



# DevWomen: Introducción a la programación

**HTML5/CSS**

Montse Ortega

12/2024



Cofinanciado por  
la Unión Europea



GOBIERNO  
DE ESPAÑA  
MINISTERIO  
DE INDUSTRIA  
Y TURISMO



Escuela de  
organización  
industrial



Fondos Europeos



AJUNTAMENT  
DE VALÈNCIA



València  
Innovation  
Capital



# Repositorio github con ejemplos

- Podeis bajar los ejemplos de este repositorio:

[https://github.com/ammont82/programacion\\_html\\_css](https://github.com/ammont82/programacion_html_css)

- Este repositorio se irá actualizando a medida que avance el curso





# Colores CSS





# Colores en CSS

- Propiedades CSS que se utilizan para cambiar el color de texto y el de fondo

Propiedad	Valor	Significado
color	COLOR	Cambia el color del texto que está en el interior de un elemento
background-color	COLOR	Cambia el color de fondo de un elemento

```
.element {  
  background-color: indigo; /* Color de fondo */  
  color: white; /* Color de texto */  
}
```





# Colores en CSS

Esquema	Descripción	Más info
Espacio de colores RGB		
red	Establece un color mediante palabra clase	<a href="https://developer.mozilla.org/es/docs/Web/CSS/color_value">https://developer.mozilla.org/es/docs/Web/CSS/color_value</a>
rgb()	Usa una función rgb (rojo, verde y azul)	Modelo para representar colores basados en la combinación de estos tres colores primarios.
#rrggbb	Notación RGB abreviada en hexadecimal	Es el formato más utilizado por los desarrolladores web.
hsl()	Usa una función hsl (color,saturación y brillo)	Modelo para representar colores basado en el incremento/decremento de la luminosidad y la saturación
hwb()	Usa una función hwb (color,claridad y oscuridad)	Modelo para representar colores basado en valores más claros o oscuros y matiz de color.





# Colores en CSS

Esquema	Descripción	Más info
Espacio de color independiente del dispositivo		
lab()	Usa una función lab() y oklab()	Las siglas LAB significan: <ul style="list-style-type: none"><li>- L: luminosidad de negro a blanco</li><li>- A: luminosidad de verde a rojo</li><li>- B: luminosidad de azul a amarillo</li></ul>
lch()	Usa una función lch() y oklch()	Las siglas LCH significan: <ul style="list-style-type: none"><li>- L: luminosidad de negro a blanco</li><li>- C: saturación</li><li>- H: tono de color</li></ul>





# Colores en CSS

- Existen valores especiales relacionados con el color:

Valor	Significado
transparent	Establece un color como completamente transparente
currentColor	Establece el mismo color que se está utilizando para el texto (tanto en CSS cómo SVG)



# Función RGB

- Las siglas RGB significan Rojo (Red), verde (green) y azul (blue)
- Hay dos formas de utilizar la función `rgb()`:

Función RGB	Descripción
<code>rgb(r g b)</code>	Valores numéricos NUMBER o porcentajes PERCENT separados por espacios
<code>rgb(r g b / a)</code>	Se añade un valor correspondiente al grado de transparencia



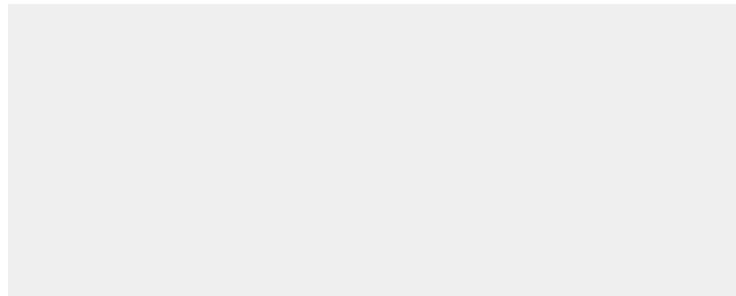


# Función RGB

rgb(**150** + **148** + **150**) = rgb(150 148 150)



rgba(**150** + **148** + **150** + 0.2) = rgba(150 148 150 0.2)





# Formato hexadecimal

- Es el más utilizado por los desarrolladores web
- Siempre empieza por una # y un conjunto de letras y números. Cada par de números simboliza el valor de un canal de RGB
- `rgb(255 0 0) => #FF0000 =>`





# Formato hexadecimal

Palabra clave	Función RGB	Hexadecimal	Hex. abreviado
red (rojo)	rgb(255 0 0)	#FF0000	#F00
black (negro)	rgb(0 0 0)	#000000	#000
cyan (azul claro)	rgb(0 255 255)	#00FFFF	#0FF
mediumpurple (lila)	rgb(147 112 219)	#9370DB	#97D

<https://www.peko-step.com/es/tool/tfcolor.html>





# La función HSL()

- Las siglas HSL significan Color (hue), Saturación (saturation) y Luminosidad (luminosity)

Función HSL	Descripción
hsl(h s l)	Número de grados en ANGLE , junto a PERCENT porcentajes de saturación y luminosidad
hsl(h s l / a)	Idem al anterior, añadiendo un canal alfa de transparencia a



# La función HSL()

HEX	RGB	HSL
#8C00FF	rgb (140 0 255)	hsl (273 100 50)



<https://www.peko-step.com/es/tool/hslrgb.html>



Cofinanciado por  
la Unión Europea



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA  
Y TURISMO

**EOI**

Escuela de  
organización  
industrial



Fondos Europeos



AJUNTAMENT  
DE VALÈNCIA

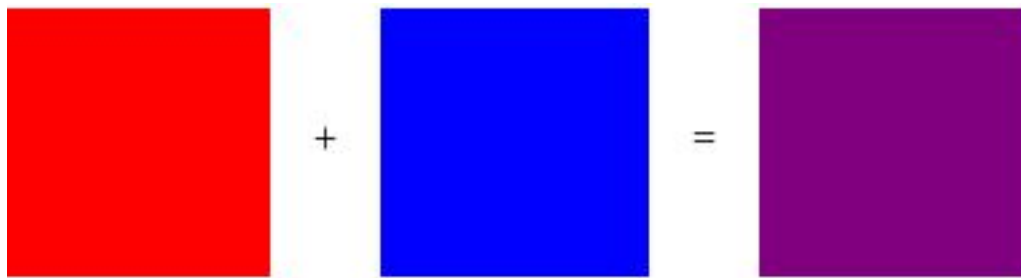


València  
Innovation  
Capital



# La función color-mix

- Esta función permite mezclar dos colores determinados en un espacio de color concreto.
- color-mix(in srgb, red, blue)





# Colores relativos

- Es una forma de establecer un color en relación a otro color diferente.
- Se pueden derivar usando **from**

```
.item {  
  width: 100px;  
  height: 100px;  
  background: rgb(from #a8201a r g b);  
}
```





# Colores relativos

- También se pueden obtener usando la función `calc()`
- Hay que partir del color de la variable
  - - color.





# LAB

- Las siglas LAB significan L (luminosidad de negro a blanco), A (luminosidad de verde a rojo) y B (luminosidad de azul a amarillo)

Función LAB	Descripción
lab(l a b)	Tres parámetros donde se puede indicar el formato con PERCENT o NUMBER
lab(l a b / t)	Idem al anterior, añadiendo un canal alfa de transparencia t

`background-color: lab(25 25 125);`





# OKLAB

- La función oklab() es una versión mejorada de la función lab(). Tienen los mismos parámetros.
- Los parámetros **A** y **B** de lab() se mueven entre -125 y 125, los parámetros **A** y **B** de oklab() se mueven entre valores -0.4 y 0.4.



# LCH

- Las siglas LCH significan L (luminosidad de negro a blanco), C (saturación) y H (tono de color)

Función LCH	Descripción
lch(l c h)	Puedes indicar formato con PERCENT o NUMBER en l y c
lch(l c h a)	Idem al anterior, añadiendo canal de transparencia a

**background-color: lch(25% 150 360);**





# OKLCH

- La función `oklch()` es una versión mejorada de la función `lch()`. Tienen los mismos parámetros.
- Los parámetros **C** y **H** de `lch()` se mueven entre 0 y 150, los parámetros **C** y **H** de `oklch()` se mueven entre valores 0 y 0.4.



# Espacios de color

- Un espacio de color es un conjunto de colores para mostrar o reproducir en un medio.
- En CSS hay un conjunto de espacios de color predefinidos.



# Espacios de color

Espacio de color	Descripción
sRGB	Espacio de color estándar creado por Microsoft y HP
srgb-linear	Idéntico a sRGB pero con diferente función de transferencia de luz
display-p3	Variación de Apple de DCI-P3
a98-rgb	Espacio de color desarrollado por Adobe
prophoto-rgb	Espacio de color desarrollado por Kodak
rec2020	Espacio de color basado en recomendación ITU-R BT.2020
xyz	Espacio de color basado en el espacio CIE 1931 XYZ
xyz-d65	Como el anterior pero usando D65
xyz-d50	Como el anterior pero usando D50



# La función color()

- Por defecto el navegador trabaja en srgb
- Para usar un espacio de color diferente podemos utilizar la función color() de CSS

```
.item-3 {  
background: color(display-p3 50% 25% 75%);  
}  
  
.item-4 {  
background: color(a98-rgb 50% 25% 75%);  
}
```





# La función color()

## Comparación de Espacios de Color

sRGB

sRGB Linear

Display P3

Adobe RGB

ProPhoto RGB

Rec.2020

XYZ

XYZ D65

XYZ D50



Cofinanciado por  
la Unión Europea



GOBIERNO  
DE ESPAÑA



MINISTERIO  
DE INDUSTRIA  
Y TURISMO



Escuela de  
organización  
industrial



Fondos Europeos



AJUNTAMENT  
DE VALÈNCIA



València  
Innovation  
Capital





# Ejercicios prácticos de colores con CSS





# Ejercicio: Tarjetas de colores personalizadas

- En este ejercicio, crearás unas tarjetas de color personalizadas.
- Objetivo
  - Aplicar colores al texto (color) y colores de fondo (background-color) a diferentes elementos HTML.
  - Practicar con valores de colores usando:
    - Nombres de colores (red, blue, green)
    - Valores HEX (#ff6347)
    - Valores RGB (rgb(255, 99, 71))
    - Valores HSL (hsl(120, 100%, 50%))



# Ejercicio: Tarjetas de colores personalizadas

- Instrucciones del Ejercicio
  - Abre el archivo HTML en tu navegador y observa las tarjetas.
  - Modifica los colores:
    - Cambia los valores de color (color del texto).
    - Cambia los valores de background-color (color de fondo) usando HEX, RGB, o HSL.
  - Añade dos tarjetas más:
    - Una tarjeta con un color de fondo amarillo.
    - Otra tarjeta con un color de fondo gris oscuro.



# Ejercicio: Tarjetas de colores personalizadas

## Ejercicio: Colores y Fondos en CSS





## Ejercicio: Espacios de color

- En este ejercicio, crearemos una comparación de espacios de color
- Objetivo
  - Crear una serie de cajas del mismo color que cada una tenga un espacio de color distinto.
  - De esta manera podemos comparar como se ven los color en los espacios de color definidos.
  - Usaremos la función **color()** que es compatible con navegadores modernos como Chrome, Safari y Edge.





# Ejercicio: Espacios de color

## Estructura HTML:

- Cada espacio de color tiene un contenedor **.color-box**.
- Los colores se muestran con etiquetas correspondientes para identificar su espacio.

## Espacios de Color en CSS:

- **srgb**: El espacio de color estándar para la web.
- **srgb-linear**: Una versión lineal de **sRGB**, sin la corrección gamma.



## Ejercicio: Espacios de color

- **display-p3**: Un espacio de color más amplio soportado por dispositivos Apple.
- **a98-rgb**: Adobe RGB, útil para impresión y fotografía.
- **prophoto-rgb**: Un espacio aún más amplio usado en fotografía profesional.
- **rec2020**: Un espacio de color diseñado para video HDR.
- **xyz, xyz-d65, xyz-d50**: Espacios de color basados en el modelo CIE 1931.



# Ejercicio: Espacios de color

## Comparación de Espacios de Color

sRGB

sRGB Linear

Display P3

Adobe RGB

ProPhoto RGB

Rec.2020

XYZ

XYZ D65

XYZ D50



Cofinanciado por  
la Unión Europea



GOBIERNO  
DE ESPAÑA



MINISTERIO  
DE INDUSTRIA  
Y TURISMO

**EOI** Escuela de  
organización  
industrial



Fondos Europeos



AJUNTAMENT  
DE VALÈNCIA



València  
Innovation  
Capital





# Imágenes y fondos en CSS



Cofinanciado por  
la Unión Europea



GOBIERNO  
DE ESPAÑA  
MINISTERIO  
DE INDUSTRIA  
Y TURISMO



Escuela de  
organización  
industrial



Fondos Europeos



AJUNTAMENT  
DE VALÈNCIA



València  
Innovation  
Capital



# La propiedad background-image

- Con esta propiedad indicamos que queremos usar una imagen de fondo

Propiedad	Valor	Significado
background-image	<b>none</b>	No utiliza ninguna imagen de fondo.
background-image	url("imagen.jpg")	Usa la imagen indicada como fondo.
background-image	image-set(...)	Indica una imagen con fallbacks.
background-image		Utiliza un gradiente de tipo lineal, radial o cónico.



# La propiedad background-image

- En el caso de que sólo se coloque el nombre de la imagen (por ejemplo, imagen.jpg), el navegador buscará la imagen en la misma carpeta donde está el archivo CSS que estamos escribiendo. Esto es lo que se llama una **ruta relativa**.





# La propiedad background-image

- En el caso de que se coloque la ruta completa, por ejemplo <https://lenguajecss.com/assets/logo.svg>, se accederá a la imagen alojada en esa dirección URL. Esto es lo que se llama **ruta absoluta**.



# La propiedad background-image

- En el caso de que no se encuentre la imagen o el valor de **background-image** se haya establecido a **none**, no se utilizará ninguna imagen de fondo, y en su lugar se mostrará el color establecido con **background-color**.



# Opciones de imagen de fondo

Propiedad	Descripción
background-repeat	Establece el <b>modo</b> en el que se repite la imagen de fondo de ser muy pequeña.
background-attachment	Indica si la imagen de fondo permanece fija o se desplaza al hacer scroll.
background-position	Establece una <b>posición</b> para la imagen de fondo, de modo que podemos moverla de sitio.
background-clip	Modo de recorte de la imagen de fondo. (Ver modelo de cajas).
background-origin	Modo de posicionamiento de la imagen de fondo. (Ver modelo de cajas).
background-size	Establece un tamaño diferente a la imagen de fondo.



# background-repeat

Valor	Descripción
<b>repeat</b>	Repite la imagen de fondo horizontal y verticalmente.
repeat-x	Repite la imagen de fondo sólo horizontalmente (eje x).
repeat-y	Repite la imagen de fondo sólo verticalmente (eje y).
space	Repite la imagen y rellena con espacio los huecos.
round	Repite la imagen y amplía cada repetición para ajustar.
no-repeat	La imagen de fondo no se repite.



# background-attachment

- La propiedad **background-attachment** especificará si la imagen de fondo seguirá el desplazamiento del usuario al hacer scroll por la página

Valor	Significado
scroll	Cuando hacemos scroll la imagen de fondo <b>se desplaza</b> .
fixed	Cuando hacemos scroll, la imagen de fondo permanece <b>fija</b> .





# background-position

- La propiedad **background-position** permite desplazar la imagen en la zona especificada por y por . Por defecto, estos valores son **0% 0%**, y pueden especificarse tanto con unidades (porcentajes, píxeles, etc...) como mediante palabras clave que representan zonas predefinidas (top, left, right, bottom y center).



# background-clip / -origin

Propiedad	Valor	Significado
background-clip	<b>border-box</b>   padding-box   content-box	Área externa afectada por el fondo
background-origin	border-box   <b>padding-box</b>   content-box	Área interna afectada por el fondo

La propiedad **background-clip** establece el área externa que afectará la imagen de fondo, mientras que **background-origin** establece el área interna que afectará la imagen de fondo.



# background-size

Valor	Significado
<b>SIZE</b>	1 parámetro. Aplica un SIZE de (ancho × <b>auto</b> ) a la imagen de fondo. Mantiene la proporción
<b>SIZE SIZE</b>	2 parámetros. Aplica un SIZE de (ancho × alto) a la imagen de fondo. Hay que vigilar la proporción.

- Por defecto, una imagen de fondo toma **automáticamente el tamaño de la imagen.**



# background-size

Valor	Significado
<b>auto</b>	No escala la imagen. Utiliza el tamaño original. Es el valor por defecto.
unidad	Indicamos el tamaño específico que queremos usar (píxels o porcentaje, por ej.).
cover	Escala el ancho de la imagen de fondo al <b>ancho</b> del elemento.
contain	Escala el alto de la imagen de fondo al <b>alto</b> del elemento.



# Atajo: background

- Es posible establecer todas estas propiedades anteriores en una sola regla de CSS a modo de atajo, y así ahorrar mucho espacio en escribir las propiedades anteriores por separado.

Atajo	Orden	Ejemplo
background	COLOR IMAGE POSITION / SIZE REPEAT ATTACHMENT ORIGIN CLIP	<pre>/* Atajo simple */ background: #fff url("imagen.jpg") top center repeat-x;  /* Atajo completo */ background: deeppink url("imagen.jpg") 0 0 / 150px space scroll padding-box padding-box;</pre>



# Ejercicio práctico de background en CSS





# Ejercicio: Tarjeta de presentación estilizada

**Objetivo:** Crear una tarjeta de presentación con un fondo decorativo que use varias capas de **background**, patrones, y efectos avanzados.



# Ejercicio: Tarjeta de presentación estilizada

## background-image:

- Usa tres capas: una imagen principal, un gradiente decorativo y un patrón repetido.

## background-size:

- Controla el tamaño de cada capa:
  - **cover** para la imagen principal y el gradiente.
  - **10px 10px** para el patrón repetido.







# Ejercicio: Tarjeta de presentación estilizada

## background-repeat:

- Evita repetir la imagen principal y el gradiente (**no-repeat**).
- Permite repetir el patrón (**repeat**).

## background-clip:

- **border-box** hace que el fondo cubra el área del borde.



# Ejercicio: Tarjeta de presentación estilizada

## Actividades adicionales:

- Cambia la imagen principal por una URL de tu elección.
- Experimenta con diferentes patrones en **repeating-linear-gradient**.
- Cambia **background-clip** a **content-box** o **padding-box** y observa el efecto.





# Ejercicio: Tarjeta de presentación estilizada





# Imágenes y fondos en CSS (2)



Cofinanciado por  
la Unión Europea



GOBIERNO  
DE ESPAÑA  
MINISTERIO  
DE INDUSTRIA  
Y TURISMO



Escuela de  
organización  
industrial



Fondos Europeos



AJUNTAMENT  
DE VALÈNCIA



València  
Innovation  
Capital



# Fondos y gradientes múltiples

- CSS permite establecer múltiples fondos, sólo separando los valores con comas.

Propiedad	Descripción
background-image: IMAGE1, IMAGE2, IMAGE3	Establece imágenes de fondo de un elemento

- La última imagen que se añade en la lista es la primera que se renderiza en la página.



# Fondos y gradientes múltiples

- Al tener varias imágenes todas las propiedades con el prefijo background- se aplican a todos los fondos
- Cuando utilizamos propiedades como background-image con valores múltiples (*sobre todo cuando se trata de valores largos*), se suelen separar en diferentes líneas, bien indentadas.





# Fondos y gradientes múltiples

- Se pueden utilizar tanto imágenes como gradientes e incluso combinarlos
- Los gradientes múltiples permiten hacer cosas increíbles con pocas líneas de código



# Fondos y gradientes múltiples

```
HTML
1 <div class="element"></div>

CSS
1 .element {
2   width: 600px;
3   height: 200px;
4   background-image:
5     radial-gradient(transparent 20%, black 22% 30%,
6     transparent 32%),
7     linear-gradient(120deg, gold, transparent, gold),
8     linear-gradient(to top, indigo, red, indigo);
9   background-size: 200px 200px, cover, cover, cover;
10 }
```







# Visibilidad de elementos

- En CSS se pueden mostrar y ocultar elementos.
- Hay 3 formas principales que destacan:

Propiedad	Descripción
display	Modifica como se muestra un elemento. Con none, lo oculta.
visibility	Modifica como se muestra un elemento. Con hidden, lo oculta, pero mantiene su espacio visual.
opacity	Modifica el nivel de transparencia del elemento con un valor de 0% a 100%.



# Propiedad display

- Permite ocultar un elemento con el valor none.
- Esto hace que el navegador ignore el elemento y lo oculte por completo.
- Se puede ocultar un elemento con la propiedad `display:none` y mostrarlo con `display:block`.





# Propiedad visibility

- Permite ocultar un elemento con el valor hidden.
- La principal diferencia con display:none es que en este caso el hueco que ocupaba el elemento se mantiene pero vacío.

Valor	Significado
visible	El elemento es visible. Valor por defecto.
hidden	El elemento no es visible (pero sigue ocupando su espacio y posición).
collapse	Sólo utilizado en tablas. El elemento se contrae para no ocupar espacio.



# Propiedad opacity

- Se suele utilizar para transiciones o animaciones.
- Permite indicar valor numéricos entre 0 y 1. También se pueden usar % de 0% a 100%.
- El grado de opacidad se aplica al elemento y a todos sus hijos.



# Función image-set()

- En CSS, existe una función image-set() que puede utilizarse para establecer fallbacks (alternativas) de imágenes
- Se puede usar según el formato de imagen, la densidad de pantalla y el ancho de banda





# Función image-set()

- Si el navegador no soporta image-set podemos usar una alternativa usando url.

```
.element {  
  background-image: url("imagen.webp");  
  background-image: image-set(  
    url("imagen.webp") 1x,  
    url("imagen-hq.webp") 2x  
  );  
}
```



# Objetos en CSS

- En CSS, se denominan **objetos** a imágenes a través de la etiqueta `<img>`, elementos multimedia a través de `<video>` u otros elementos como `<textarea>` o `<input>`
- Muchas veces estos objetos tienen un tamaño inicial que no encaja con el uso que queremos darle o no se adapta a las cajas o contenedores que usamos



# Objetos en CSS

- Algunas propiedades en CSS nos permiten modificar ciertos aspectos de muchos de estos elementos, pudiendo darle estilo y adaptarlos.

Propiedad	Valores	Descripción
object-fit	<b>fill</b>   contain   cover   none   scale-down	Modo en que se adapta el elemento.
object-position	PERCENT PERCENT	Posición (x,y) del elemento.
object-view-box	SHAPE	Región del elemento se visualizará





# Propiedad object-fit

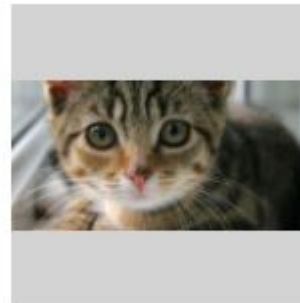
- La propiedad object-fit nos va a permitir cambiar el modo en el que se rellena o adapta una imagen `<img>` en su contenedor padre o en si mismo.

Valor	Descripción
fill	«Rellena» la imagen ocupando todo el espacio. Habitualmente, se produce un estiramiento de la imagen.
cover	Mantiene proporción, <b>cubriendo</b> lo máximo posible, sin que queden huecos sin cubrir. Suele ocultar partes.
contain	Mantiene proporción, <b>conteniendo</b> el máximo posible de la imagen. Suelen sobrar partes.
none	Mantiene el aspecto natural.
scale-down	Similar a contain, pero si la imagen es más pequeña, no la amplía.



# Propiedad object-fit

```
.item-1 { object-fit: fill; }  
.item-2 { object-fit: cover; }  
.item-3 { object-fit: contain; }  
.item-4 { object-fit: scale-down; }  
.item-5 { object-fit: none; }
```





# Propiedad object-position

- La propiedad object-position que nos sirve para utilizar junto a la propiedad object-fit y cambiar la posición donde aparece la imagen.
- Se pueden indicar top, left, right, bottom o center para indicar en qué zona centrar la imagen

Valor	Descripción
<b>50% 50%</b>	Por defecto, la imagen está centrada tanto en X como en Y.
PERCENT PERCENT	Se puede indicar un porcentaje para colocarlo en el eje correspondiente.



# Propiedad object-view-box

- La propiedad object-view-box nos permite indicar al navegador la región visible de un elemento a visualizar, es decir, su viewbox (caja de visualización).
- Para ello, utilizaremos por ejemplo, la función inset() para determinar la región que recortaremos



# Procesamiento de imágenes

- La orientación en la que se muestra una imagen o como se renderiza cuando hacemos una imagen más grande, es algo configurable mediante las siguientes propiedades CSS

Propiedad	Valor	Descripción
image-rendering	<b>auto</b>   smooth   high-quality   crisp-edges   pixelated	Algoritmo de escalado a utilizar.
image-orientation	<b>from-image</b>   none   ANGLE   [flip]	Orientación de la imagen.
aspect-ratio	<b>auto</b>   NUMBER/NUMBER   NUMBER	Define la proporción de aspecto.



# Propiedad image-rendering

- Permite indicar al navegador como se debería renderizar una imagen, y que algoritmo de reescalado se debería aplicar.

Valor	Descripción	Soporte
<b>auto</b>	Por defecto, el navegador decide que algoritmo de escalado utilizar.	
smooth	La imagen es escalada con algoritmos que priorizan la apariencia, ideal para fotos.	✗ <u>Poco</u>
high-quality	Idéntico a smooth, pero con mejor calidad. Utilizar cuando la calidad se degrada.	✗ Poco
crisp-edges	Enfocado en preservar contrastes y bordes de imagen. Ideal para líneas definidas y bocetos.	✓ <u>Bueno</u>
pixelated	Usa algoritmo de "vecinos más próximos", orientado a preservar pixel art bien definido.	✓ <u>Bueno</u>



# Propiedad image-orientation

- Permite rotar una imagen respecto a la información que se encuentra en sus metadatos EXIF, o por el contrario, desactivarla y mostrarla tal cual aparece en la propia imagen

Valor	Descripción
<b>from-image</b>	Usa la orientación de la imagen indicada en sus metadatos EXIF.
none	No se usa ninguna rotación adicional. Se muestra tal cual.



# Ejercicios prácticos de gradientes e imágenes







## Ejercicio: Fondos múltiples

**Objetivo:** Crear un HTML con **fondos múltiples** usando la propiedad **background-image**. El ejemplo debe superponer varias imágenes de fondo o combinar imágenes con colores o gradientes.



# Ejercicio: Fondos múltiples

## Resultado esperado:

La imagen base cubre completamente el fondo del contenedor.

La imagen superpuesta aparece sobre la base, claramente visible en el centro del contenedor.



Cofinanciado por  
la Unión Europea



Fondos Europeos



AJUNTAMENT  
DE VALÈNCIA



València  
Innovation  
Capital



# Ejercicio: Fondos múltiples

## Explicación del código:

background-position:

- La imagen base sigue centrada (center).
- La imagen superpuesta está centrada pero desplazada verticalmente (center 50px) para que sea más evidente.



Cofinanciado por  
la Unión Europea



GOBIERNO  
DE ESPAÑA



MINISTERIO  
DE INDUSTRIA  
Y TURISMO

**Eoi** Escuela de  
organización  
industrial



Fondos Europeos



AJUNTAMENT  
DE VALÈNCIA



València  
Innovation  
Capital



## Ejercicio: Fondos múltiples

background-size:

- La imagen base cubre todo el contenedor con cover.
- La imagen encima tiene un tamaño fijo de 100px x 100px para garantizar que no se escale fuera del rango visible.



Cofinanciado por  
la Unión Europea



Fondos Europeos



AJUNTAMENT  
DE VALÈNCIA



València  
Innovation  
Capital



## Ejercicio: Fondos múltiples

Bordes del contenedor:

- Se agregó un borde (border: 1px solid black) para delimitar claramente el área visible.



Cofinanciado por  
la Unión Europea



GOBIERNO  
DE ESPAÑA  
MINISTERIO  
DE INDUSTRIA  
Y TURISMO



Escuela de  
organización  
industrial



Fondos Europeos



AJUNTAMENT  
DE VALÈNCIA



València  
Innovation  
Capital



# Ejercicio: Fondos múltiples





## Ejercicio: Visibilidad de elementos

**Objetivo:** Crear un HTML para entender las diferencias entre visibility, opacity, y display en CSS. El ejemplo incluye tres elementos que cambiarán su visibilidad al hacer clic en botones.





# Ejercicio: Visibilidad de elementos

## 1. Elementos HTML:

- Hay tres cajas (div) con diferentes colores (red, green, blue).
- Cada caja está asociada a una funcionalidad de visibilidad.







# Ejercicio: Visibilidad de elementos

## 2. Clases CSS:

- hidden: Oculta el elemento con visibility: hidden. El elemento no es visible pero ocupa espacio.
- transparent: Hace al elemento transparente con opacity: 0. El elemento sigue ocupando espacio.



## Ejercicio: Visibilidad de elementos

- removed: Elimina el elemento completamente del flujo con display: none. No ocupa espacio ni es interactuable.

### 3. Botones y Funciones JavaScript:

- Cada botón activa o desactiva una clase CSS usando `classList.toggle()`:



## Ejercicio: Visibilidad de elementos

- Toggle Visibility: Alterna la clase hidden en box1.
- Toggle Opacity: Alterna la clase transparent en box2.
- Toggle Display: Alterna la clase removed en box3.



# Ejercicio: Visibilidad de elementos

## Ejercicio: Visibilidad con CSS



**Box 1**



**Box 2**



**Box 3**

Toggle Visibility (Box 1)

Toggle Opacity (Box 2)

Toggle Display (Box 3)





## Ejercicio: Uso de image-set

**Objetivo:** Crear un HTML que use image-set para mostrar imágenes diferentes según la resolución de pantalla

Usar background-image, background-size y background-position para probar la función





## Ejercicio: Uso de image-set

- background-image: Usa image-set para definir dos versiones de la imagen:
  - Una para pantallas estándar (1x).
  - Otra para pantallas de alta resolución como Retina o 4K (2x).





## Ejercicio: Uso de image-set

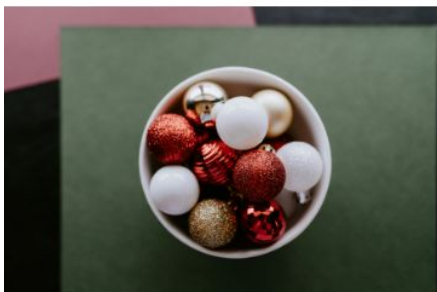
- background-size: cover -> Asegura que la imagen cubra completamente el contenedor.
- background-position: center -> Centra la imagen dentro del contenedor.
- Imágenes:
  - una para una resolución de pantalla alta
  - otra para una resolución más pequeña



# Ejercicio: Uso de image-set

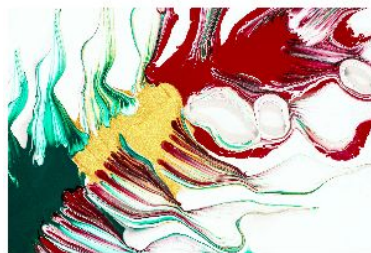
## Ejercicio: Uso de image-set

Esta caja cambia la resolución de la imagen de fondo según la pantalla. Si estás en un dispositivo con alta resolución, se cargará una imagen más nítida.



## Ejercicio: Uso de image-set

Esta caja cambia la resolución de la imagen de fondo según la pantalla. Si estás en un dispositivo con alta resolución, se cargará una imagen más nítida.



Cofinanciado por  
la Unión Europea



GOBIERNO  
DE ESPAÑA



MINISTERIO  
DE INDUSTRIA  
Y TURISMO

**EOI** Escuela de  
organización  
industrial



Fondos Europeos



AJUNTAMENT  
DE VALÈNCIA



València  
Innovation  
Capital





## Ejercicio: Galería de imágenes

**Objetivo:** crear una página que muestra una galería de imágenes. Estas imágenes tendrán diferentes configuraciones de ajuste (object-fit), posición (object-position) y una simulación de la propiedad SVG viewBox utilizando object-fit en imágenes dentro de un contenedor.





# Ejercicio: Galería de imágenes

## Práctica:

- Modifica las propiedades de las imágenes individualmente para aplicar diferentes valores de object-fit, object-position, y simular object-view-box:

1. Para la primera imagen:

```
.image-container:nth-child(1) img {  
  object-fit: cover;  
}
```



# Ejercicio: Galería de imágenes

2. Para la segunda imagen:

```
.image-container:nth-child(2) img {  
  object-fit: contain;  
}
```

3. Para la tercera imagen:

```
.image-container:nth-child(3) img {  
  object-fit: none;  
  object-position: center top;  
}
```





# Ejercicio: Galería de imágenes

## Resultado esperado

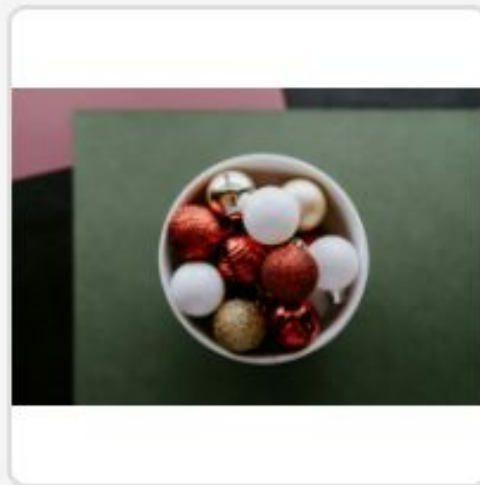
Deberías ver una galería con tres imágenes, cada una ajustada de manera diferente:

- object-fit: cover hace que la imagen llene completamente el contenedor recortando lo necesario.
- object-fit: contain asegura que la imagen se ajuste al contenedor sin recortarse.
- object-fit: none y object-position muestran la imagen en su tamaño original y posicionada según los valores proporcionados.



# Ejercicio: Galería de imágenes

## Galería de imágenes





# Ejercicio: Procesamiento de imágenes

**Objetivo:** crear un HTML que muestre imágenes distintas con distintos renderizados usando las propiedades image-rendering, image-orientation y aspect-ratio.





# Ejercicio: Procesamiento de imágenes

## Descripción:

- **image-rendering:** Se utiliza para controlar cómo se renderizan las imágenes, en este caso con el valor pixelated para enfatizar un efecto pixelado en imágenes de arte píxel.
- **image-orientation:** Se aplica para rotar imágenes automáticamente, en este ejemplo con un giro de 90 grados.
- **aspect-ratio:** Define proporciones específicas para contenedores, como en el div con relación 16:9.



# Ejercicio: Procesamiento de imágenes

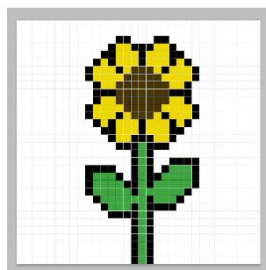


Image con image-rendering: pixelated.



Image con image-orientation: 90deg.



Este div tiene un aspecto fijo de 16:9.





# Gradientes o degradados





# Gradiente o degradado

- Es básicamente un color que cambia a otro color de forma gradual o brusca.
- Existen 3 funciones de gradientes: linear-gradient(), radial-gradient() y conic-gradient()

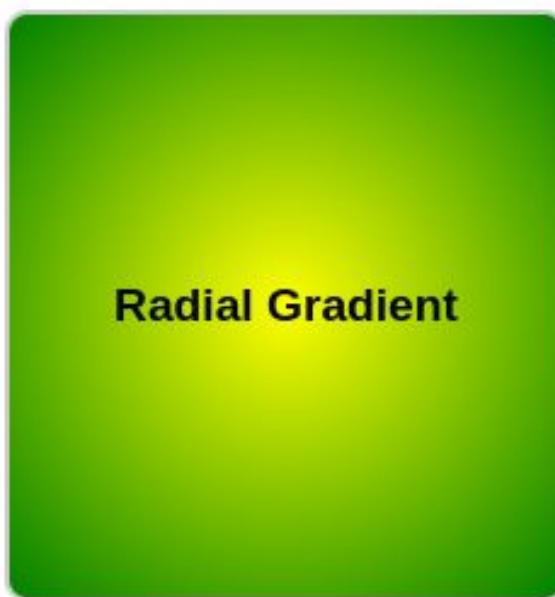




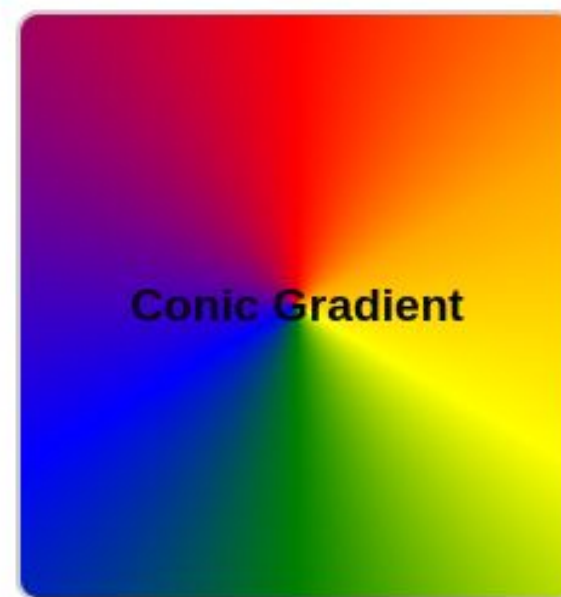
# Gradiente o degradado



**Linear Gradient**



**Radial Gradient**



**Conic Gradient**





# Gradiente o degradado

Función de gradiente	Significado
linear-gradient()	Define un gradiente lineal
radial-gradient()	Define un gradiente radial
conic-gradient()	Define un gradiente cónico

- Piensa que los navegadores, internamente, crean una imagen con el gradiente indicado en código. Por esa razón utilizamos background-image y no un background-gradient o un background-color.



# Gradientes repetibles

- Las funciones para crear gradientes repetitivos se basan en añadir el prefijo `repeating-` a las funciones de gradientes anteriores.

Función de gradiente	Significado
<code>repeating-linear-gradient()</code>	Define un gradiente lineal repetitivo
<code>repeating-radial-gradient()</code>	Define un gradiente radial repetitivo
<code>repeating-conic-gradient()</code>	Define un gradiente cónico repetitivo



# Gradientes lineales

- Permite crear fondos degradados que van en una dirección y cambian de un color a otro

Función y modalidad	Tipo de gradiente
linear-gradient( COLOR,COLOR , ... )	Gradiente de colores (hacia abajo).
linear-gradient(DIRECTION ,COLOR ,COLOR , ... )	Gradiente con dirección específica.
linear-gradient( DIRECTION,COLOR SIZE , COLOR SIZE , ... )	Se indica donde comienza a cambiar el color.
linear-gradient(DIRECTION,COLOR SIZE SIZE ,COLOR SIZE SIZE, ... )	Se indica inicio y fin de cada color.



# Gradientes lineales

- Hay más parámetros para hacerlos más flexibles:

Parámetros	Valor
DIRECTION	Dirección que tendrá el gradiente lineal. El valor por defecto es to bottom.
ANGLE	La dirección también se puede indicar con un número de grados
COLOR	Uno de los colores del gradiente
COLOR SIZE	Opcional: Se puede indicar a qué altura se encuentra el centro del color que lo precede.
COLOR SIZE SIZE	Opcional: Se puede indicar a qué altura comienza y acaba el color que lo precede.



# Gradientes lineales

```
.element {  
  width: 600px;  
  height: 100px;  
  background-image: linear-gradient(blue, red);  
}
```







# Gradientes lineales

- Por defecto las distancias entre colores se ajustan automáticamente con proporciones equilibradas. Si se especifica un tamaño SIZE (% o píxeles) justo después del color se puede variar la posición en la que aparece el color.
- También se puede indicar donde empieza un color y donde acaba



# Gradientes lineales repetitivos

- Si usamos `repeating-linear-gradient()` podemos indicar un patrón de colores que se van a repetir continuamente.
- El truco está en añadir el parámetro `SIZE` detrás de cada color. Así el navegador se encargará de completarlos y repetirlos.



# Gradientes radiales

- Tipo de degradado con formas circulares o elípticas. Hay que usar la función radial-gradient()
- Hay diferentes modalidades. Como mínimo hay que escoger entre circle y ellipse (que es la opción por defecto).



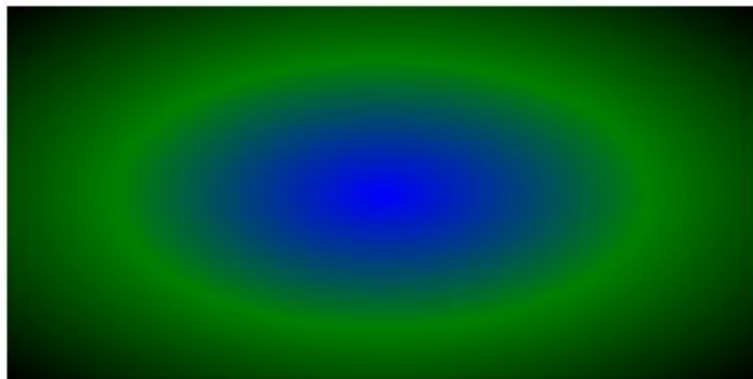
# Gradientes radiales

Función y modalidad	Tipo de gradiente
radial-gradient(COLOR,COLOR , ...)	Básico entre colores.
radial-gradient(forma,COLOR ,COLOR , ...)	Con forma circular o elíptica.
radial-gradient(forma SIZE,COLOR ,COLOR , ...)	Con tamaño o dimensión.
radial-gradient(forma SIZE at ubicación,COLOR ,COLOR , ...)	Con colocación inicial.
radial-gradient(forma SIZE at ubicación, COLOR SIZE ,COLOR SIZE , ...)	Con tamaños entre colores.
radial-gradient(forma SIZE at ubicación, COLOR SIZE SIZE ,COLOR SIZE SIZE , ...)	Con tamaños de inicio y fin.

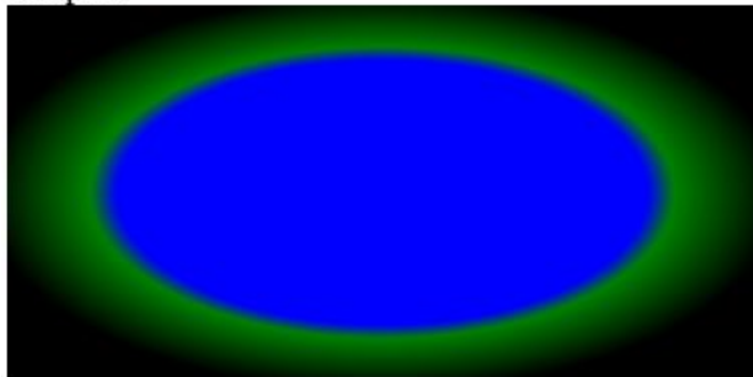


# Gradientes radiales

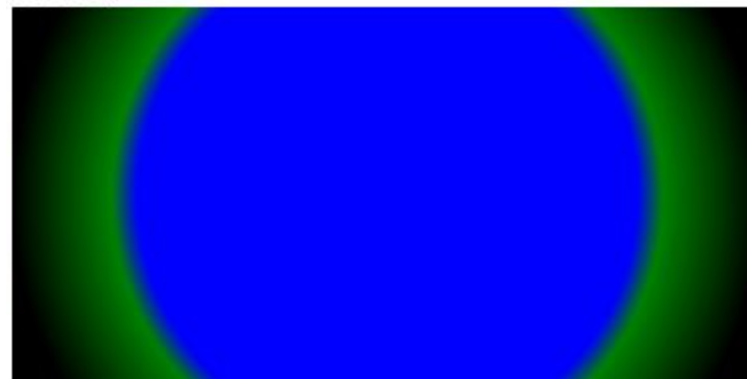
Basic:



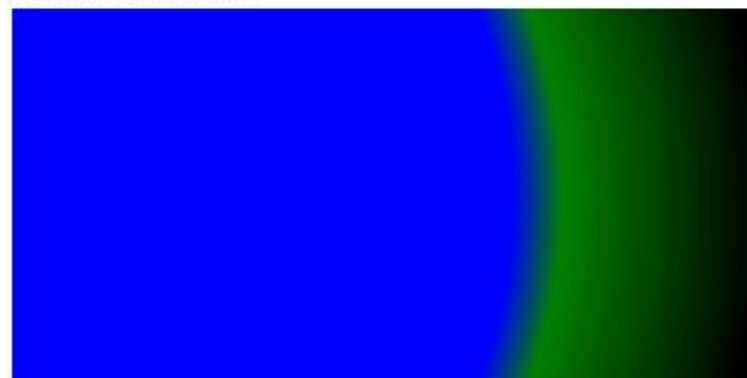
Ellipse:



Circle:



Circle detailed:





# Gradientes radiales

- Hay que indicar la forma que queremos:
  - circle SIZE: Crea un gradiente radial con forma circular
  - ellipse SIZE SIZE: Crea un gradiente radial con forma de elipse
- El tamaño SIZE puede ser de distintos valores.





# Gradientes radiales: forma

Parámetros	Valor
SIZE	Indicamos el <b>radio</b> del círculo (tamaño fijo). En la <b>elipse</b> , son dos valores (horizontal y vertical).
farthest-corner	Intenta cubrir hasta la <b>esquina más lejana</b> . Es el valor por defecto si no se indica.
closest-corner	Intenta cubrir hasta la <b>esquina más cercana</b> . Ídem al anterior si es un <b>elipse</b> .
farthest-side	Intenta cubrir hasta el <b>lado más lejano</b> .
closest-side	Intenta cubrir hasta el <b>lado más cercano</b> . Ídem al anterior si es un <b>elipse</b> .

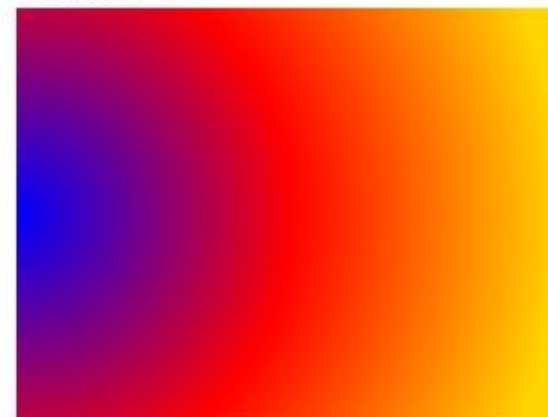


# Gradientes radiales: posición

Parámetros	Valor
SIZE SIZE	Indica la ubicación en <b>x</b> e <b>y</b> de la forma del gradiente, por ejemplo, con porcentajes.
[ubicación]	<b>center</b>   top   left   right   bottom   top left   top right   bottom left   bottom right

```
/* Gradiente circular de 250px de radio */  
.circle {  
  background: radial-gradient(circle 200px  
    at 0 50%, blue, red, gold);  
}
```

Circle:







# repeating-radial-gradient()

- La función repeating-radial-gradient() para indicar un patrón de colores que se repetirá continuamente.
- El truco está en que los tamaños SIZE indicados a continuación del color, teniendo en cuenta que no debe llegar al 100%, y el navegador se encargará de completarlos y repetirlos



# Gradientes cónicos

- Son parecidos al gradiente radial pero hacen referencia a una figura cónica
- La función conic-gradient() permite crear gradientes cónicos.



# Gradientes cónicos

Función y modalidad	Tipo de gradiente
conic-gradient(COLOR,COLOR , ...)	Básico entre colores.
conic-gradient(from ANGLE ,COLOR SIZE,COLOR SIZE , ...)	Con posición de color.
conic-gradient(from ANGLE at ubicación,COLOR SIZE,COLOR SIZE , ...)	Con ubicación del inicio.
conic-gradient(from ANGLE at ubicación,COLOR SIZE SIZE,COLOR SIZE SIZE, ...)	Con posición de inicio y fin

- Mediante la palabra clave **from** podemos indicar los grados desde los cuales comenzamos el gradiente cónico, que por defecto, se establece en 0deg.



# Gradientes cónicos

- Se pueden indicar varios tipos de unidades:

Dirección	Grados	Gradianes	Radianes	Giro
Arriba	from 0 ó from 360deg	from 0 ó from 400grad	from 0 ó from rad	from 0 ó from 1turn
Arriba-derecha	from 45deg	from 50grad	from 0.78rad ( $\pi/4$ )	from 0.125turn
Derecha	from 90deg	from 100grad	from 1.57rad ( $\pi/2$ )	from 0.250turn
Abajo-derecha	from 145deg	from 150grad	from 2.35rad	from 0.375turn
Abajo	from 180deg	from 200grad	from 3.14rad ( $\pi$ )	from 0.5turn
Abajo-izquierda	from 215deg	from 250grad	from 3.92rad	from 0.625turn
Izquierda	from 270deg	from 300grad	from 4.71rad ( $2\pi$ )	from 0.75turn
Arriba-izquierda	from 325deg	from 350grad	from 5.49rad	from 0,875turn

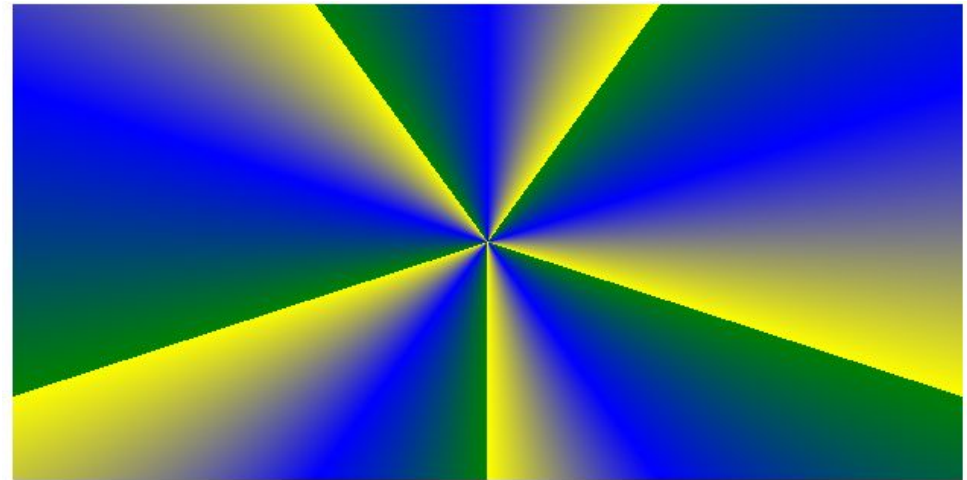




# repeating-conic-gradient()

- Es para indicar que se repite el gradiente

```
.element {  
  width: 500px;  
  height: 250px;  
  background:  
    repeating-conic-gradient(green  
    10%, blue 20%, yellow 30%);  
}
```





# Ejercicios de gradientes





# Ejercicio: gradiente lineal

- **Objetivo:** Crear un HTML y un CSS que tengan un div con un gradiente lineal de fondo. El gradiente lineal se puede usar en fondos, textos, bordes, etc.

Con este ejercicio se trabaja la función `linear-gradient()` con distintos parámetros para ver cómo funciona.



# Ejercicio: gradiente lineal

- **Explicación:**

- Gradiente lineal: En la propiedad background del .element, utilizamos linear-gradient(to bottom, #ff7e5f, #feb47b):
  - linear-gradient: Define que el fondo será un gradiente.
  - to bottom: Especifica que el gradiente va de arriba a abajo.





# Ejercicio: gradiente lineal

- **Explicación:**

- #ff7e5f y #feb47b: Son los colores que forman el gradiente. El gradiente comienza con el color #ff7e5f (un tono rosado) y termina con el color #feb47b (un naranja suave).





# Ejercicio: gradiente lineal

- **Personalización:**

Puedes cambiar los colores del gradiente modificando los valores de #ff7e5f y #feb47b por cualquier otro color que prefieras y puedes cambiar la dirección del gradiente (por ejemplo, de izquierda a derecha con `to right`).



# Ejercicio: gradiente lineal

¡Hola,  
Mundo!





# Ejercicio: gradientes radiales

- **Objetivo:** Crear un HTML y un CSS que tengan un div con un gradiente radial de fondo. El gradiente radial se puede usar en fondos, textos, bordes, etc.

Con este ejercicio se trabaja la función `radial-gradient()` con distintos parámetros para ver cómo funciona.



# Ejercicio: gradientes radiales

- **Explicación de gradiente radial:**
  - radial-gradient: Define que el fondo será un gradiente radial.
  - circle: Especifica que el gradiente tendrá forma circular.
  - #ff7e5f y #feb47b: Son los colores que forman el gradiente.



# Ejercicio: gradiente lineal

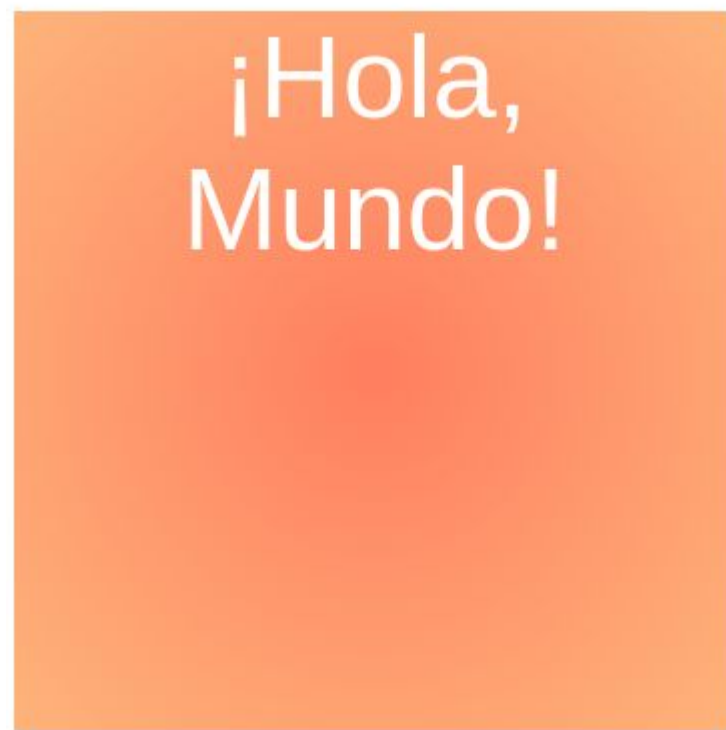
- **Personalización:**

Puedes cambiar los colores del gradiente modificando los valores de #ff7e5f y #feb47b por cualquier otro color que prefieras y puedes cambiar la forma del gradiente por ellipse.





# Ejercicio: gradientes radiales





# Ejercicio: gradientes cónicos

- **Objetivo:** Crear un HTML y un CSS que tengan un div con un gradiente cónico de fondo. El gradiente radial se puede usar en fondos, textos, bordes, etc.

Con este ejercicio se trabaja la función `conic-gradient()` con distintos parámetros para ver cómo funciona.





# Ejercicio: gradientes cónicos

- **Explicación de gradiente cónico:**
  - conic-gradient: Define que el fondo será un gradiente cónico.
  - from 0deg: Especifica el ángulo desde donde empieza el gradiente (en este caso, desde la parte superior del círculo, a 0 grados).
  - #ff7e5f y #feb47b: Son los colores que forman el gradiente.



# Ejercicio: gradientes cónicos

