

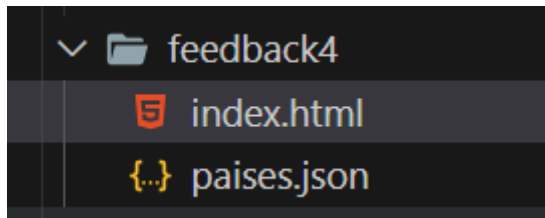
FEEDBACK 4 DESARROLLO ENTORNO CLIENTE

David Herrero Estévez – DAW 2

CONTENIDO

Estructura de directorios.....	3
Contenido fichero json	3
Contenido de index.html.....	4
Descripción de la funcionalidad	5
Capturas de pantalla de la aplicación funcionando.	6

ESTRUCTURA DE DIRECTORIOS



CONTENIDO FICHERO JSON

```
{
  "países": [
    {
      "nombre": "España",
      "ciudades": ["Madrid", "Barcelona", "Sevilla"]
    },
    {
      "nombre": "Francia",
      "ciudades": ["París", "Marsella", "Lyon"]
    },
    {
      "nombre": "Italia",
      "ciudades": ["Roma", "Milán", "Florencia"]
    },
    {
      "nombre": "Alemania",
      "ciudades": ["Berlín", "Múnich", "Hamburgo"]
    },
    {
      "nombre": "Reino Unido",
      "ciudades": ["Londres", "Manchester", "Edimburgo"]
    },
    {
      "nombre": "Portugal",
      "ciudades": ["Lisboa", "Oporto", "Coímbra"]
    },
    {
      "nombre": "Suecia",
      "ciudades": ["Estocolmo", "Gotemburgo", "Malmö"]
    }
  ]
}
```

Con la ayuda de chatgpt se ha generado un fichero json con varios países añadiendo 3 ciudades a cada país.

CONTENIDO DE INDEX.HTML

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js">
  script>
  <script>
    function cargaValores(donde,valores){
      var d=document.getElementById(donde);
      for (let i=0;i<valores.length;i++){
        var o=document.createElement("option");
        o.text=valores[i];
        o.value=valores[i];
        d.appendChild(o);
      }
    }
  }
```

Incluimos la librería de jquery directamente desde googleapis

```

10 function cargaValores(donde,valores){
11     var d=document.getElementById(donde);
12     for (let i=0;i<valores.length;i++){
13         var o=document.createElement("option");
14         o.text=valores[i];
15         o.value=valores[i];
16         d.appendChild(o);
17     }
18 }
19 $(document).ready(function(){
20     //inicialmente cargamos los países
21
22     $.getJSON("países.json", function(data) {
23         // La variable 'data' contendrá el contenido del archivo JSON
24         console.log(data);
25         // Crear un array asociativo
26         var paísesCiudades = {};
27
28         // Recorrer la lista de países y construir el array asociativo
29         $.each(data.países, function(index, país) {
30             paísesCiudades[país.nombre] = país.ciudades;
31         });
32         console.log(paísesCiudades);
33         cargaValores("países",Object.keys(paísesCiudades));
34         cargaValores("ciudades",Object.values(paísesCiudades["España"]));
35         $("#países").change(function(){
36             $("#ciudades").empty();
37             cargaValores("ciudades",Object.values(paísesCiudades[$(this).val()]));
38         });
39     });

```

DESCRIPCIÓN DE LA FUNCIONALIDAD

Cuando el documento está cargado, usando jquery obtenemos los datos de países.json , y pasamos los datos cargados a la función anónima.

Esa función recorre los datos con la función “.each” y va creando el array asociativo de países con sus ciudades correspondientes.

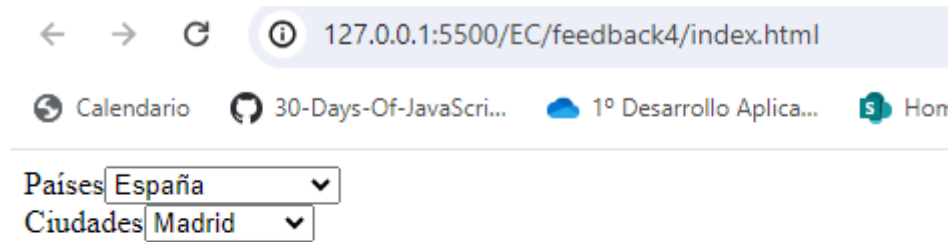
Una vez obtenido el array asociativo (realmente es un mapa), llamamos a la función cargaValores.

Primero pasándole como parámetro países, y le pasamos las claves del mapa creado anteriormente.

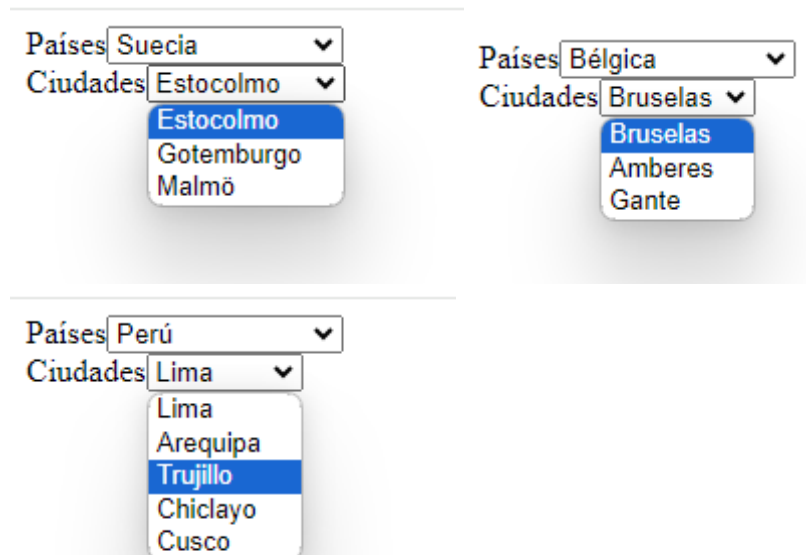
Posteriormente llamamos a la misma función pasándole como parámetro el id del select ciudades, con los valores (values) del mapa creado anteriormente.

Finalmente usando jquery indicamos que cuando se cambie el valor seleccionado en el select países, se vacíe el select de ciudades, y se cargue de nuevo pasándole las ciudades correspondientes al país seleccionado.

CAPTURAS DE PANTALLA DE LA APLICACIÓN FUNCIONANDO.



Valores que se muestran al cargar la pantalla



Distintos valores que toma el select (combo) de ciudades.