Ejercicios AJAX / XHR

Nociones

Crear un elemento

const | let nombreVariable = document.createElement("etiquetaHTML");

Todos los elementos creados deberíamos añadirlos luego a algún elemento existente en el HTML con su método appendChild (para añadir al final), insertBefore (para añadir antes de un elemento), …

Ejemplo:

```
const img = document.createElement("img");
document.getElementById("body").appendChild(img);
```

Modificar atributos

Si queremos modificar un atributo, como src, id, alt, …. Usaremos:

elemento.nombreAtributo = valor:

Ejemplo:

```
const img = document.createElement("img");
img.src = "imágenes/foto.png";
```

Añadir una o varias clases clases

elemento.classList.add("nombreClase");

elemento.classList.add("nombreClase", "nombreClase", …);

Ejemplo:

```
const img = document.createElement("img");
img.classList.add("d-block");
```

Eliminar una o varias clases

elemento.classList.remove("nombreClase");

elemento.classList.remove("nombreClase", "nombreClase", …);

Ejemplo:

```
const img = document.getElementById("cuadro");
img.classList.remove("d-block", "m-2");
```

Alternar una clase

SI la tiene, la quita, si no la tiene, la pone.

elemento.classList.toggle("nombreClase");

Ejemplo:

```
const img = document.getElementById("cuadro");
img.classList.toggle("d-none");
```

Ejercicios AJAX / XHR

Modificar una propiedad de estilo

elemento.style.nombrePropiedad = valor;

Ejemplo:

```
document.getElementById("cuadro").style.color = "red";
```

Ejercicio Mensaje

Usar esta API para mostrar el título y cuerpo del mensaje (title y body), sustituyendo número por el número que introduzca el usuario en el cuadro de texto:

https://jsonplaceholder.typicode.com/posts/**numero**

Estos datos deben crearse en un div tal como aparece en el documento HTML comentado.

Ejercicio Mensajes

Parecido al anterior, sin cuadro de texto, pero ahora mostrar todos los mensajes que devuelve esta URL:

```
https://jsonplaceholder.typicode.com/posts
```

Ejercicio Usuarios

Ahora debemos mostrar todos los usuarios que devuelve esta URL:

```
https://jsonplaceholder.typicode.com/users
```

Ejercicio Chistes

Usar esta API para mostrar un chiste:

https://v2.jokeapi.dev/joke/Any?lang=es

Debemos tener en cuenta que en ocasiones el chiste tiene dos partes, setup y delivery y otras solo una parte, joke.

Ejercicio Gatos

Usar esta API para ir mostrando una foto de un gato al pulsar el botón:

```
https://api.thecatapi.com/v1/images/search
```

Horóscopo

Usar esta API para mostrar el horóscopo del signo elegido, debiendo cambiar signo por el signo elegido por el usuario (en minúsculas):

https://horoscope-app-api.vercel.app/api/v1/get-horoscope/daily?sign=**SIGNO**&day=TODAY

También debe cambiarse la imagen por la del signo zodiacal elegido.

Población

Usar esta API para mostrar los datos de la población de la ciudad escrita por el usuario:

https://countriesnow.space/api/v0.1/countries/population/cities

La ciudad debe pasarse como una petición POST de esta forma:

```
{
    "city": "ciudadDeseada"
}
```

Recetas

Mostrar todas las recetas creando divs tal como aparece en el documento HTML comentado. La imagen en la receta debe aparecer de fondo del div, usando la propiedad style de JavaScript y no tal como se muestra en el div existente:

https://dummyjson.com/recipes

Hacer que el botón se desactive hasta que termine la petición, de forma correcta o no. Para simular un tiempo largo, la URL anterior admite el parámetro delay con los milisegundos de espera:

https://dummyjson.com/recipes?delay=1000 (cambiar 1000 por lo deseado)

Tráfico

Mostrar todos los avisos de tráfico de Vigo, clonando el nodo que está incluido en la página HTML (está oculto por defecto), clonando el elemento section que está en la página HTML:

https://datos.vigo.org/data/trafico/avisos-trafico-gl.json