

DISPLAY

Esta propiedad le indica al navegador como presentar la caja dentro del flujo.

Tiene una gran variedad de valores, pero nosotros solo aplicaremos:

- None
- Block
- Inline
- Inline-block

NONE

Con este valor, la caja desaparece por completo

BLOCK

Ocupa todo el ancho del contenedor padre

INLINE

Ocupa solo el ancho del contenido. No admite width y height

INLINE-BLOCK

Ocupa el ancho del contenido, pero podemos asignar la propiedad width y height para establecer el tamaño de la caja.

PRACTICA

Para esta práctica vamos a crear dos archivos:

Crear una carpeta de nombre display y adentro de ella crear los siguientes archivos:

- Display.html
- Displat.css

Vamos con el marcado.

```
<header>
  <h1 id="uno">Hola</h1>
  <h1 id="dos">Mundo</h1>
  <h1 id="tres">Desde CSS3</h1>
</header>
```

El resultado es



Cada etiqueta `<h1></h1>` ocupa una línea del ancho total de su padre. Esto se debe a que tienen por defecto el atributo **display:block**.

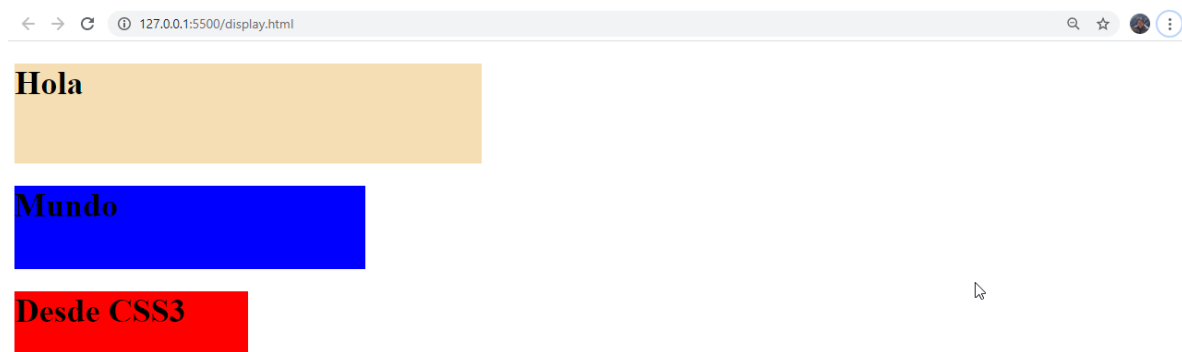
Bien ahora vamos a aplicar algunas reglas a cada una de las cajas.

```
#uno {
  width: 40%;
  height: 6rem;
  background-color: wheat;
}
```

```
#dos {
  width: 30%;
  height: 5rem;
  background-color: blue;
}
```

```
#tres {
  width: 20%;
  height: 4rem;
  background-color: red;
}
```

El resultado es el siguiente



Como observa, cada una ocupa una línea aunque hay espacio a la derecha de cada caja, por definición la caja siguiente no puede ocupar el espacio restante, debe desplegarse en la línea siguiente. Todo esto por definición del flujo de presentación.

Ahora vamos a cambiar el atributo `display:block` a `display:inline`

NOTA

El atributo **`display:block`** por defecto, así que ponerlo o no el resultado es el mismo. Pero caba aclarar que hay etiquetas HTML que su valor por defecto es **`display:inline`**.

Cambiemos el primer titular

```
#uno {
  width: 40%;
  height: 6rem;
  background-color: wheat;
  display: inline;
}
```

El resultado es



El ancho como el alto de la caja ya no es el mismo, esto se debe a que la dimensión de la caja se ajusta al tamaño del contenido y omite las propiedades `width` y `height`.

Ahora aplicamos al segundo titular

```
#dos {
  width: 30%;
  height: 5rem;
  background-color: blue;
  display: inline;
}
```

El resultado es



Ahora ya dos titulares comparten una sola línea y se ajustan a su contenido.

Finalmente aplicamos lo mismo para el tercer titular y queda

```
#tres {
  width: 20%;
  height: 4rem;
  background-color: red;
  display: inline;
}
```

El resultado es



Con esto hemos visto lo que hace los valores **block** e **inline** del atributo **display**.

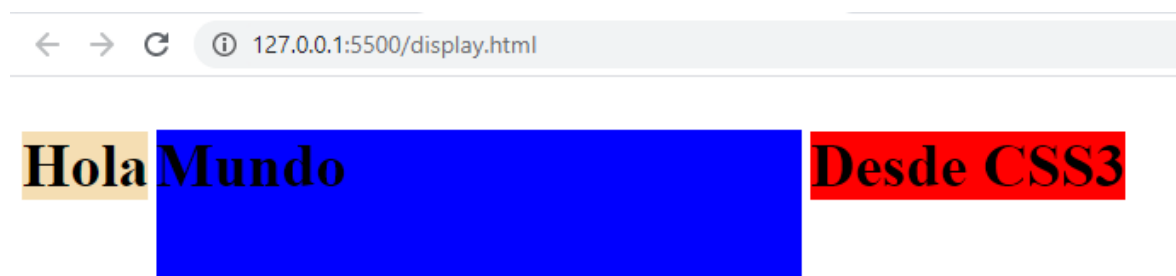
Ahora bien, entonces para que nos sirve ¿ **display: inline-block** ?

Comentamos que width y height no funciona con display:inline. Pero si queremos darle una dimensión pero al mismo tiempo tenerlos en la misma línea, para eso es esta combinación.

Vamos a aplicar inline-block al segundo titular

```
#dos {
  width: 30%;
  height: 5rem;
  background-color: blue;
  display: inline-block;
}
```

Y el resultado es



Ahora si podemos ajustar la dimensión de la caja a nuestra necesidad compartiendo una sola línea.

