|  |  |
| --- | --- |
| **Desarrollo de Aplicaciones en Entorno Cliente**  **2º Ciclo Superior de Desarrollo de Aplicaciones Web**  **2022-2023** | **barcoFomento.png** |
| **Tarea 4. jQuery. AJAX.** |  |

1. Si se hace clic en los elementos *li* que pertenezcan a la clase *country* se conmuta la clase *enabled*. Es decir, se añade la clase *enabled* al elemento si no está presente y se elimina si lo está.
2. Si se hace clic en los nodos con identificador *Spain* o *USA* se muestra una alerta indicando el nombre del país que se ha pulsado, “*Spain*” o “*United States of America*”, respectivamente.
3. **(**Implementar los botones removeAlert.png para borrar los países activados cuando se pulsen, teniendo en cuenta que existe un botón para cada continente.
4. Empleando la caja de texto situada en la parte superior derecha de la página, implementar un filtro por nombre de tal modo que, cada vez que se teclee en la caja de texto, se muestran únicamente aquellos países cuyo nombre contenga el texto escrito.

Por ejemplo, si se introduce “w”, se deben mostrar únicamente “Norway”, “Sweden” y “Switzerland”. Si a continuación se teclea “e”, se muestra solamente “Sweden”.

**Observaciones:**

Emplee el evento *keyup* para la detección de cambios en la caja de texto. El filtrado puede basarse en ocultar los países que no coincidan con la búsqueda; emplee el método .hide() para ocultar elementos. Puede ayudarse del método $.*grep()* para filtrar los países. Revise los ejemplos que emplean todos los métodos anteriores.

1. Implementar los botones settings.png para que recarguen todos los países de cada continente cuando se pulsen, teniendo en cuenta que existe un botón para cada continente. La recarga de la información se realiza a través de AJAX mediante una petición GET al script getCountires.php. Como se puede comprobar en el código del script, éste recibe un parámetro “continent” cuyo valor es el nombre del continente cuyos países se pretende recargar (“Europe” o “NorthAmerica”).

**Observaciones:**

Emplee la función $.get() para gestionar la petición AJAX y JSON.parse() para interpretar la respuesta. Revise los ejemplos que emplean los métodos anteriores.

Con el propósito de evitar errores CORS, recuerde probar scripts que involucren AJAX solicitando las páginas a un servidor web, introduciendo en su navegador URLs locales (<http://localhost/tarea4.html>) en lugar de realizar peticiones de archivos ([C:\Users\alumno\tarea4.html](about:blank)).

A tal efecto se recomienda el empleo de Apache integrado en XAMPP. Antes de probar código que incluya AJAX, compruebe que su navegador responde a peticiones del script getCountries.php. Por ejemplo, <http://localhost/getCountries.php?continent=Europe>