

Manipulando elementos con JavaScript

Práctica integradora

Objetivo

Como hemos visto, podemos implementar JavaScript del lado del servidor (**back-end**). Sin embargo, es importante poder añadir características interactivas a nuestro sitio web —por ejemplo: juegos, eventos que ocurren cuando los botones son presionados o los datos son introducidos en los formularios, efectos de estilo dinámicos, animaciones, y mucho más—. Para esto, deberemos utilizar JavaScript, pero del lado del cliente (**front-end**). Veamos qué posibilidades nos brinda.

¡Buena suerte! 🤙👍 ✨



Micro desafío - Paso 1:

Utilizaremos de base el siguiente [proyecto creado con Express](#) (recordemos instalar todas las dependencias del proyecto, ejecutando el comando **npm install** 😊). Además, aprovecharemos la base de datos [movies db](#) (no olvidemos activar el servicio de MySQL en nuestro equipo). De esta manera, todo funcionará correctamente. No debemos asustarnos si no visualizamos nada cuando ejecutemos **npm test** para activar el servidor y, luego, en el browser ejecutemos: <http://localhost:3001/>. Lo que ocurre es que el contenedor padre de la página posee una propiedad de CSS llamada: **display : none**. Una vez realizado todos los pasos anteriores, debemos hacer lo siguiente:

- Crear un archivo JavaScript: **/public/js/index.js**, y vincularlo con el archivo: **/views/index.ejs**.
- Desde el archivo **/public/js/index.js**, capturar los siguientes elementos: **<main>**, **<h2>**, **<a>** y **<p>**, ubicados en el archivo: **/views/index.ejs**.
- Haciendo uso del **prompt**, indicar al usuario que: **"Ingrese su nombre"**.
- En caso de que el usuario no coloque su nombre, a la etiqueta **<h2>**, se le debe agregar la palabra "Invitado". En caso contrario, se le debe agregar el nombre que el usuario ingresó.
- Agregar a la etiqueta **<h2>** el estilo **uppercase**.
- A la etiqueta **<a>**, colocarle el estilo correspondiente para que asuma el siguiente color: **#E51B3E**.
- Mediante el **confirm**, preguntar al usuario **"¿Desea colocar un fondo de pantalla?"**. Si la respuesta es afirmativa por parte del usuario, agregar al **<body>** la clase **"fondo"**.
- A todos los párrafos que fueron capturados, asignar a los pares la clase: **"destacadoPar"**. Y a los impares agregar la clase: **"destacadoImpar"**.
- Finalmente, establecer como visible a la etiqueta **<main>** en el browser o navegador, aplicando el estilo: **display : block**.



Micro desafío - Paso 2:

- Crear un archivo JavaScript: `/public/js/moviesList.js`, y vincularlo con el archivo: `/views/moviesList.ejs`.
- Desde el archivo `/public/js/moviesList.js`, capturar los siguientes elementos: `<body>` y `<h1>`, ubicados en el archivo `/views/moviesList.ejs`.
- Haciendo uso del **prompt**, preguntar al usuario: “¿Desea modo oscuro?”. Si la respuesta es afirmativa, agregar a la etiqueta `<body>` un color de fondo: `#7f7f7f`, y la clase: `fondoMoviesList`.
- Agregar a la etiqueta `<h1>` el mensaje: “LISTADO DE PELÍCULAS”.
- Agregar a la etiqueta `<h1>` los siguientes estilos:
 - Color de la fuente: `white`.
 - Color de fondo: `teal`.
 - Relleno: `20px`.



Micro desafío - Paso 3:

- Crear un archivo JavaScript: `/public/js/moviesAdd.js`, y vincularlo con el archivo: `/views/moviesAdd.ejs`.
- Desde el archivo `/public/js/moviesAdd.js`, capturar los siguientes elementos: `<h1>`, `<section>` y `<article>`, ubicados en el archivo: `/views/moviesAdd.ejs`.
- Agregar a la etiqueta `<h1>` el mensaje: “AGREGAR PELÍCULAS”.
- Agregar a la etiqueta `<h1>` la clase: “`titulo`”.
- Agregar al artículo la clase: “`fondoTransparente`”.
- Agregar a la sección la clase: “`fondoCRUD`”.



Bonus Track:

Si logramos realizar toda la práctica, una buena idea es replicar el proceso para las vistas: `moviesEdit.ejs` y `moviesDetail.ejs`. Podemos ejecutar sobre ellas todos los ajustes que deseemos.

Conclusión

Con esta práctica pudimos comprobar la ventaja de manipular el **DOM**, ya que permite a los programadores disponer de un control muy preciso sobre la estructura del documento HTML que están trabajando. Las funciones que proporciona DOM permiten añadir, eliminar, modificar y reemplazar cualquier nodo de cualquier documento de forma sencilla.

No quedan dudas del poder que tiene JavaScript sobre nuestro front-end. Más adelante, veremos cómo incorporar interacción a nuestro sitio web, haciendo uso de los eventos. Estas son acciones u ocurrencias que suceden en el sistema que estamos programando, y que este nos informa para poder responder de alguna manera, si así lo deseamos. 🤖🚀

¡Hasta la próxima!