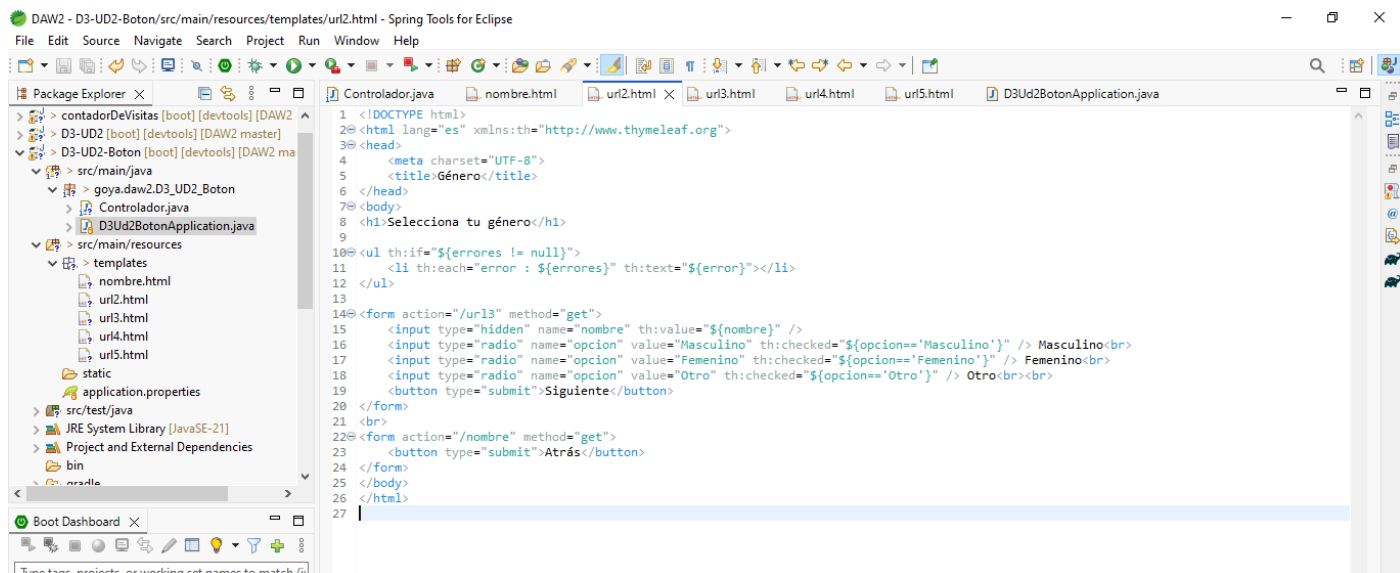
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21

<!DOCTYPE html>
<html lang="es" xmlns:th="http://www.thymeleaf.org">
<head>
<meta charset="UTF-8">
<title>Nombre</title>
</head>
<body>
<h1>Introduce tu nombre</h1>

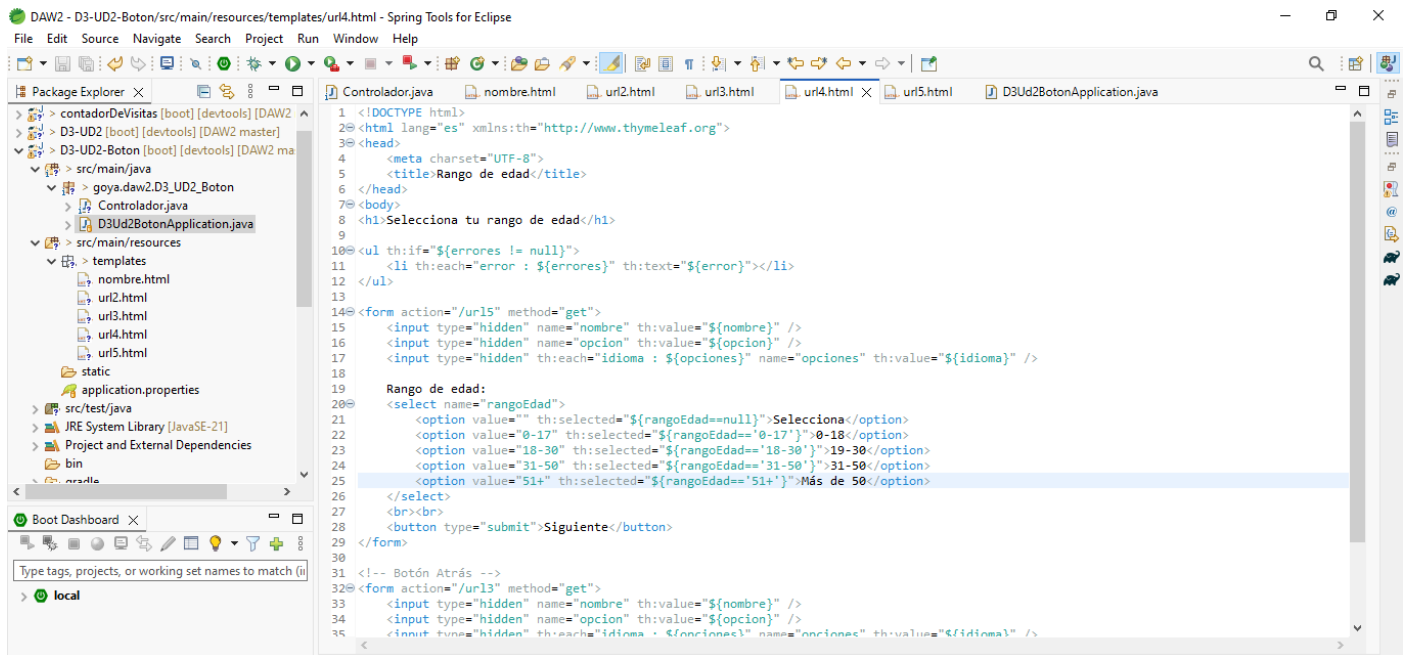
<ul th:if="${errores != null}">
<li th:each="error : ${errores}" th:text="${error}"></li>
</ul>

<form action="/url2" method="get">
<input type="text" name="nombre" th:value="${nombre}" />
<br><br>
<button type="submit">Siguiete</button>
</form>
</body>
</html>
```

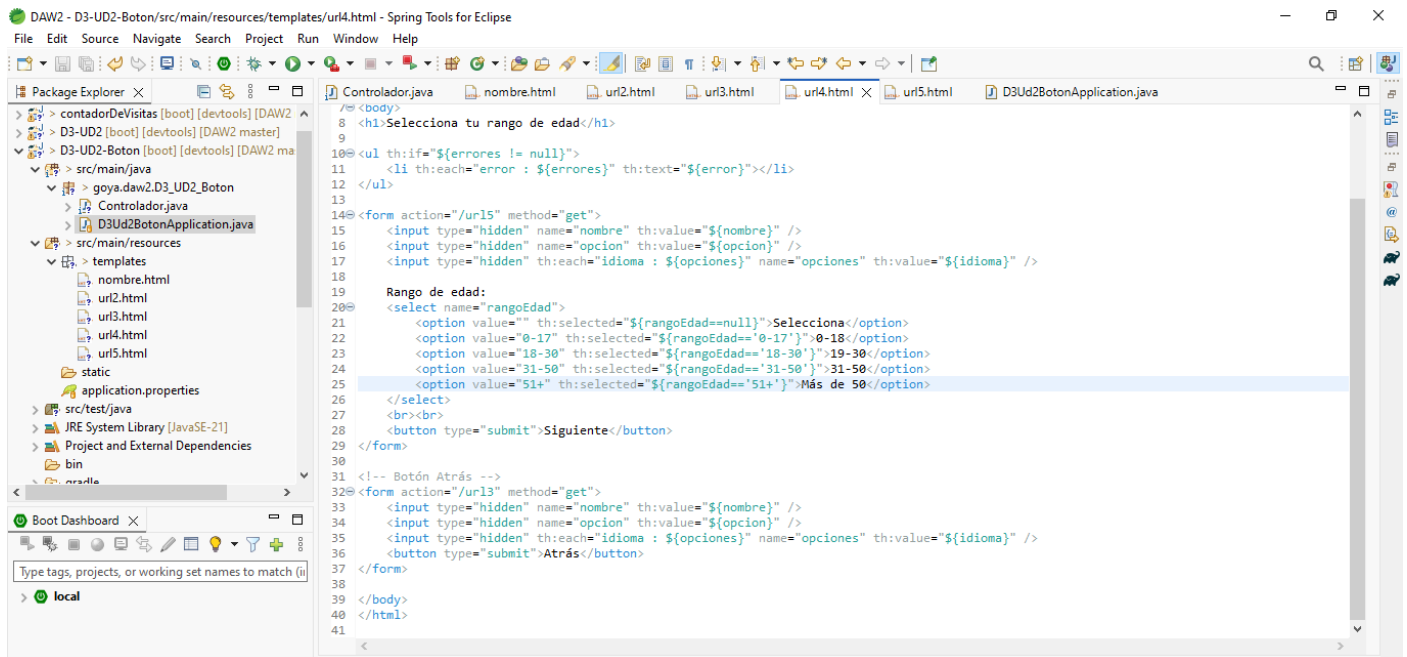

Creamos el HTML llamado url2.html con botón Atrás



Creamos el HTML llamado url4.html con botón Atrás

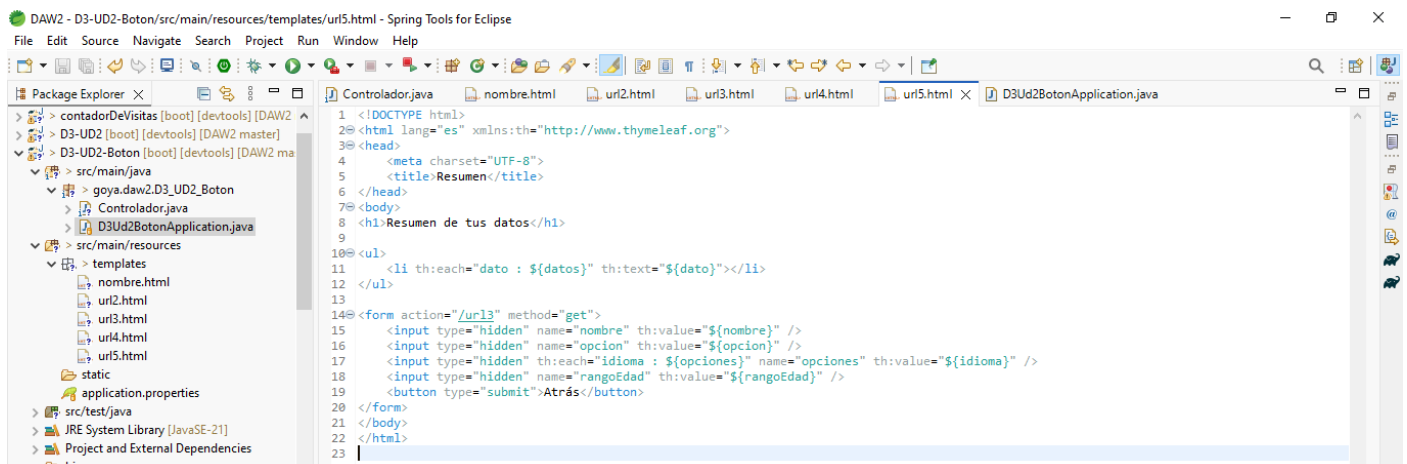


```
1 <!DOCTYPE html>
2 <html lang="es" xmlns:th="http://www.thymeleaf.org">
3 <head>
4 <meta charset="UTF-8">
5 <title>Rango de edad</title>
6 </head>
7 <body>
8 <h1>Selecciona tu rango de edad</h1>
9
10 <ul th:if="${errores != null}">
11 <li th:each="error : ${errores}" th:text="${error}"></li>
12 </ul>
13
14 <form action="/url5" method="get">
15 <input type="hidden" name="nombre" th:value="${nombre}" />
16 <input type="hidden" name="opcion" th:value="${opcion}" />
17 <input type="hidden" th:each="idioma : ${opciones}" name="opciones" th:value="${idioma}" />
18
19 Rango de edad:
20 <select name="rangoEdad">
21 <option value="" th:selected="${rangoEdad==null}">Selecciona</option>
22 <option value="0-17" th:selected="${rangoEdad=='0-17'}">0-18</option>
23 <option value="18-30" th:selected="${rangoEdad=='18-30'}">19-30</option>
24 <option value="31-50" th:selected="${rangoEdad=='31-50'}">31-50</option>
25 <option value="51+" th:selected="${rangoEdad=='51+'}">Más de 50</option>
26 </select>
27 <br><br>
28 <button type="submit">Siguiente</button>
29 </form>
30
31 <!-- Botón Atrás -->
32 <form action="/url3" method="get">
33 <input type="hidden" name="nombre" th:value="${nombre}" />
34 <input type="hidden" name="opcion" th:value="${opcion}" />
35 <input type="hidden" th:each="idioma : ${opciones}" name="opciones" th:value="${idioma}" />
```



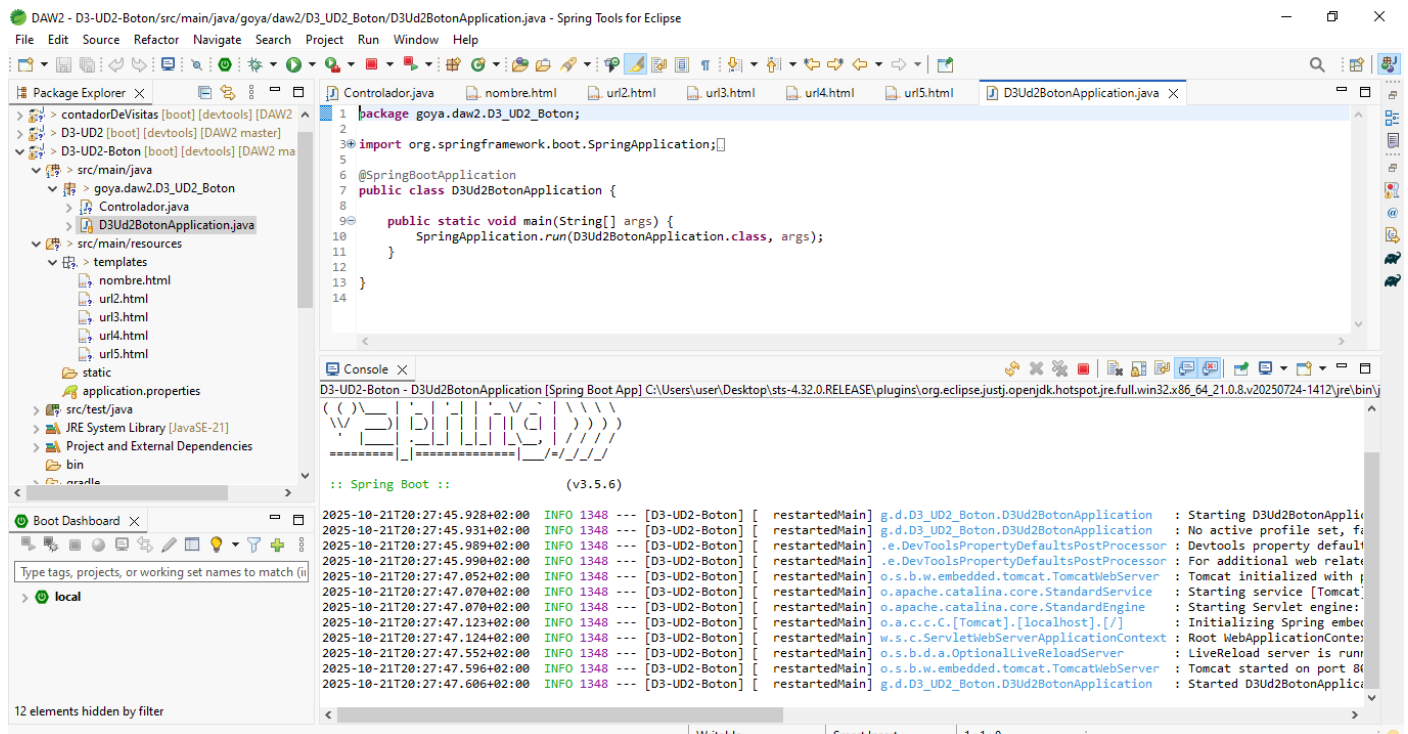
```
1 <!-- Botón Atrás -->
2 <form action="/url3" method="get">
3 <input type="hidden" name="nombre" th:value="${nombre}" />
4 <input type="hidden" name="opcion" th:value="${opcion}" />
5 <input type="hidden" th:each="idioma : ${opciones}" name="opciones" th:value="${idioma}" />
6 <button type="submit">Atrás</button>
7 </form>
8 </body>
9 </html>
```

Creamos el HTML llamado url5.html con botón Atrás



```
1 <!DOCTYPE html>
2 <html lang="es" xmlns:th="http://www.thymeleaf.org">
3 <head>
4 <meta charset="UTF-8">
5 <title>Resumen</title>
6 </head>
7 <body>
8 <h1>Resumen de tus datos</h1>
9
10 <ul>
11 <li th:each="dato : ${datos}" th:text="${dato}"></li>
12 </ul>
13
14 <form action="/url3" method="get">
15 <input type="hidden" name="nombre" th:value="${nombre}" />
16 <input type="hidden" name="opcion" th:value="${opcion}" />
17 <input type="hidden" th:each="idioma : ${opciones}" name="opciones" th:value="${idioma}" />
18 <input type="hidden" name="rangoEdad" th:value="${rangoEdad}" />
19 <button type="submit">Atrás</button>
20 </form>
21 </body>
22 </html>
```

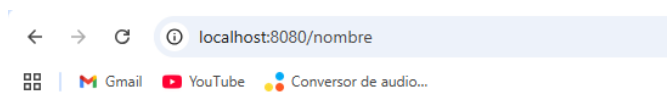
Iniciamos



Comprobamos en el navegador

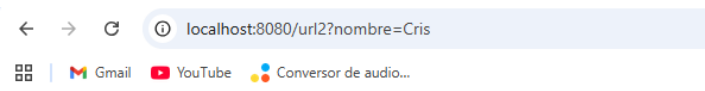
<http://localhost:8080/nombre>

Nos pide el nombre



Introduce tu nombre

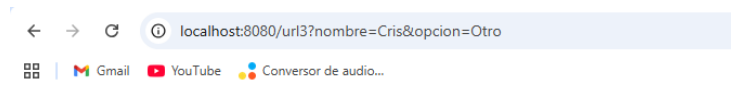
El género



Selecciona tu género

- ☐ Masculino
☐ Femenino
☐ Otro

Idiomas

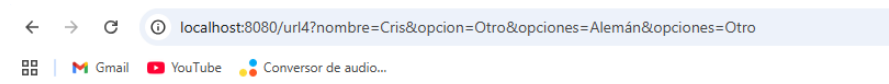


Selecciona los idiomas que hablas

- ☐ Español
- ☐ Inglés
- ☒ Alemán
- ☐ Francés
- ☒ Otro

Siguiente
Atrás

Edad



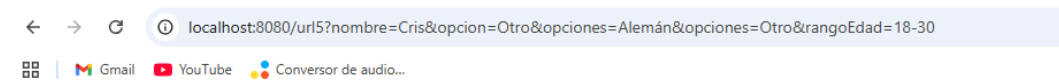
Selecciona tu rango de edad

- El rango de edad es obligatorio

Rango de edad: Selecciona ▼

Siguiente
Atrás

Nos muestra el resultado



Resumen de tus datos

- Nombre: Cris
- Género: Otro
- Idioma: Alemán
- Idioma: Otro
- Rango de edad: 18-30

Atrás

<https://github.com/Cristina200317/PracticasEntornoServidor.git>