

UT4_7.- HOJAS DE ESTILO. AVANZADO - IMÁGENES

RESULTADOS DE APRENDIZAJE ASOCIADOS
<ol style="list-style-type: none">1. Reconoce las características de lenguajes de marcas analizando e interpretando fragmentos de código.2. Utiliza lenguajes de marcas para la transmisión de información a través de la Web analizando la estructura de los documentos e identificando sus elementos
CRITERIOS DE EVALUACIÓN
<ul style="list-style-type: none">- De RA1 – desde CEA hasta CEK- De RA2 – desde CEA hasta CEH

UT4_7.- HOJAS DE ESTILO. AVANZADO - IMÁGENES

Índice de contenido

1.- Introducción.....	2
2.- Imágenes responsivas.....	3
3.- Centrado de imágenes.....	4
4.- Filtros.....	5
5.- Clip-path.....	6

1.- Introducción

Las imágenes son uno de los elementos más importantes dentro del diseño web, la correcta utilización de las imágenes, puede ser la diferencia entre una buena página y una página no tan buena.

Hay muchos aspectos que debemos tener en cuenta cuando elegimos una imagen:

- color
- tamaño
- posición
- espacio que ocupa, etc.

Esto son los factores que debemos sopesar siempre antes de añadir una imagen a nuestra página.

2.- Imágenes responsivas

Otro elemento importante es hacer que nuestras imágenes tengan un **comportamiento responsivo**. Este tema es tratado de formas muy diferentes, nosotros vamos a hacerlo de una forma muy sencilla utilizando la propiedad **max-width**, que hace que nuestras imágenes tengan un comportamiento responsivo.

max-width:100%;

Lo que vamos a conseguir es que la imagen se adapte al tamaño de su contenedor.

Por defecto las imágenes son elementos en línea, esto puede que nos cause algún que otro problema, normalmente **lo que se suele hacer convertir la imagen, o su contenedor, en elemento de bloque**.

Otra forma de hacer que nuestra imagen sea responsiva cuando el contenedor no tiene un tamaño proporcional al tamaño de la imagen es **object-fit** que lo vamos a utilizar para especificar cómo se debe cambiar el tamaño del **img** (también se puede utilizar con **video**) para que se ajuste a su contenedor.

object-fit	
Valores	fill contain cover none scale-down
Descripción	Define cómo se debe cambiar el tamaño de o <video> para que se ajuste a su contenedor.

- ✓ **fill:** valor por defecto
- ✓ **contain:** El contenido se ajustará hasta rellenar de forma horizontal o vertical el contenedor sin deformarse
- ✓ **cover:** El contenido se ajustará hasta rellenar de forma horizontal y vertical el contenedor sin deformarse
- ✓ **none:** El contenido no se redimensiona y mantiene su tamaño original mostrando solo el trozo de las dimensiones especificadas
- ✓ **scale-down:** Selecciona el menor de la comparación entre none y contain.

3.- Centrado de imágenes

Si queremos centrar **horizontalmente** una imagen, lo haremos en función del tipo de elemento que tengamos, vamos a realizar el centrado de una forma u otra.

- Si no hemos modificado el tipo de elemento, la **imagen** será un elemento **de línea** y lo centraremos con **text-align**.

```
text-align:center;
```

- Si hemos convertido la **imagen** a un **elemento de bloque** centraremos la imagen con **margin**:

```
margin: 0 auto;
```

- Si lo que necesitamos es **centrar verticalmente** la imagen respecto a un texto, las cosas son un poco más complejas, **la mejor solución es utilizar flex**. Si queremos centrar la imagen sola o con un texto que sólo tenga una línea podemos utilizar **vertical-align**

vertical-align: middle

4.- Filtros

La propiedad filter nos permite aplicar una transformación a una imagen, existen diferentes tipos de filtros, con diferentes formatos.

filter	
Valores	none blur brightness contrast drop-shadow grayscale hue-rotate invert opacity saturate sepia
Descripción	Nos permite aplicar un filtro sobre una imagen.

- ✓ **blur(px)**: desenfoca la imagen
- ✓ **brightness(%)**: Ajusta el brillo de la imagen siendo 1 el original. De 0 a 1 dan oscuridad y de 1 a n dan sobreexposición
- ✓ **contrast(%)**: Ajusta el contraste de la imagen siendo 1 el original. De 0 es negro y de 1 a n dan más contraste
- ✓ **drop-shadow(h-shadow v-shadow blur color)**: Aplica una sombra paralela a la imagen
- ✓ **grayscale(%)**: Convierte la imagen a escala de grises, 0 es el original y 1 sería blanco y negro completamente
- ✓ **hue-rotate(deg)**: Añade matiz de color a la imagen. Se da un valor en grados según la rueda cromática. El valor máximo es 360deg
- ✓ **invert(%)**: Invierte el color de la imagen. Saca un negativo. 0 es el valor por defecto y 1 es totalmente invertida
- ✓ **opacity(%)**: Controla la opacidad de la imagen. 1 es el valor por defecto y 0 es transparente

- ✓ **saturate(%):** Controla la saturación de color de la imagen. 1 es el valor por defecto, 0 es totalmente desaturada y por encima de 1 se sobresaturada
- ✓ **sepia(%):** Aplica un tono sepia a la imagen. 0 es el valor por defecto y 1 es totalmente sepia

Estos filtros se pueden combinar de la siguiente manera:

```
.efecto:hover{
  filter: brightness(200%) contrast(200%);
}
```



5.- Clip-path

En CSS podemos aplicar una máscara que nos permite ocultar partes de una caja.

clip-path	
Valores	circle ellipse inset contrast drop-shadow grayscale hue-rotate invert opacity saturate sepia
Descripción	Nos permite aplicar una mascar a una capa.

- ✓ **circle():** dibuja un círculo, puede tener una medida fija, circle(100px) o podemos especificar un **centro** con at, circle(100px at medidaX medidaY | palabrasClaveX palabrasClaveY)
- ✓ **ellipse():** dibuja una elipse, funciona igual que el círculo, pero en este caso tenemos que especificar 2 centros, el horizontal y el vertical

- ✓ **inset():** dibuja un borde transparente por dentro de la caja
 - inset(all | Y X | top X bottom | top left bottom right)


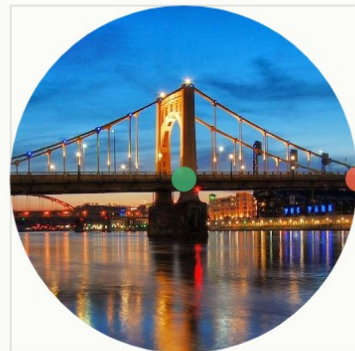
Existe la opción de redondear las esquinas, para ello después de los valores de inset pondremos round

 - inset(round all |
top-left & bottom-right top-right & bottom-left |
top-left top-right & bottom-left bottom-right
top-left top-right bottom-right bottom-left)
- ✓ **polygon():** Especifica una serie de puntos (mínimo 3) para definir la zona visible siendo el punto 0 0 la esquina superior izquierda los puntos se especifican por parejas de ejeX y ejeY separados por comas polygon(0 0, 100% 0, 0 100%)



Podemos utilizar esta propiedad de una manera mucho más cómoda:

<https://bennettfeely.com/clippy/>



```
clip-path: circle(50% at 50% 50%);
```