

# UD3: Maquetación web con CSS

Lenguajes de  
marcas y sistemas  
de gestión de  
información

# Posición de un elemento

- Los elementos se posicionan en CSS siguiendo el modelo de caja.
- Además de modificar el tipo de *display* de un elemento, los márgenes y el relleno, existen otras propiedades CSS que nos permiten posicionarlo fuera de su posición habitual.
- Una de estas propiedades para posicionar un elemento es *position*, cuya modificación nos permite posicionar relativamente un elemento.
- Position tiene 5 posibles valores: *static*, *relative*, *absolute*, *fixed* y *sticky*.

# position:static;

- El valor `static` es el que tienen por defecto todos los elementos del documento HTML.
- Indica que la posición se determina por la estricta aplicación del modelo de caja.
- Por tanto, aplicar la propiedad `position:static;` no aplicará ningún cambio sobre el posicionamiento de un elemento, salvo en el caso de que previamente le hubiéramos cambiado la propiedad `position` a otro valor.
- No le afectan las propiedades `top`, `bottom`, `left` o `right`.

# position:static;

```
<!DOCTYPE html>

<html>
  <head>
    <style>
      div {
        position: static;
        border: 1px solid blue;
        width: 400px;
        height: 200px;
        background-color: cyan;
      }
    </style>
  </head>
  <body>
    <p>El siguiente <div> tiene
    posicionamiento <em>static</em></p>
    <div>Este elemento div tiene posicionamiento
    <strong>static</strong>, que es el que tienen
    por defecto todos los elementos HTML</div>
  </body>
</html>
```

El siguiente <div> tiene posicionamiento *static*

Este elemento div tiene posicionamiento **static**, que es el que tienen por defecto todos los elementos HTML

# position:relative;

- Un elemento con posicionamiento `relative` se posiciona con respecto a su posición por defecto (`static`).
- Queda el “hueco” dejado por su posición por defecto.
- Podemos usar las propiedades `top`, `bottom`, `left` y `right` para alejarlo de su posición por defecto.

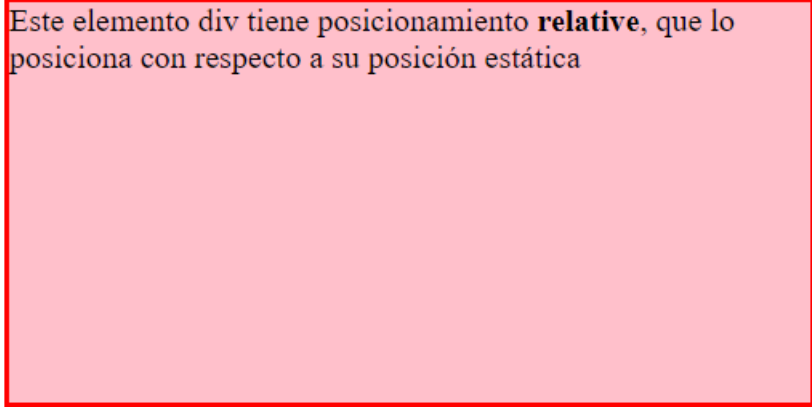
# position:relative;

```
<html>
  <head>
    <style>
      div{
        position: relative;
        left: 100px;
        top: 250px;
        width: 400px;
        height: 200px;
        border: 3px solid red;
        background-color: pink;
      }
    </style>
  </head>
  <body>
    <p>El siguiente <div> tiene
    posicionamiento <em>relative</em></p>
    <div>Este elemento div tiene posicionamiento
    <strong>relative</strong>, que lo posiciona con
    respecto a su posición estática</div>
    <p>Como puede observarse, ha dejado un hueco
    correspondiente a su posicionamiento estático
    </p>
  </body>
</html>
```

El siguiente <div> tiene posicionamiento *relative*

Como puede observarse, ha dejado un hueco correspondiente a su posicionamiento estático

Este elemento div tiene posicionamiento **relative**, que lo posiciona con respecto a su posición estática

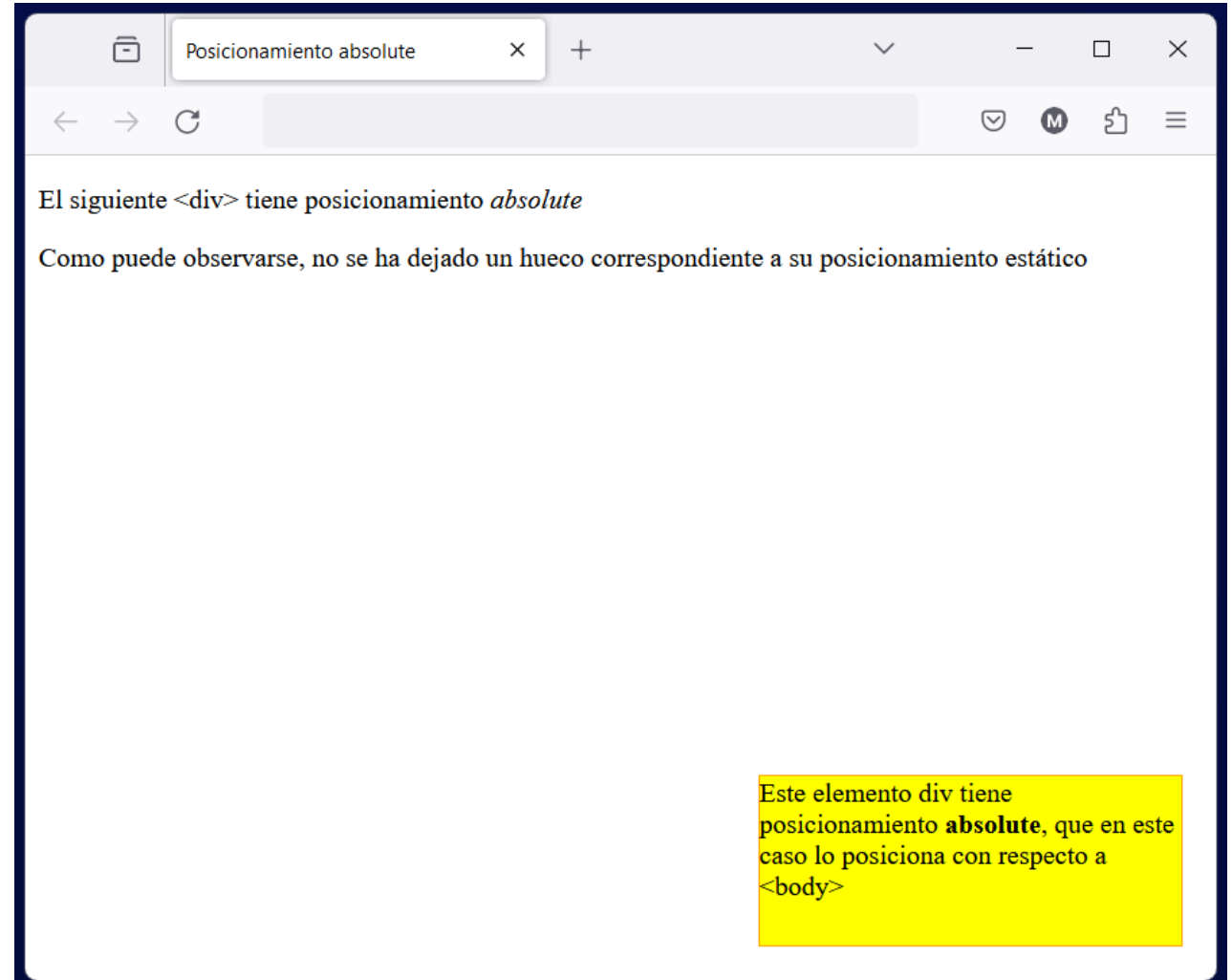


# position:absolute;

- Un elemento con posicionamiento `absolute` se posiciona con respecto a su elemento “antepasado” con posicionamiento diferente de `static`. Si no existe un elemento antepasado con estas características, se posiciona con respecto al elemento `<body>`.
- No deja ningún “hueco” en su posición por defecto.
- Podemos usar las propiedades `top`, `bottom`, `left` y `right` para alejarlo de su posición por defecto.

# position:absolute;

```
<html>
<head>
  <style>
    div{
      position: absolute;
      bottom: 20px;
      right: 20px;
      width: 250px;
      height: 100px;
      border: 1px solid orange;
      background-color: yellow;
    }
  </style>
</head>
<body>
  <p>El siguiente <div> tiene
  posicionamiento <em>absolute</em></p>
  <div>Este elemento div tiene posicionamiento
  <strong>absolute</strong>, que en este caso lo
  posiciona con respecto a <body></div>
  <p>Como puede observarse, no se ha dejado un
  hueco correspondiente a su posicionamiento
  estático</p>
</body>
</html>
```



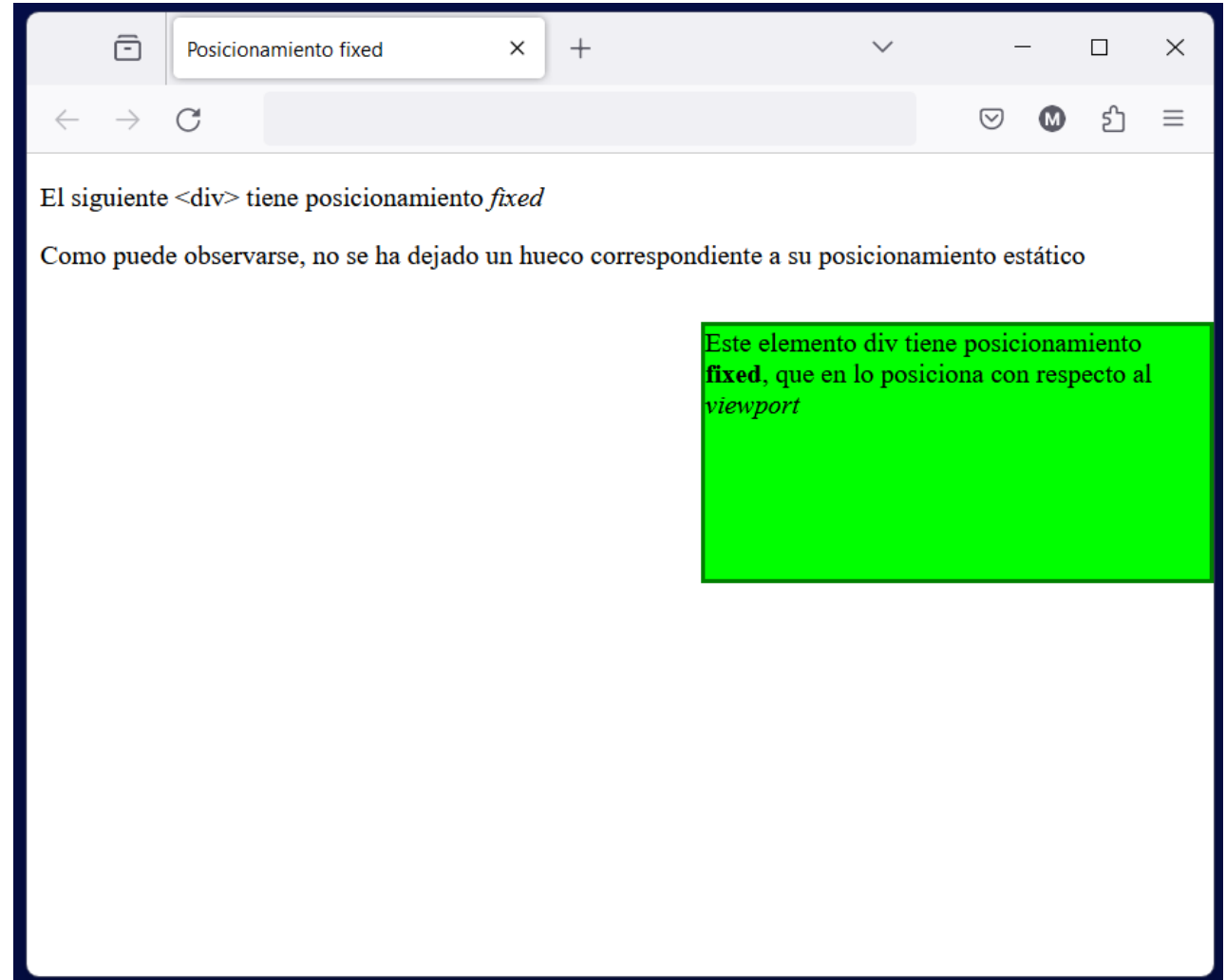


# position:fixed;

- Un elemento con posicionamiento `fixed` se posiciona con respecto al *viewport*, es decir, el área de visión que el navegador muestra al usuario. Como está vinculado al *viewport*, no le afecta el *scroll* de la página, quedando siempre “fijado” en esa posición.
- No deja ningún “hueco” en su posición por defecto.
- Podemos usar las propiedades `top`, `bottom`, `left` y `right` para alejarlo de su posición por defecto.
- Para apreciar su comportamiento frente a un posicionamiento `absolute` que se posicione respecto a `<body>` es necesario hacer *scroll* en la página.

# position:fixed;

```
<html>
<head>
  <style>
    div {
      position: fixed;
      right: 0px;
      top: 100px;
      width: 300px;
      height: 150px;
      border: 3px solid green;
      background-color: lime;
    }
  </style>
</head>
<body>
  <p>El siguiente <div> tiene
  posicionamiento <em>fixed</em></p>
  <div>Este elemento div tiene posicionamiento
  <strong>fixed</strong>, que en lo posiciona con
  respecto al <em>viewport</em></div>
  <p>Como puede observarse, no se ha dejado un
  hueco correspondiente a su posicionamiento
  estático</p>
</body>
</html>
```

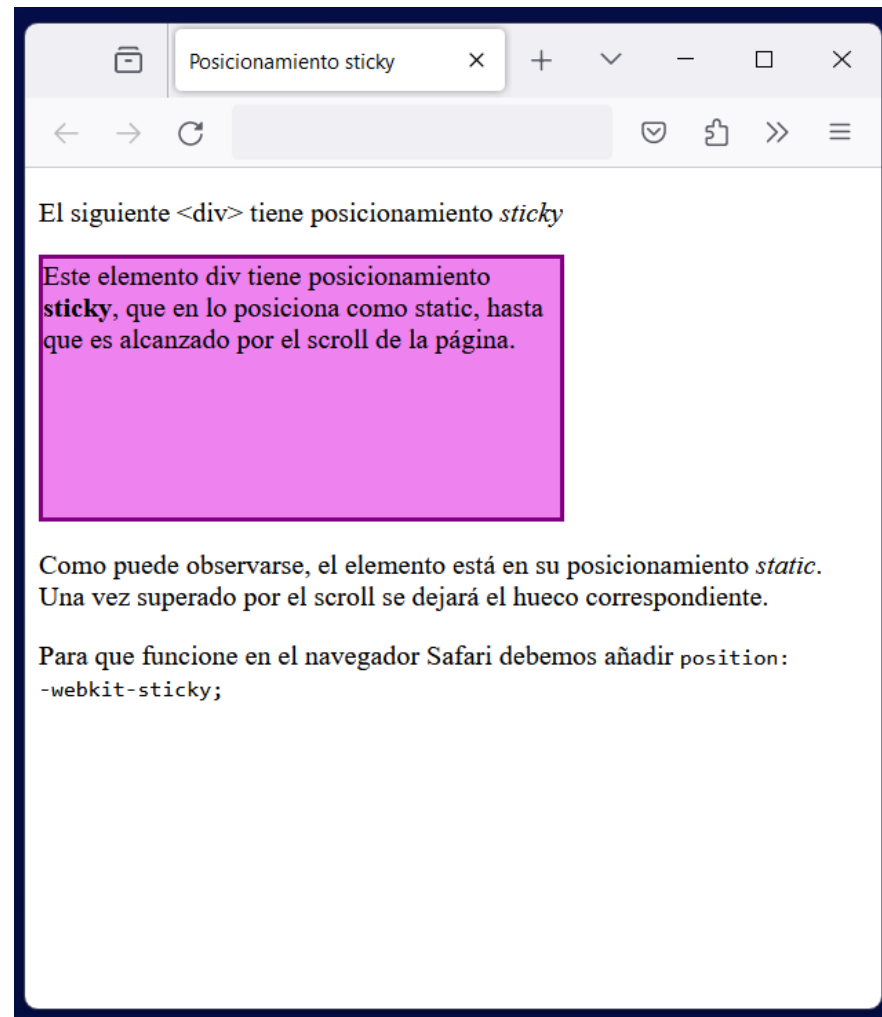


# position:sticky;

- Un elemento con posicionamiento `sticky` se posiciona de forma igual a `static` hasta que el *scroll* de la página llega hasta un punto de desplazamiento (indicado con las propiedades `top`, `bottom`, `left` y `right`) en el cual se queda anclado al `viewport`.
- Ocupa su posición por defecto, y cuando se desplaza deja su “hueco”.
- Podemos usar las propiedades `top`, `bottom`, `left` y `right`.
- Para apreciar su comportamiento frente a un `static` es necesario hacer *scroll* en la página.

# position:sticky;

```
<head>
  <style>
    div{
      position: sticky;
      width: 300px;
      height: 150px;
      top: 0;
      border: 3px solid purple;
      background-color: violet;
    }
  </style>
</head>
<body>
  <p>El siguiente <div> tiene
  posicionamiento <em>sticky</em></p>
  <div>Este elemento div tiene posicionamiento
  <strong>sticky</strong>, que en lo posiciona
  como static, hasta que es alcanzado por el
  scroll de la página.</div>
  <p>Como puede observarse, el elemento está en
  su posicionamiento <em>static</em>. Una vez
  superado por el scroll se dejará el hueco
  correspondiente.</p>
  <p>Para que funcione en el navegador Safari
  debemos añadir <code>position: -webkit-sticky;
  </code></p>
</body>
```



# La propiedad z-index

- Como hemos visto, con elementos con posicionamiento `relative`, `absolute`, `fixed` y `sticky` es posible que los elementos se superpongan en la página.
- Para indicar el orden de superposición de los elementos (cual es más superficial o profundo) usamos la propiedad `z-index`.
- Si asignamos un `z-index` de 1 a un elemento y de 5 a otro, el de valor 5 aparecerá por encima en caso de que ambos coincidan.