

# Práctica librerías gráficas de JavaScript

## (D3 JS y Google Chart)

1. Introducción a las librerías gráficas de JavaScript
2. Realización de un ejemplo práctico
3. Conclusión y opinión personal

- 1) Vamos a trabajar con las librerías gráficas de JavaScript, concretamente con D3 y Google Chart, estas librerías nos permiten transmitir datos a los usuarios de nuestra web de forma clara, concisa y atractiva.

Los gráficos son una gran herramienta para la visualización de datos ya que son fáciles de leer y de interpretar, mientras que para los desarrolladores son sencillos de implementar.

Además tienen un techo de personalización muy alto porque podemos añadir animaciones e interactividad en nuestras gráficas, de esta forma los usuarios no solo asimilan y aprenden los datos que les mostramos, sino que también pueden jugar con ellas y disfrutarlas visualmente.

La librería D3 es una opción muy potente para realizar todo tipo de gráficos en JavaScript, combina el elemento canvas con el de SVG y la manipulación del DOM, de esta forma se puede crear una alta gama de diagramas. Tiene un abanico muy grande de posibilidades dadas las herramientas con las que trabaja por lo que quizá sea un poco compleja y requiera un estudio extenso para utilizarla en su pleno potencial.

Por otro lado tenemos Google Chart, con la que creamos gráficos interactivos y herramientas para datos utilizando la API de Google Visualization de Google Chart. La manipulación de esta API es bastante sencilla y ofrece una personalización intuitiva de las opciones de tu gráfico y como se renderiza.

- 2) Para la realización de un ejemplo práctico crearemos un archivo sencillo de html para ambos casos, en el deberemos implementar una etiqueta 'script' para importar la librería a nuestro proyecto.

Una vez tenemos nuestro html creamos un 'div' que nos permita englobar nuestro gráfico, dándole un 'id' para enlazarlo desde nuestro código JavaScript. Desde este último en un archivo index.js que enlazaremos a nuestro index.html, trabajaremos con unos datos de pega como son las edades de un muestreo de personas inventado.

En cada uno de los ejemplos trabajaremos con los métodos de la librería en cuestión, en el caso de D3 trabajaremos creando, definiendo y enlazando un elemento SVG donde aparecerá nuestro gráfico, en este caso, un histograma sencillo. Una vez hagamos el proceso de definir las características de nuestro SVG con los datos a trabajar, las barras y los ejes del histograma, procederemos a dibujarlo en nuestra página con el método `svg.node()`.

En el ejemplo práctico de Google Chart dibujaremos un histograma también con los mismos datos, para ello definiremos una constante objeto que definirá nuestro histograma y tendrá una serie de propiedades para personalizarlo, como el conjunto de datos a trabajar, el id del div donde mostrarlo y otros aspectos visuales como las dimensiones, el color, etc. Con esto definiremos una constante `init` que actuará de función callback llamada al cargarse la página, aquí es donde utilizando la API de Google Visualization determinaremos qué tiene que mostrar y cómo.

Por último introducimos las líneas que nos permiten acceder a la librería para cargarla y dibujar nuestro gráfico llamando a nuestra función `init`.

3) En conclusión ambas librerías son bastante sencillas de usar para ejemplos sencillos, Google Chart siendo la más sencilla de las dos ya que D3 trabaja con SVG y es subjetivamente interesante aprender a manipular esa herramienta, pero Google Chart resulta más fácil de leer como primer contacto, además de ser más sencillo implementar animación e interactividad al no trabajar puramente con el DOM como D3. Para un uso sencillo y no demasiado en profundidad de las librerías de JavaScript para visualizar datos me parece que Google Chart es una mejor opción.