

PROYECTO01_EVA2: API File

- Se pide una aplicación que abra uno o más archivos de texto mostrando por pantalla su nombre y contenido.

Funcionamiento:

- Crearemos uno o más archivos con cualquier editor de texto plano. Escribiremos un cierto texto en él. Lo guardaremos como uno de estos tres formatos: **csv**, **lst** y **txt**.
- Nuestra aplicación abrirá ese fichero con un botón adecuado para tal labor. Mostrando por pantalla el resultado.
- La aplicación iniciará con una única línea javascript en el archivo HTML llamada `iniciaListeners()` que se encargará únicamente de gestionar el/los eventos.

Indicaciones:

- El código html deberá dar opción a abrir únicamente los tipos de archivos señalados.
- El código JavaScript deberá ser investigado e implementado consultando los documentos necesarios. En los siguientes enlaces encontrarás información detallada sobre el modo en que deberás hacerlo.

- ✚ <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input/file>
- ✚ https://developer.mozilla.org/en-US/docs/Web/API/File_API/Using_files_from_web_applications
- ✚ <https://developer.mozilla.org/en-US/docs/Web/API/FileReader>
- ✚ <https://developer.mozilla.org/en-US/docs/Web/API/FileReader/FileReader>

- Nos hará falta conocimientos básicos en gestión de eventos con *listeners*. Consulta este código inscrito en una etiqueta BODY como ejemplo de uso:

```
<p>Uno</p><p>Dos</p><p>Tres</p><p>Cuatro</p>
<script>
  function miFuncion(ev) { //ev es el objeto Evento creado
    console.log(ev.target.textContent); /*target es el elemento
  }           //donde se produjo el evento (en este caso -> evento click)*/
  let parrafos = document.querySelectorAll("p");
  for (let parrafo of parrafos) {
    parrafo.addEventListener("click", miFuncion); //Solo el nombre
  }
</script>
```

Si quieres ampliar la información sobre gestión de eventos por *Listeners* (lo veremos mejor en el tema siguiente) puedes consultar la página:

https://www.w3schools.com/jsref/met_element_addeventlistener.asp

Salida:

- La aplicación creará los elementos html (DOM) donde aparezca la información solicitada: nombre y contenido del/los ficheros abiertos.
- El código resultante se organizará en ficheros `index.html`, `script.js` y `estilos.css` (este último no es necesario). Estos dos últimos en sus respectivas carpetas JS y CSS.

Salida documentada:

- ✓ Es importante la correcta documentación de métodos (JSDoc) y líneas de código que se consideren importantes y/o complejas.
- ✓ Debido a que este proyecto está orientado a la búsqueda de información y posterior implementación del código, se requerirá una correcta explicación de API, objetos y métodos usados. Elabora un documento en formato *markdown* donde se expliquen de modo breve pero paso a paso cómo has hecho el proyecto.
- ✓ Nomenclatura de variables y métodos clara, breve y adecuada.

Entrega (2 opciones):

- 🔗 Enviar enlace github con el proyecto al usuario **belisoc**. El nombre del proyecto será **PROYECTO01EVA2_API**.
- 🔗 *moodleSed*.