

Teoria Llenguatge de fulls d'estils

CSS (*Cascading Style Sheets*)

Índex

1. Introducció	3
2. CSS Bàsic	3
2.1 Com incorporar CSS a un fitxer HTML	4
2.1.1 Forma 1: mitjançant l'element <code><style></style></code>	4
2.1.2 Forma 2: mitjançant l'element <code><link></link></code>	4
2.2 Sintaxis	4
2.2.1 Comentaris	5
2.3 Selector	5
2.3.1 Selector per element	5
2.3.2 Selector per ID	6
2.3.3 Selector per classe	7
2.3.4 Selector universal	7
2.3.5 Agrupació de selectors	8
3. Colors	8
3.1 Colors del text	8
3.2 Color de fons	8
4. Bordes	11
5. Marges	11
6. Responsive Web Design	13
6.1 Imatges responsive	13
6.2 Text responsive	13
7. Elements flotants	13
8. Estructura d'un projecte real	16
9. Fonts personalitzades	17
10. Filtratge d'imatges	20
Referències	21

1. Introducció

- **Què és CSS?** CSS significa *Cascading Style Sheet* o lo que és lo mateix: full d'estil en cascada. Pensat per escriure la presentació (format o estil), dels documents HTML o XML. CSS descriu com ha de ser renderitzat un element estructurat en una pantalla.
- No és un llenguatge de programació o un llenguatge de marques, és un llenguatge pensat per donar estil a un full, per tant és un llenguatge de fulls d'estil.
- El CSS es pot fer de moltes maneres, també explicat breument en el document de *Llenguatges de marca - HTML*, mitjançant *frameworks*, com és el cas del [bootstrap](#), mitjançant *floats*, que és una nova forma d'empaquetar una pàgina web, o mitjançant [Flexbox](#), un sistema de layout senzill i ajustable per a qualsevol sistema i resolució.

Una vegada afegim un codi CSS en el nostre projecte web, el que estem fent és diversificar de la següent manera:

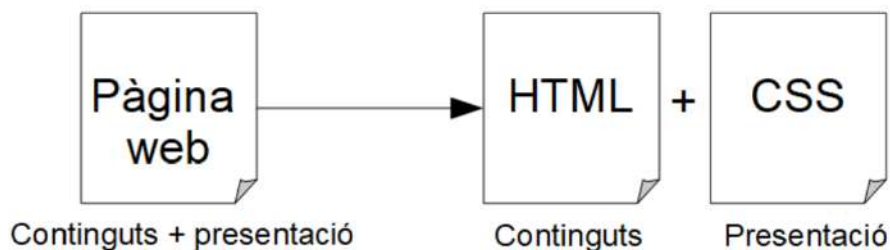


Diagrama 1: combinació d'arxiu comú d'una pàgina web.

2. CSS Bàsic

El CSS és una forma d'assignar estil a un document HTML. El CSS es pot combinar dintre d'un arxiu que conté codi en HTML o directament que sigui un fitxer a part que faci referència a l'arxiu HTML al qual vol donar un estil particular.

CSS va sorgir, ja que HTML mai va pensar en donar-li un estil a la pàgina, HTML es va crear per descriure el contingut d'una pàgina web, com per exemple, títols, paràgrafs, etc. No és fins a la versió 3.2, d'HTML (HTML3.2), on es comencen a incloure etiquetes o elements com el `<font></font>`, per canviar colors o l'etiqueta `<style></style>`.

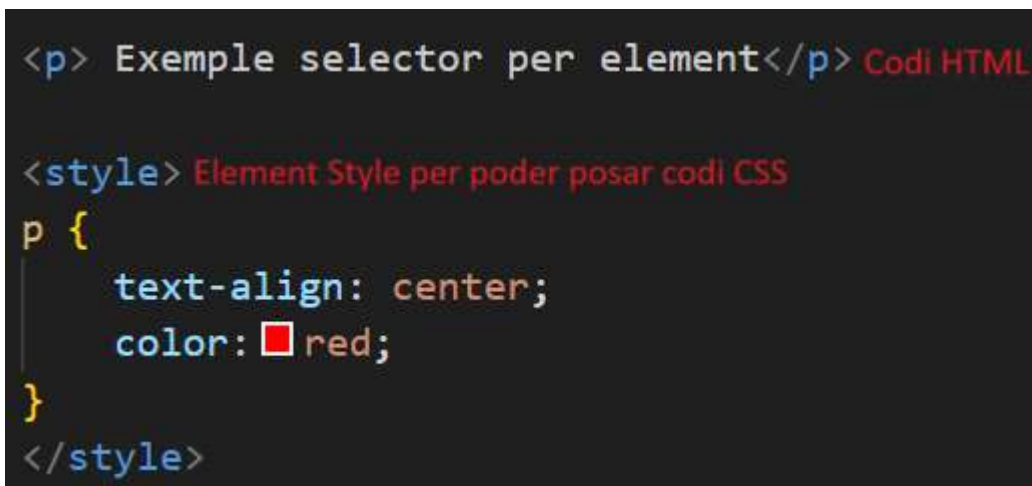
2.1 Com incorporar CSS a un fitxer HTML

Com s'ha comentat abans, existeixen dues formes d'afegir format CSS a un fitxer HTML.

2.1.1 Forma 1: mitjançant l'element `<style></style>`

Podem afegir el codi CSS, dintre d'un element `<style></style>`, però és un mètode del qual tampoc es pot abusar gaire, és a dir, és útil per fer proves o afegir format a una petita part de la pàgina, però no és la forma més òptima per fer-ho.

En aquest cas, tal i com es presenta a la **imatge 1**, es barreja el codi HTML i el codi CSS en un únic document, per motius obvis no és la millor opció, però és útil i es pot utilitzar.



```
<p> Exemple selector per element</p> Codi HTML

<style> Element Style per poder posar codi CSS
p {
  text-align: center;
  color: red;
}
</style>
```

Img 1: Barreja de codi HTML i CSS amb l'element *style*

2.1.2 Forma 2: mitjançant l'element `<link></link>`

Aquesta és la millor manera d'incorporar un codi CSS en un arxiu amb codi HTML. S'utilitza l'element `<link></link>`, amb el qual enllacem ambdós fitxers (l'HTML i el CSS). Fent-ho així aconseguim que s'importi tot el format sense necessitat de barrejar codis.



```
<p> Exemple selector per element</p> Codi HTML

<link rel="stylesheet" href="C:\Users\joelv\Desktop\Exemples CSS-HTML\Ex1_SelectorElement.css">
```

Tipus d'enllac: full d'estil Ruta on es troba el full d'estil

Img 2: Importar arxiu CSS amb l'element *link*.

2.2 Sintaxis

Normalment, qualsevol conjunt de codi CSS, està format per un **selector** i un **bloc de declaració**.

- **Selector:** és el que apunta a l'element HTML que es desitja dissenyar.
- **Bloc de declaració:** conté una o més declaracions separades per punt i coma.
- Cadascuna d'aquestes declaracions inclou un nom de propietat CSS i un valor, separat per dos punts.
- Una declaració CSS sempre termina amb un punt i coma i els blocs de declaració estan limitats per dues claus `{..}`.
- El CSS es programa en general, és a dir, el CSS es programa per blocs HTML, paràgrafs, articles, títols, etc i es dissenya de forma general en el fitxer CSS, llavors qualsevol codi que tingui un paràgraf, un article o un títol tindrà el format especificat pel codi CSS.

2.2.1 Comentaris

Els comentaris en CSS, s'acostumen a fer mitjançant: `/* comentari */` per tant tot el que va dintre d'aquestes línies de codi, no s'executarà, només serà text informatiu pel desenvolupador.

2.3 Selector

Els selectors s'utilitzen, per associar els elements HTML als que volem donar estil. Tal i com s'ha explicat abans, un fitxer CSS dona l'estil al contenidor HTML que selecciona a partir del selector (títols, capçaleres, paràgrafs, etc).



Img 3. Conjunt de regles de sintaxis CSS

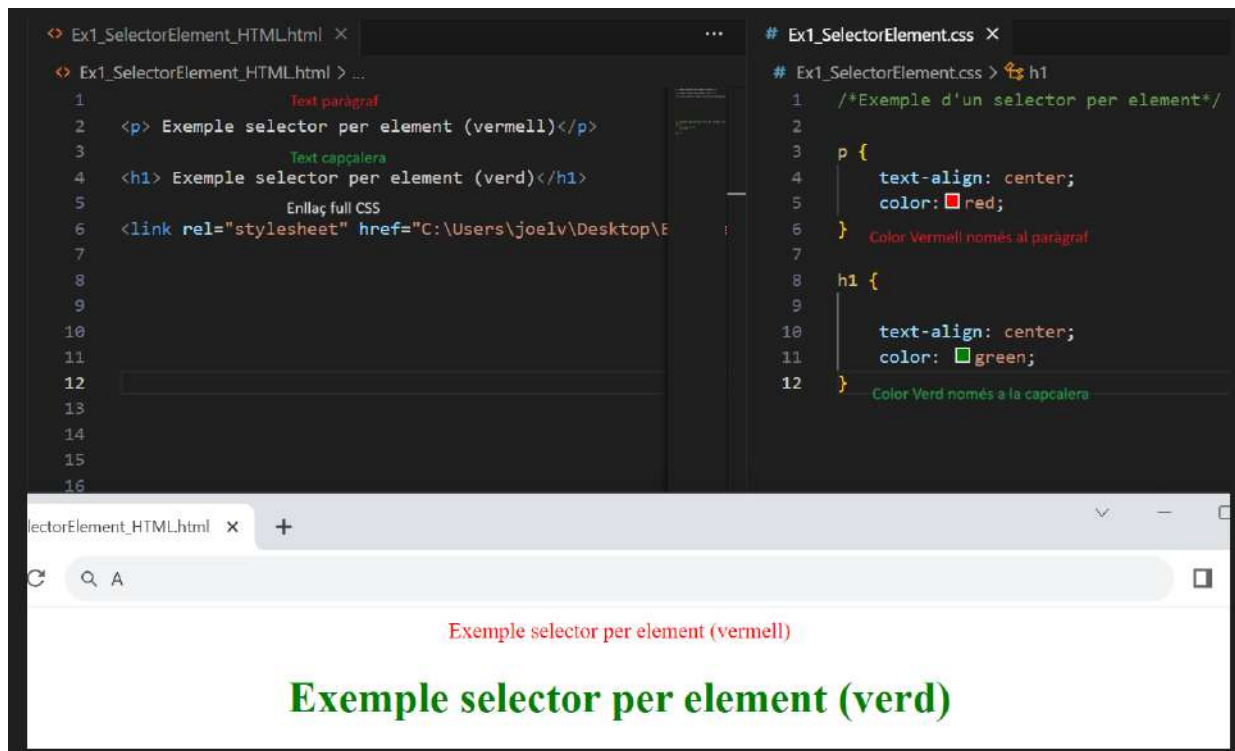
Podem dividir els selectors en diverses categories:

- **Simple selector:** seleccionen un element basat en el nom, id o classe.
- **Combinator selector:** seleccionen elements basats en relacions amb altres selectors. És a dir, agafa diferents tipus a l'hora: Títols, paràgrafs, etc. També anomenat universal.
- **Pseudo-class selector:** selecciona en base a cert estat.
- **Attribute selector:** selecciona elements basats en un atribut o el valor d'un atribut.

2.3.1 Selector per element

Aquest tipus de selector només aplicarà el format descrit al fitxer HTML, per l'element al qual se li atribueix el format al fitxer CSS. És a dir, aplicarà el format als paràgrafs `<p></p>` sempre que el format es defineixi entre `p {...}` o a una capçalera `<h1></h1>`, sempre que el format es defineixi entre `h1 {...}` i així amb qualsevol element HTML, sempre que s'utilitzi l'element adient al CSS.

A continuació es pot veure un exemple d'un paràgraf vermell (es defineix vermell només aquest element) i una capçalera verda (es defineix verd només aquest element), i el codi HTML i CSS emprats. (**Imatge 4**).



Imatge 4. Exemple de selector per element.

2.3.2 Selector per ID

El ID és un codi d'identificació que es poden posar en els elements, per conèixer més ràpidament cadascun dels elements. Per assignar un codi a un element, per exemple un paràgraf s'ha d'atorgar un ID: `<p id="identificació_element"> text </p>`. d'aquesta manera, ara al CSS ja no farem referència al paràgraf com una `p{..}`, sinó que farem referència amb l'identificador: `#identificació_element {...}`

```
<p id="paragraf2"> Exemple selector per ID </p>

#paragraf2{
    color: magenta;
}
```

Img 5: exemple identificador HTML i CSS.

2.3.3 Selector per classe

D'igual manera que el **ID**, ara tenim la classe, es defineix igual però amb l'atribut **class**. La classe serveix per agrupar diferents elements dintre d'un mateix format, per exemple, si és la part de la pàgina web i la seva localització, o les lletres de la pàgina web que són d'un color. És com l'ID, però agrupant elements. Per posar una classe en un element i després assignar-li un format s'ha de: `<h2 class = "nom_classe">text</h2>` i ara per assignar format CSS: `.nom_classe {...}`.

```
<h2 class = "pàgina1">Exemple selector per classe</h2>

.pàgina1 {
    color: blue;
}
```

Img 6: exemple classe HTML i CSS.

2.3.4 Selector universal

El selector universal permet donar-li un mateix format a qualsevol element que hi hagi en el fitxer HTML. Per determinar que és universal, s'ha d'escriure en el CSS: `*{...}` i això agafarà qualsevol element que estigui definit al fitxer HTML.



Exemple selector per element (vermell)

Exemple selector per element (verd)

Exemple selector per ID

Exemple selector per classe

Img 7. exemple selector universal.

2.3.5 Agrupació de selectors

L'agrupació de selectors està dissenyat per a poder agrupar diversos elements sense la necessitat de que siguin TOTS, com seria el cas dels selector universal. Per a fer una agrupació de selectors, només s'ha de posar els elements, separats per una coma, abans de les claus. És a dir: **p, h1, h2 {....}**

3. Colors

Els colors, en HTML/CSS, es poden especificar, directament amb el nom del color en anglès, per colors bàsics, però si ja es busca un color específic o una combinació de colors, s'ha d'especificar aquest mitjançant valors: [RGB](#), [HEX](#), [HSL](#), [RGBA](#) o [HSLA](#). Per saber més sobre aquests colors i com importar-los al vostre codi, podeu consular la referència 5.

En HTML es pot donar format, afegint colors, d'algunes parts de la pàgina, com el text, el fons, el contorn d'un text, etc.

3.1 Colors del text

La possibilitat de canviar diversos colors del text, com el color de les lletres, de fons o del contorn, es pot realitzar directament amb HTML, sense la necessitat d'utilitzar un CSS.

- **Color de fons:** serveix per afegir un color de fons al text, com si estigués subratllat. Per a poder dur a terme aquest canvi de color, s'ha de definir a l'element, un **<element style="background-color:nomdelcolor;">**.

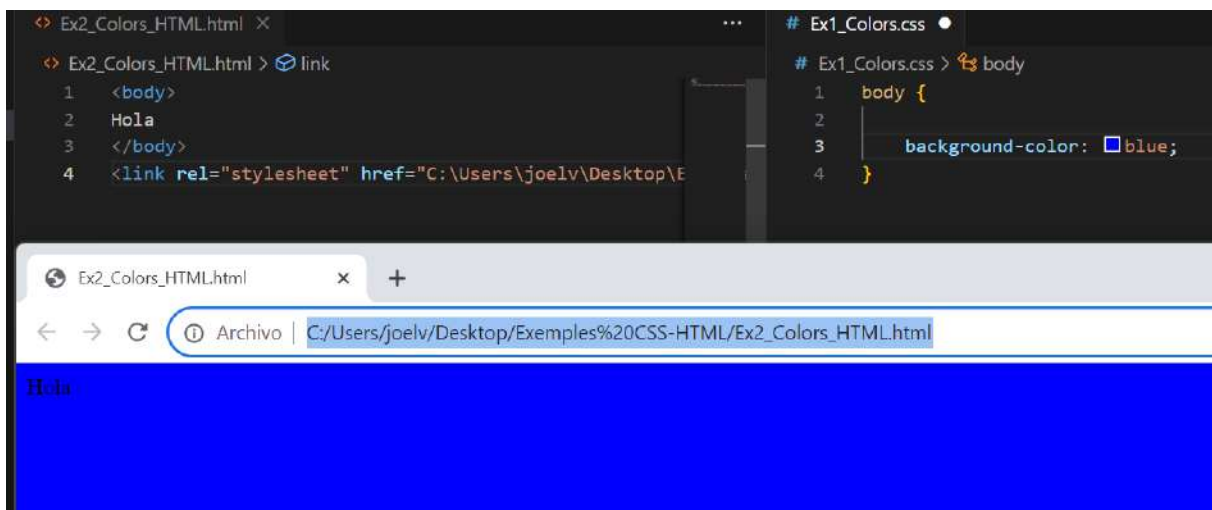
Teoria de llenguatge de fulls d'estil - CSS (*Cascading Style Sheets*)

- **Color de text:** canviar el color de text pot servir per agrupar temàtiques, per decorar, etc. En aquest cas únicament s'haurà de definir el color a l'atribut *style*, **<element style="color:nomdelcolor;">**.
- **Color de contorn:** el text d'HTML es pot utilitzar una propietat, *border*, de l'atribut *style*, per canviar el color del contorn del text. Per poder-ho fer s'ha d'afegir la comanda: **<element style = "border=nomdelcolor;">**.

3.2 Color de fons

Es pot canviar el color de fons de moltes coses, en un codi HTML, per exemple el color de fons de pàgina, del fons d'una imatge, etc. Però per fer-ho necessitarem l'ajuda del CSS. Tenim diversos tipus:

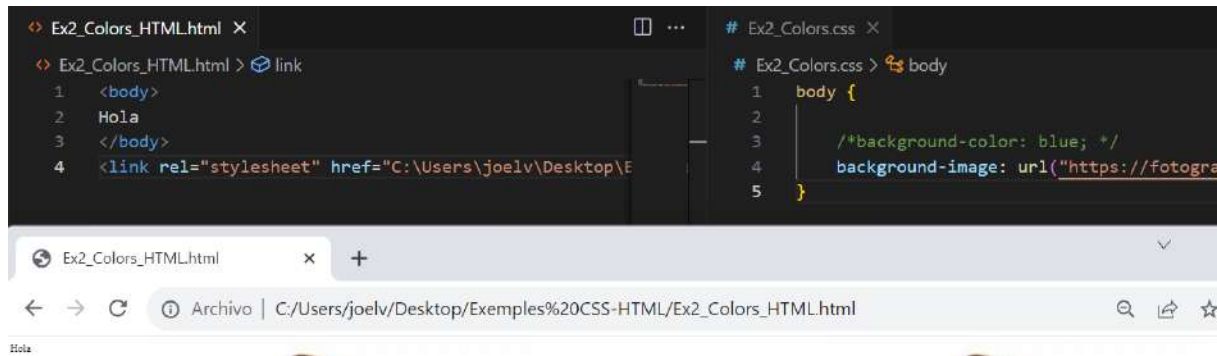
- **background-color:** Aquesta propietat permet posar el fons de la pàgina d'un color. Per això s'ha d'afegir dintre de l'element **body{...}**, per exemple:



Img 8. Canvi del color de fons mitjançant CSS.

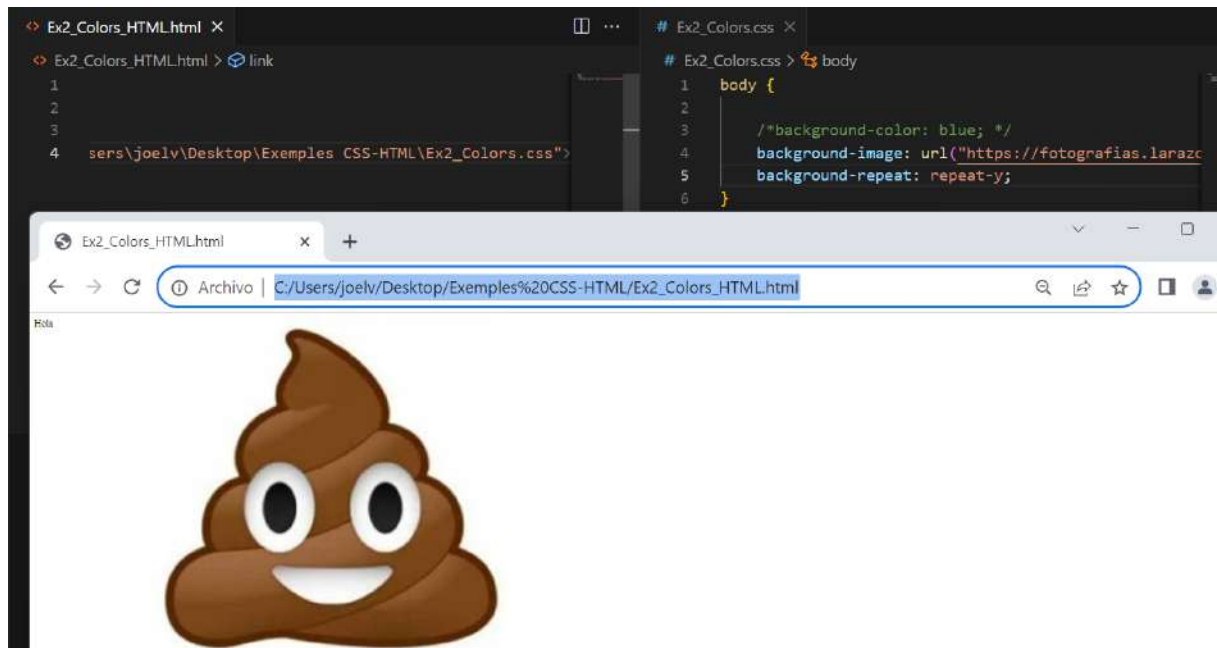
- **background-image:** en aquest cas, aquesta propietat permet posar una imatge de fons. Aquesta es repetirà, tantes vegades com sigui necessari fins omplir la pàgina sencera. Per poder posar la imatge, amb aquest mètode, només podrem agafar enllaços d'imatges d'internet, no funciona amb un ruta local del teu equip.

Teoria de llenguatge de fulls d'estil - CSS (*Cascading Style Sheets*)



Img 9. Afegir una foto com a fons de la pàgina.

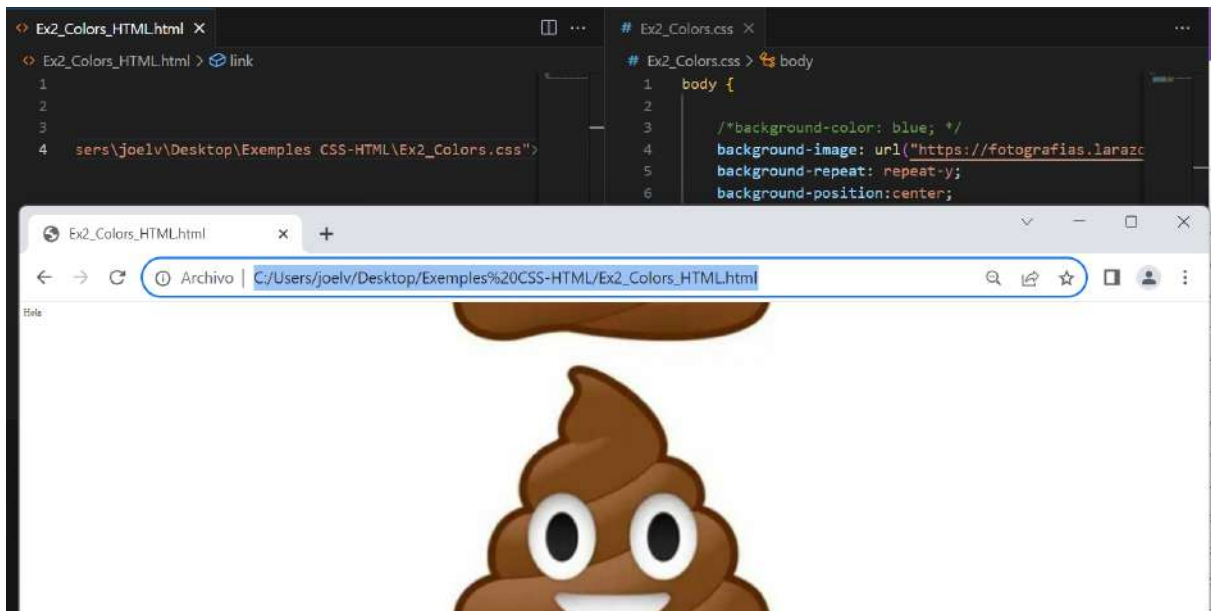
- **background-repeat:** per configurar la repetitivitat de la imatge de fons, que és escalable i s'afegeixen tantes imatges com gran sigui la pantalla amb la que estas accedint a la pàgina. Per a espaiar més les vegades que es repeteix la imatge. En aquest cas, afegint **repeat-y**, aconseguim que no es repeteixi en l'eix X, només en l'eix Y. En el cas que no volem que es repeteixi: **no-repeat**. I encara hi han més opcions.



Img 10. Modificant la repetitivitat de la imatge de fons.

Teoria de llenguatge de fulls d'estil - CSS (*Cascading Style Sheets*)

- **background-position:** amb aquesta opció es pot canviar la posició de la imatge de fons, és a dir, es pot centrar al mig, per exemple. En aquest cas haurem d'escriure: **background-position: center;**.



Img 11. Modificant la posició on es troba la imatge al fons de la pàgina.

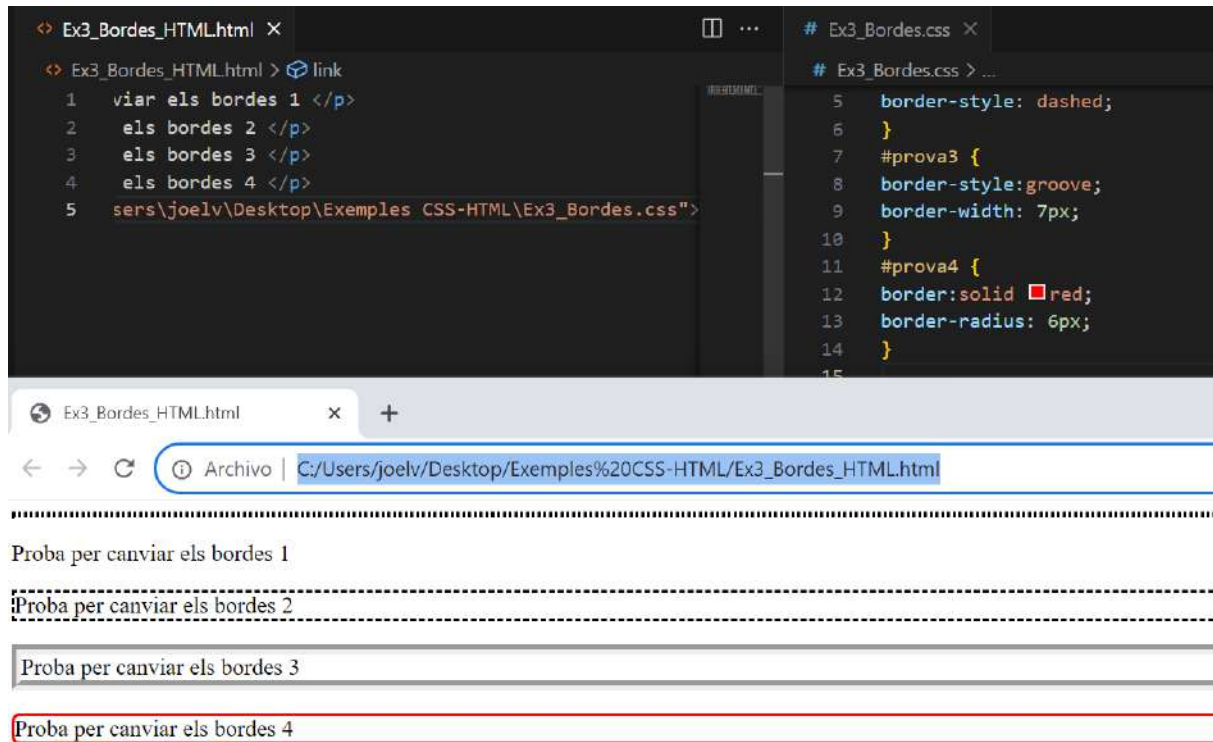
- **background-attachment:** aquesta opció permet fixar la imatge i que estigui fixa quan es fa *scroll* de la pàgina. Per deixar la imatge immòbil, s'ha d'escriure: **background-attachment: fixed;**.

4. Bordes

En aquest apartat, s'especifica com posar un borde en un text, un contorn, i canviar-li la forma, el color, la mida, etc.

Les diferents opcions que dona CSS per modificar i configurar els bordes són:

- **border-style:** mitjançant aquesta declaració, es pot canviar el tipus de línia: de punts, de ratlles, línia doble... **border-style: tipusdeborde;**.
- **border-width:** per modificar el gruix de la línia: **boder-width: 7px;**.
- **border-radius:** es poden modificar les cantonades i arrodonir-les: **border-radius: 7px;**.
- **border:** amb *border*, es pot canviar per exemple, el color: **border:solid red;**



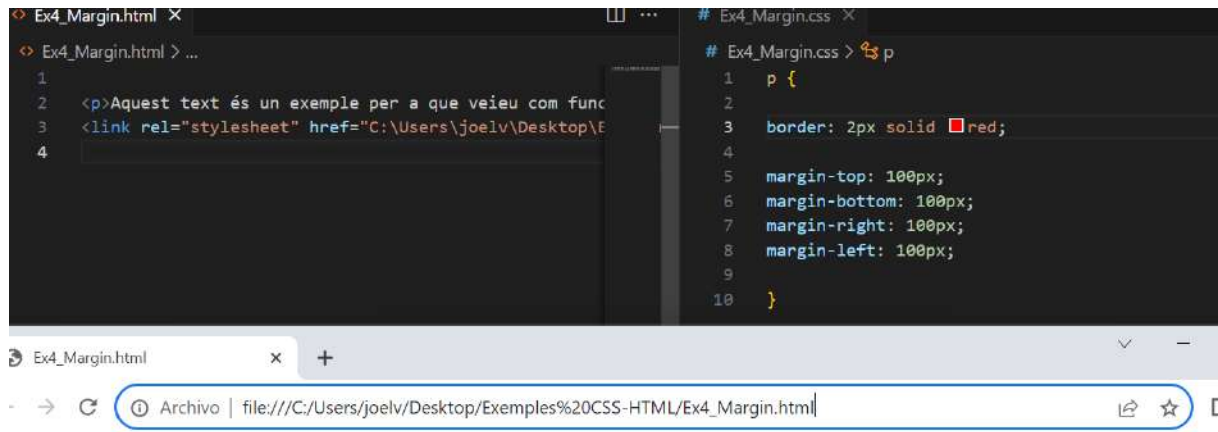
Img 12: bordes i tipus de bordes.

5. Marges

També es poden definir els marges, és a dir, des d'on fins a on hi haurà text. A més es pot definir que et canviï el color o que et marqui quina és la zona dels marges. Es pot definir de diverses maneres:

- **Forma llarga:** definint cadascuna de les 4 mesures de marges, és a dir: **margin-top: valor px, margin-bottom: valor px, margin-right: valor px, margin-left: valor px.** el que quedaria com:

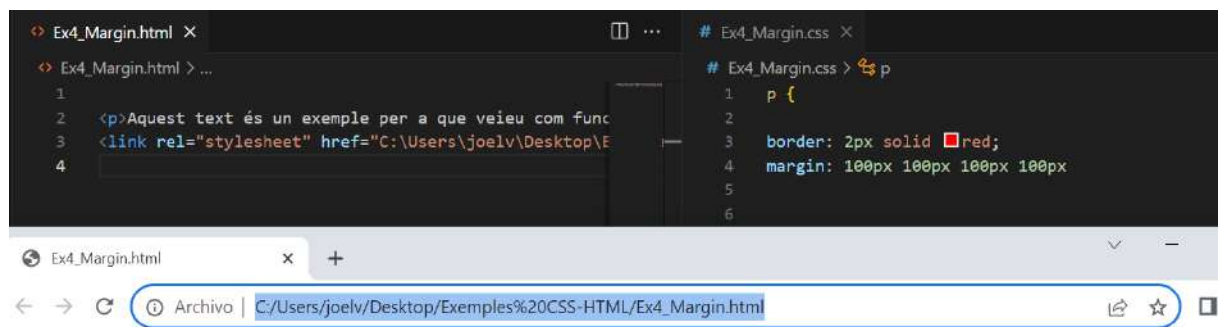
Teoria de llenguatge de fulls d'estil - CSS (*Cascading Style Sheets*)



Aquest text és un exemple per a que veieu com funcionen els marges.

Img 13: modificació dels marges del text. Es fa un bordejat per a que es pugui veure millor.

- **Forma curta:** amb la forma curta, només hem d'afegir un **margin** i a continuació d'aquest les 4 mesures, **top**, **bottom**, **right** i **left**.



Aquest text és un exemple per a que veieu com funcionen els marges.

Img 14: modificació dels marges del text. Forma curta.

6. Responsive Web Design

El *Responsive Web Design* és dissenyar o crear la teva pàgina web, a partir d'HTML i CSS, de forma que automàticament sigui rescalable i s'adapti para que es pugui visualitzar correctament en qualsevol dispositiu des del qual es visiti la web: tauleta, televisió, ordinador, mòbil, etc.

Per a que una pàgina sigui *responsive* s'haurà d'afegir, a l'HTML, el següent **<meta>**:

<meta name="viewport" content="width=device-width, initial-scale=1.0">

6.1 Imatges *responsive*

En el cas de les imatges, per a que una imatge escalesi automàticament, quan l'afegim a la pàgina web, podem posar-li un *style* amb el 100% de l'amplada de la imatge i així autoescalarà: ****. Això pot comportar un problema, ja que si la imatge supera la seva mida d'origen, és possible que aquesta al adaptar-se a la mida de la pantalla, es pugui veure pixelada, per aquest motiu, la podem fer *responsive* però sense que superi la seva mida original màxima: ****

També podem fer una imatge *responsive*, si afegim els **** i les **<source media...>**, dintre d'un element **<picture></picture>**. Aquest element reescala automàticament totes les imatges que es convoquin dintre.

6.2 Text *responsive*

Per a que un text s'adapti a la forma de la pantalla amb la que s'està consultant la pàgina web, s'haurà d'afegir un atribut *style* dintre de l'element (paràgraf, capçalera, etc), aquest atribut *style* quedaria de la següent forma: **<h1 style = "font-size:10vw"> text </h1>**. Hi han molts *frameworks* que ja ens permeten un disseny *responsive*, com pot ser *bootstrap*.

7. Elements flotants

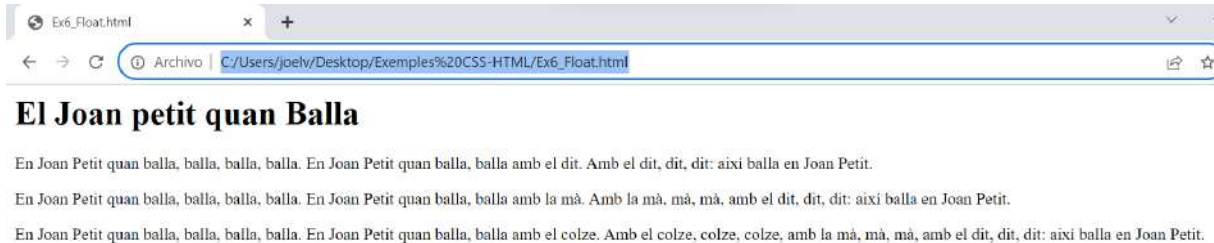
Originàriament, els elements flotants [6], estaven pensats per flotar imatges dintre de blocs de text, és a dir, la típica imatge que està al marge dret del text i és envoltada pel text. La propietat *float* llavors es va convertir en una de les eines més utilitzades per crear dissenys multicolumna en les pàgines web. Actualment s'utilitza menys, ja que han sorgit d'altres formes de poder realitzar dissenys multicolumna, com el *flexbox* o el *grid*.

Teoria de llenguatge de fulls d'estil - CSS (*Cascading Style Sheets*)

Per a entendre el que és un element flotant, fem un exemple senzill on tenim un text, llarg, la lletra d'una cançó i apliquem un CSS, que crearà un quadrat justament al costat del text, aquest quadrat, per exemple, simula una imatge.

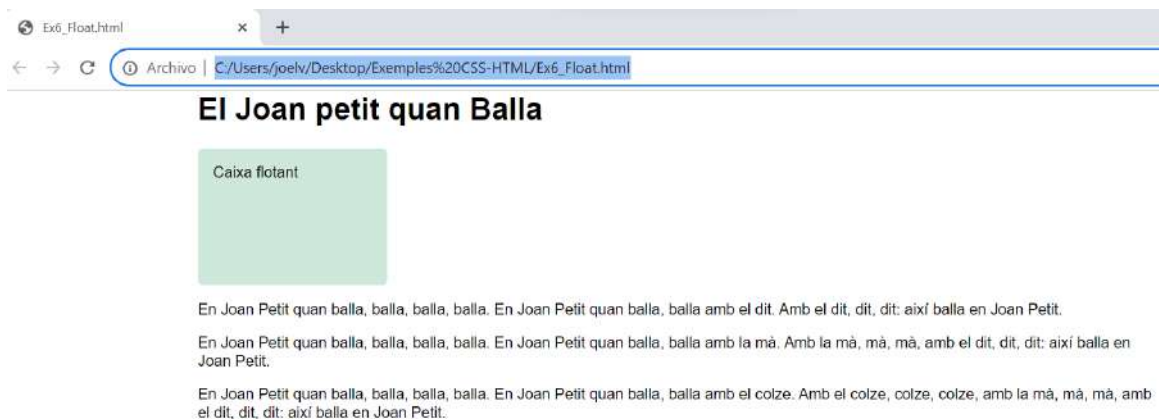
Ara farem un recull d'imatges per a que s'entengui:

- Arxiu HTML sense CSS:



Img 15: Arxiu HTML sense format.

- Arxiu HTML amb CSS, s'ha creat una caixa de color, però aquesta no s'ha flotat. Podem veure una capsa entre el títol i el text. A més es pot veure com la font del text, la mida, etc ha canviat.



Img 16: Pàgina amb la capsa afegida (sense flotar) i amb el format definit al CSS.

- A continuació, podem veure com afegint en el CSS, *float* i *margin-right* es pot col·locar la caixa blava allà on es vulgui. En aquest cas, hem decidit que la caixa estigui a l'esquerra, per tant: **float: left;** i **margin-right: 15px;** que és la distància entre la capsa i el text de la dreta, aquests 15 px.



Img 17: Pàgina amb la capsa flotant al marge esquerra.

Encara es poden afegir d'altres caixes, per exemple, una que agafi només el primer paràgraf, per poder dur a terme aquesta nova caixa, s'ha de crear un nou element a l'arxiu CSS, aquest arxiu conté la forma i el color d'aquesta nova caixa i òbviament, amb el nom d'aquest element, podem definir sota quin text anirà. Al CSS també es defineix que aquell troç de text canvia de color, per poder veure-ho bé amb el contrast de color blau del fons.

El Joan petit quan Balla



Img 18: Aquest quadre no ha d'estar flotant, ja que ha d'anar sota un text.

Amb els flotants es poden afegir noves línies de codi al CSS, per dur a terme diferents tasques, per exemple:

Nom	Descripció	Exemple
Netejar <i>floats</i>	Es poden netejar les caixes float, és a dir, eliminar el color, però deixar en mode <i>transparent</i> el contingut. Per fer això, afegirem al CSS, l'element: clear: left/right/both, en funció d'on tenim el <i>float</i>.	Aquesta imatge és una captura de pantalla d'un navegador web. El títol de la pàgina és "El Joan petit quan Balla". A sota del títol hi ha una caixa rectangular de color verd clar amb el text "Caixa flotant". A continuació, hi ha tres paràgrafs de text que comencen amb "En Joan Petit quan balla, balla, balla, balla." i continuen amb frases com "amb el dit", "amb la mà" i "amb el colze".

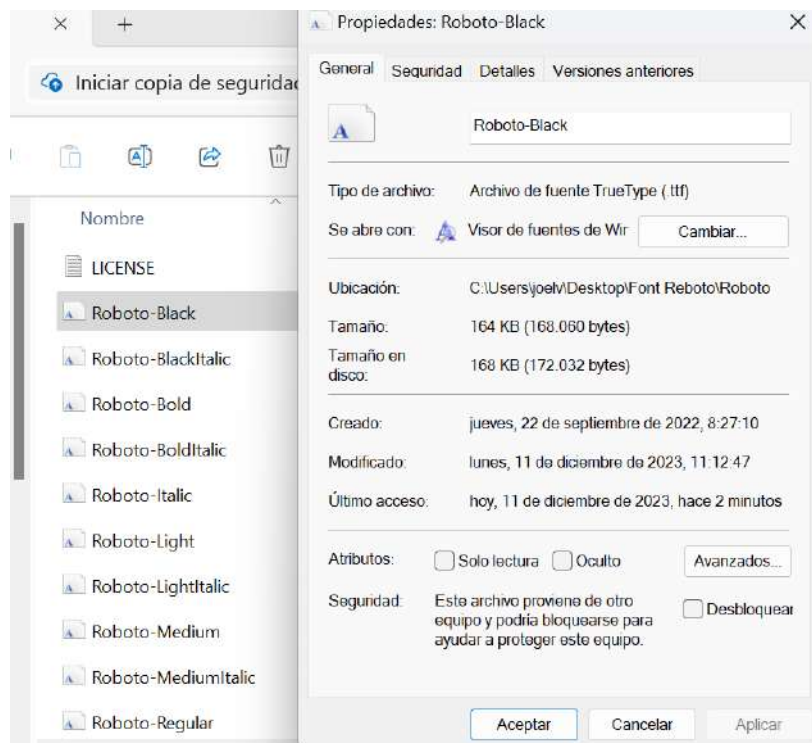
Taula 1: eliminar una forma flotant.

8. Estructura d'un projecte real

Normalment, els projectes segueixen una estructura bàsica, que es podrà complicar més o menys en funció del tipus de projecte.

L'estructura bàsica, és la següent:

- **Carpeta de fonts:** on anirem posant les fonts (tipus de lletra), ja hi han pàgines especialitzades on poder consultar [7] i descarregar aquestes fonts, per exemple amb l'eina gratuïta [Google Fonts](#), et pots descarregar aquestes fonts amb l'extensió **.ttf (TrueTypeFont)**. I afegir aquests arxius a la carpeta **fonts** del projecte.



Img 19: Fonts descarregades de Google Fonts.

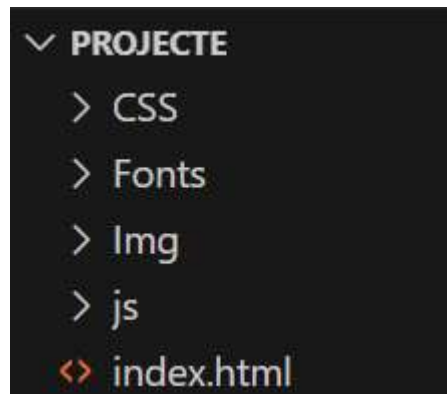
- **Carpeta Img:** en aquesta carpeta ficarem totes les imatges que s'utilitzaran a la pàgina web.
- **Carpeta JS:** la carpeta JS, és per posar dintre els arxius [javascript](#), en cas que vulguis que la teva pàgina sigui dinàmica (no estàtica, és a dir, com si fos un llibre), amb animacions, etc. Hauras de tenir arxius *javascript* i aniràn dintre d'aquesta carpeta.
- **Carpeta CSS:** pels arxius CSS, també hi ha una carpeta específica on guardarem tots. Dintre de la carpeta CSS, normalment hi ha un fitxer, anomenat **style.css**, amb el format de la pàgina.
- **Index.html:** el fitxer principal de la pàgina web, normalment sempre s'anomena **index.html**, si s'utilitza [visual studio code](#), com a editor de codi, si obrim un arxiu nou i escrivim: **html:5** automàticament crearà l'estructura HTML bàsica.



```
<> index.html •
<> index.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Document</title>
7  </head>
8  <body>
9
10 </body>
11 </html>
```

Img 20: Estructura HTML bàsica.

Per tant, l'estructura de carpetes i fitxer mínima, que ha de tenir un projecte és la següent:



```
▼ PROJECTE
  > CSS
  > Fonts
  > Img
  > js
  <> index.html
```

Img 21: Estructura de carpetes bàsica.

9. Fonts personalitzades

Una de les coses que pot ser més importants en una pàgina web, fent referència al disseny de la mateixa, és la **font de lletra** (tipografia de lletra) que s'utilitza. Per defecte, quan es pica un codi amb **HTML** aquesta font és **Times New Roman** una lletra bàsica que s'utilitza en molts editors de text com a font predeterminada.

Existeix la forma de descarregar o/i enllaçar una nova font per a donar una imatge més professional a la nostra pàgina web. Per aquest motiu accedirem a la pàgina [Google Fonts](#), tot i que hi ha d'altres que també es poden utilitzar. Aquesta pàgina és un cercador de fonts molt utilitzat pels desenvolupadors web.

Existeixen dues opcions per incorporar la font de lletra en el nostre projecte la opció 1, mitjançant un enllaç i un troç de codi que la mateixa pàgina ens facilita i l'altra opció 2, que és descarregar els arxius de la font, afegir-los a la nostra estructura de carpeta i enllaçant aquests arxius al nostre codi CSS.

1 font family selected



Whereas recognition of the inherent dignity

Img 22. Dues opcions per afegir una nova font.

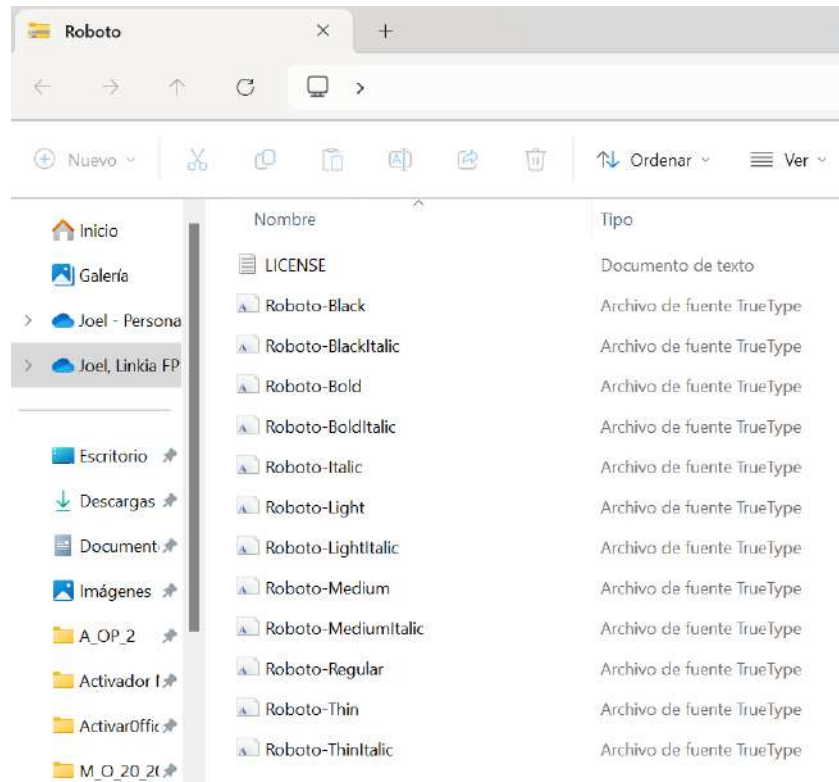
- **Opció 1:** per poder agafar el tros de codi. Hi ha un tros per enllaçar amb **html** i un altre per enllaçar amb **css**, simplement s'ha de copiar i pegar cadascú on toca i ja tindrem la font: ([Enllaç demostratiu](#))



Img 23. Tros d'HTML i tros de CSS.

- **Opció 2:** Amb aquesta opció, hem de clicar en la part 2 de la imatge 22 i es descarrega una carpeta .zip, amb el nom de la font que estem descarregant. Aquesta carpeta, una vegada l'hem descomprimit, l'hem de guardar al a carpeta **fonts** de la nostra estructura de projectes i cridar-la en local, des de l'arxiu CSS.

Teoria de llenguatge de fulls d'estil - CSS (*Cascading Style Sheets*)



Img 24. Carpeta de fonts descarregada.

Tot i haver dues formes diferents de poder afegir una nova font a una de les nostres pàgines web, sempre es recomana utilitzar la primera opció. (**Opció 1**).

10. Filtratge d'imatges

El llenguatge amb el que es realitzen fitxers CSS, ens permet fer un tractament bàsic i simple de les imatges que s'inclouen a la pàgina. Aquest tractament es basa en una sèrie de filtres. Aquests filtres es posen dintre de l'apartat **img {.....}**.

Nom	Descripció
blur(px)	Aplica un desenfocament a la imatge, quan major sigui el nombre que posem a px major serà el desenfocament.
brightness(%)	Ajusta la brillantor de la imatge, quan més gran sigui el % més brillant serà la llum de la imatge.
contrast (%)	Quan més petit sigui el % més fosca serà la imatge. Normalment, una imatge, sense aplicar cap filtre, té un 100% de contrast, a partir d'aquí es pot anar baixant per obtenir més fosc.
hue-rotate(deg)	Aplica una rotació a la imatge on deg són els graus que aquesta rota, és a dir, de 0 a 360°.
invert(%)	Inverteix la imatge, on 0% és una imatge sense filtratge i 100% una inversió total d'aquesta.
opacity(%)	Aplica un nivell de opacitat a la imatge, normalment 0% és una opacitat transparent i un 100% és la imatge sense haver-ne aplicat cap filtre.
saturate(%)	Aplica un nivell de saturació a la imatge, 0% és la imatge sense res de saturació i un 100% és la imatge completament saturada.

Taula 2: Filtres CSS per a imatges.

```
img {
  filter: grayscale(100%);
  filter: blur(34);
  filter: brightness(55%);
  filter: saturate(90%);
}
```

Img 25: exemple d'una imatge filtrada.

Referències

1. ¿Qué es *BOOTSTRAP* y para qué sirve? ¿Cómo funciona y se usa? | Kiko Palomares. (s. f.).
<https://kikopalomares.com/clases/que-es-bootstrap-y-para-que-sirve-como-funciona-y-se-usa>
2. *Bootstrap Grid Basic*. (s. f.).
https://www.w3schools.com/bootstrap/bootstrap_grid_basic.asp
3. CSS | MDN. (2023, 13 marzo). MDN Web Docs.
<https://developer.mozilla.org/es/docs/Web/CSS>
4. ¿Qué es CSS y para que sirve? | Kiko Palomares. (s. f.).
<https://kikopalomares.com/clases/que-es-css-y-para-que-sirve>
5. Marco, B. S. (s. f.). *Colores. CSS. páginas web HTML y hojas de estilo CSS*. Bartolomé Sintes Marco. www.mclibre.org.
<https://www.mclibre.org/consultar/htmlcss/css/css-color.html>
6. *Floats - Aprende desarrollo Web* | MDN. (2023, 21 julio). MDN Web Docs.
https://developer.mozilla.org/es/docs/Learn/CSS/CSS_layout/Floats
7. Barron, B. (2021, 8 abril). *Tu guía completa de fuentes HTML (o fuentes web)*. Kinsta®. <https://kinsta.com/es/blog/html-fuentes/>
8. *Browse Fonts - Google Fonts*. (s. f.). Google Fonts. <https://fonts.google.com/>
9. *W3Schools.com*. (s. f.-c). https://www.w3schools.com/cssref/css3_pr_filter.php