

DWC\examenFInal\asincronia\promesasaniadas.html

```
1 <!DOCTYPE html>
2 <html lang="es">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta http-equiv="X-UA-Compatible" content="IE=edge">
7   <meta name="viewport" content="width=device-width, initial-scale=1.0">
8   <style>
9     main{display: flex;
10     flex-direction: column;
11     gap: 2rem;
12   }
13
14   form {
15     display: flex;
16     flex-direction: column;
17     gap: 2rem;
18   }
19
20   fieldset {
21     display: flex;
22     flex-direction: column;
23     gap: 2rem;
24   }
25
26   fieldset>div {
27     display: flex;
28     flex-direction: row;
29     gap: 2rem;
30   }
31
32   .radio {
33     display: block;
34   }
35 </style>
36
37 </head>
38
39 <body>
40   <header>
41
42   </header>
43
44   <main>
45
46
47     <div id="container">
48       <p id="rdo"></p>
49     </div>
50   </main>
51   <script>
52     const rdo = document.querySelector('#rdo');
53     const contenedor =document.querySelector('#container');
54
55     const buscarDatos = () => {
56       return new Promise ((resolve, reject) => {
57         setTimeout(() => {
```

```
58     let exito = Math.random() > 0.1 ? true : false;
59     if (exito){
60         rdo.textContent = 'Datos encontrados';
61         setTimeout(() => {
62             resolve(['pequeño', 'cabaret', 'ambulante']);
63         }, 2000);
64     }
65     else{
66         reject ("Hubo un error");
67     }
68 }
69 }, 2000);
70 });
71 }
72
73
74 const procesarDatos = (datos) => {
75     return new Promise((resolve) => {
76         setTimeout(() => {
77             const resultados = datos.map(nombre => nombre.toUpperCase());
78             resolve (resultados);
79         }, 2000);
80     });
81 }
82
83 const mostrarResultados = (resultados) => {
84     return new Promise((resolve) => {
85         setTimeout(() => {
86             resultados.forEach(resultado =>{
87                 //contenedor.createElement;
88                 let parr = document.createElement('p');
89                 parr.textContent = resultado;
90                 contenedor.append(parr);
91             });
92             resolve();
93         }, 2000)
94     });
95 }
96
97 buscarDatos()
98 .then(datos => {
99     rdo.textContent = 'Datos procesados';
100     return procesarDatos(datos)})
101 .then(resultados => {
102     rdo.textContent = 'Resultados'
103     return mostrarResultados(resultados)})
104 .catch(error => {
105     let err = document.createElement('p');
106     err.textContent = "ha ocurrido algo malo";
107     contenedor.append(err);
108 })
109 .finally(()=>{
110     setTimeout(() => {
111         let parr = document.createElement('p');
112         parr.textContent = 'Proceso terminado';
113         container.append(parr);
114     }, 1000);
115 });
116
117 </script>
```

```
118 | </body>  
119 |  
120 | </html>
```