

# Incorporación de Imágenes, videos y embebidos

iframes, embed, video, img, figure

Julian F. Latorre

September 20, 2024

# Agenda

Imágenes en HTML

Videos en HTML

Contenido Embebido

Optimización de Contenido Multimedia

Accesibilidad en Contenido Multimedia

Práctica

Conclusión

# La etiqueta `img`

- ▶ Sintaxis básica:

```

```

- ▶ Atributos importantes:

- ▶ **src**: Ruta de la imagen
- ▶ **alt**: Texto alternativo
- ▶ **width** y **height**: Dimensiones

## Formatos de imagen comunes

- ▶ JPEG: Fotografías y imágenes con muchos colores
- ▶ PNG: Gráficos, logos, imágenes con transparencia
- ▶ GIF: Animaciones simples
- ▶ SVG: Gráficos vectoriales escalables

# Responsive Images

- ▶ Usar CSS para imágenes responsive:

```
img {  
    max-width: 100%;  
    height: auto;  
}
```

# La etiqueta ¡figure!

- Encapsula contenido multimedia con descripción:

```
<figure>  
    
  <figcaption>Leyenda de la imagen</figcaption>  
</figure>
```

# La etiqueta ¡video!

## ► Sintaxis básica:

```
<video width="320" height="240" controls>  
  <source src="video.mp4" type="video/mp4">  
  Tu navegador no soporta el elemento video.  
</video>
```

## ► Atributos: **controls, autoplay, loop, muted**

## Formatos de video soportados

- ▶ MP4: Ampliamente soportado
- ▶ WebM: Formato abierto y libre
- ▶ Ogg: Otro formato libre



## Poster para videos

- Usar el atributo **poster** para mostrar una imagen previa:

```
<video poster="imagen_previa.jpg">  
  ...  
</video>
```

## La etiqueta `iframe`

- ▶ Permite incrustar contenido de otras fuentes:

```
<iframe src="https://www.youtube.com/embed/  
VIDEO_ID"  
width="560" height="315"  
frameborder="0"  
allowfullscreen>  
</iframe>
```

- ▶ Usos comunes:
  - ▶ Videos de YouTube
  - ▶ Mapas de Google
  - ▶ Tweets de Twitter
  - ▶ Posts de Facebook

## Seguridad con iframes

- ▶ Usar el atributo **sandbox** para restringir capacidades
- ▶ Establecer una política de seguridad de contenido (CSP)

## La etiqueta `<embed>`

- ▶ Otra forma de insertar contenido externo:

```
<embed src="archivo.swf"  
       type="application/x-shockwave-flash"  
       width="300" height="200">
```

- ▶ Menos flexible que `<iframe>`

# Compresión de imágenes

## ► Herramientas útiles:

- TinyPNG
- ImageOptim
- Squoosh

# Lazy Loading

- Cargar imágenes y videos solo cuando son necesarios:

```

```

## Formatos modernos de imagen

- ▶ WebP: Mejor compresión que JPEG y PNG
- ▶ AVIF: Aún más eficiente, pero con menor soporte

## Texto alternativo para imágenes

- ▶ Siempre proporcionar un atributo **alt** descriptivo:

```

```



# Subtítulos y transcripciones para videos

- ▶ Usar la etiqueta `<track>` para añadir subtítulos:

```
<video controls>
  <source src="video.mp4" type="video/mp4">
  <track kind="subtitles"
        src="subtitulos.vtt"
        srclang="es"
        label="Español">
</video>
```

## Ejercicio práctico

Crear una página web que incorpore:

- ▶ Una galería de imágenes usando `¡figure¿` y `¡figcaption¿`
- ▶ Un video con controles y poster
- ▶ Un mapa de Google embebido usando `¡iframe¿`
- ▶ Una imagen con lazy loading

## Resumen

- ▶ Imágenes: `¡img!`, formatos, responsive
- ▶ Videos: `¡video!`, formatos, atributos
- ▶ Contenido embebido: `¡iframe!`, `¡embed!`
- ▶ Optimización y accesibilidad

## Próximos pasos

- ▶ Practicar la incorporación de elementos multimedia
- ▶ Explorar técnicas avanzadas de optimización
- ▶ Mantener el enfoque en la accesibilidad

¿Preguntas?



¡Gracias por su atención!