



UNIVERSITAT DE
BARCELONA

Facultat d'Informació
i Mitjans Audiovisuals

Introducción a Highcharts JS

Rubén Alcaraz Martínez

2026



Créditos

© Rubén Alcaraz Martínez, Grupo de innovación docente Adaptabit,
<https://www.ub.edu/adaptabit/>



Este recurso de aprendizaje se encuentra bajo una licencia Creative Commons Reconocimiento – No comercial – Compartir Igual 4.0

Highchart es una biblioteca de JavaScript gratuita para la creación y el diseño, pero se requiere una licencia de Highcharts para su publicación y uso en entornos comerciales.

Highcharts y el resto de los nombres de productos, marcas comerciales y logotipos mencionados son propiedad de Highsoft o de sus respectivos titulares. Su uso en este material tiene únicamente fines educativos y no implica ningún tipo de afiliación, patrocinio o respaldo por parte de los titulares de los derechos.

Febrero de 2026

Sumario

Créditos	1
Sumario	2
Glosario	3
Introducción	4
Instalación	5
Inicializar un gráfico	5
Accesibilidad	15
El módulo <i>accessibility</i>	15
El módulo <i>exporting</i>	18
El módulo <i>export-data</i>	19
El módulo <i>pattern-fill</i>	22
El módulo <i>sonification</i>	24
Código fuente	26
Referencias	27

Glosario

- **Ayudas técnicas:** herramientas, dispositivos o software que permiten a las personas con discapacidad percibir, comprender, navegar e interactuar con contenidos y servicios digitales en igualdad de condiciones. Por ejemplo, lectores o magnificadores de pantalla.
- **Biblioteca de software:** conjunto de ficheros, funciones y estilos listos para reutilizar para añadir funcionalidades a un sitio web o aplicación sin tener que programarlas desde cero.
- **Biblioteca de JavaScript:** tipo concreto de biblioteca de software escrita en JavaScript, pensada para usarse en páginas web. Permite añadir comportamientos interactivos (gráficos, formularios dinámicos, validaciones, etc.) usando pocas líneas de código. Highcharts es un ejemplo.
- **Canvas:** elemento de HTML que permite dibujar gráficos directamente en píxeles mediante JavaScript.
- **JSON:** formato de texto estructurado para representar datos (nombres y valores), fácil de leer por personas y por programas. Se usa habitualmente para pasar datos a bibliotecas de JavaScript, como los valores que se muestran en un gráfico.
- **SVG:** formato gráfico basado en vectores, no en píxeles, que se dibuja mediante código. Permite que los gráficos escalables, nítidos en cualquier tamaño y mucho más accesibles.
- **WebGL:** tecnología basada en JavaScript que permite usar la tarjeta gráfica (GPU) del ordenador para renderizar gráficos complejos en 2D y 3D dentro del navegador.

Introducción

Highcharts¹ es una biblioteca de JavaScript basada en los estándares SVG y canvas/WebGL, pensada para la creación de visualizaciones interactivas para la Web. Permite representar una gran variedad de gráficos, mapas, cuadros de mando o diagramas Gantt, entre otros muchos, integrables directamente en páginas HTML.

Su principal objetivo es facilitar la visualización de datos sin necesidad de conocimientos avanzados de programación, ofreciendo una sintaxis relativamente sencilla y una amplia colección de ejemplos reutilizables.

Highcharts es desarrollado por la empresa Highsoft AS, con sede en Noruega. El proyecto se inició en 2009, impulsado por el desarrollador noruego Torstein Hønsi, con la intención de crear una alternativa moderna y reutilizable a los gráficos estáticos utilizados habitualmente en la mayoría de los sectores profesionales.

Desde su origen, Highcharts se ha caracterizado por contar con una documentación extensa y bien estructurada, un enfoque claro en la calidad visual y la interacción, y, de forma destacada, una apuesta temprana por la accesibilidad web.

Highcharts funciona bajo un modelo de licencia comercial, aunque permite su uso gratuito en proyectos no comerciales, contextos educativos y proyectos personales sin ánimo de lucro.

Finalmente, y a diferencia de otras bibliotecas como D3, Highcharts cuenta con una solución comercial en la nube denominada Everviz² que ofrece un editor visual a través del cual la implementación de este tipo de visualizaciones se democratiza al dejar de ser necesarios conocimientos sobre HTML, CSS o JavaScript.

Empresas de todo el mundo como Google, Netflix, SAP, o Samsung, entre otras muchas, utilizan Highcharts. Dentro del ámbito de la comunicación científica, hace algunos años se incorporaron en los productos de ScienceDirect (Gies, 2018).

En este manual introductorio se muestra el proceso de integración de la biblioteca principal de Highcharts, la generación de diversos gráficos, así como la integración de diversas características de accesibilidad.

¹ <https://www.highcharts.com/>

² <https://www.everviz.com>

Instalación

Partiendo de la base de que se desea incorporar un gráfico dentro de una página HTML, lo primero que será necesario hacer es incluir una llamada a la biblioteca de Highcharts en la cabecera del documento HTML.

```
1 <!doctype html>
2 <html lang="es">
3 <head>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1">
6   <title>Mi primer gráfico con Highcharts</title>
7   <script src="https://code.highcharts.com/highcharts.js"></script>
8 </head>
9 <body>
10
11 </body>
12 </html>
```

En la línea 7 del código anterior se incorpora la última versión disponible de la biblioteca. No obstante, si se desea garantizar la máxima compatibilidad a largo plazo, quizá sea preciso utilizar la versión específica con la que se esté trabajando en cada momento. Por ejemplo, si se desea utilizar la versión, 10.1, la llamada sería la siguiente:

<https://code.highcharts.com/12.5/highcharts.js>

Inicializar un gráfico

Para inicializar un gráfico, Highcharts necesita, en primer lugar, un elemento HTML que funcione contenedor, y lo habitual y recomendado es usar un `<div>`, ya que se trata de un elemento neutro sin ningún tipo de semántica asociada.

Dentro de ese contenedor genérico en el que no hay contenido, se cargará dinámicamente el gráfico que se generará en pasos posteriores. Mediante CSS será necesario también especificar, como mínimo, el alto (*height*) y el ancho (*width*) del elemento `<div>` o, dicho de otra manera, el tamaño del gráfico.

En el ejemplo siguiente se genera un elemento `<div>` con un ancho del 100% para que se adapte dinámicamente al ancho del contenedor en el que se encuentre el elemento `<div>` dentro del que se cargará el gráfico; la propiedad *max-width* con el valor 900px, la cual es opcional y permite que el gráfico no

crezca demasiado (no más de 900px en pantallas muy grandes; y una altura fija de 420px para simplificar y hacer predecible ese bloque en la página.³

```
1 <!doctype html>
2 <html lang="es">
3 <head>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1">
6   <title>Mi primer gráfico con Highcharts</title>
7   <script src="https://code.highcharts.com/highcharts.js"></script>
8   <style>
9     #grafico{
10      width: 100%;
11      max-width: 900px;
12      height: 420px;
13    }
14  </style>
15 </head>
16 <body>
17   <div id="grafico"></div>
18 </body>
19 </html>
```

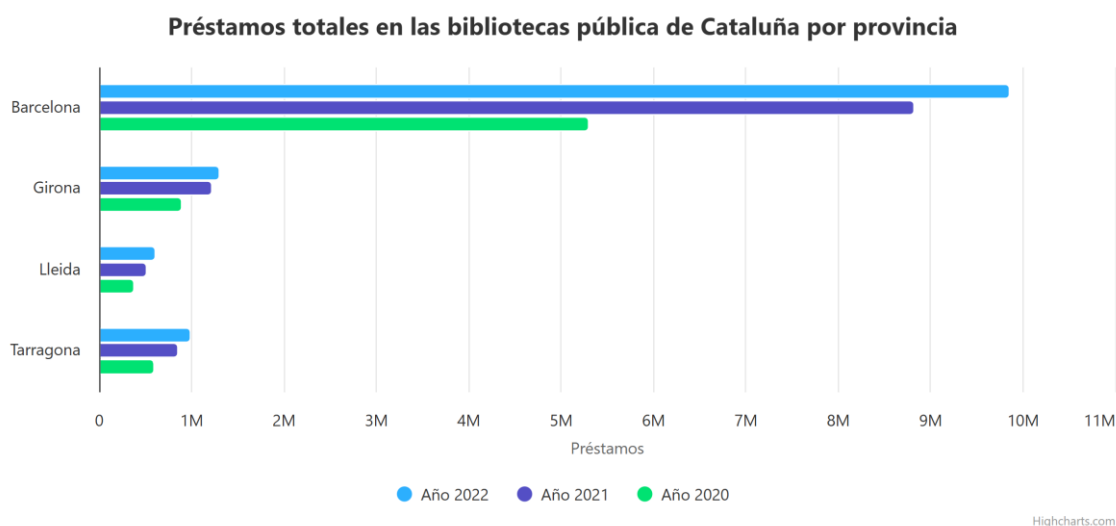
El gráfico se inicializa mediante una etiqueta `<script></script>` dentro de la que se invocará el constructor de Highcharts: `Highcharts.chart()`. Un constructor es una función especial que se utiliza para crear objetos y que funciona como una "plantilla" que permite instanciar múltiples veces un mismo objeto con las mismas propiedades y métodos, pero con distintos valores. En el contexto de uso de Highcharts, un objeto es un gráfico, un mapa o cualquier otro tipo de visualización, pero también algunos de los elementos que los conforman. Por su parte, las propiedades se definen en formato JSON y permiten almacenar los datos en bruto, así como definir la apariencia de la visualización.

³ Siempre es recomendable incluir el código CSS en un fichero independiente. No obstante, para facilitar la lectura de los ejemplos que acompañan a este manual, se incluyen directamente en la misma página HTML.

```
<!doctype html>
<html lang="es">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Mi primer gráfico con Highcharts</title>
    <script src="https://code.highcharts.com/highcharts.js"></script>
    <style>
      #grafico{
        width: 100%;
        max-width: 900px;
        height: 420px;
      }
    </style>
  </head>
  <body>
    <div id="grafico"></div>

    <script>
      Highcharts.chart('grafico', {
        chart: {
          type: 'bar'
        },
        title: {
          text: 'Préstamos totales en las bibliotecas pública de Cataluña por provincia'
        },
        xAxis: {
          categories: ['Barcelona', 'Girona', 'Lleida', 'Tarragona'],
        },
        yAxis: {
          title: {
            text: 'Préstamos'
          },
        },
        legend: {
          align: 'center',
          verticalAlign: 'bottom'
        },
        series: [{
          name: 'Año 2022',
          data: [9847040, 1301179, 599029, 984589]
        }, {
          name: 'Año 2021',
          data: [8821117, 1216586, 506732, 856157]
        }, {
          name: 'Año 2020',
          data: [5293704, 884194, 370970, 587963]
        }
      ]
    });
  </script>
</body>
</html>
```

En el código anterior, se observan diferentes objetos y algunas de sus propiedades más elementales. Por ejemplo, el objeto *chart* se define mediante la propiedad *type*, indicando que se desea crear un gráfico de barras (*type*: 'bar'). El objeto *title* se define mediante la propiedad *text*, indicando el valor del título del gráfico. Los objetos *xAxis* y *Axis* incluyen la propiedad *categories* y el objeto *title* con la propiedad *text*, en las que se almacena un vector con los valores del eje x y un título para el eje y, respectivamente. El objeto *legend* se define mediante las propiedades *align* y *verticalAlign* que permiten determinar la posición de la leyenda. Finalmente, el objeto *series* almacena un vector con grupos de propiedades *name* (nombre de la serie) y *data* (valores de la serie) para cada categoría del eje x. El resultado del gráfico se muestra a continuación.

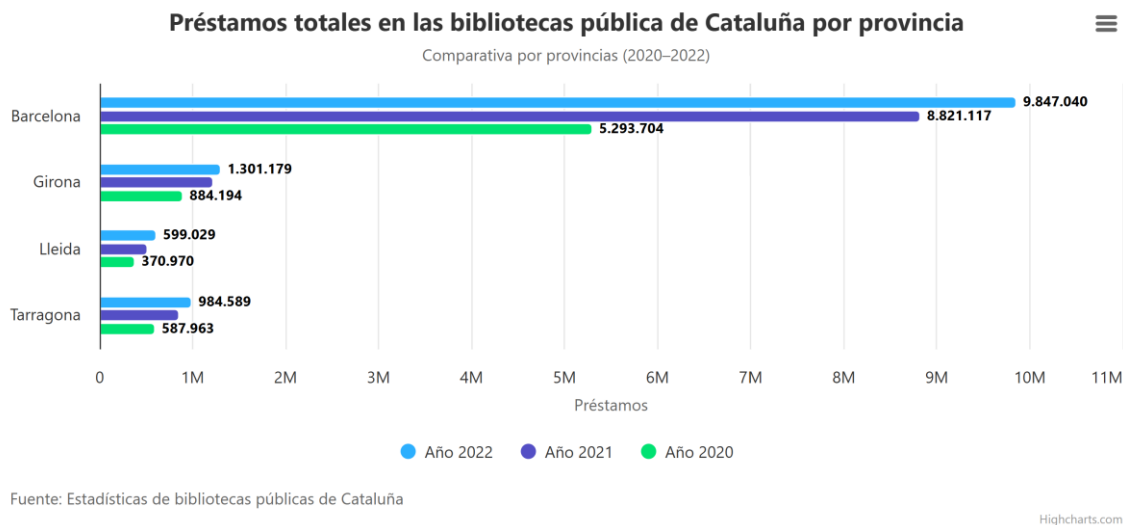


A continuación, se muestran objetos adicionales con los que es posible añadir nuevos elementos al gráfico.

- *subtitle* permite añadir un subtítulo o cualquier otra información que se considere relevante para el lector como, por ejemplo, la fuente de los datos. Para ello, se utiliza la propiedad *text*.
- *caption* para añadir un pie de figura. Como en el caso anterior, se utiliza la propiedad *text*.
- El objeto *plotOptions* permite mostrar etiquetas con los valores de cada barra o punto de forma permanente, sin necesidad de situar el cursor del ratón o el foco del teclado sobre cada elemento. Estas etiquetas se activan mediante la opción *dataLabels*.

```
Highcharts.chart('grafico', {
  chart: {
    type: 'bar'
  },
  accessibility: {
    highContrastMode: 'on'
  },
  title: {
    text: 'Préstamos totales en las bibliotecas pública de Cataluña por provincia'
  },
  subtitle: {
    text: 'Comparativa por provincias (2020-2022)'
  },
  caption: {
    text: 'Fuente: Estadísticas de bibliotecas públicas de Cataluña'
  },
  plotOptions: {
    series: {
      dataLabels: {
        enabled: true
      }
    }
  },
},
```

En la siguiente imagen se muestra el resultado de añadir un subtítulo, un pie de imagen y las etiquetas con los valores de cada barra.



Para personalizar la apariencia del gráfico es posible utilizar, entre otras opciones, las que continúan:

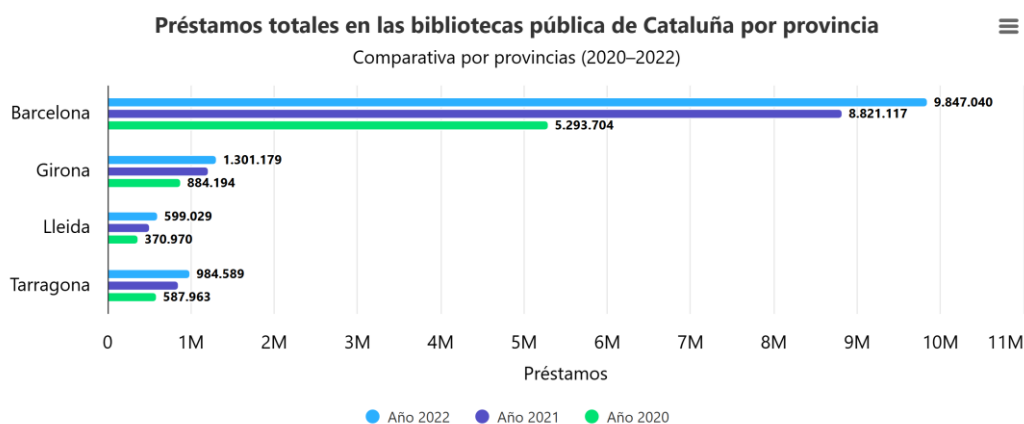
- El objeto *labels* se puede utilizar dentro de los ejes (*xAxis* e *yAxis*) para personalizar las etiquetas de sus valores. Para ello, se aplica la propiedad *style* con la que es posible insertar propiedades CSS.

```
xAxis: {
  categories: ['Barcelona', 'Girona', 'Lleida', 'Tarragona'],
  labels: {
    style: {
      fontSize: '16px',
      color: '#000000'
    }
  }
},
yAxis: {
  title: {
    text: 'Préstamos',
    style: {
      fontSize: '16px',
      color: '#000000'
    }
  },
  labels: {
    style: {
      fontSize: '16px',
      color: '#000000'
    }
  }
},
```

El objeto *style* se puede utilizar para personalizar otros elementos del gráfico. Por ejemplo, en el código siguiente, el subtítulo.

```
subtitle: {
  text: 'Comparativa por provincias (2020-2022)',
  style: {
    fontSize: '16px',
    color: '#000000'
  }
},
```

A continuación, se muestra la personalización aplicada hasta el momento.



Fuente: Estadísticas de bibliotecas públicas de Cataluña

Highcharts.com

La propiedad `gridLineColor` del objeto `yAxis` permite modificar el color gris sutil por uno con más contraste si la cuadrícula se considera importante para la lectura del gráfico. Un mayor contraste en las líneas de la cuadrícula puede mejorar la lectura del gráfico, pero debe usarse con moderación para no sobrecargar visualmente la visualización.

```
gridLineColor: '#666666',
title: {
  text: 'Préstamos',
  style: {
    fontSize: '16px',
    color: '#000000'
  }
},
```

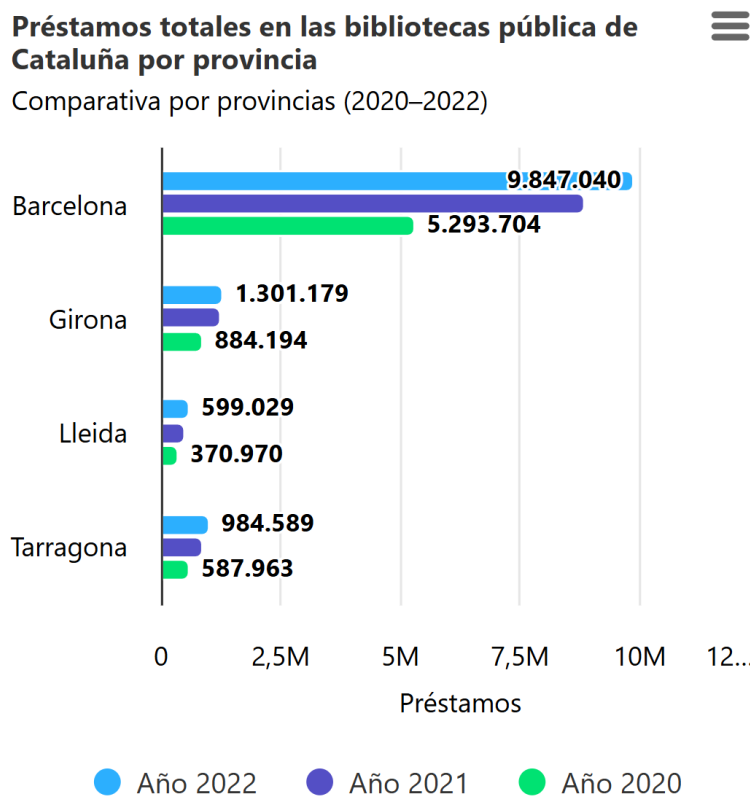
El objeto `responsive` permite definir reglas similares a las *media queries* de CSS para que el gráfico se adapte automáticamente a distintos tamaños de pantalla. Estas reglas se aplican a partir de puntos de ruptura (*breakpoints*), que indican en qué condiciones debe modificarse la configuración del gráfico.

```
responsive: {
  rules: [{
    condition: {
      maxWidth: 600
    },
    chartOptions: {
      xAxis: {
        labels: {
          style: {
            fontSize: '12px'
          }
        }
      },
      yAxis: {
        title: {
          text: 'Préstamos',
          style: {
            fontSize: '12px'
          }
        },
        labels: {
          style: {
            fontSize: '12px'
          }
        }
      },
      subtitle: {
        style: {
          fontSize: '12px'
        }
      }
    }
  ]
}
```

En el ejemplo anterior, el bloque *responsive* define una regla responsiva para que el gráfico se adapte automáticamente cuando el ancho del contenedor del gráfico es de 600 px o menos. Cuando se cumple esta condición, se activa el sistema de reglas responsivas de Highcharts.

chartOptions indica qué cambios se aplican cuando se cumple la condición. Es decir: "si el contenedor mide ≤ 600 , entonces cambia estas opciones".

Concretamente, se reduce el tamaño de las etiquetas del eje X (nombres de provincias) y del eje Y ("Préstamos") a 12px. Reduce también el tamaño de las etiquetas numéricas del eje Y a 12px. Y, finalmente, hace lo mismo con el tamaño del subtítulo.



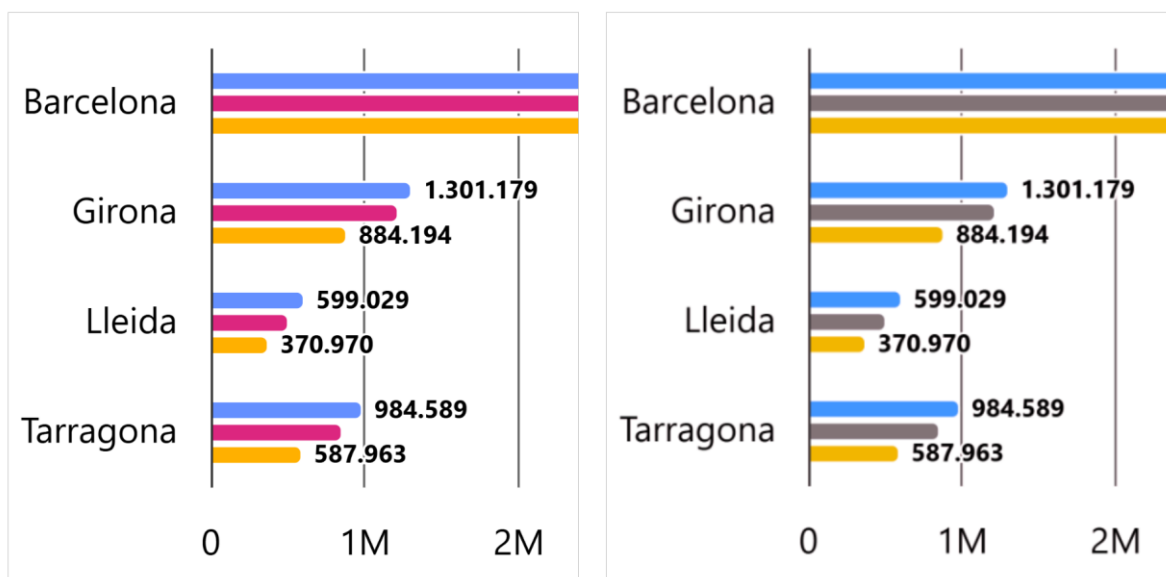
Fuente: Estadísticas de bibliotecas públicas de Cataluña

Highcharts.com

Para modificar el color de cada barra es posible utilizar la propiedad *color* tal y como se muestra a continuación. Es importante seleccionar paletas de colores accesibles que permitan a todo el mundo diferenciar de forma eficiente cada color.

```
series: [{
  name: 'Año 2022',
  data: [9847040, 1301179, 599029, 984589],
  color: '#648FFF'
}, {
  name: 'Año 2021',
  data: [8821117, 1216586, 506732, 856157],
  color: '#DC267F'
}, {
  name: 'Año 2020',
  data: [5293704, 884194, 370970, 587963],
  color: '#FFB000'
}],
```

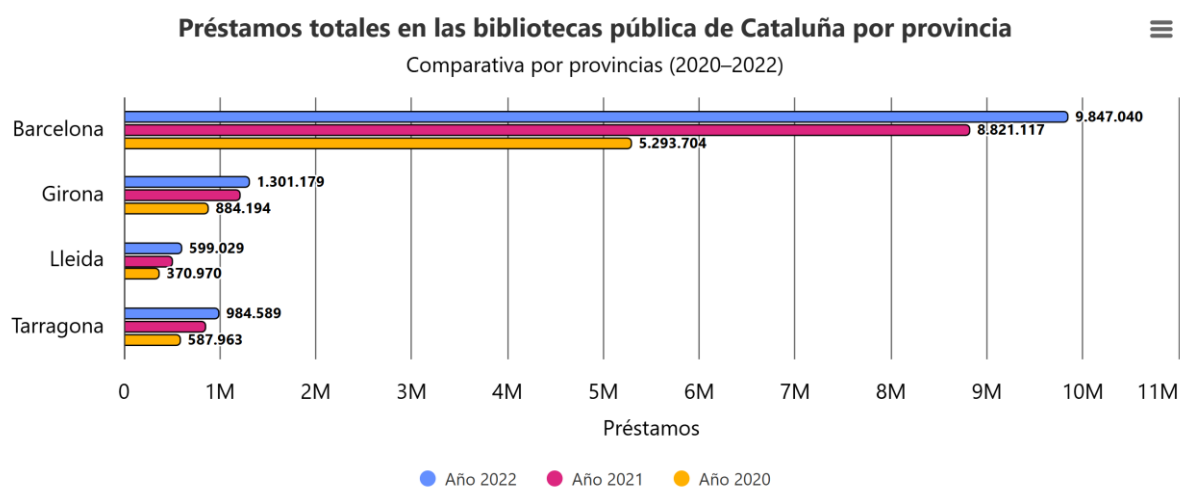
En la siguiente imagen, en la izquierda se pueden apreciar los colores seleccionados, mientras que en la derecha se ven los colores simulados tal y como los vería una persona con deuteranopia, una de las formas en las que se presenta el daltonismo.



Para mejorar el contraste entre las barras y el fondo también es posible añadir un borde sólo a su alrededor mediante las propiedades *borderColor* y *borderWidth* del objeto *plotOptions*.

```
plotOptions: {  
  series: {  
    dataLabels: {  
      enabled: true  
    },  
    borderColor: '#000000',  
    borderWidth: 1  
  }  
},
```

El resultado puede verse en la siguiente imagen:



Fuente: Estadísticas de bibliotecas públicas de Cataluña

Highcharts.com

Accesibilidad

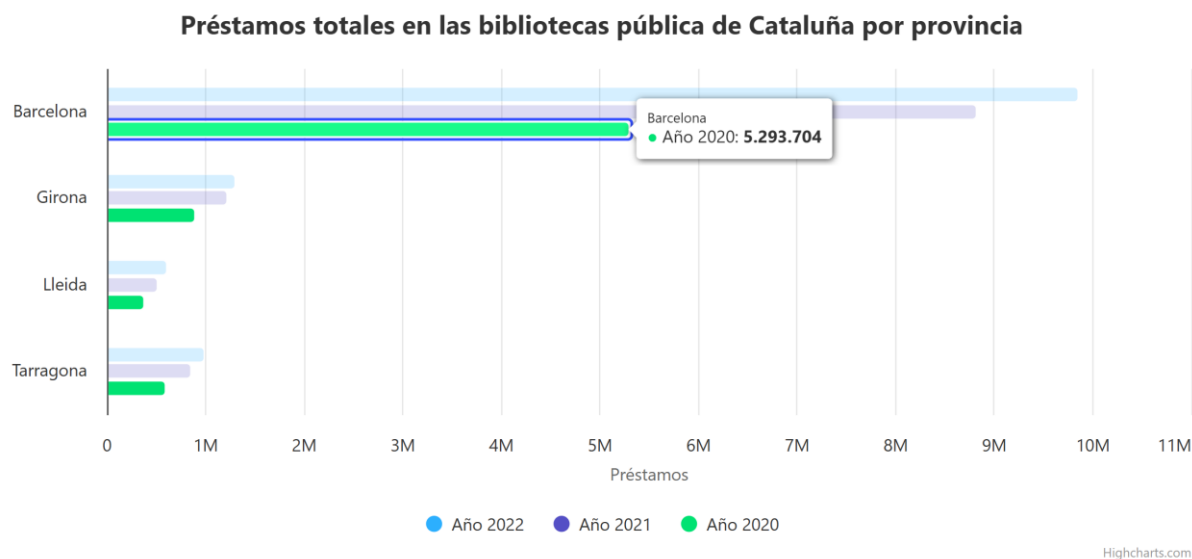
Como hemos comentado anteriormente, Highcharts incorpora múltiples características pensadas para incrementar la accesibilidad de los gráficos que se pueden generar con esta biblioteca. Algunas son nativas o, incluso, inherentes a las tecnologías que implementa. Por ejemplo, el hecho de utilizar SVG permite redimensionar el gráfico tanto como sea necesario sin perder calidad. Además, al tratarse todos los componentes del gráfico de elementos HTML, es posible personalizar hasta el último detalle de cada uno de ellos, implementar opciones de personalización o, incluso, resultar mucho más compatibles con las ayudas técnicas que permiten personalizar la apariencia de las páginas web (familia y tamaño de la fuente, color, interlineado...). A continuación, se recogen algunas de las características de accesibilidad adicionales que ofrece Highcharts.

El módulo *accessibility*

Para instalar el módulo *accessibility* basta con incorporar cargarlo justo a continuación de la biblioteca Highcharts tal y como se muestra en la siguiente imagen.

```
1 <!doctype html>
2 <html lang="es">
3 <head>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1">
6   <title>Mi primer gráfico con Highcharts</title>
7   <script src="https://code.highcharts.com/highcharts.js"></script>
8   <script src="https://code.highcharts.com/modules/accessibility.js"></script>
9 </head>
10 <body>
11   <div id="grafico">
12     <img alt="Gráfico de ejemplo" data-bbox="140 650 850 750"/>
13   </div>
14 </body>
15 </html>
```

Al añadir el módulo *accessibility* se da soporte automático a la navegación con el teclado, la cual puede personalizarse mediante el objeto *keyboardNavigation*. También se añade automáticamente un borde para ver con mayor facilidad el elemento interactivo con el foco del teclado.

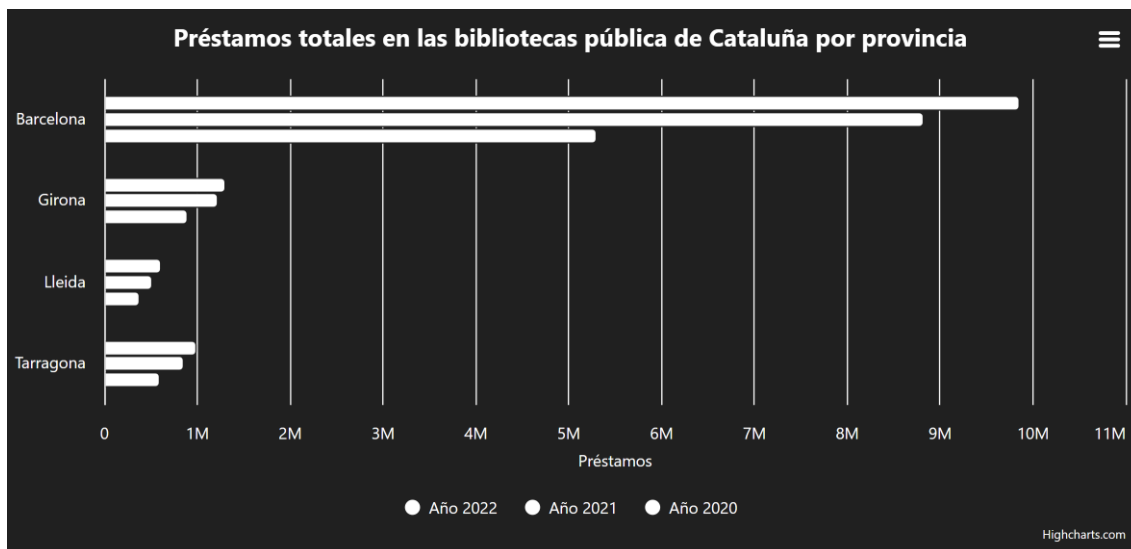


Una vez añadido el módulo podemos crear un objeto *accessibility* y personalizar múltiples propiedades. Por ejemplo, *highContrastMode* hace que se active automáticamente un estilo en alto contraste si el usuario tiene activado un tema de alto contraste en su sistema operativo.

Se puede crear el objeto *accessibility* tal y como se muestra en la siguiente figura, por ejemplo, justo después de definir el tipo de gráfico. Dentro del objeto añadimos las propiedades. En el ejemplo siguiente, aparece la propiedad *highContrastMode* con el valor *on*.

```
<script>
  Highcharts.chart('grafico', {
    chart: {
      type: 'bar'
    },
    accessibility: {
      highContrastMode: 'on'
    },
  },
```

El resultado visual en el caso de activar alguno de los temas de alto contraste de Windows se puede apreciar en la siguiente imagen.



El módulo *accessibility* también cuenta con propiedades para añadir descripciones largas al gráfico. Estas descripciones deberían ser alternativas completas que permitan comprender el gráfico y disponer de la misma información que se transmite de forma visual.

La propiedad *description* permite añadir una descripción larga que no aparecerá en la página, pero que sí estará disponible para los lectores de pantalla. A continuación, se muestra un ejemplo.

```

accessibility: {
  highContrastMode: 'on',
  description: 'Gráfico de barras horizontales muestra el número total de préstamos realizados en las bibliotecas públicas de Cataluña, desglosado por provincia y por año, para el periodo comprendido entre 2020 y 2022. En el eje vertical se representan las provincias de Barcelona, Girona, Lleida y Tarragona, mientras que el eje horizontal muestra el número de préstamos en valores absolutos. Cada provincia presenta tres barras, correspondientes a los años 2020, 2021 y 2022, lo que permite comparar tanto la evolución temporal como las diferencias territoriales. En la provincia de Barcelona se registran los valores más altos en los tres años analizados. En 2020 se contabilizan 5.293.704 préstamos, cifra que aumenta en 2021 hasta 8.821.117, y alcanza su máximo en 2022 con 9.847.040 préstamos. La provincia de Girona presenta valores considerablemente inferiores a los de Barcelona, aunque con una tendencia similar. En 2020 se registran 884.194 préstamos, que aumentan en 2021 hasta 1.216.586, y en 2022 hasta 1.301.179 préstamos. En Lleida, los valores son los más bajos del conjunto. En 2020 se contabilizan 370.970 préstamos, en 2021 506.732, y en 2022 599.029 préstamos, mostrando también un incremento progresivo a lo largo del periodo. Por su parte, la provincia de Tarragona presenta cifras intermedias entre Girona y Lleida. En 2020 se registran 587.963 préstamos, en 2021 856.157, y en 2022 984.589 préstamos. En todas las provincias se observa un incremento del número de préstamos año a año, con un descenso relativo en 2020 y una recuperación progresiva en 2021 y 2022. El gráfico permite comparar visualmente tanto la distribución territorial de los préstamos como su evolución temporal, destacando el peso predominante de la provincia de Barcelona en el conjunto del sistema bibliotecario público catalán.'
},

```

Una alternativa es utilizar la propiedad *linkedDescription*. Esta otra propiedad permite asociar el gráfico con una descripción externa escrita en HTML, que se integra de forma semántica en la experiencia de accesibilidad del gráfico. En lugar de introducir el texto directamente dentro de la configuración de Highcharts, esta opción enlaza el gráfico con un contenido ya existente en la página mediante un identificador (id), normalmente un párrafo o un bloque de texto más extenso.

Este enfoque tiene varias ventajas importantes. En primer lugar, la descripción puede escribirse con HTML completo y semántico, utilizando párrafos, listas, énfasis tipográfico u otros elementos estructurales que facilitan la comprensión del contenido, especialmente para usuarios de lectores de pantalla. Highcharts no “interpreta” ese HTML ni lo modifica, sino que simplemente establece una relación entre el gráfico y ese bloque de texto, de modo que el lector de pantalla puede acceder a él como parte de la información asociada al gráfico.

Además, al tratarse de contenido visible y externo al gráfico, la descripción enlazada es más fácilmente reutilizable y mantenible. El mismo texto puede servir como explicación general para todos los usuarios, no solo para quienes utilizan ayudas técnicas, y puede ser indexado por buscadores o referenciado desde otros contextos del documento. Esto evita duplicar información en diferentes lugares y refuerza la coherencia entre la visualización y su explicación textual.

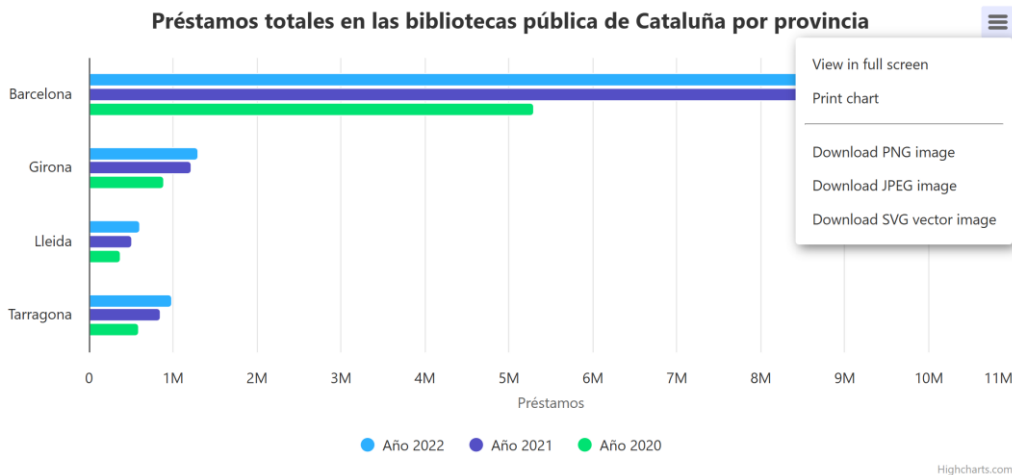
```
<body>
  <div id="grafico"></div>
  <p id="desc-grafico">Gráfico de barras horizontales muestra el número total de préstamos realizados en las bibliotecas públicas de Cataluña, desglosado por provincia y por año, para el periodo comprendido entre 2020 y 2022. En el eje vertical se representan las provincias de Barcelona, Girona, Lleida y Tarragona, mientras que el eje horizontal muestra el número de préstamos en valores absolutos. Cada provincia presenta tres barras, correspondientes a los años 2020, 2021 y 2022, lo que permite comparar tanto la evolución temporal como las diferencias territoriales. En la provincia de Barcelona se registran los valores más altos en los tres años analizados. En 2020 se contabilizan 5.293.704 préstamos, cifra que aumenta en 2021 hasta 8.821.117, y alcanza su máximo en 2022 con 9.847.040 préstamos. La provincia de Girona presenta valores considerablemente inferiores a los de Barcelona, aunque con una tendencia similar. En 2020 se registran 884.194 préstamos, que aumentan en 2021 hasta 1.216.586, y en 2022 hasta 1.301.179 préstamos. En Lleida, los valores son los más bajos del conjunto. En 2020 se contabilizan 370.970 préstamos, en 2021 506.732, y en 2022 599.029 préstamos, mostrando también un incremento progresivo a lo largo del periodo. Por su parte, la provincia de Tarragona presenta cifras intermedias entre Girona y Lleida. En 2020 se registran 587.963 préstamos, en 2021 856.157, y en 2022 984.589 préstamos. En todas las provincias se observa un incremento del número de préstamos año a año, con un descenso relativo en 2020 y una recuperación progresiva en 2021 y 2022. El gráfico permite comparar visualmente tanto la distribución territorial de los préstamos como su evolución temporal, destacando el peso predominante de la provincia de Barcelona en el conjunto del sistema bibliotecario público catalán.
  </p>
  <script>
    Highcharts.chart('grafico', {
      chart: {
        type: 'bar'
      },
      accessibility: {
        highContrastMode: 'on',
        linkedDescription: '#desc-grafico'
      }
    });
  </script>
</body>
```

El módulo *exporting*

El módulo *exporting* añade al gráfico un menú de exportación que, por defecto, se sitúa en la esquina superior derecha. Dentro de este menú se añaden opciones para ver el gráfico en pantalla completa, imprimirlo y exportarlo en formato PNG, JPEG y SVG.

Para instalar el módulo cárgalo justo después de la biblioteca y el módulo *accessibility*.

```
1 <!doctype html>
2 <html lang="es">
3 <head>
4 <meta charset="utf-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1">
6 <title>Mi primer gráfico con Highcharts</title>
7 <script src="https://code.highcharts.com/highcharts.js"></script>
8 <script src="https://code.highcharts.com/modules/accessibility.js"></script>
9 <script src="https://code.highcharts.com/modules/exporting.js"></script>
```

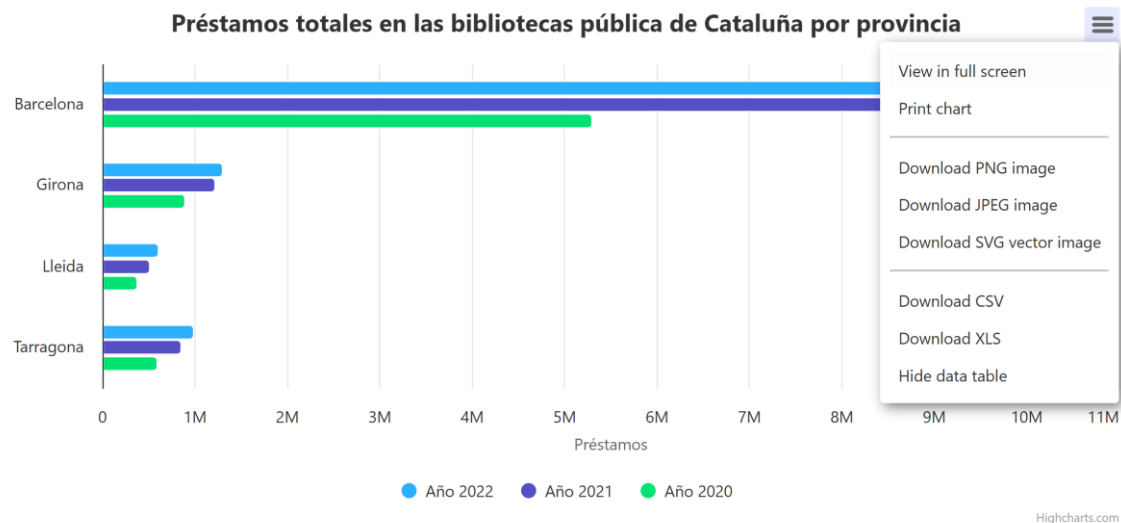


El módulo *export-data*

El módulo *export-data* añade opciones para acceder y descargar los datos del gráfico, entre las que se incluye la posibilidad de descargar los datos en bruto en formato CSV o XLS, así como una tabla que, de activarse, se genera bajo el gráfico.

Para instalar el módulo cárgalo justo después de los anteriores.

```
1 <!doctype html>
2 <html lang="es">
3 <head>
4 <meta charset="utf-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1">
6 <title>Mi primer gráfico con Highcharts</title>
7 <script src="https://code.highcharts.com/highcharts.js"></script>
8 <script src="https://code.highcharts.com/modules/accessibility.js"></script>
9 <script src="https://code.highcharts.com/modules/exporting.js"></script>
10 <script src="https://code.highcharts.com/modules/export-data.js"></script>
```



Category	Año 2022	Año 2021	Año 2020
Barcelona	9,847,040	8,821,117	5,293,704
Girona	1,301,179	1,216,586	884,194
Lleida	599,029	506,732	370,970
Tarragona	984,589	856,157	587,963

Por defecto, la tabla no incorpora estilos. Highcharts inserta la tabla dentro de un contenedor con clase `.highcharts-data-table` y un elemento `<table>`. Sabiendo esto podemos personalizar sus estilos para hacerla más legible y accesible.

```
.highcharts-data-table {
  margin-top: 12px;
  overflow-x: auto;
  -webkit-overflow-scrolling: touch;
}
.highcharts-data-table table {
  width: 100%;
  border-collapse: collapse;
  font-family: Arial, Helvetica, sans-serif;
}
.highcharts-data-table caption {
  text-align: left;
  font-weight: 600;
  padding: 8px 0;
}
```

```
.highcharts-data-table th,
.highcharts-data-table td {
  padding: 8px 10px;
  border: 1px solid #000;
  text-align: left;
  vertical-align: top;
}
.highcharts-data-table thead th {
  background: #d0d0d0;
}
.highcharts-data-table tbody tr:nth-child(even) {
  background: #fafafa;
}
.highcharts-data-table tbody tr:hover {
  background-color: #eef3ff;
}
```

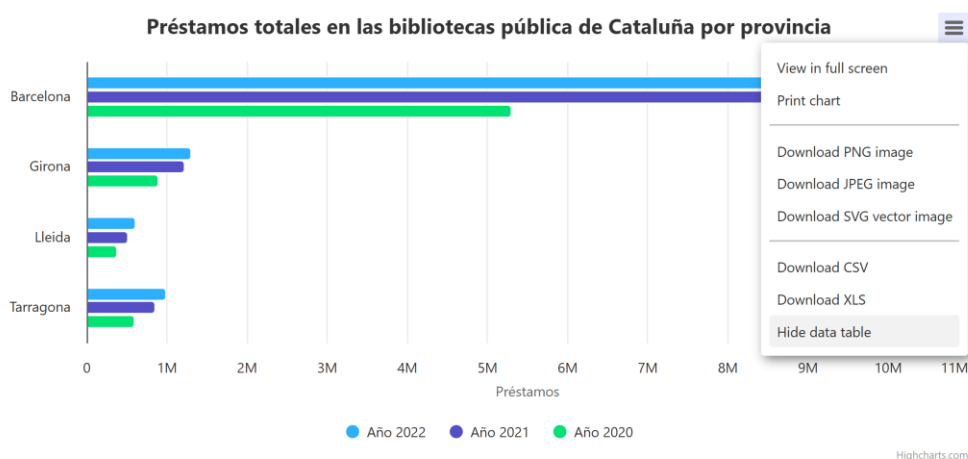
En el ejemplo anterior:

- `.highcharts-data-table` es la caja (contenedor) que Highcharts crea alrededor de la tabla.
- × Le damos un `margin-top: 12px;` para dejar un espacio de 12 píxeles entre el gráfico y la tabla, para que no queden pegados.

- × Con *overflow-x: auto*; si la tabla es más ancha que la pantalla (por ejemplo, en móvil), aparece scroll horizontal solo cuando hace falta.
 - × *-webkit-overflow-scrolling: touch*; mejora el desplazamiento horizontal en algunos dispositivos móviles como los de Apple.
- *.highcharts-data-table table* es la tabla dentro del contenedor anterior.
- × Con *width: 100%*; la tabla ocupa todo el ancho disponible dentro del contenedor y se adapta de forma responsiva.
 - × *border-collapse: collapse*; hace que los bordes de las celdas se unan y no se vean "dobles".
 - × *font-size: 0.95rem*; justa el tamaño del texto a un poco más pequeño que el texto normal (pero sigue siendo legible).
 - × *font-family: Arial, Helvetica, sans-serif*; define fuentes sin serifa para la tabla.
- *.highcharts-data-table caption* es el título de la tabla (por defecto, el mismo que el del gráfico).
- × *text-align: left*; lo alinea a la izquierda.
 - × *font-weight: 600*; lo pone en negrita moderada
 - × *padding: 8px 0*; añade espacio vertical arriba y abajo (8px) para que respire. El 0 indica que no añade espacio a izquierda/derecha.
- *.highcharts-data-table th* y *.highcharts-data-table td* son los estilos para las celdas de encabezado y las celdas de valores.
- × *padding: 8px 10px*; añade espacio dentro de cada celda: 8px arriba/abajo y 10px izquierda/derecha.
 - × *border: 1px solid #ddd*; dibuja un borde fino de color negro (#000) alrededor de cada celda.
 - × *text-align: left*; alinea el texto a la izquierda.
 - × *vertical-align: top*; si una celda tiene varias líneas, el contenido se queda arriba, no centrado verticalmente.

- `.highcharts-data-table thead th` selecciona los `<th>` que están en la parte superior de la tabla (encabezados).
 - × `background: #f6f6f6`; les añade un fondo gris para diferenciar encabezados de los datos.
- `.highcharts-data-table tbody tr:nth-child(even)` selecciona las filas pares (2, 4, 6, ...) dentro del cuerpo de la tabla.
 - × `background: #fafafa`; les aplica un fondo casi blanco para que la tabla sea más fácil de leer (guía visual)
- `.highcharts-data-table tbody tr:hover` se activa cuando el usuario pone el ratón encima de una fila. Cambia el fondo a un azul claro para indicar la posición del ratón.

El resultado puede observar en la siguiente imagen:



Préstamos totales en las bibliotecas pública de Cataluña por provincia

Category	Año 2022	Año 2021	Año 2020
Barcelona	9,847,040	8,821,117	5,293,704
Girona	1,301,179	1,216,586	884,194
Lleida	599,029	506,732	370,970
Tarragona	984,589	856,157	587,963

El módulo *pattern-fill*

Con el módulo *pattern-fill* es posible rellenar las barras (u otros elementos) de los gráficos con patrones que faciliten la diferenciación de cada una de ellas a las personas daltónicas o para facilitar su impresión en blanco y negro. El módulo incluye varios patrones prediseñados, aunque también pueden dibujarse con SVG patrones personalizados.

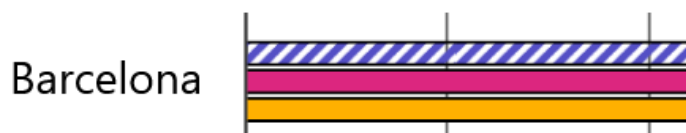
Para instalar el módulo basta cargarlo como los anteriores:

```
1 <!doctype html>
2 <html lang="es">
3 <head>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1">
6   <title>Mi primer gráfico con Highcharts</title>
7   <script src="https://code.highcharts.com/highcharts.js"></script>
8   <script src="https://code.highcharts.com/modules/accessibility.js"></script>
9   <script src="https://code.highcharts.com/modules/exporting.js"></script>
10  <script src="https://code.highcharts.com/modules/export-data.js"></script>
11  <script src="https://code.highcharts.com/modules/pattern-fill.js"></script>
```

El módulo *pattern-fills* cuenta con 10 patrones predefinidos. Para utilizar uno u otro basta con indicar su identificador (1-10) como valor de la propiedad *patternIndex* en lugar de indicar directamente el valor.

```
series: [{
  name: 'Año 2022',
  data: [9847040, 1301179, 599029, 984589],
  color: {
    patternIndex: 1
  }
}, {
  name: 'Año 2021',
  data: [8821117, 1216586, 506732, 856157],
  color: '#DC267F'
}, {
  name: 'Año 2020',
  data: [5293704, 884194, 370970, 587963],
  color: '#FFB000'
}],
```

El resultado se muestra en la siguiente imagen:



En este caso, el ejemplo que continua lo aplica sobre un gráfico de líneas.

```
series: [{
  name: 'Girona',
  data: [884194, 1216586, 1301179],
  color: '#DC267F',
  dashStyle: 'solid'
}, {
  name: 'Lleida',
  data: [370970, 506732, 599029],
  color: '#FFB000',
  dashStyle: 'dash'
}, {
  name: 'Tarragona',
  data: [587963, 856157, 984589],
  color: '#785EF0',
  dashStyle: 'dot'
}],
```

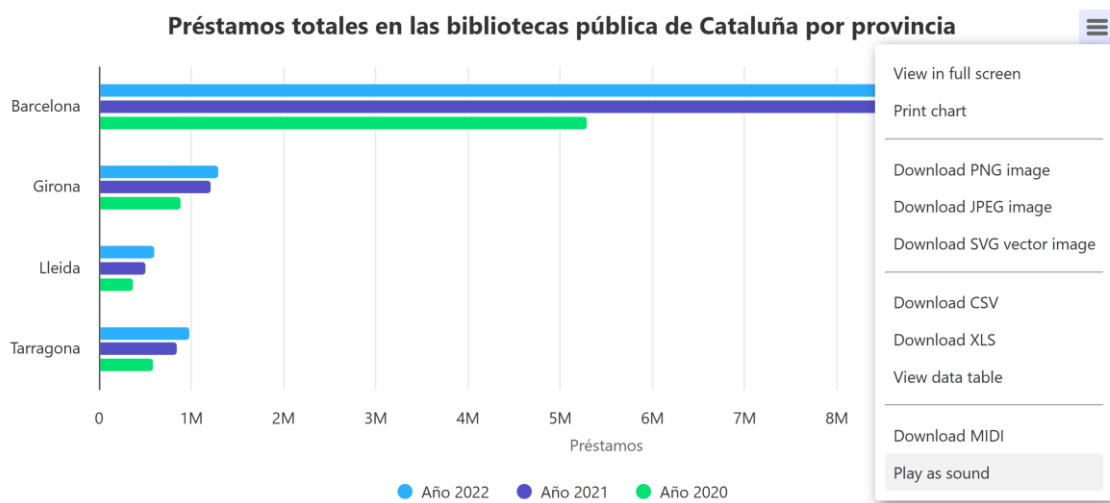
El módulo *sonification*

El módulo *sonification* permite reproducir los datos de un gráfico como sonido, facilitando el análisis y mejorando la accesibilidad, especialmente para personas con discapacidad visual. Concretamente, Highcharts añade la capacidad de reproducir los datos del gráfico como audio, asociar valores numéricos a propiedades del sonido (tono, duración, volumen) y permitir que el usuario escuche la evolución de los datos punto a punto.

Esto permite a algunas personas con discapacidad visual poder interpretar patrones y tendencias mediante el oído. Concretamente, resulta de utilidad para percibir cambios bruscos, picos o tendencias los cuales pueden observarse fácilmente como variaciones sonoras. Por parto, es de utilidad para gráficos de líneas (evolución temporal) o percibir series ordenadas (valores crecientes/decrecientes), pero tiene menos sentido en gráficos puramente categóricos sin orden lógico.

Para que el módulo funcione es imprescindible cargar previamente los módulos *accessibility* y *exporting*. Tras cargarlo, Highcharts añade automáticamente el botón *Play as sound* dentro del menú del gráfico. También se da la opción de descargar un fichero MIDI con el mismo sonido.

```
1 <!doctype html>
2 <html lang="es">
3 <head>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1">
6   <title>Mi primer gráfico con Highcharts</title>
7   <script src="https://code.highcharts.com/highcharts.js"></script>
8   <script src="https://code.highcharts.com/modules/accessibility.js"></script>
9   <script src="https://code.highcharts.com/modules/exporting.js"></script>
10  <script src="https://code.highcharts.com/modules/export-data.js"></script>
11  <script src="https://code.highcharts.com/modules/sonification.js"></script>
```





Código fuente

En los ficheros *grafico-barras.html* y *grafico-lineas.html* disponibles junto a este manual se encuentra el código fuente utilizado en los ejemplos expuestos.

Referencias

Alcaraz Martínez, Rubén (2024). Visualizaciones de datos accesibles con Highcharts JS. *Infonomy*, 2(3), 279-291.

<https://doi.org/10.3145/infonomy.24.035>

Gies, Ted (2018). The Science Direct accessibility journey: A case study.

Learned Publishing, 31(1), 69-76. <https://doi.org/10.1002/leap.1142>

Highcharts (2025). Highcharts documentation.

<https://www.highcharts.com/docs>