
IMÁGENES Y MULTIMEDIA

Una imagen se podría definir como una representación gráfica que muestra cómo es una entidad, figura o cosa. El problema de las imágenes viene, como casi siempre, cuando se deben presentar en resoluciones diferentes y hay que buscar soluciones adaptativas que no provoquen pérdida de información ni relación de aspecto.

En este capítulo vamos a ver cómo definir contenidos multimedia de forma adecuada, cómo hacer las imágenes adaptativas y receptivas y cómo hacer que cualquier contenido multimedia sea usable y accesible.

5.1 TIPOS DE IMÁGENES

Existen múltiples tipos de imágenes, pero los más extendidos son las imágenes de mapas de bits o rasterizadas y las imágenes vectoriales.

Las **imágenes rasterizadas** están formadas por píxeles y son, básicamente, la mayoría de las imágenes que se pueden encontrar en Internet. En general, ocupan más espacio en disco que las vectoriales y, dependiendo del formato de compresión que se utilice, pueden variar su tamaño, sin embargo, su gran inconveniente es que, cuando se amplían o reducen, se produce una pérdida de calidad debido, fundamentalmente a los procesos extrapolación o interpolación que intervienen en dichas ampliaciones y/o reducciones.

Mientras que la interpolación puede provocar un efecto de imagen borrosa y algo desenfocada, la extrapolación puede, y de hecho lo hace la mayoría de las veces, provocar un efecto de pixelado que, básicamente, se produce por mostrarlas a un tamaño en el que los píxeles individuales se vuelven visibles a simple vista.

Entre los diferentes formatos de compresión que existen o se aplican para las imágenes rasterizadas se encuentran el formato WEBP, PNG, JPEG o GIF y entre los diferentes recursos que se pueden utilizar para manipularlas se encuentran Adobe Photoshop y GIMP.

Las **imágenes vectoriales** no están formadas por píxeles, sino que están compuestas por unas líneas (rectas o curvas) que se generan entre dos puntos de control y que se calculan a partir de una fórmula matemática. Esta es la principal razón por la que las imágenes vectoriales suelen ocupar mucho menos espacio, en comparación con las rasterizadas, y de por qué al ampliarlas o reducirlas no se produce pérdida de calidad ni efecto de pixelado.

Entre los diferentes formatos de compresión que existen o se aplican para las imágenes vectoriales se encuentran el formato SVG, EPS o AI y, entre los diferentes recursos que se pueden utilizar para manipularlas se encuentran Adobe Illustrator y CorelDraw.

5.2 ELEMENTOS DISPONIBLES EN HTML5

HTML5 es un lenguaje versátil que permite manejar audio, video e imágenes adaptativas de múltiples formas, no obstante, los elementos más recurrentes siguen siendo AUDIO, VIDEO e IMG.

5.2.1 Elemento audio

El elemento AUDIO especifica que el contenido que se va a representar es un sonido o música. Aunque actualmente existen varios formatos de sonido, entre los que podemos encontrar el MP3, WAV o OGG, el único compatible con todos los navegadores es el MP3.

```
<audio controls>
<source src="colibri.mp3" type="audio/mpeg">

El navegador no soporta la etiqueta audio.
</audio>
```

Entre los atributos que admite en su configuración, hay que destacar la propiedad **AUTOPLAY**, para indicar si se debe poner automática en modo reproducción, **CONTROLS**, para especificar si se deben mostrar o no los controles de parar, reanudar, siguiente, etcétera, **LOOP**, para indicar si el audio se debe repetir de manera continuada cuando termine su reproducción, **MUTED**, para indicar si el volumen o salida de audio debe estar silenciada y **SRC**, que especifica la URL del archivo de audio a cargar.

5.2.2 Elementos figure y figcaption

El elemento FIGURE especifica que el contenido que se va a representar es una ilustración, diagrama, fotografía, listado de códigos o algo similar. Para describir el contenido del FIGURE puede ser descrito a través del elemento FIGCAPTION.

```
<figure>

```

```
<figcaption>Ejemplo de diagrama de Gantt</figcaption>
</figure>
```

Cabe destacar que el elemento **FIGURE** es un elemento de sección que está excluido del esquema principal del documento por considerarse que su propósito es introducir contenidos externos y que no tiene por qué estar formado por un único elemento de contenido. De hecho, es frecuente verlo para definir un conjunto de elementos multimedia que representan una única entidad que se desea, se interprete, como una única figura.

5.2.3 Elemento **img**

El elemento **IMG** especifica que el contenido que se va a representar es una imagen. Este elemento imagen no es incrustado en el documento, aunque sí que se reserva un espacio de retención para la imagen.

```

```

El elemento **IMG** no se debería utilizar si el contenido que muestra no está relacionado directamente con el contenido del documento, es decir, sólo se debe utilizar cuando su representación sea significativa para el contenido del documento.

También es importante saber que la inserción de imágenes en un documento puede afectar al rendimiento global y a la accesibilidad, por lo que se deben definir de forma precisa utilizando todos los atributos necesarios.

Entre los atributos que admite en su configuración, se deben destacar **ALT**, que especifica el texto descriptivo que se debe mostrar cuando la imagen no esté disponible, **HEIGHT** y **WIDTH**, que especifican la altura y anchura en píxeles de la imagen dentro del documento y **SRC** y que especifica la URL del archivo a cargar.

5.2.4 Elemento **picture**

El elemento **PICTURE** fue diseñado con la idea de proporcionar soporte nativo a imágenes responsive o adaptativas. En general, se utiliza de forma conjunta con el elemento **SOURCE** y **IMG** para ofrecer las diferentes alternativas de la imagen en distintos escenarios o resoluciones.

```
<picture>
  <source srcset="./img/land-desktop.png" media="(min-width: 1680px)" />
  <source srcset="./img/land-laptop.png" media="(min-width: 1366px)" />
  <source srcset="./img/land-tablet.png" media="(min-width: 640px)" />
  <source srcset="./img/land-mobile.png" media="(min-width: 360px)" />

  
</picture>
```

Cuando se definen todos los elementos, el agente de usuario seleccionará, entre todos los elementos secundarios **SOURCE**, el que mejor coincida con el escenario actual. Si no

encuentra una coincidencia que se ajuste lo suficientemente, o no soporta el elemento **PICTURE**, lo que se representará será la imagen asociada al elemento **IMG**.

5.2.5 Elemento **source**

El elemento **SOURCE** permite especificar los recursos alternativos de medios que están disponibles para ser gestionados por los elementos **AUDIO**, **PICTURE** y **VIDEO**.

Estos recursos serán seleccionados de forma automática por el agente de usuario en función del tipo de medio, códec o consulta de medios.

```
<picture>
<source srcset="/img/land-laptop.png" media="(min-width: 1366px)" />
<source srcset="/img/land-tablet.png" media="(min-width: 900px)" />
<source srcset="/img/land-mobile.png" media="(min-width: 768px)" />
</picture>
```

Entre los atributos que admite en su configuración, se deben destacar los siguientes:

5.2.5.1 **SRCSET**

Especifica una lista de imágenes, separadas por coma, a seleccionar según sea el medio, resolución, ... Cada elemento de esta lista se compone de una URL, un descriptor de ancho seguido de la letra **W** minúscula (por ejemplo, 360w o 480w) y un descriptor de densidad seguido de la letra **X** minúscula (por ejemplo, 2x).

Aunque las opciones de descriptor de ancho y descriptor de densidad son opcionales, al menos, una de ellas siempre debe estar presente. Es un elemento obligatorio cuando está definido dentro de una estructura **PICTURE**.

5.2.5.2 **MEDIA**

Especifica la consulta de medios que se debería cumplir para poder ser aplicado el recurso. Sigue las mismas normas y validaciones que las consultas de medios definidas por la regla **@MEDIA**.

5.2.5.3 **TYPE**

Especifica el tipo MIME del recurso. Todos los posibles valores que puede tomar este atributo están disponibles en la dirección <http://www.iana.org/assignments/media-types/>.

5.2.6 Elemento **video**

El elemento **VIDEO** especifica que el contenido que se va a representar es una película, cortometraje o cualquier otro contenido de video. Aunque actualmente existen varios formatos de video, entre los que podemos encontrar el MP4, AVI, WEBM o OGG, el más compatible es el formato en MP4.

```
<video controls>
<source src=" el-quinto-elemento.mp4" type="audio/mp4">

El navegador no soporta la etiqueta video.
</video>
```

Entre los atributos que admite en su configuración, hay que destacar la propiedad **AUTOPLAY**, para indicar si se debe poner automática en modo reproducción, **CONTROLS**, para especificar si se deben mostrar o no los controles de parar, reanudar, siguiente, etcétera, **LOOP**, para indicar si el audio se debe repetir de manera continuada cuando termine su reproducción, **MUTED**, para indicar si el volumen o salida de audio debe estar silenciada, **HEIGHT** y **WIDTH**, que especifican la altura y anchura en píxeles de la imagen dentro del documento y **SRC**, que especifica la URL del archivo de audio a cargar.

Si el elemento VIDEO no puede reproducir ninguno de los vídeos propuestos con sonido o el agente de usuario indica que no es posible la reproducción de sonido, el elemento TRACK podrá adquirir un papel importante. Esto es así porque, el elemento TRACK especifica una pista adicional que servirá como descripción textual para el elemento VIDEO.

5.3 PROPIEDADES DISPONIBLES EN CSS

CSS es un lenguaje que posee una gran variedad de propiedades para el manejo y manipulación de imágenes. A continuación, se muestran la mayor parte de ellas, si no todas.

5.3.1 Propiedad background-attachment

Especifica si la imagen establecida por BACKGROUND-IMAGE debe desplazarse con el resto del documento o debe quedarse fija. Entre sus posibles valores podemos encontrar **SCROLL**, que indica cómo se desplazará la imagen con el documento y es el valor por defecto, **FIXED**, que especifica que la imagen debe mantenerse fija, es decir, sin responder al desplazamiento de la página o documento y **LOCAL**, que indica que la imagen debe desplazarse con el contenido del elemento al que está asociada.

5.3.2 Propiedad background-clip

Especifica cómo debe extenderse el fondo, gradiente o imagen dentro del elemento actual.

Entre sus posibles valores podemos encontrar **BORDER-BOX**, que indica que el fondo debe extenderse incluyendo el borde del elemento y es el valor por defecto, **PADDING-BOX**, que indica que el fondo debe extenderse sin incluir el borde del elemento y **CONTENT-BOX**, que indica que el fondo debe extenderse hasta dónde empieza el espacio útil para el contenido del elemento sin incluir el margen interno.

5.3.3 Propiedad background-image

Especifica una o varias imágenes o gradientes para un elemento. Entre sus posibles valores podemos encontrar **[URL]**, que indica la dirección de la imagen que se establecerá como fondo, **NONE**, que es el valor por defecto e indica que no se aplique fondo alguno y los posibles valores de **LINEAR-GRADIENT**, **RADIAL-GRADIENT**, **REPEATING-LINEAR-GRADIENT** y **REPEATING-RADIAL-GRADIENT**, comentados anteriormente en el apartado de funciones gráficas del capítulo de Introducción al CSS.

5.3.4 Propiedad background-origin

La propiedad **BACKGROUND-ORIGIN** funciona de forma similar a la propiedad **BACKGROUND-CLIP** y especifica la posición de origen de una imagen de fondo.

Entre sus posibles valores podemos encontrar **BORDER-BOX**, que indica que la imagen empezará en la esquina superior izquierda del borde, **PADDING-BOX**, que es el valor por defecto e indica que la imagen empezará en la esquina superior izquierda del límite del margen interno y **CONTENT-BOX**, que indica que la imagen empezará en la esquina superior izquierda del límite del contenido.

5.3.5 Propiedad background-position

Especifica la posición inicial del fondo. Entre sus posibles valores podemos encontrar **[POSICIÓN]**, que indica un valor de posicionamiento. Puede ser una combinación de palabras clave como son **LEFT TOP**, **LEFT CENTER**, **LEFT BOTTOM**, **CENTER TOP**, **CENTER CENTER**, **CENTER BOTTOM**, **RIGHT TOP**, **RIGHT CENTER**, **RIGHT BOTTOM**, o un valor establecido en una de las medidas permitidas de CSS.

A continuación, se muestra una ilustración con cada uno de los significados:

LEFT TOP	CENTER TOP	RIGHT TOP
LEFT CENTER	CENTER CENTER	RIGHT CENTER
LEFT BOTTOM	CENTER BOTTOM	RIGHT BOTTOM

5.3.6 Propiedad background-repeat

Especifica si la imagen establecida como fondo debe repetirse y cómo debe hacerlo. Entre sus posibles valores podemos encontrar **REPEAT**, que es el valor por defecto e indica que la imagen debe repetirse tanto horizontal, como verticalmente, **REPEAT-X**, que indica que la imagen debe repetirse sólo horizontalmente, **REPEAT-Y**, que indica que la imagen debe repetirse sólo verticalmente, **NO-REPEAT**, que indica que la imagen NO debe repetirse, **SPACE**, que indica que la imagen debe repetirse tanto como sea posible, siempre y cuando, no se deforme ni se corte y **ROUND**, que indica que la imagen debe repetirse para llenar el espacio del elemento, aunque eso implique que sea deformada.

5.3.7 Propiedad background-size

Especifica el tamaño del fondo.

Entre sus posibles valores podemos encontrar **AUTO**, que es el valor por defecto e indica que el tamaño de la imagen debe ser igual al tamaño original, **COVER**, que indica que el tamaño de la imagen debe ajustarse para cubrir todo el contenedor o elemento, aunque eso implique que la imagen se corte por los extremos, **CONTAIN**, que indica que el tamaño de la imagen debe ajustarse para asegurarse de que sea totalmente visible. Cuando este valor se utiliza, lo normal es que se generen espacios en blanco en alguno de los extremos del elemento y **[VALOR]**, que indica un valor establecido en una de las medidas permitidas de CSS.

5.3.8 Propiedad object-position

Especifica dónde se debe colocar el elemento con respecto a su elemento padre o contenedor. Entre sus posibles valores podemos encontrar:

- **FILL**: indica que el elemento será ajustado al tamaño del contenedor, aunque este deba ser deformado, si así se requiere. Es el valor por defecto
- **CONTAIN**: indica que el elemento será ajustado con respecto al tamaño del contenedor, pero guardando la relación de aspecto para que entre todo su contenido en el espacio disponible.
- **COVER**: indica que el elemento será ajustado con respecto al tamaño del contenedor, pero guardando la relación de aspecto para llenar el espacio disponible. Este valor puede hacer que se corte información por los extremos.
- **NONE**: indica que el elemento NO será ajustado ni deformado.
- **[POSICIÓN]**: indica un valor de posicionamiento. Puede ser una combinación de palabras clave como son LEFT TOP, LEFT CENTER, LEFT BOTTOM, CENTER TOP, CENTER CENTER, CENTER BOTTOM, RIGHT TOP, RIGHT CENTER, RIGHT BOTTOM, o un valor establecido en una de las medidas permitidas de CSS.

NOTA

Esta propiedad sólo es aplicable a los elementos IMG y VIDEO.

5.4 IMÁGENES RECEPTIVAS Y ADAPTATIVAS

Cada vez más, accedemos a los contenidos web desde muy diferentes dispositivos con distintos tamaños y resoluciones. Esto provoca que los diseñadores y desarrolladores

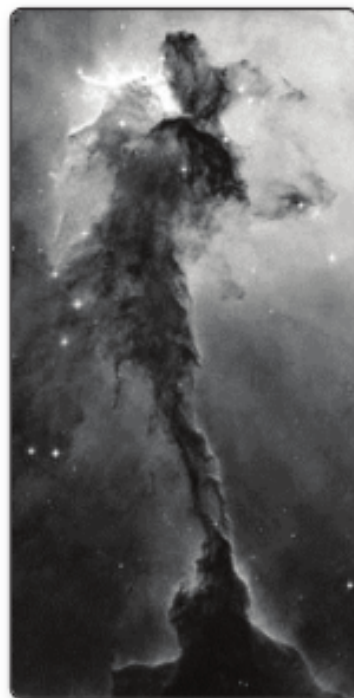
tengan que ingeniárselas para mostrar los contenidos de forma que no pierdan información, calidad o relación de aspecto.

Hasta no hace tanto, era habitual ver las imágenes deformadas o con espacios en “blanco” alrededor, lo que provocaba sensación de mala calidad, mal gusto o una imagen corporativa descuidada. Pero entonces, apareció el concepto de diseño receptivo o adaptativo y, con él, varias técnicas de adaptación de contenidos que trataban de conseguir que las imágenes se viesen de forma adecuada.

Aunque, a primera vista, no es la misma la información dependiendo del dispositivo en el que se muestra la imagen, toda la información está disponible. Esto es posible gracias a la implementación de funcionalidades adicionales que permiten, entre otras cosas, agrandar o empequeñecer la imagen o captar cualquier punto de esta a través de un desplazamiento.

Dicho esto, las imágenes receptivas o adaptativas pueden conseguirse, fundamentalmente, a través de varios métodos o técnicas, que suelen implementarse de forma combinada.

5.4.1 Adaptación de imágenes mediante propiedades CSS



Si lo que se desea es hacer que un elemento IMG se muestre de forma adaptable, la manera más sencilla de conseguir esto es utilizar las propiedades WIDTH, HEIGHT y OBJECT-FIT de forma combinada.

En general, podríamos decir que todas las imágenes deben estar establecidas al cien por cien del ancho del contenedor y con asignación de altura automática. Esto provocará que las imágenes se muestren correctamente, independientemente de su relación de

aspecto y tamaño. Sin embargo, si el contenedor no se corresponde, en proporción, con el tamaño de la imagen, lo más probable es que la imagen se corte o se salga fuera de los límites marcados por el contenedor.

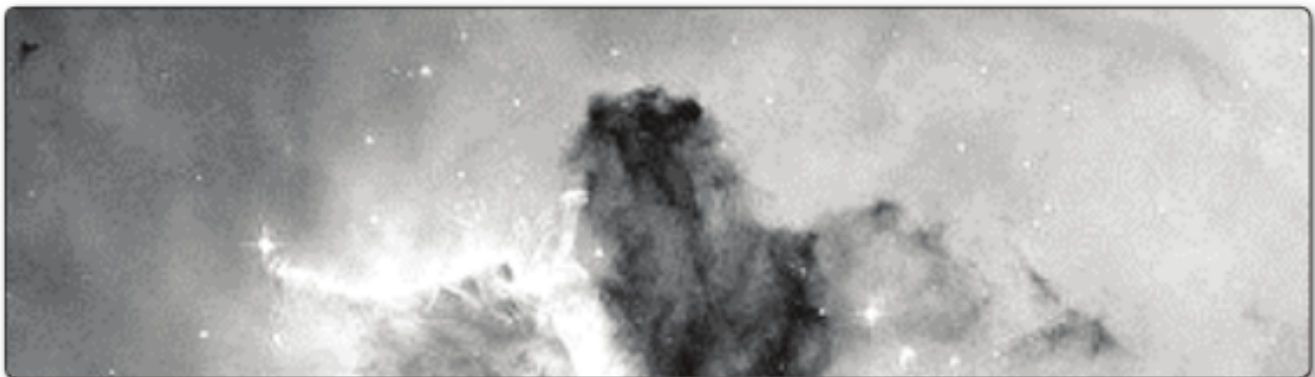
Supongamos una situación en la que se desea presentar la imagen de la derecha en un contenedor de 1920x250 píxeles únicamente utilizando un elemento IMG y CSS. Una posibilidad podría ser escribir este código:

```
<style>
div { display: block; height: 250px; position: relative; width: 100%; }
img { height: auto; object-fit: none; width: 100%; }
</style>

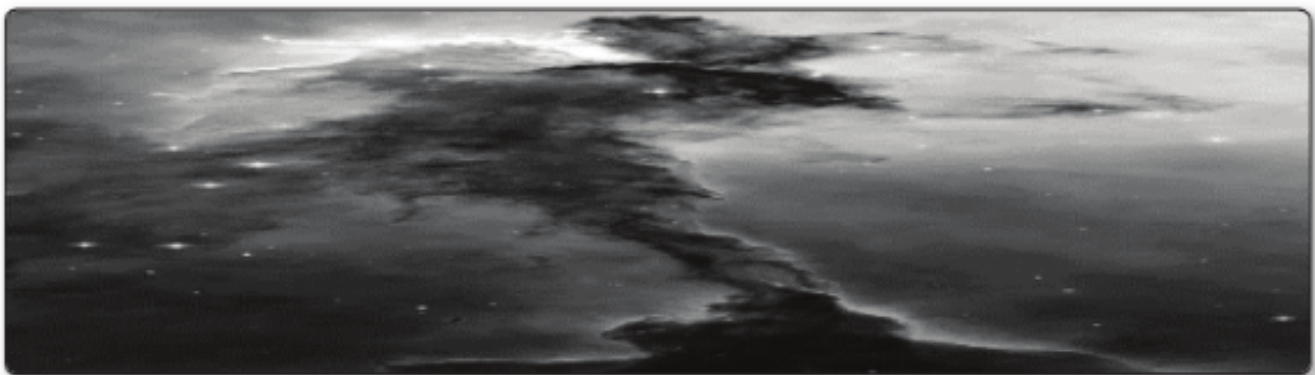
<div>

</div>
```

El resultado de este código sería algo como:



Si ahora, en el elemento IMG, en vez de establecer la propiedad HEIGHT a AUTO, la establecemos al 100%, el resultado sería:

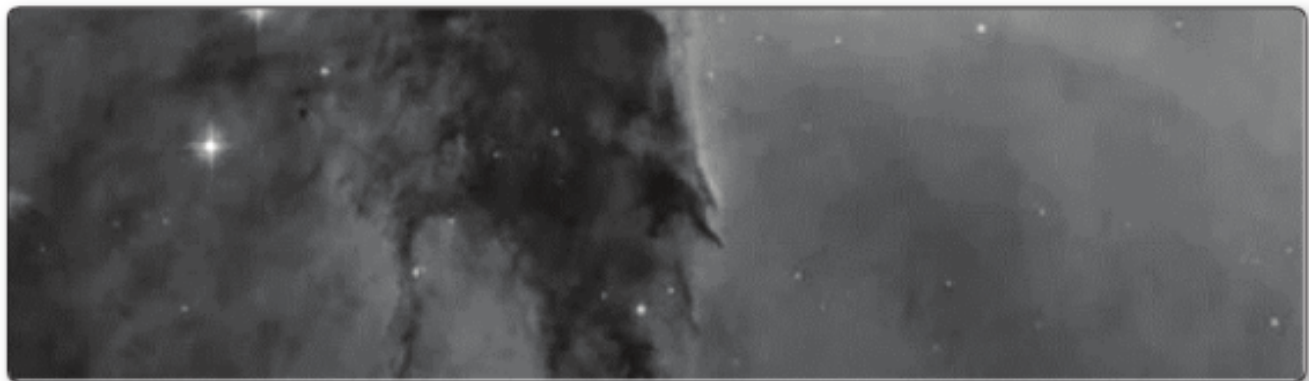


Como se puede observar, la imagen se ha deformado para ajustarse a las proporciones del contenedor donde se encuentra incrustado. Sin embargo, si ahora, en el elemento IMG, establecemos la propiedad OBJECT-FIT a CONTAIN, el resultado será:



Si con la propiedad `OBJECT-FIT` establecida a `CONTAIN`, aumentamos el alto del contenedor, la imagen se irá agrandando a la nueva altura. No obstante, si la altura fuese mayor que la requerida por la imagen, lo que se producirá es un espacio en blanco entre la imagen y el siguiente elemento.

Ahora, si en el elemento `IMG`, establecemos la propiedad `OBJECT-FIT` a `COVER`, el resultado será muy diferente:



Si observamos la imagen anterior con detenimiento, lo que veremos es que la imagen se ha ajustado al 100% del ancho disponible y se ha centrado con respecto al alto del contenedor mostrándonos sólo 250 píxeles desde el punto central de la imagen.

Al final, todo esto se podría reducir en que los contenedores deben estar ajustados, en proporción, a las imágenes que se van a mostrar y bajo la orientación del dispositivo donde se vayan a reproducir.

5.4.2 Adaptación de imágenes mediante consultas de medios

Si lo que se desea es hacer que se cargue una u otra imagen a través de CSS y en función de la resolución, la manera más sencilla de conseguir esto es utilizar las variaciones de la propiedad `BACKGROUND`, en combinación con la regla `@MEDIA`.

Supongamos una situación en la que tenemos cuatro versiones de una misma imagen y, lo que se desea hacer es presentar la imagen como fondo de un elemento DIV, pero con la condición de que se cargue una u otra versión en función de la resolución.

Una posibilidad podría ser el siguiente código:

```
<style>
.banner {
background-image: url(imagen-640x360.jpg);
border-bottom: 1px solid rgba(0,0,0,0.1);
color: #fff;
display: block;
height: 100vh;
position: relative;
text-align: center;
width: 100%;
}

@media (min-width: 800px) {
.banner { background-image: url(imagen-1280x720.jpg); }
}

@media (min-width: 1400px) {
.banner { background-image: url(imagen-1920x1080.jpg); }
}

@media (min-width: 2000px) {
.banner { background-image: url(imagen-2560x1440.jpg); }
}
</style>

<div class="banner"><div>
```

Basándonos en disciplina de Mobile First, lo que se conseguirá con esta solución es definir la imagen de menor resolución que se desea cargar y, según se vaya detectando que el dispositivo admite una mayor resolución, se irá sobrescribiendo la propiedad BACKGROUND-IMAGE para seleccionar la imagen que más se ajusta al escenario actual.

5.4.3 Selección de imágenes mediante IMG y SRCSET

El elemento IMG, como hemos visto anteriormente, dispone de un atributo denominado SRCSET que permite seleccionar el origen a presentar, mediante una sintaxis similar a las consultas de medios.

Supongamos, como antes, una imagen de la que disponíamos hasta cuatro versiones. Lo primero que nos podría venir a la cabeza es definir un elemento IMG que fuese actualizando el atributo SRC a través de un lenguaje como JavaScript. Como solución podría estar bien, no obstante, definir un elemento IMG con varios atributos SRCSET es una solución mucho más simple y de menor coste en rendimiento.


```

```

En el código anterior seleccionamos diferentes imágenes en función del ancho disponible, sin embargo, si, por la razón que fuese, necesitásemos también mostrar la imagen en función de su densidad, lo único que haría falta es añadir el indicador de densidad seguido de la letra X.

```

```

Pero, ¿qué pasa si lo que se desea es también manipular el tamaño de la imagen? Pues para ello, disponemos del atributo SIZES.

```

```

Al insertar en un documento HTML esta declaración, lo que se está indicando al sistema es que se dispone de una imagen con tres posibles tamaños, los cuales, serán seleccionados si se cumplen las condiciones marcadas por SRCSET. En otras palabras, si el tamaño de la ventana gráfica (el VIEWPORT) tiene, cómo máximo el tamaño establecido por el parámetro W, su imagen se seleccionará.

Otra cosa es el parámetro SIZES, el cual, indica exactamente qué tamaño se desea que tenga la imagen en relación con el tamaño de la ventana gráfica expuesta en la condición de medios. En nuestro ejemplo, la expresión entre paréntesis es la condición de medios y, el valor a su derecha, el tamaño que se desea que tenga la imagen en esas circunstancias. Por ejemplo, si el tamaño de la ventana es menor de 800 píxeles, la imagen deberá tener un ancho de 100% del tamaño de la ventana.

En lo referente al atributo SRC, debe establecerse siempre porque, aunque las imágenes se seleccionen por SRCSET, puede que el agente de usuario no soporte esta característica y, si no está definido, no se mostrará imagen alguna.

i NOTA

El atributo SIZES, no funcionará si el elemento IMG tiene asignada una anchura o altura, independientemente de si se establece por estilos en línea o por reglas CSS. Es decir, si se desea utilizar esta técnica, las propiedades HEIGHT y WIDTH del elemento IMG deben estar establecidas a INITIAL o INHERIT.

5.4.4 Selección de imágenes mediante PICTURE y SOURCE

Como comentamos anteriormente, el elemento PICTURE fue diseñado con la idea de proporcionar soporte nativo a imágenes responsive o adaptativas. Para conseguir este objetivo, lo habitual es recurrir al elemento SOURCE, el cual permite las mismas capacidades que el atributo SRCSET del elemento IMG.

Supongamos, como antes, una imagen de la que disponíamos hasta cuatro versiones. Para conseguir nuestro objetivo mediante el uso de PICTURE y SOURCE, lo que podría hacer es algo como lo siguiente:

```
<picture>
<source srcset="imagen-2560x1440.jpg" media="(min-width: 2000px" />
<source srcset="imagen-1920x1080.jpg" media="(min-width: 1360px" />
<source srcset="imagen-1280x720.jpg" media="(min-width: 768px" />


</picture>
```

Al insertar en un documento HTML esta declaración, lo que se está indicando al sistema es que se dispone de una imagen con cuatro posibles tamaños, los cuales, serán seleccionados si se cumplen las condiciones marcadas por el atributo MEDIA. En otras palabras, si el tamaño de la ventana gráfica (el VIEWPORT) tiene, como máximo el tamaño establecido por el atributo MEDIA, la imagen definida en el atributo SRCSET se seleccionará.

Es importante destacar que, las declaraciones del elemento SOURCE deben estar en un orden determinado en función de su condición de medios. Si la condición de medios, el atributo MEDIA, está aplicando a través de MIN-WIDTH, como es este último ejemplo, deben declararse de mayor a menor resolución. Pero, si la condición de medios que se está aplicando va definida a través de MAX-WIDTH, deben declararse de menor a mayor.

Para poder comparar y verlo mejor, vamos a ver cómo sería la declaración del elemento PICTURE anterior, pero utilizando la condición de MAX-WIDTH, en vez de MIN-WIDTH.

```
<picture>
<source srcset="imagen-640x360.jpg" media="(max-width: 768px" />
<source srcset="imagen-1280x720.jpg" media="(max-width: 1360px" />
<source srcset="imagen-1920x1080.jpg" media="(max-width: 2000px" />


</picture>
```

Aunque, el elemento SOURCE admite el atributo SIZES, éste, no se aplicará porque el elemento PICTURE lo ignora cuando SOURCE es un descendiente directo. Sin embargo, si, por la razón que fuese, necesitásemos también mostrar la imagen en función de su densidad, lo único que haría falta es añadir el indicador de densidad seguido de la letra X.

```

```

En lo referente al atributo SRC, debe establecerse siempre porque, aunque las imágenes se seleccionen por SRCSET, puede que el agente de usuario no soporte esta característica y, si no está definido, no se mostrará ninguna imagen.

5.5 VIDEOS RECEPTIVOS

Aunque es posible hacer una manipulación de igual forma que el elemento PICTURE, es decir, estableciendo varios elementos SOURCE con sus respectivas resoluciones dentro de una estructura VIDEO, lo habitual es que sólo se establezcan unos pocos estilos y sea el usuario el que escoja la resolución del video que desea ver.

De hecho, muchos sistemas multimedia suelen precargar el vídeo de menor tamaño y añadir un desplegable con las diferentes resoluciones disponibles. Con esta posibilidad, el usuario puede escoger la resolución que más se ajusta a sus deseos y se cubren algo más los requerimientos de usabilidad y accesibilidad web.

Dicho esto, una posible forma de hacer que este tipo de recursos se vuelvan adaptables a cualquier dispositivo es hacer lo siguiente:

```
video {
width: 100%;
display: block;
object-fit: cover;
height: 100vh;
}
```

```
<video src="/4703.mp4"
type="video/mp4"
loop=""
muted=""
controls=""
autoplay="">
```

Tu navegador no admite el elemento `<video>`. Puedes descargar el video desde la URL <https://www.pexels.com/video/beautiful-timelapse-of-the-night-sky-with-reflections-in-a-lake-857251/>


```
</video>
```


Si nos fijamos en el código anterior, sólo hace falta manejar las propiedades del objeto, como si de una imagen se tratase, es decir, establecer el objeto al tamaño máximo deseado (en este caso al 100% del ancho y alto de la ventana gráfica) y habilitar el ajuste del elemento con respecto a su contenedor con la propiedad `OBJECT-FIT`.

Por desgracia, no siempre es posible agregar contenidos multimedia a través de esta estructura y debemos recurrir a otros métodos.

Este es el caso de YouTube o Vimeo, los cuales requieren de un elemento `OBJECT` o `IFRAME` para poder incrustarse en nuestra página. Ambos casos, se verán con detalle, cuando lleguemos al capítulo de iframes y objetos.

5.6 PRACTICA Y JUEGA

Maquetación CSS – Parte 2	Código QR
Corrige el CSS solicitado y juega a cambiar el HTML para realizar tu propia personalización. https://codepen.io/pefc/pen/abaJdqq	

Maquetación CSS – Parte 3	Código QR
Corrige el CSS solicitado y juega a cambiar el HTML para realizar tu propia personalización. https://codepen.io/pefc/pen/bGxqqEP	