



MULTIMEDIA

¿Qué aporta Html, CSS, Javascript en mejorar carga de imágenes?





Multimedia - Imágenes

Veamos que nos aporta :

- **HTML**
- **Javascript**
- **CSS**

en la carga de imágenes



Multimedia - Imágenes - HTML

Las opciones para insertar imágenes en html son:

- ****
- **<map>**
- **<area>**
- **<picure>**



Multimedia - Imágenes - HTML -

Atributos: src, alt, style y float

- **src**

- Fichero → ``
- Carpeta → ``
- Servidor → ``
- Animadas → ``
- Como link →

```
<a href="default.asp">  
    
</a>
```



Multimedia - Imágenes - HTML -

Atributos: src, alt, width / height y float

- **alt**

- Da un texto alternativo a la imagen. **Mejora la accesibilidad**
 - ``

- **style**

- Especifica ancho y alto de la imagen, si no se especifican la página web puede parpadear mientras se carga la imagen
 - ``
 - ``
- Es preferible poner ancho y alto dentro de style para que las hojas de estilo no cambien los valores.



Multimedia - Imágenes - HTML -

Atributos: src, alt, width / height y float

- **float**

- Permite que la imagen flote hacia la derecha o hacia la izquierda de un texto

```
<p>  
  
A paragraph with a floating image.  
</p>
```

A paragraph with a floating image.



```
  
A paragraph with a floating image.  
</p>
```



A paragraph with a floating image.



Multimedia - Imágenes - HTML - <map>

Es una imagen donde se crean áreas clicables. Soporta:

- Atributos globales de HTML
- Atributos de eventos de HTML

Ejemplo:

```

```

```
<map name="workmap">  
  <area shape="rect" coords="34,44,270,350" alt="Computer" href="computer.htm">  
  <area shape="rect" coords="290,172,333,250" alt="Phone" href="phone.htm">  
  <area shape="circle" coords="337,300,44" alt="Cup of coffee"  
href="coffee.htm">  
</map>
```



Multimedia - Imágenes - HTML - <area>

Define una zona clicable en una imagen:

- Atributos globales de HTML
- Atributos de eventos de HTML
- Epecíficos:
 - alt → texto alternativo; `alt="Computer"`
 - coords → coordenadas de la zona clicable; `coords="34,44,270,350"`
 - download → descarga target cuando se haga click en la zona
 - `<area shape="rect" coords="0,0,82,126" alt="Sun" href="info_about_the_sun.htm" download="sun">`
 - href → link del target y hreflang indica el idioma del URL
 - media → qué medio/dispositivo está optimizada la URL de destino
 - shape → la forma del área: default, rect, circle, poly `<area shape="rect"`
 - target → dónde abre el documento: `<area target="_blank|_self|_parent|_topframeName">`
 - type → tipo de MIME `<area type="media_type">`



Multimedia - Imágenes - HTML - <picture>

Define un contenedor con varias fuentes de imágenes:

- En lugar de tener una imagen ampliada o reducida según el ancho de la ventana gráfica, se pueden diseñar varias imágenes para llenar mejor el espacio. ventana gráfica del navegador.

```
<picture>  
  <source media="(min-width:650px)" srcset="img_pink_flowers.jpg">  
  <source media="(min-width:465px)" srcset="img_white_flower.jpg">  
    
</picture>
```

- Atributos globales de HTML
- Atributos de eventos de HTML



Multimedia - Imágenes - HTML - trucos

1. **Cargar color promedio:** color medio entre todos los pixeles → disimula carga de imagen

```
<div style="background-color: #333">  
    
</div>
```

¿Cómo calcular el color promedio?

- Enlace online: [Average color](#)
- Ejecutar en consola:

```
convert tu_imagen.jpg -format %c -depth 8 histogram:info:histogram_image.txt  
sort -n histogram_image.txt | tail -1
```

Devuelve algo como: 39356: (11, 10, 18) #0B0A12 srgb(11,10,18)



Multimedia - Imágenes - HTML - trucos

1. **Mostrar imagen de baja resolución** y cuando la web esté cargada sustituir por la imagen real. (Haciendo uso de javascript)

En nuestro html:

```

```

Nuestro javascript:

```
function init() {  
    var imgDefer = document.getElementsByTagName('img');  
    for (var i = 0; i < imgDefer.length; i++) {  
        if(imgDefer[i].getAttribute('data-src')) {  
            imgDefer[i].setAttribute('src',imgDefer[i].getAttribute('data-src'));  
        }  
    }  
}  
window.onload = init;
```



Multimedia - Imágenes - HTML - trucos

2 Usar atributo srcset para definir la fuente:

```

```

- **2x** → dispositivos de buena resolución: pantallas o teles.
- **3x** → dispositivos de gran resolución: móviles, tablets

- image.jpg (original) = 720 x 500px
- image@**2x**.jpg = 1440 x 1000px
- image@**3x**.jpg = 2160 x 1500px



Multimedia - Imágenes - HTML - trucos

3 Explotando el atributo <picture>

```
<picture>  
  <source srcset="image-468.jpg" media="(min-width: 468px)">  
  <source srcset="image-600.jpg" media="(min-width: 600px)">  
  <source srcset="image-1024.jpg" media="(min-width: 1024px)">  
    
</picture>
```

Traducción. Dispositivos con:

- El ancho mínimo de **468px** use la imagen **image-468.jpg**
- Si el ancho mínimo es **600px** se usa **image-600.jpg**
- Si el ancho mínimo es **1024px** se usa **image-1024.jpg**
- Si no se llega a cumplir ninguna de esas reglas, se usa lo que hay en la etiqueta img.



Multimedia - Imágenes - HTML - trucos

3 Explotando el atributo <picture>

```
<picture>
  <source srcset="image-468.jpg 2x, image-468@2x.jpg 3x" media="(min-width: 468px)">
  <source srcset="image-600.jpg 2x, image-600@2x.jpg 3x" media="(min-width: 600px)">
  <source srcset="image-1024.jpg 2x, image-1024@2x.jpg 3x" media="(min-width:
1024px)">
  
</picture>
```

Traducción. Dispositivos con:

- El ancho mínimo de 468px use la imagen image-468.jpg
- Si el ancho mínimo es 600px se usa image-600.jpg
- Si el ancho mínimo es 1024px se usa image-1024.jpg
- Si no se llega a cumplir ninguna de esas reglas, se usa lo que hay en la etiqueta img.



Multimedia - Imágenes - HTML - trucos

4 Preload - Prefetch

Informamos al navegador de qué cosas cargar y cuando.

Preload → carga instantánea, según se lee html.

Prefetch → la descarga no tiene tanta prioridad

```
<link rel="preload" href="image.jpg" as="image" media="(min-width: 460px)">  
<link rel="prefetch" href="other-image.jpg" as="image" media="(min-width: 468px)">
```

Atributos:

- **rel** ayuda a definir si usar preload o prefetch.
- **href** define la ruta del archivo.
- **as** el tipo de archivo.
- **media** para definir en qué tamaño de pantalla va a cargar la imagen.



Multimedia - Imágenes - HTML - trucos

4 Preload - Prefetch *EJEMPLO*

```
<!DOCTYPE html>
<html>
  <head>
    <!-- Preload & Prefetch -->
    <link rel="preload" href="image.jpg" as="image" media="(min-width: 320px)">
    <link rel="prefetch" href="other-image.jpg" as="image" media="(min-width: 320px)">
  </head>
  <body>
    <header>
      <!-- Imagen 1 -->
      
    </header>

    <section>...</section>

    <footer>
      <!-- Imagen 2 -->
      
    </footer>
  </body>
</html>
```

- No sobrecarga.
- También usado para ficheros css y javascript

SABER MÁS



Multimedia - Imágenes - CSS - @media

```
.element {  
  background-image: url(image.jpg);  
}
```

```
/* Consultamos si la pantalla es de buena resolución */  
@media (min-device-pixel-ratio: 2), (min-resolution: 192dpi) {  
  .element {  
    background-image: url(image@2x.jpg);  
  }  
}
```

```
/* Consultamos si la pantalla es de alta resolución */  
@media (min-device-pixel-ratio: 3), (min-resolution: 288dpi){  
  .element {  
    background-image: url(image@3x.jpg);  
  }  
}
```



Multimedia - Imágenes - CSS - @media

Podría mezclar condiciones:

/* Imagen para dispositivos de alta resolución y el ancho mínimo de 468px */

```
@media (min-device-pixel-ratio: 2) and (min-width: 468px), (min-resolution:  
192dpi) and (min-width: 468px), {  
  .element {  
    background-image: url(image-468@2x.jpg);  
  }  
}
```



Multimedia - Imágenes - CSS - image-set

Lo anterior podemos usarlo con: image.set. [Soportado por](#)

```
.element {  
  background-image: image-set(  
    url(image.jpg) 1x,  
    url(image@2x.jpg) 2x,  
    url(image@3x.jpg) 3x  
  );  
}
```

```
.element {  
  background-image: image-set(  
    url(image.png) 1x,  
    url(image@2x.png) 2x,  
    url(image@3x.png) 3x  
  );  
}
```

Practiquemos ...

