



# MAQUETACIÓN CON CAPAS

# LINEAS GENERALES



## ¿QUE ES UNA CAPA?

Comúnmente llamados “**div**”  
son **cajas** en las que  
podemos incluir diferentes  
elementos como imágenes,  
textos, otros div....

### ATRIBUTOS

(GENERALES)

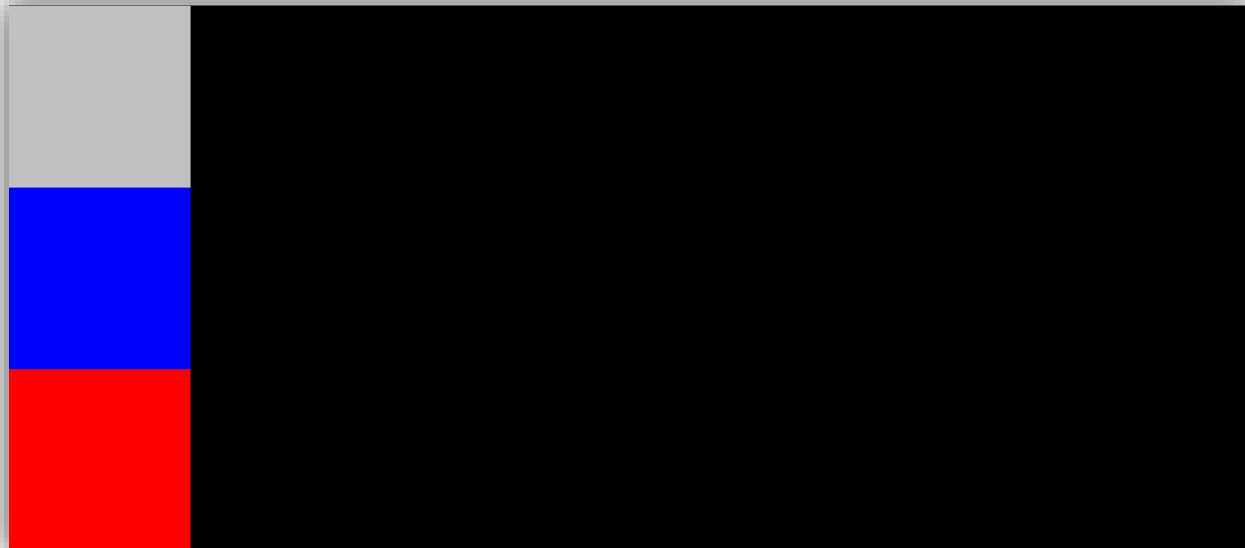
Width | Height | Position

### ATRIBUTOS

(PARA POSICIONAR)

Top | Bottom | Left | Right

## EJEMPLO



# ATRIBUTOS DE POSICIÓN



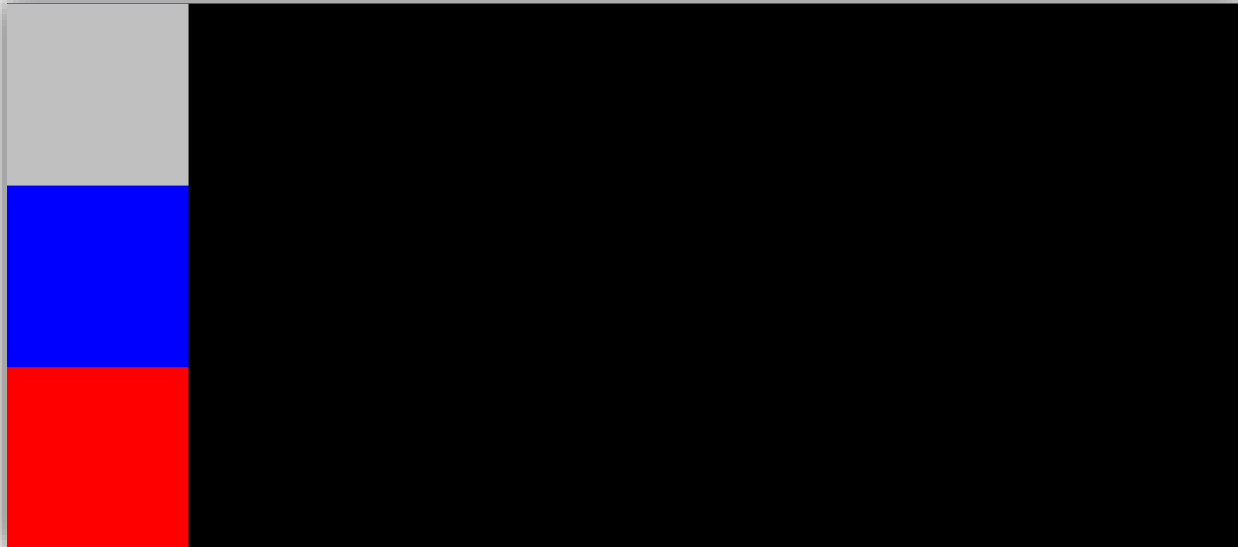
# POSITION : STATIC

Es el valor por **defecto**

Los elementos se colocan según el **flujo del HTML**

**No** se ven afectados por las propiedades top, bottom, left y right

## EJEMPLO





## POSITION : ABSOLUTE

Los elementos **dejan de seguir el flujo normal** del HTML por lo que puede llegar a provocar solapamientos

### EJEMPLO

Las 3 cajas están solapadas

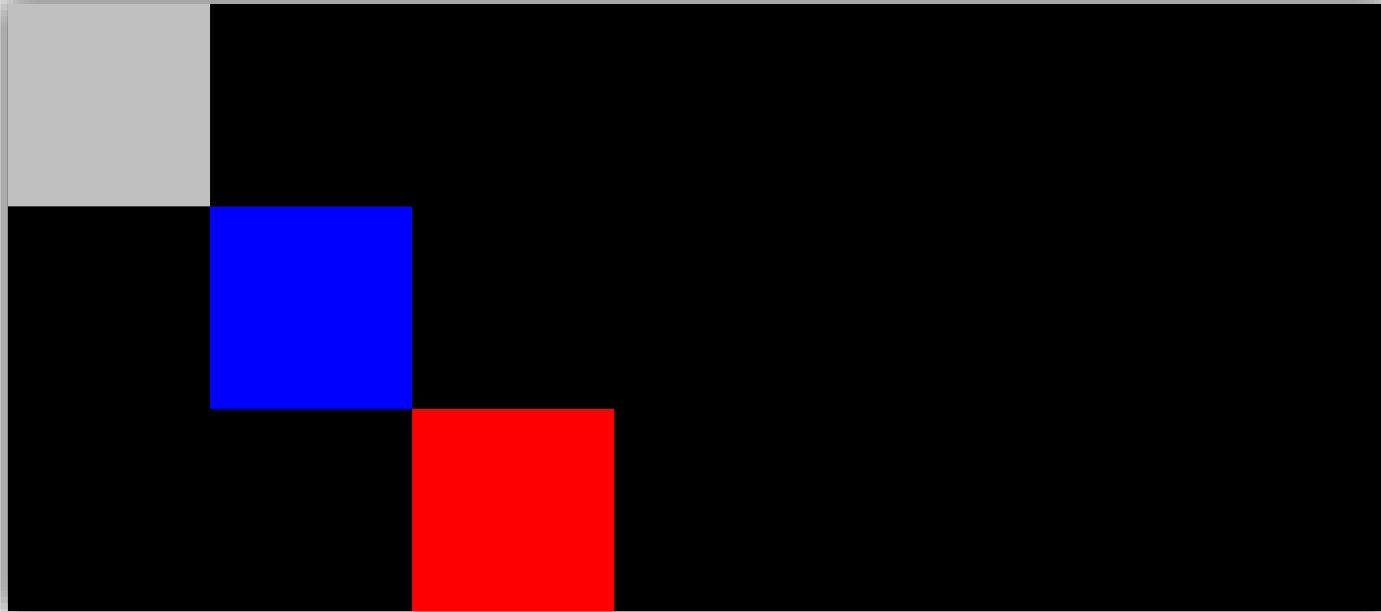




# POSITION : RELATIVE

Los elementos se colocan en función del flujo del HTML **en ese momento**

## EJEMPLO

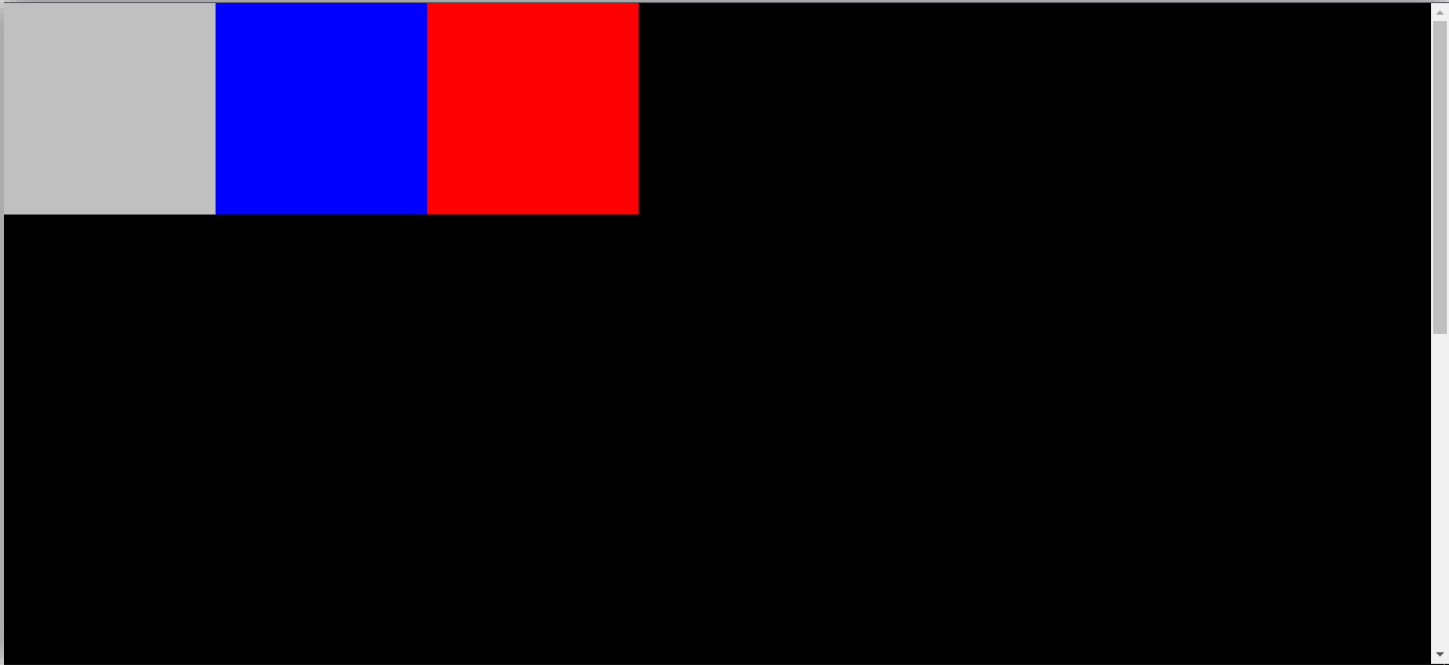




## POSITION : FIXED

Los elementos se mantendrán **siempre fijos** en la posición establecida

### EJEMPLO





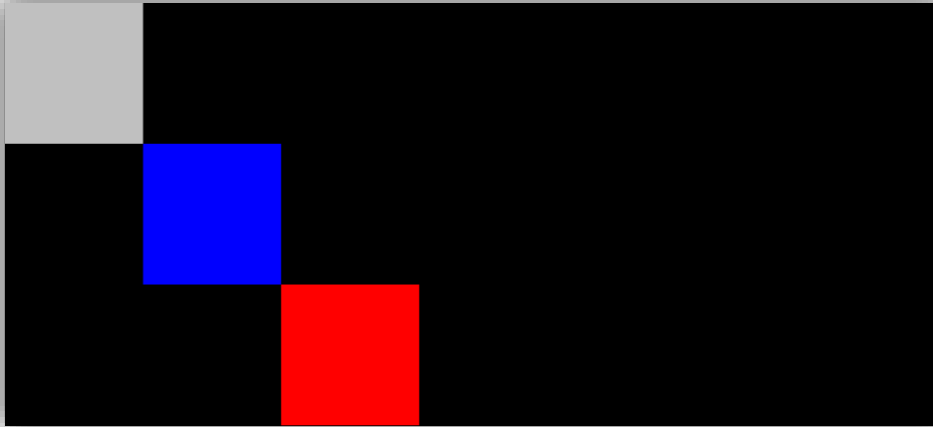


## POSITION : STICKY

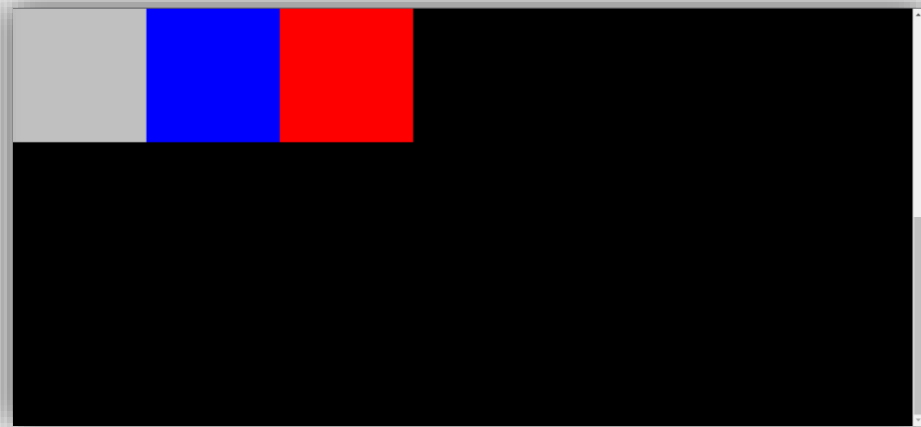
Los elementos se comportarán como el posicionamiento relativo hasta el momento en que se haga scroll y se quedará como posicionamiento fijo

Para que se aplique el elemento debe tener una de las propiedades: Top | Bottom | Left | Right

### EJEMPLO



*Inicio*



*Después del scroll*

# **SUPERPOSICIÓN DE CAJAS**



# ATRIBUTO : z-index

Nos permite indicar la **profundidad** del elemento

**Valores posibles:** 0 (nivel más bajo) en adelante (se permiten negativos pero no es lo común)

Valor más alto = elemento por encima

Si el elemento tiene una **position : static no funcionará** (podrías usar valores negativos como una alternativa)

## EJEMPLO



# FLOTADOS Y POSICIONES



# FLOTAR Y POSICIONAR

## Atributo: float / clear

**Float:** nos permite mover un elemento a la izquierda o derecha teniendo el contenedor padre

Los elementos flotados no siguen el flujo normal del HTML. Esto provoca que las cajas no flotadas se comporten como si las flotantes no estuviesen ahí.

**Valores:** none: valor por defecto, provoca que el elemento no sea flotante

left: flotar elemento a la izquierda

right: flotar elemento a la derecha

**Clear:** limpia el área adyacente al elemento flotante para que los siguientes elementos a él comiencen en su posición normal.

**Valores:** left: el elemento es empujado por debajo de los elementos flotados a la izquierda

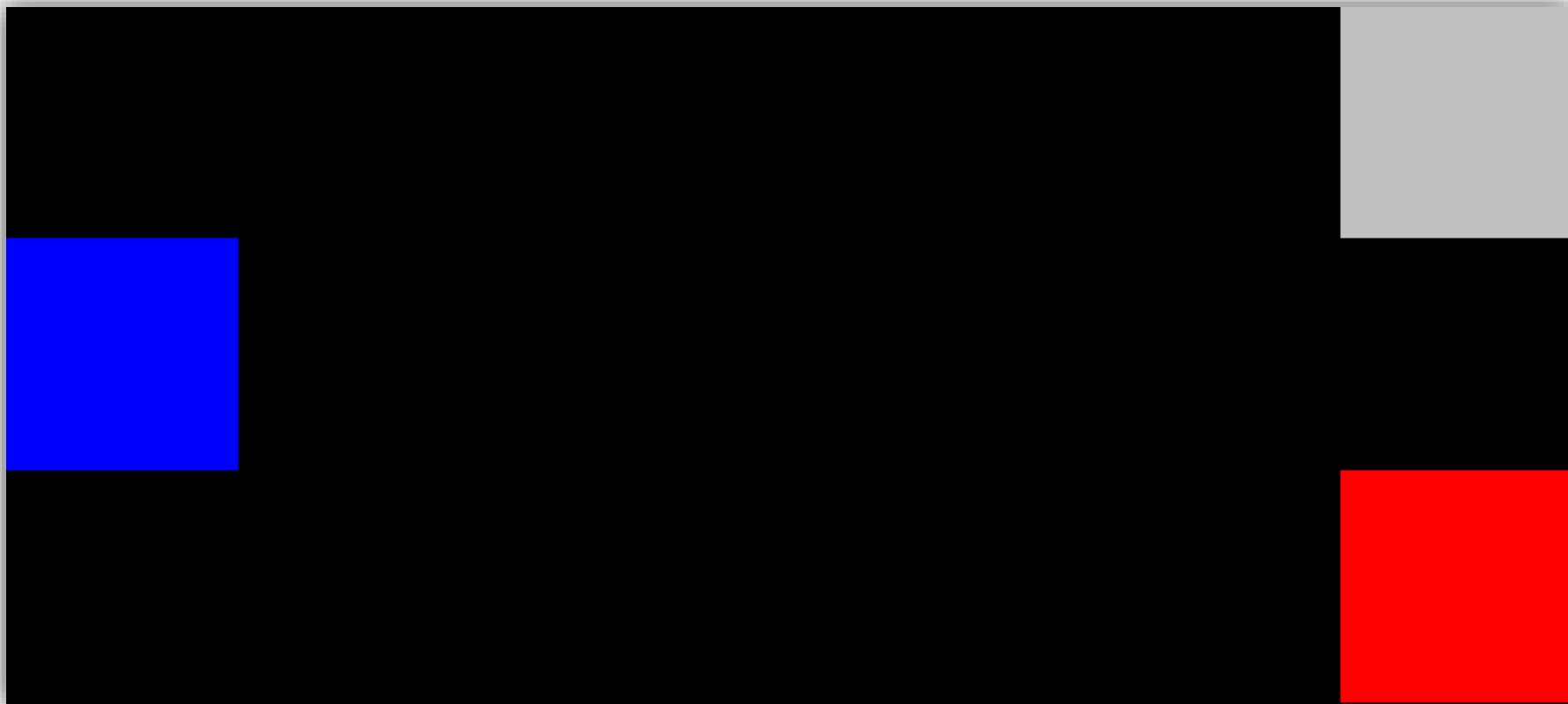
right: el elemento es empujado por debajo de los elementos flotados a la derecha

both: el elemento es empujado por debajo de los elementos flotados a la derecha e izquierda

none: valor por defecto, el elemento no es empujado por debajo de ningún elemento flotado



## EJEMPLOS



# **CENTRAR ELEMENTOS**

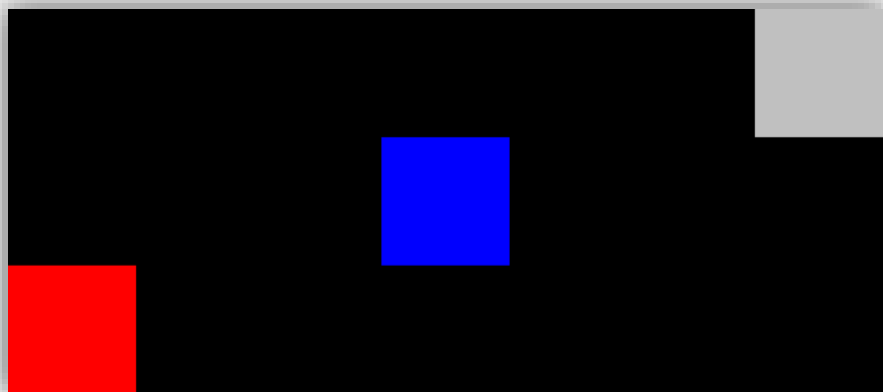


# ALINEAR CONTENEDORES

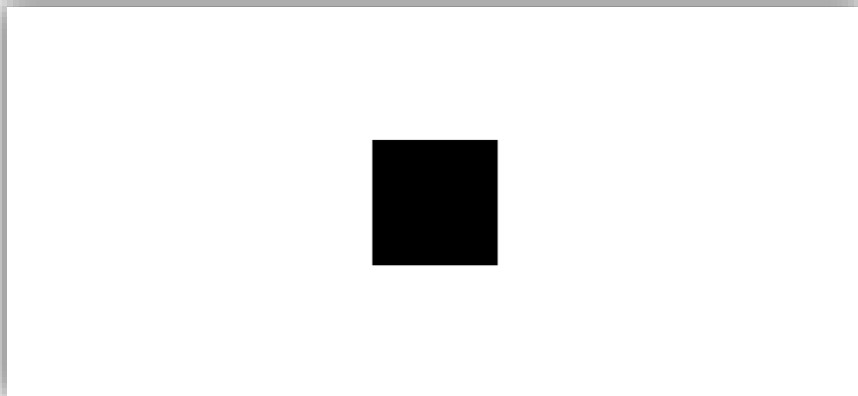
**Alineación horizontal:** usaremos margin left y right con el valor auto

**Alineación vertical:** casi nunca se busca la posición vertical por su complejidad, la forma de suplirlo es jugando con los margin de los diferentes divs. En caso de querer alinear verticalmente se puede usar la posición absoluta

## EJEMPLOS



*Horizontal*



*Vertical*



# CONTENIDO EXTRA



# RESPONSIVE

## ¿Qué es un diseño responsive?

Adaptar la apariencia de las páginas web al dispositivo que se esté utilizando para visitarlas

## ¿Cómo se codifica?

Utilizando **porcentajes**

A través de las **media queries** (módulos CSS) -> **@media ("propiedad"){ }**

(Se pueden incluir varias propiedades)

# MEJORANDO EL RESPONSIVE

## Viewport

A través de esta etiqueta **meta**, comunicamos al navegador que el responsive codificado es **también viable para móvil**. Si no lo incluimos, las media queries no se procesarán correctamente



**GRACIAS POR  
SU ATENCIÓN**