







Nos han dado acceso a una BBDD de usuarios a través de una API REST de usuarios. Tenemos un problema, y es que no sabemos el género de las personas que hemos dado de alta, ni la edad y además, las direcciones son inventadas, pero la GEOLOCALIZACIÓN si es correcta, ya que se hace de forma automática.

Todo la parte de la Vista de la aplicación está finalizada, pero hay que terminar ciertos aspectos del Javascript.

Lo que se busca es:

- [2 punto] Que se carguen todos los nombres de los usuarios en un select:



Para obtgener la información debemos recurrir a la URL siguiente:

https://jsonplaceholder.typicode.com/users

El código HTML resultante debería ser como el siguiente

```
coption >Selectiona...
coption >Selecciona...
coption class="" id="" value="">
coption class="" id="" value="1">Leanne Graham
coption class="" id="" value="2">Ervin Howell
coption class="" id="" value="3">Clementine Bauch
coption class="" id="" value="4">Patricia Lebsack</ption>
coption class="" id="" value="5">Chelsey Dietrich</ption>
coption class="" id="" value="6">Mrs. Dennis Schulist
coption class="" id="" value="6">Kurtis Weissnat
coption class="" id="" value="8">Nicholas Runolfsdottir V</option>
coption class="" id="" value="9">Glenna Reichert
coption class="" id="" value="10">Clementina DuBuque
```

- [2 puntos] Que al seleccionar un usuario, se carguen todos sus datos en un div con id="info".













Para conseguir saber el género, la edad y la ciudad a través del primer nombre de cada usuario, tenemos que hacer tres peticiones independientes.

- [1 punto] Por un lado tenemos la siguiente URL para la edad:

https://api.genderize.io?name=[nombre]

- [1 punto] Por otro tenemos la siguiente para el sexo. En función de el sexo que prediga la API, pondremos como foto, male.png o female.png.

https://api.agify.io/?name=[nombre]

- [1 punto] Y por último, tendremos la siguiente:

https://geocode.xyz/[lat],[lng]?json=1

para el que necesitaremos la longitud y latitud obtenidas con la información de cada usuario. Esta API devuelve un JSON como el siguiente:

```
'geonumber": "4076544375625",
"longt": "0.00000",
"matches": null,
"error": {
  "description": "23. Your request did not produce any results.",
 "code": "008"
},
"suggestion": {
  "south": {
   "longt": "-48",
   "city": "Endurance Canyon",
   "latt": "-69.5"
 "territory": {},
 "altname": {},
 "region": {},
  "subregion": {},
 "north": {
   "longt": {},
   "prov": {},
   "city": {},
   "distance": {},
   "latt": {}
 "iso": {},
 "eez": {}
 latt": "0.00000"
```









Donde deberemos quedarnos con la ciudad sugerida (norte o sur).

El resultado HTML obtenido, debería ser como el siguiente:

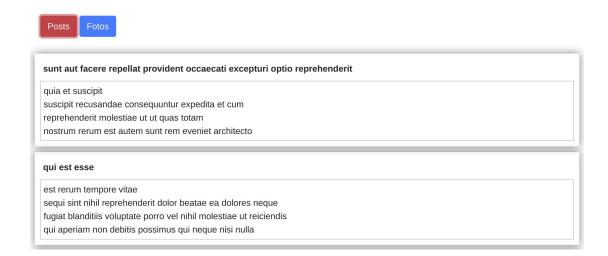
```
<div id="info">
   <div id="foto">
       <imq src="img/female.png">
   <div class="titulo">Nombre</div>
   <div class="descripción">Leanne Graham</div>
   <div class="titulo">Edad</div>
   <div class="descripción">41 años</div>
   <div class="titulo">email</div>
   <div class="descripción">
       <a href="mailto:Sincere@april.biz">Sincere@april.biz</a>
   </div>
   <div class="titulo">Ciudad</div>
   <div class="descripción">Zeewolf Fracture Zone</div>
   <div class="titulo">Web</div>
   <div class="descripción">
       <a target="_blank" href="hildegard.org">hildegard.org</a>
   </div>
```

Hay que tener en cuenta que tanto el e-mail como la web, serán enlaces.

Una vez obtenidas la ficha, veremos que aparecen dos botones, que ya existen, pero están ocultos con la clase .oculto. Cuando carguemos un usuario, deberemos eliminar la clase del div con id="botonera".

Cada uno de estos botones hará una llamada a la API de los usuarios, mostrando:

- [2 puntos] En el primer caso los posts que el usuario seleccionado haya hecho y que se cargarán en el div con id="posts".











Estos posts constan de un título (title), y un cuerpo (body). La estructura HTML resultante es la siguiente.

- [2 puntos] Y en el segundo caso cargaremos los álbumes a través de la siguiente URL:

https://jsonplaceholder.typicode.com/users/[idUser]/posts

Estos álbumes se cargarán en el div con id="albumes".



que generará una estructura com la siguiente:

```
id="albums">

<a class="album" id="1" href="#">sunt aut facere repellat provident occaecati excepturi optio
reprehenderit</a>
<a class="album" id="2" href="#">qui est esse</a>
<a class="album" id="3" href="#">ea molestias quasi exercitationem repellat qui ipsa sit aut</a>
<a class="album" id="4" href="#">eum et est occaecati</a>
<a class="album" id="4" href="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#">pesf="#"
```

- [1 punto] Al pulsar sobre cada uno de estos álbumes, se caragarán las fotos respectivas utilizando el siguiente enlace:

https://jsonplaceholder.typicode.com/albums/[idAlbum]/photos

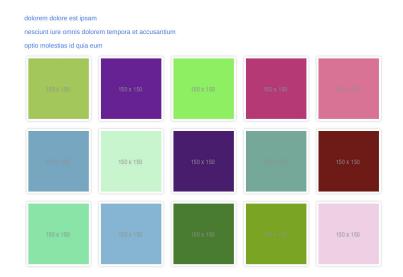
Y mostrará las fotos de cada album en el div con id="fotos"











La estructura HTML resultante, será como la que se muestra a continuación.

Con toda esta información nos piden resolver la aplicación en las próximas 3 horas, ya que saldrá a producción en breve.

Los puntos que se obtendrán por cada parte están indicados a lo largo de todo el documento.

Añadir comentarios en el código, por favor!