

Fons, marges i vores

Background

El **background** o fons d'un element HTML és el que apareix *darrere* del text. Tot i que CSS permet aplicar un fons a qualsevol tipus d'element HTML, s'utilitza principalment en elements a nivell de bloc.

Els fons només s'apliquen a l'element objectiu. Però tenint en compte que **la majoria d'elements HTML tenen un fons transparent**, aplicar un fons al `body` semblarà que s'aplica a tots els elements.

background-color

Valor per defecte: `transparent` .

Heretat pels fills: no.

Com ja hem tractat les diferents maneres de definir un color amb CSS, aplicar un color de fons és senzill:

```
body{ background-color: #ff0000;}
```

Tot l'element s'omplirà amb un color de fons **vermell**. Tingueu en compte que heu de triar sempre un color de text adequat per mantenir el vostre contingut fàcil de llegir.

background-image

Com que els colors llisos no solen ser suficients, CSS permet aplicar **imatges** com a fons per als elements.

L'aplicació d'una imatge de fons només requereix especificar la seva URL:

```
body{ background-image: url(images/diagonal-pattern.png);}
```

El comportament de la imatge (com es repeteix, on es col·loca, com es mida) està definit per altres propietats de `background` que ara veurem.

Tingueu en compte que l'element HTML **no canviarà la mida** per adaptar-se a la imatge, ja que la imatge és purament decorativa.

 o background-image?

L'element HTML `` és per a imatges que formen part del **contingut**, mentre que les imatges de fons CSS són purament **decoratives**.

De vegades es complica diferenciar entre contingut i estil. Algunes tècniques visuals són més fàcils d'aconseguir amb fons CSS. Pregunteu-vos si la imatge que utilitzeu és essencial per a la pàgina. Si és així, utilitzeu l'element ``.

Degradats

CSS també permet definir **degradats de color** com a imatges de fons, en 2 formes diferents:

- `linear-gradient` per a gradients en una sola direcció, en forma rectangular
- `radial-gradient` per a gradients en totes direccions, en forma circular

Els degradats de fons es consideren **imatges de fons**:

```
body{ background-image: linear-gradient(white, grey);}
```

background-position

Per defecte, una imatge de fons es repetirà indefinidament. Podeu especificar la seva **background-position**, escollint un valor `x` horitzontal i un valor `y` vertical.

Per a cada valor, podeu utilitzar:

- valors de píxels `px`
- percentatges, relatius a les dimensions de l'element HTML
- paraules clau com `center`, `left`, `bottom` ...

```
body{ background-position: right bottom;}
```

background-repeat

Per defecte, una imatge de fons es repetirà indefinidament. Podeu triar que es repeteixi només horitzontalment, només verticalment o no.

```
/* Només horitzontalment */  
body{ background-repeat: repeat-x;}  
/* Només verticalment */  
body{ background-repeat: repeat-y;}  
/* La imatge de fons només apareixerà una vegada */  
body{ background-repeat: no-repeat;}
```


Altres propietats de background

Hi ha altres `propietats de background` que podeu fer servir:

- `background-attachment`
- `background-size`

border I

Com que un element HTML es representa com a rectangle, pot tenir fins a **4 vores**: superior, inferior, esquerra i dreta. Podeu establir una vora a tots els costats alhora o a cada costat individualment.

Una vora CSS té 3 propietats:

- `border-color` definit utilitzant una unitat de color.
- `border-style` amb valors com `solid`, `dotted`, `dashed` ... ([tots els valors de border-style a w3schools](#))
- `border-width` definit utilitzant una unitat de mida].

També té 4 possibles costats: `top`, `bottom`, `left` i `right`.

border II

La propietat `border` permet definir les 3 propietats alhora de tota la vora d'un element:

```
div{ border: 1px solid yellow;}
```

És equivalent a:

```
div{  
    border-width: 1px;  
    border-style: solid;  
    border-color: yellow;  
}
```

border-top, border-bottom, border-left, border-right

També existexen versions per quan es vol especificar tan sols, per exemple, la vora inferior:

```
div{ border-bottom: 1px solid yellow;}
```

És equivalent a:

```
div{  
    border-bottom-width: 1px;  
    border-bottom-style: solid;  
    border-bottom-color: yellow;  
}
```

padding

El **marge** és l'espai entre la *vora* d'un element i el seu *contingut*.

```
div { padding: 20px;}
```

La quantitat d'espai es pot definir mitjançant qualsevol de les unitats de mida. Com que es troba **entre** la *borde* i el *contingut*, és més fàcil visualitzar l'espai interior amb una vora aplicada

```
div {  
  padding: 20px;  
  border: 1 px blau sòlid;  
}
```

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

Igual que amb `border`, el `padding` es pot configurar *individualment* per a qualsevol dels 4 costats.

```
p{ padding-bottom: 20px;}
```

També es pot definir els 4 valors de `padding` de manera agrupada:

```
/* top = 20, right = 10, bottom = 5, left = 2 */  
p{ padding: 20px 10px 5px 2px;}
```

margin

Si el `padding` afegeix espai *dins* d'un element (entre la seva vora i el seu contingut), els marges afegeix espai *exterior* entre un element i *altres elements*.

```
div { margin: 20px;}
```

Igual que amb `padding` i `border`, existeixen les propietats `margin-top`, `margin-bottom`, `margin-left`, `margin-right` i també es poden definir els 4 marges amb valors diferents fent servir `margin`:

```
/* top = 20, right = 10, bottom = 5, left = 2 */  
div { margin: 20px 10px 5px 2px;}
```

Combinant els marges

Posem un títol i un subtítol.

```
.title{ margin-bottom: 30px;}  
.subtitle{ margin-top: 15px;}
```

```
<h1 class="title">Això és un títol</h1>  
<h2 class="subtitle">Això és un subtítol</h2>
```

El marge entre els dos elements serà **30px** , i **no** **45px** . Això és perquè els marges que es *toquen* entre si es **fussionen**!

Es pot considerar el marge d'un element com l'**espai mínim** que vol mantenir-se *allunyat* d'altres elements.

Escollint entre margin i padding

Aquesta pregunta sorgeix quan no s'aplica cap vora ni fons. De fet: si s'estableix una vora o un fons en qualsevol dels elements, la representació visual serà diferent. Però si no n'hi ha cap, i tenint en compte que els marges i els farciments són *transparentes*, el resultat tindrà el mateix aspecte.

Pregunteu-vos *per què* esteu espaiant les coses. És per permetre que el contingut interior respire més? O és permetre que tot l'element respire més? És farciment en el primer cas, marge en el segon.

A més, tenint en compte com es poden **fusionar** els marges.