



# MODELO DE CAJAS

---

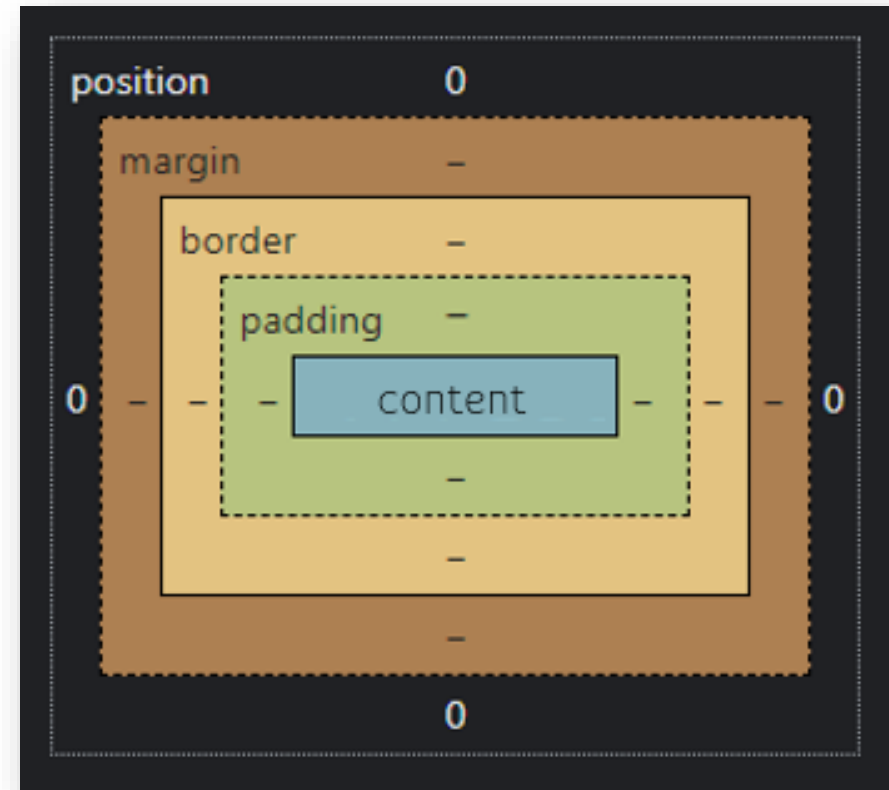
NO HAY NADA QUE LEER AQUÍ,  
PRÉSTAME ATENCIÓN COMPI

# ¿QUÉ ES EL MODELO DE CAJAS?

---

Es el concepto de que cada elemento de la página es una caja rectangular con 5 propiedades:

- ❖ Height = Alto
- ❖ Width = Ancho
- ❖ Border = Borde
- ❖ Padding = Relleno
- ❖ Margen = Margin



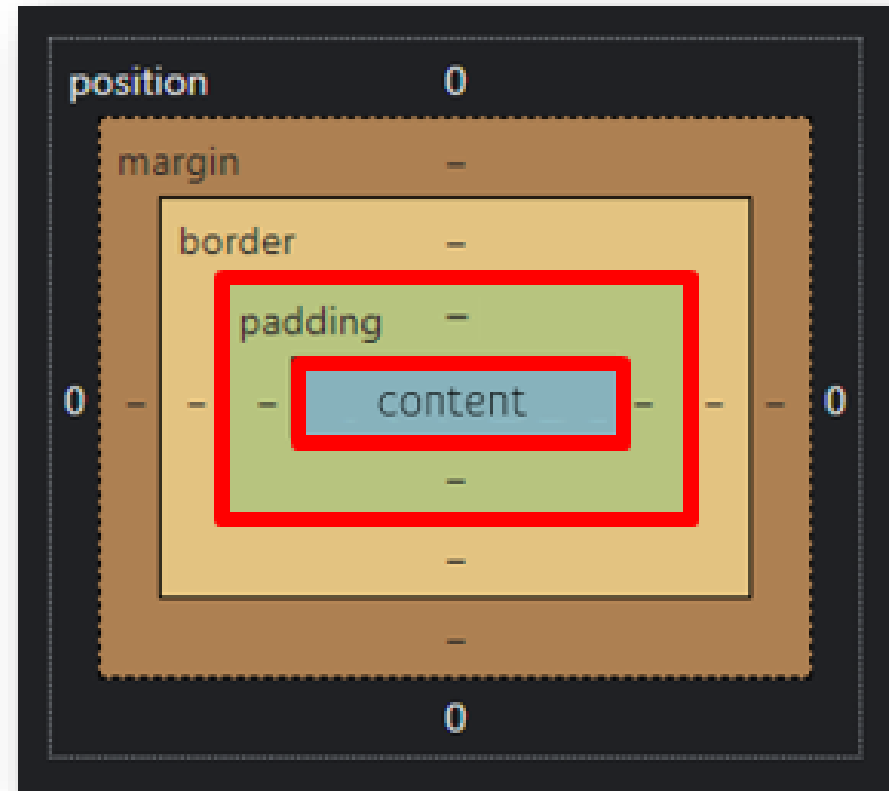
PADDING (RELLENO)

# PADDING (RELLENO)

---

Es el espacio entre el borde y el contenido.

Se suma a los atributos width y height. Si tengo un div de 50px de ancho y alto y le pongo un padding de 10px, pasará a tener un ancho y alto de 60px.



# SINTAXIS DE PADDING

---

❖ Se declara con “padding:”

❖ Puede tener...

- **1 valor** (el padding es el mismo por todos los lados):
  - padding : 10px;
- **2 valores** (el primer valor es para arriba y abajo y el segundo para los lados):
  - padding : 10px 5px;
- **4 valores** (cada valor para cada lado):
  - padding : 10px 5px 7px 4px;

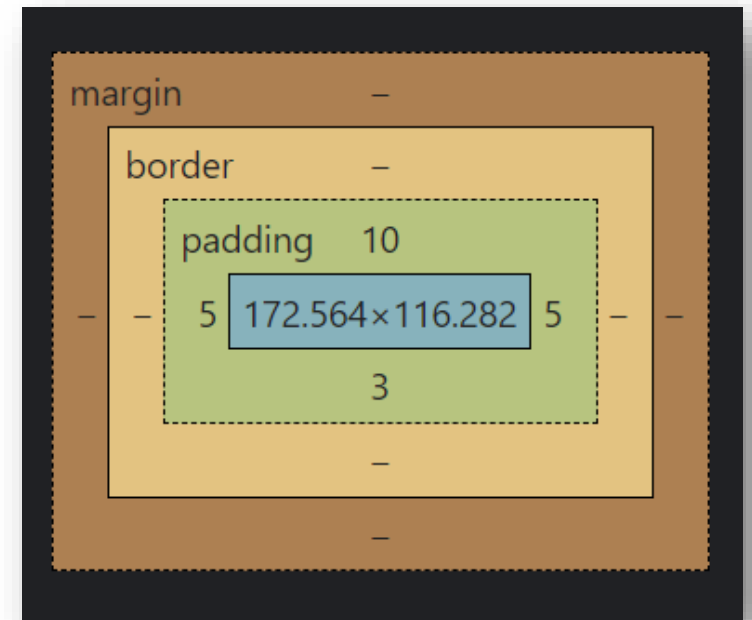
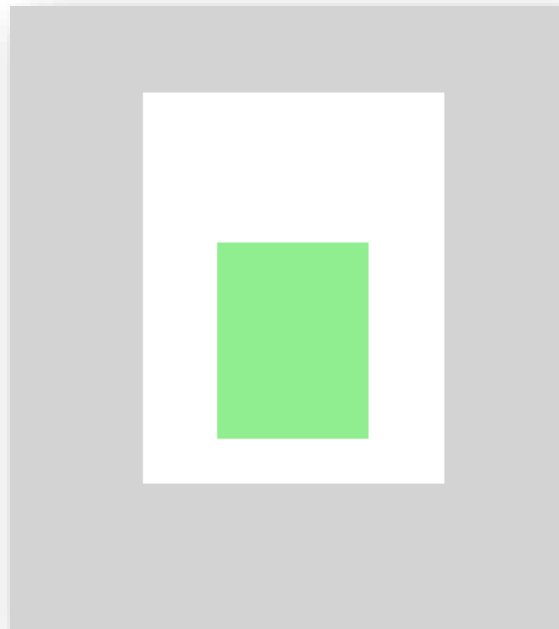
Peeeeeeero también se puede declarar con:

- padding-bottom
- padding-right
- padding-left
- padding-top

# ¿PERO PROFE Y SI LE PONGO SOLO 3 VALORES QUE PASA? PROFEEEEEEEE

---

```
div{  
  background-color: white;  
  width: 50%; height: 50%;  
  padding: 10px 5px 3px;  
}
```



MARGIN (MARGEN EXT.)

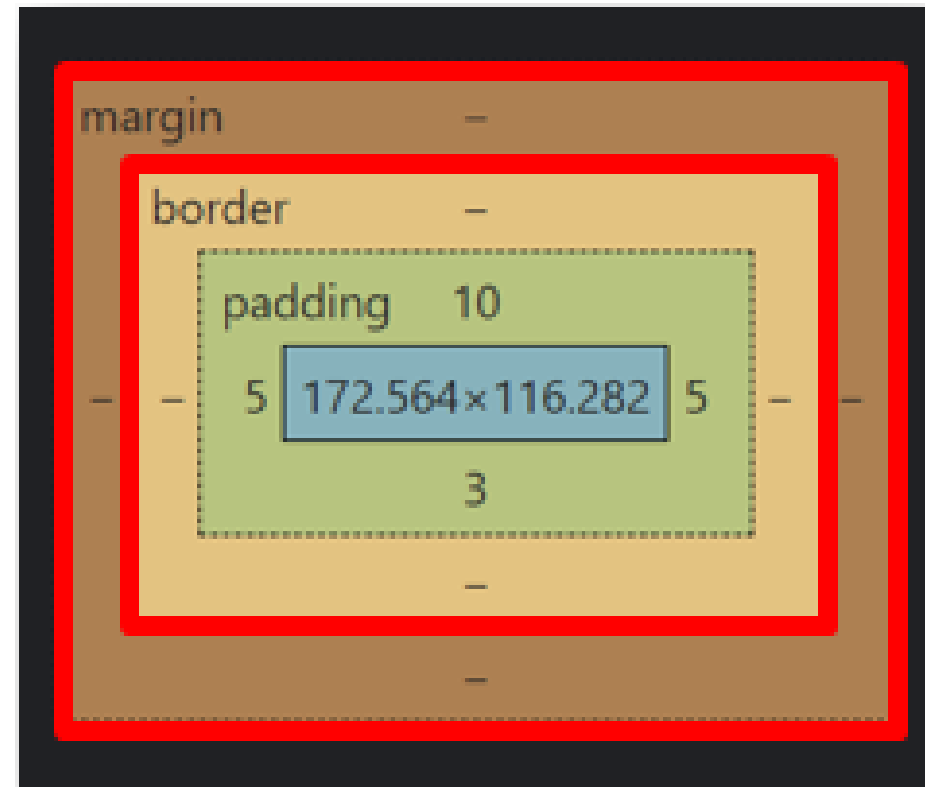
# MARGIN (MARGEN EXT)

---

Es el espacio entre el **borde** y cualquier element exterior.

Al igual que el padding, se **suma a los atributos width y height**. Si tengo un div de 50px de ancho y alto y le pongo un margin de 10px, pasará a tener un ancho y alto de 60px.

;





# SINTAXIS DE MARGIN

---

❖ Se declara con “margin:”

❖ Puede tener...

- **1 valor** (el margin es el mismo por todos los lados):
  - margin : 10px;
- **2 valores** (el primer valor es para arriba y abajo y el segundo para los lados):
  - margin : 10px 5px;
- **4 valores** (cada valor para cada lado):
  - margin : 10px 5px 7px 4px;

Peeeeeeeero también se puede declarar con:

- margin-bottom
- margin-right
- margin-left
- margin-top

# ¿CUÁL ES EL TRUCO DEL AMENDRUCO?

---

Margin es la única propiedad del modelo de cajas que admite valores negativos. Se puede poner `margin: -5px;`

	DISPLAY:BLOCK	DISPLAY:INLINE	DISPLAY: INLINE-BLOCK
Margin	Todos los lados	Solo por los lados	Solo arriba y abajo

BORDER (BORDE)

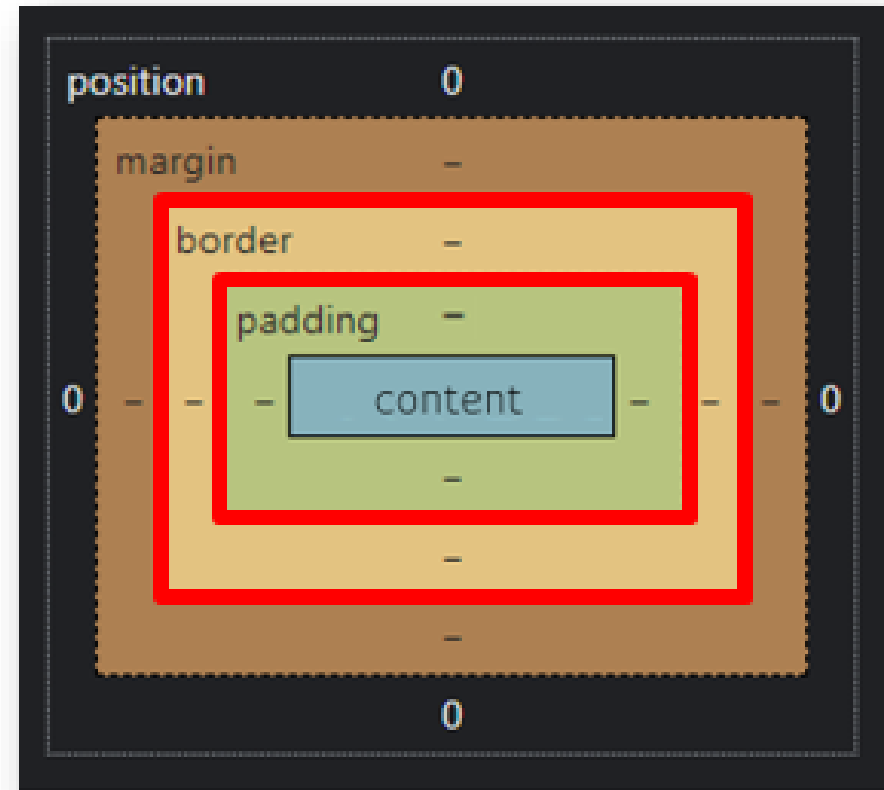
# BORDER

---

Es el espacio entre el **margin** y el **padding**.

Al igual que el padding y margin, se suma a los atributos **width** y **height**. Si tengo un div de 50px de ancho y alto y le pongo un borde de 10px, pasará a tener un ancho y alto de 60px.

;



# SINTAXIS DE BORDER

---

❖ El borde tiene 3 atributos:

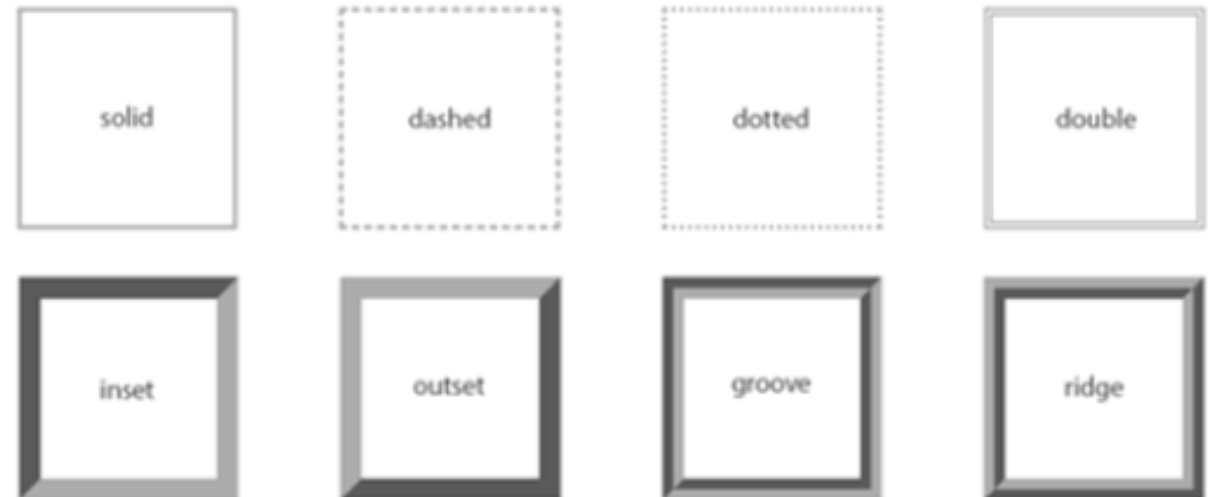
- Grosor (border-width)
- Color (border-color)
- Estilo (border-style)

❖ Cada una se puede declarar con hasta 4 valores, como el padding y margin.

- EJEMPLO: border-color: red blue;
- EJEMPLO: border-style: inset outset;
- EJEMPLO: border-width: 10px 5px;

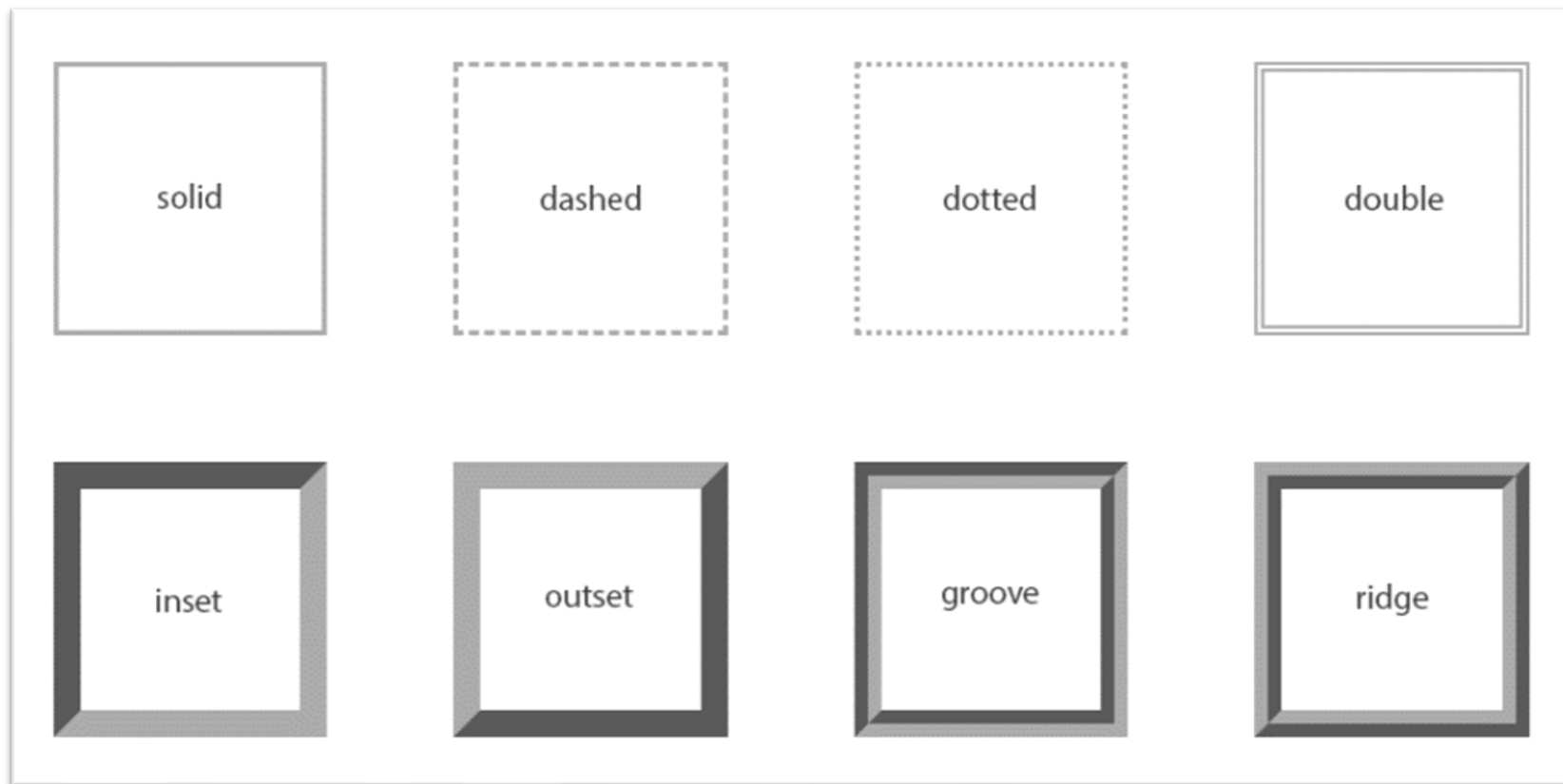
❖ Se declara con "border:"

- Se debe indicar anchura, estilo y color en cualquier orden separado por espacios:
  - EJEMPLO: border: solid black 10px;



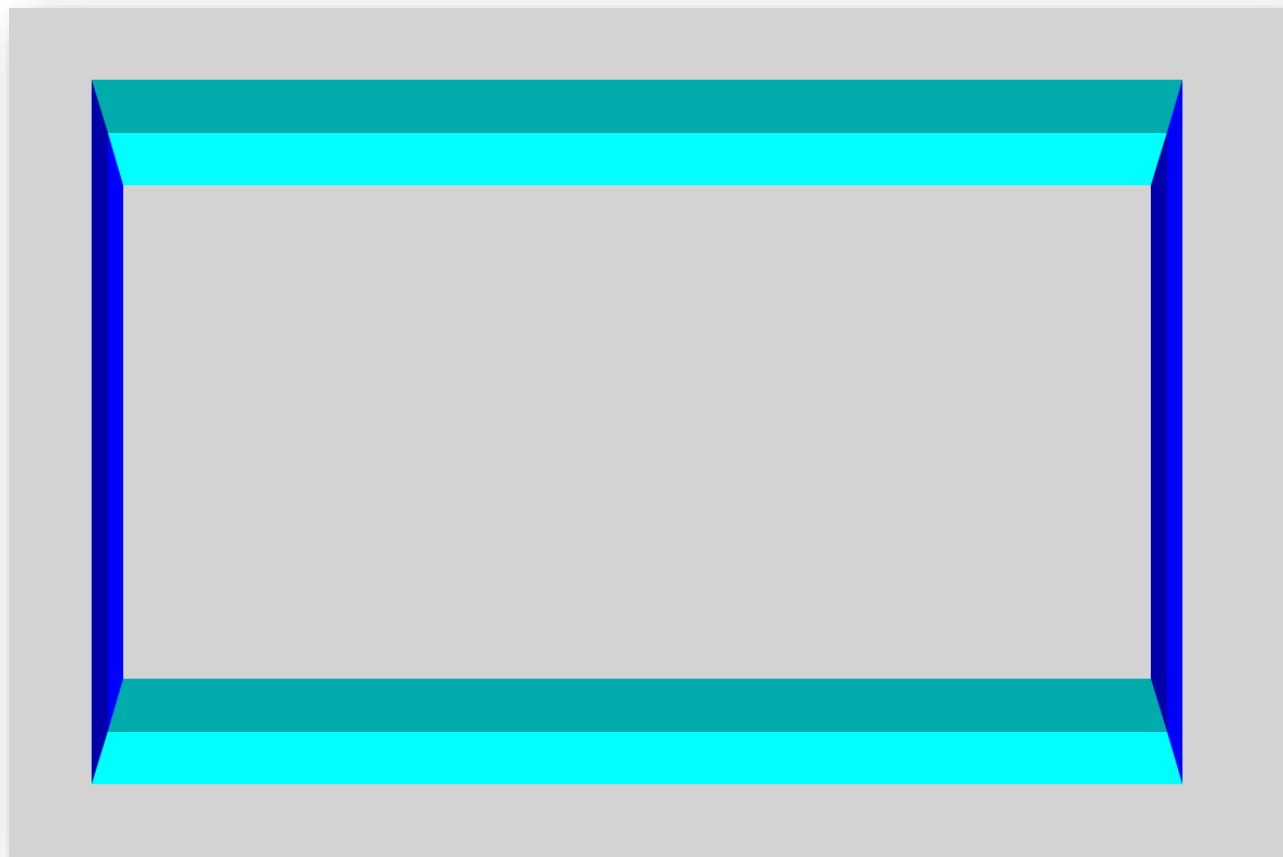
# TIPOS DE BORDES

---



```
div{  
  width: 50%; height: 50%;  
  border-width: 50px 15px;  
  border-style: groove;  
  border-color: ■aqua ■blue;  
}
```

```
div {  
  width: 50%; height: 50%;  
  border-width: 50px 15px;  
  border-style: groove;  
  border-color: aqua blue;  
}
```



# BOX-SHADOW

---

## ❖ Box-shadow:

Se declara con  
"Box-shadow:" y se  
acompaña de los siguientes  
valores sin comas:

### box-shadow:

longitud-x  
Longitud-y  
difuminado (opcional)  
color (opcional)  
inset / outset / inherit

```
5px 5px 10px rgba(0,0,0,0.3)
```

```
0 1px 3px rgba(0,0,0,0.8)
```

```
0 0 15px hsla(100,50%,50%,1)
```

```
5px 5px 0 rgba(0,0,0,0.2)
```

```
inset 5px 3px 4px rgba(0,0,0,0.4)
```

```
inset 0 0 20px #fff
```

```
inset 1px 1px 150px #000
```

```
inset 0 0 0 10px #10b7ec,  
inset 0 0 0 14px #fff
```

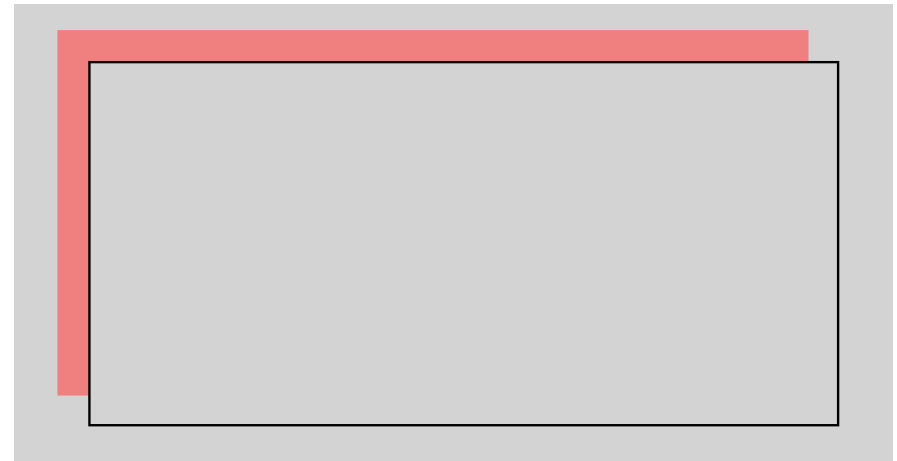
```
5px 5px 0 #fff,  
10px 10px 0 rgba(0,0,0,0.2)
```



# PROFEEEEEE, ¿SE PUEDE PONER LA SOMBRA PARA EL OTRO LADO?

---

```
div{  
  width: 50%; height: 50%;  
  border:solid □black 2px;  
  box-shadow: -20px -20px ■lightcoral;  
}
```



# BORDER IMAGE

---

## ❖ Border-image:

Se declara con “**Border-image:**” y se acompaña de los siguientes valores sin comas:

**Border-image :**

“url”

slice (valor numérico)

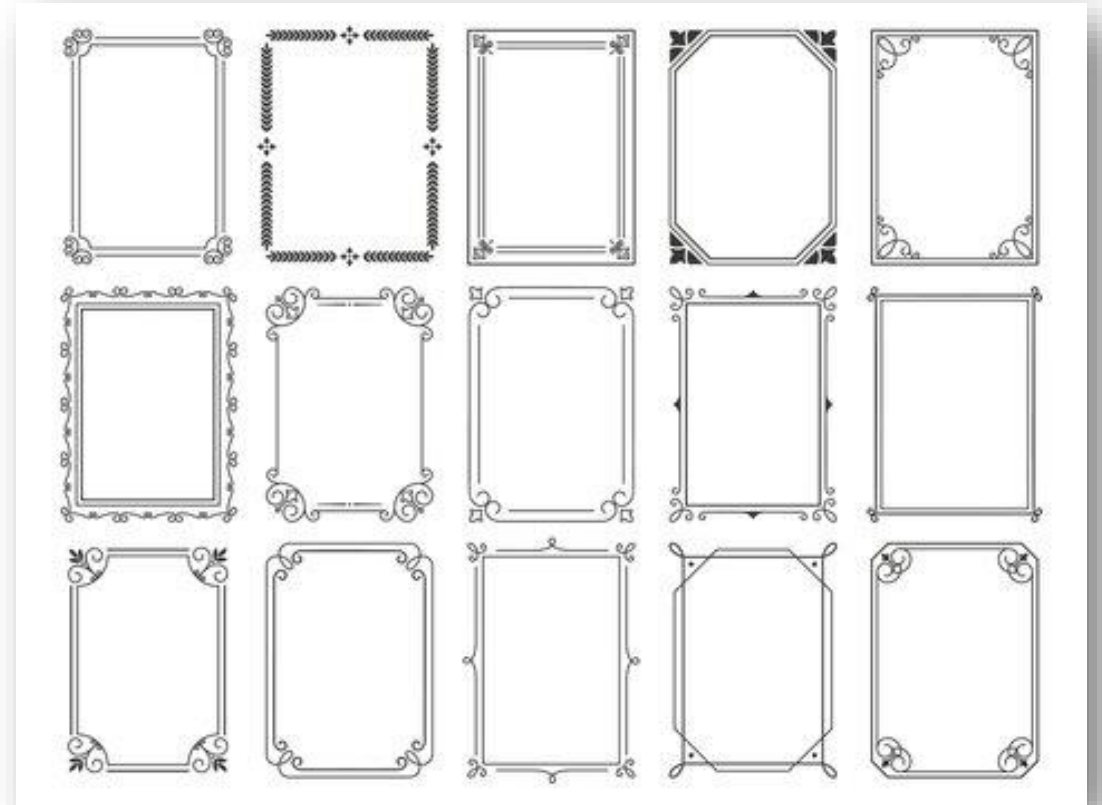
anchura (valor numérico)

cuánto sobresale (valor numérico)

Repeat / rounded / stretch

(Repetir la imagen / redondearla / estirla)

**EJEMPLO:** `border-image: url(border.png) 30% round;`



# BORDER-RADIUS

---

## ❖ Border-radius:

Se declara con

“Border-radius :” y tiene la misma sintaxis que el padding y margin

`border-radius-top-left`



`border-radius: 5px`



`border-radius: 1%`



`border-radius: 20px 0`



`border-radius-top-right`



`border-radius: 25px`



`border-radius: 10%`



`border-radius: 40px 0`



`border-radius-bottom-right`



`border-radius: 35px`



`border-radius: 25%`



`border-radius: 80px 0`



`border-radius-bottom-left`



`border-radius: 50px`



`border-radius: 50%`



`border-radius: 100px 0`



# BOX-SIZING: BORDER-BOX

---

Box-sizing: border-box es una propiedad común en el modelo de cajas (de todos todos los elementos) que hace que los atributos padding, margin, border no se sumen al width y height.

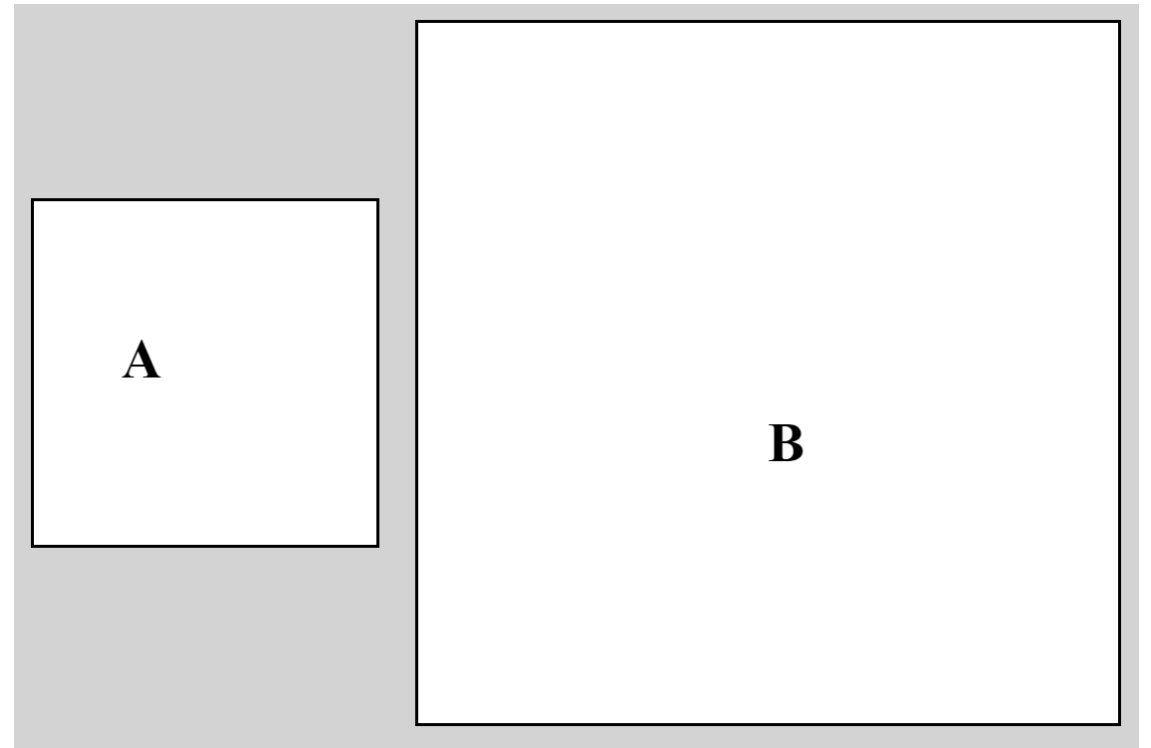
## EJEMPLO:

Si tengo un <div> de "width: 200px" con "padding:20px" y "margin:5px" sin "box-sizing: border-box" tendría un width de 40+10+200, pero con "box-sizing: border-box" el width se ajusta a 200px, no se suma.

# ¿PERO TÍCHER, SI LE PONGO UN “WIDTH:200PX” Y UN “PADDING:200PX”?

---

```
div.a{  
  width: 200px; height: 200px;  
  border:solid □black 2px;  
  box-sizing: border-box;  
  padding:50px; margin: 10px;  
}  
  
div.b{  
  width: 200px; height: 200px;  
  border:solid □black 2px;  
  box-sizing: border-box;  
  padding:200px; margin: 10px;  
}
```



WIDTH Y HEIGHT

# ANCHO Y ALTO (WIDTH Y HEIGHT)

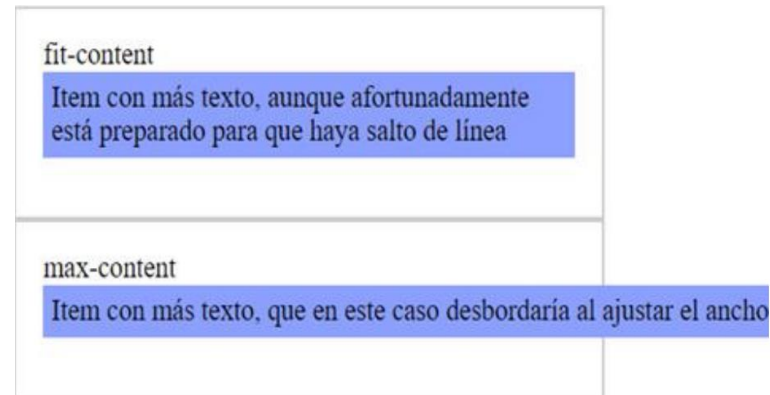
---

Solo se pueden modificar el ancho y el alto de **cualquier elemento** que tenga **posición absoluta** (`position:absolute`),

`width/height`:

- `fit-content`
  - Se adapta al contenido.
- `Min-content`
  - Ajusta al mínimo posible.
- `Max-content`
  - Ajusta a la altura del contenido, aunque desborde.

- `Min/Max-height` y `Min/Max-width`
  - Sirven para delimitar el tamaño máximo y mínimo que puede tener un elemento. EJEMPLO: `min-width: 500px`;



`min-content`

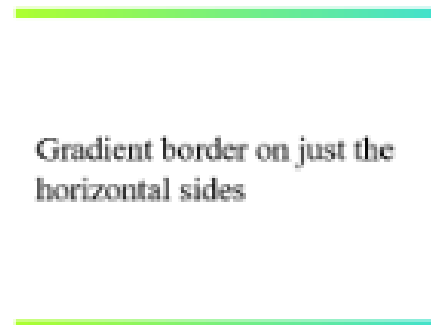
Item con más texto, aunque en este caso se amplía el alto al ajustar el ancho al mínimo contenido

# EJERCICIOS

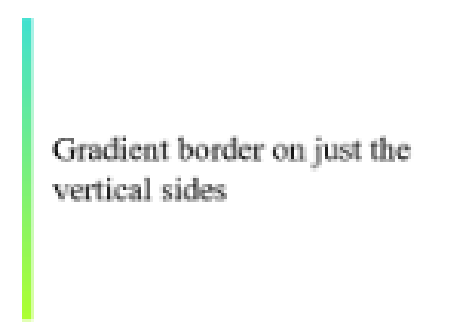




Full gradient border



Gradient border on just the horizontal sides



Gradient border on just the vertical sides

