

# Bootcamp Full Stack

## CSS avanzado



# Tipografías con CSS



# Propiedades de fuente

## Font-family

Es una propiedad que hace referencia a la tipografía. Actualmente, es posible trabajar con varios elementos alternativos para poder utilizar la tipografía que guste sin la limitación anterior.

Incluso, podemos sugerir con qué tipografía trabajar si el navegador tiene problemas para encontrar la primera opción tipográfica..

# Google Fonts

# Google Fonts

## Uso de un CDN en Google Fonts

Un **CDN** es un *Content Delivery Network*. El trabajo con este tipo de metodología permite **incorporar un recurso a nuestra interfaz sin necesidad de descargarlo.**

Si bien existen múltiples recursos posibles, mayormente encontramos el trabajo con **Google Fonts** como el **más usual**.

La página es <https://fonts.google.com/> y nos permite incorporar cualquier tipografía disponible en el inmenso catálogo que nos ofrece.

Al elegir una tipografía podemos encontrar también diversas variantes. No sólo eso sino debemos entender si el trabajo se está realizando con una tipografía variable o en su defecto con una fija.



## Ejemplo

Busquemos dentro de la página de Google Fonts la tipografía **Texturina**. Lo haremos utilizando el buscador en la parte superior.

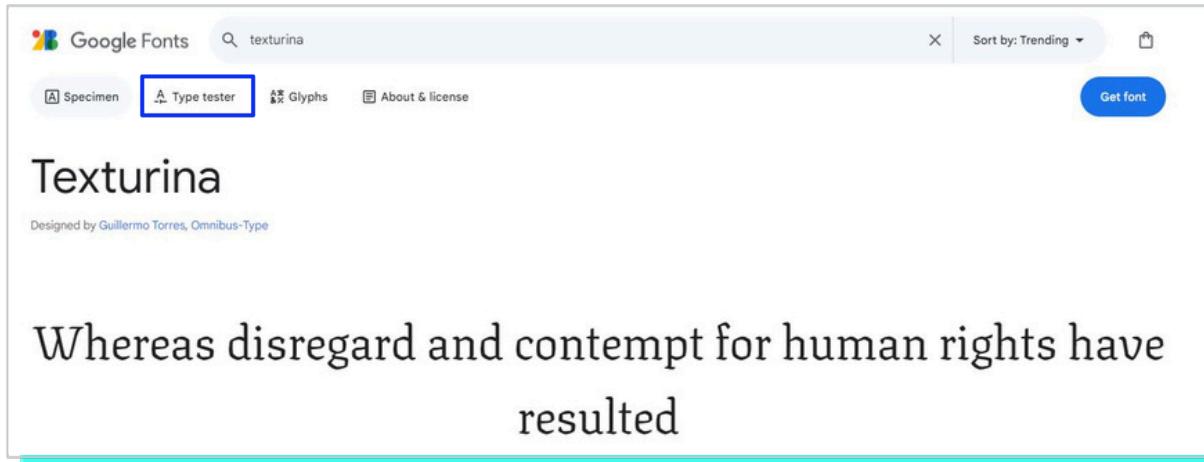
Esta tipografía de ejemplo es variable y podemos notarlo porque tal descripción aparece al lado del nombre.

The screenshot shows two instances of the Google Fonts search interface. The top instance has a search bar containing 'texturina' with a magnifying glass icon. Below it are sorting options ('Sort by: Trending') and a shopping cart icon. A large blue arrow points from the search bar of the top instance down to the search bar of the bottom instance. The bottom instance also has a search bar with 'texturina' and a magnifying glass icon, along with sorting and shopping cart options. It includes a 'Filters' button. Below the search bar, it says '1 of 1603 families'. On the right, there's a link 'About these results' with an info icon. At the bottom, it shows a preview of the 'Texturina' font in a sentence: 'Everyone has the right to freedom of thought, conscience and religion; everyone also has the right to change his religion and...'. The word 'Variable (3 axes)' is highlighted with a blue box next to 'Texturina' in the preview list.

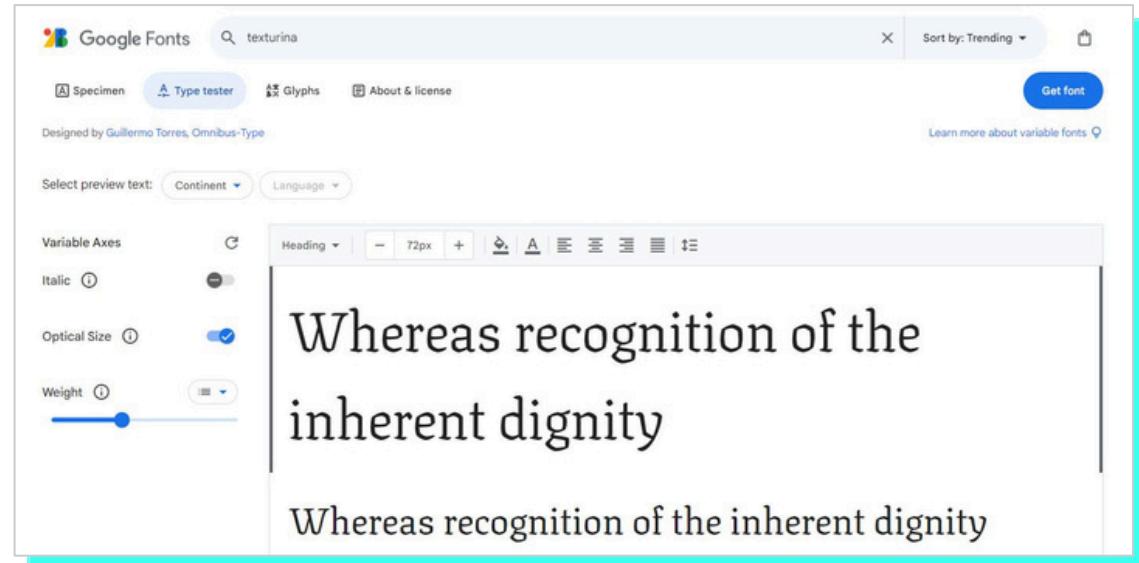
Para conocer más detalle tanto en el caso de la tipografía variable como fija sugerimos:

1. Hacerle **clic en la tipografía** en cuestión.

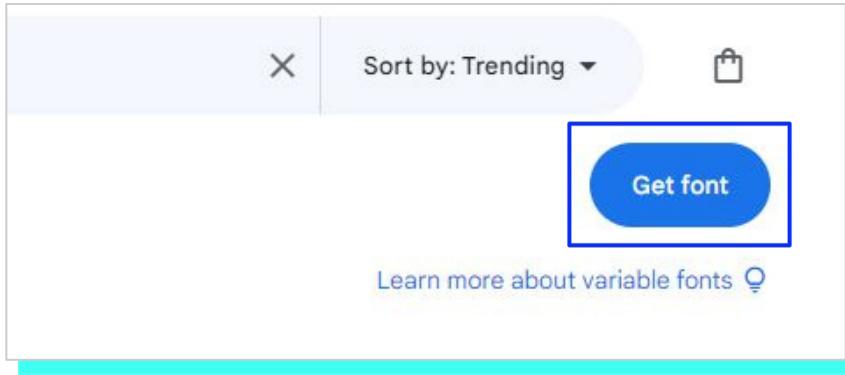
2. Ir a la sección **Type tester**.



3.Dentro de la sección **Type tester**, tenemos un *range* que nos permite indagar sobre cada eje trabajado (*weight*, *stretch*, *width*, *style*) y los valores posibles de esa tipografía. Por ejemplo, en el caso de Texturina esta trabaja con los ejes ***optical size***, ***style (italic)*** y ***weight***.



4.Para poder utilizar la tipografía sólo debemos hacer clic en > ***Get font***.



**Nota:** siempre debemos saber qué variantes tiene una tipografía. Por ejemplo, si quiero utilizar el *eje width* en esta tipografía por más que lo intente no podrá porque la misma no lo posee.

## ¿Cómo vincular la tipografía en nuestro CSS?

1. Luego de hacer clic en > **Get font**, vamos a obtener la pantalla que mostramos a continuación. Debemos luego elegir la opción > **Get embed code**.

2. Si por el contrario trabajamos con > **Download** esta puede ser una buena alternativa si se quiere trabajar con la tipografía en un programa de diseño.



### 3. En la **pantalla de vinculación** o ***embed code***

tenemos dos posibilidades que podemos trabajar vinculadas con este curso (si bien hay otras relacionadas con otras plataformas como Flutter o Android: si es de tu interés te sugerimos realizar el curso de Introducción a Android).

4. En este curso se trabajará con la opción import dado qué es la más ágil y conveniente para trabajar dado que se incluye directamente en el CSS.

Web      Android      iOS      Flutter

<link>  @import

Embed code in the `<head>` of your html

```
<style>
@import url('https://fonts.googleapis.com/css2?family=Texturina:ital,opsz,wght@0,12..72,100..900;
1,12..72,100..900&display=swap');
</style>
```

Copy code

En nuestra hoja de estilos pegaremos el texto copiado sin incluir la etiqueta `<style></style>` dado que no estamos trabajando en nuestro HTML y sumaremos un ; al final para qué no de error.

Notamos que las tipos variables como Texturina traen todos los valores trabajando con un rango, por ejemplo, en el eje wght (weight) qué vemos en la importación el valor va desde el 100 al 900 , es decir en un solo fichero-archivo- traemos todos los valores.



```
index.html # estilos.css X
css > # estilos.css > p
1  @import url('https://fonts.googleapis.com/css2?family=Texturina:ital,opsz,wght@0,12..72,100..900;1,12..72,100..900&display=swap');
```

Gracias a esto podemos, por ejemplo, utilizar los valores deseados en diferentes elementos sin necesidad de elegir o traer uno por uno.

```
✓ body {  
    font-family: "Texturina", serif;  
    font-optical-sizing: auto;  
}  
  
✓ h1 {  
    font-weight: 800;  
}  
  
✓ p {  
    font-weight: 300;  
}
```



# Pero... ¿qué es @import?

Vinculación dentro de la propia hoja de estilo

**@import** actúa igual que la **vinculación link**, pero a diferencia de ésta, **puede colocarse dentro de un archivo.css**.

De esta manera, ante cualquier cambio, **hace más rápido el mantenimiento de una interfaz**. Alcanza con ir a un archivo vinculado con todos los **.html** y modificar el elemento distinto. Por ejemplo, en el caso de **cambiar la tipografía de un sitio web entero**.

```
@import url('estilos.css')
```

# Profundizando: ejes tipográficos

## Uso de ejes desde Google Fonts

En Google Fonts, encontramos **tipografías fijas y otras variables.**

Primero debemos entender que **las tipografías tienen ejes.** Por ejemplo:

- italic (eje de inclinación / *slant*)
- bold (eje de peso / *weight*)
- condensed (eje de ancho / *width*)
- bold-italic (eje de inclinación y peso)

Cuando trabajamos con los diferentes ejes y sus valores, por ejemplo en el weight ir de 400 a 700, debemos utilizar o llamar todos los archivos necesarios para poder hacerlo.

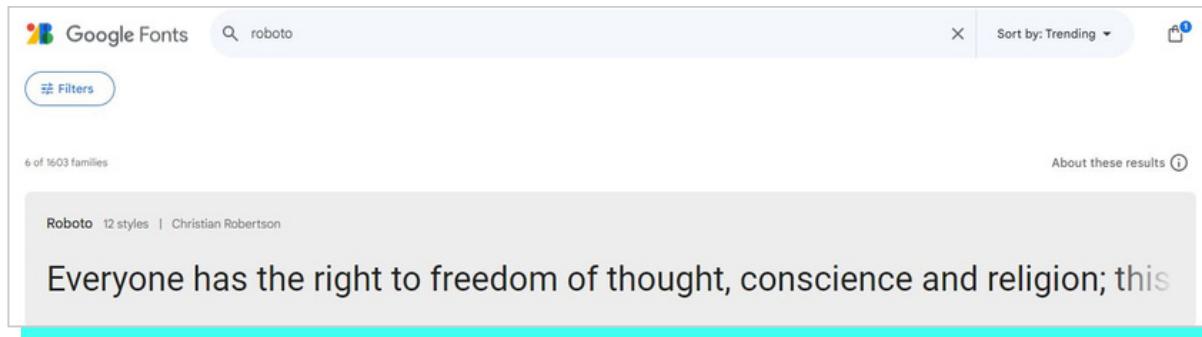
**Nota:** como hemos visto hay más ejes posibles, incluso un diseñador puede inventar un eje con un nombre que de elección. Por ejemplo a través de illustrator, figma o incluso plataformas como [FontStruct](#) o [Prototypo](#).



## Tipografías no variables o fijas: ejemplo Roboto

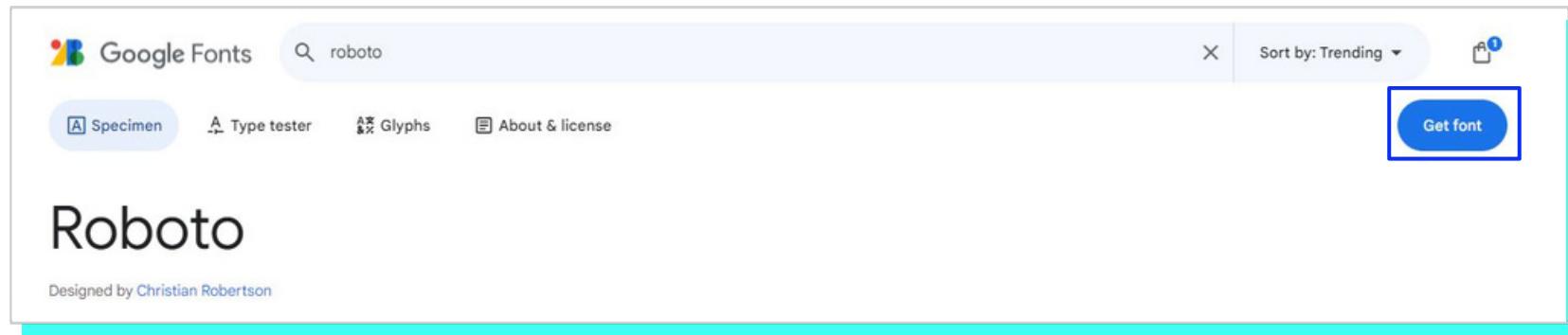
Una tipografía fija, como Roboto, obliga a vincular todas las variantes que se necesitan una por una. Nos damos cuenta que una tipografía es fija porque al hacerle clic nos muestra los números de estilo, por ejemplo.

**Nota:** a diferencia de una tipografía variable, como la anteriormente vista, que con un mismo archivo se pueden usar todas las variantes. De esta manera, se utilizarán menos recursos y será más fácil la implementación.



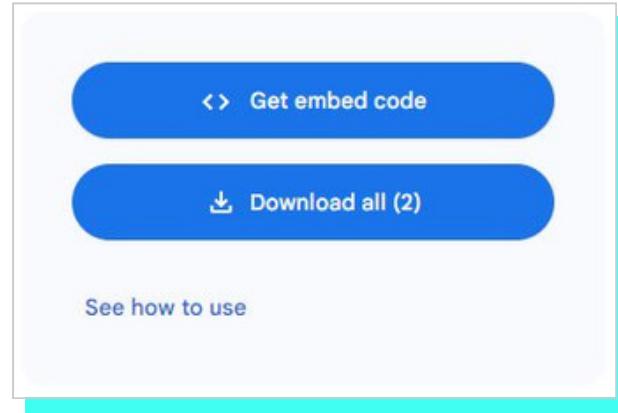
## ¿Cómo implementar una tipografía fija?

Luego para poder utilizar la tipografía fija elegida debemos hacer clic en > **Get font**.



Esto nos llevará a la pantalla donde al hacerle clic a > **Get embed code** igual a las tipografías variables, nos llevará a linkear o importar (en este curso trabajaremos con *Import*) la misma.

**La diferencia de la tipografía variable a la fija, es que en la última debemos decidir que ejes y valores utilizaremos.** Esto no depende de una elección al azar, sino de lo que el diseñador eligió para trabajar en proyecto presentado al cliente y aprobado por este.



A través de > **Change styles** podemos decidir con cuál quedarnos.

The screenshot shows the Google Fonts interface for the Roboto font family. On the left, a preview window displays the text "Whereas recognition of th" in various styles. Below the preview, a list of 12 font styles is shown in a grid:

Thin 100	Light 300	Regular 400	Medium 500	Bold 700
Black 900	Thin 100 Italic	Light 300 Italic	Regular 400 Italic	
Medium 500 Italic	Bold 700 Italic	Black 900 Italic		

A blue rectangular box highlights the "Change styles" button, which is located below the preview and above the list of styles. To the right of the preview, there is a "Web" tab and several other tabs for "Android", "iOS", and "Flutter". Under the "Web" tab, there are two radio button options: "<link>" and "@import", with "@import" being selected. Below these options is a code editor containing the CSS code for embedding the font. A "Copy code" button is located to the right of the code editor. At the bottom, there is a section titled "Roboto: CSS classes" with two defined classes:

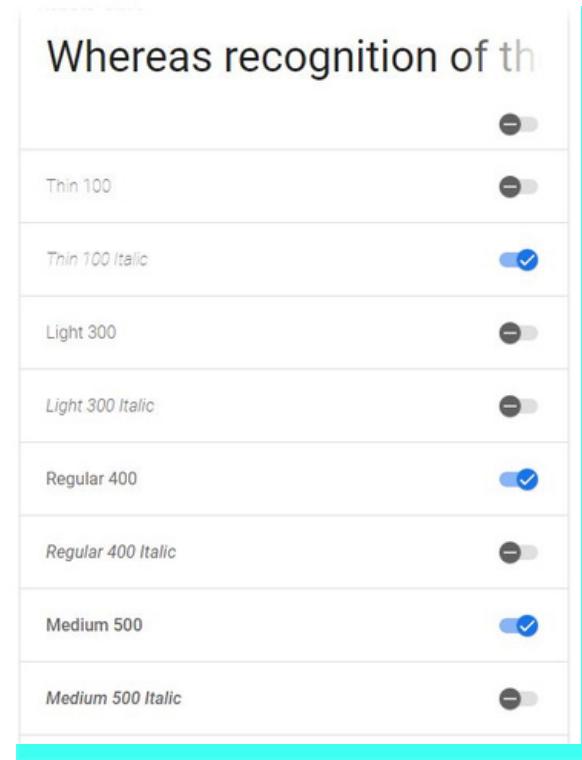
```
.roboto-thin {  
    font-family: "Roboto", sans-serif;  
    font-weight: 100;  
    font-style: normal;  
}  
  
.roboto-light {  
    font-family: "Roboto", sans-serif;  
    font-weight: 300;  
    font-style: normal;  
}
```

## Pantalla *Change styles*

Esta pantalla nos permite con la selección correspondiente marcar qué valor o eje trabajar por ejemplo el diseñador eligió trabajar sólo con thin italic, regular y medium. Por lo que elegiremos sólo estos valores.

### ¿Por qué NO llevar todas las variantes?

A diferencia de las tipografías variables, el llevarnos innecesariamente variantes no utilizadas o valores no solicitados por el diseño generarán más peso e inconvenientes al momento de trabajar.



## ¿Notaron algo en el import anterior?

En los ejemplos queremos mostrar cómo se pueden también trabajar dos tipografías al mismo tiempo.

Por ejemplo, no hemos dejado de trabajar con Texturina y ahora decidimos sumar Roboto, por eso el import muestra ambas.

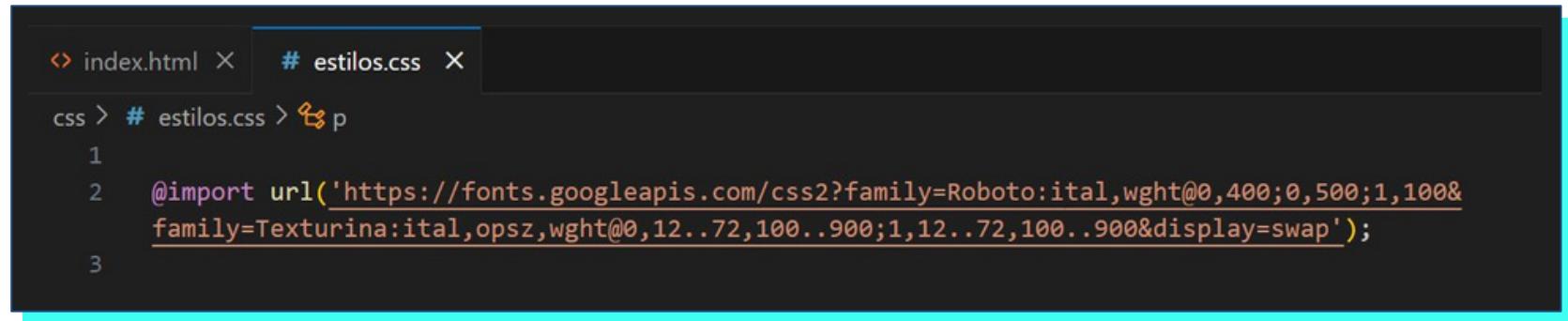
The screenshot shows a user interface for managing fonts across different platforms. At the top, there are tabs for Web, Android, iOS, and Flutter. The Web tab is currently selected, indicated by a blue underline. Below the tabs, there are two radio button options: <link> (unselected) and @import (selected, indicated by a blue circle). A section titled "Embed code in the <head> of your html" contains the following CSS code:

```
<style>
@import url('https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,400;0,500;1,100&family=Te
xturina:ital,opsz,wght@0,12..72,100..900;1,12..72,100..900&display=swap')
</style>
```

At the bottom right of this code block is a "Copy code" button with a clipboard icon.

## ¿Cómo utilizarlo?

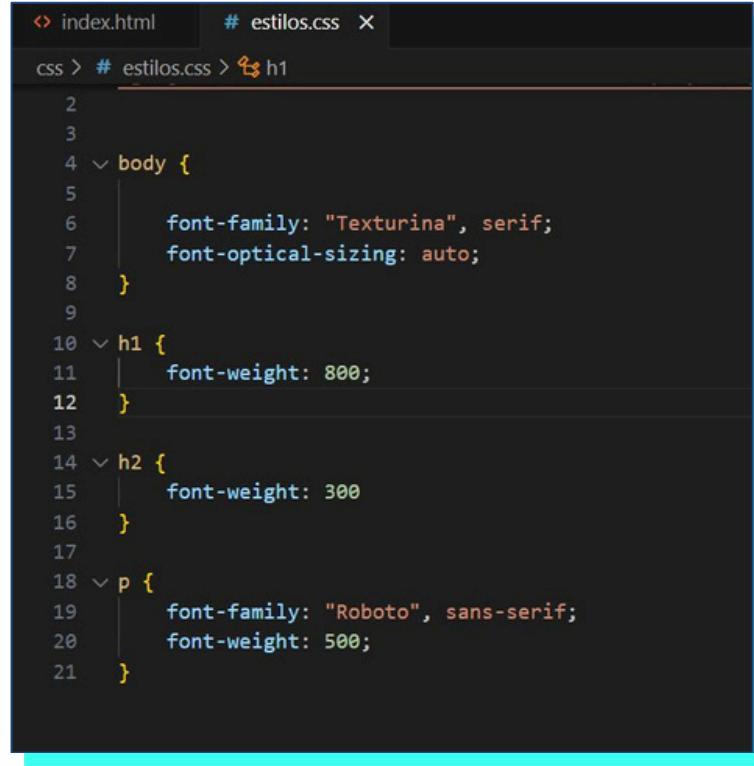
Coparemos y pegaremos el import en nuestro sin hacer uso de la etiqueta **<style> </style>** y sumando el ; para no generar un error.



The screenshot shows a code editor with two tabs: 'index.html' and '# estilos.css'. The '# estilos.css' tab is active, displaying the following CSS code:

```
css > # estilos.css > p
1
2 @import url('https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,400;0,500;1,100&
family=Texturina:ital,opsz,wght@0,12..72,100..900;1,12..72,100..900&display=swap');
3
```

Dado que hemos trabajado sobre un ejemplo real, donde por lo general se trabaja con más de una tipografía, por ejemplo, una para los elementos *display (headers)* y otra para los textos comunes, en este caso vamos a tener un diseño a replicar donde el **h1** y **h2** heredan la tipografía **Texturina** del *body* con sus elecciones específicas, y luego, los párrafos se trabajan con la tipografía **Roboto** y uno de los valores importados (revisar *slides* anteriores), en este caso *medium* o 500.



```
index.html # estilos.css x  
css > # estilos.css > h1  
2  
3  
4 <body {  
5  
6     font-family: "Texturina", serif;  
7     font-optical-sizing: auto;  
8 }  
9  
10 <h1 {  
11     font-weight: 800;  
12 }  
13  
14 <h2 {  
15     font-weight: 300  
16 }  
17  
18 <p {  
19     font-family: "Roboto", sans-serif;  
20     font-weight: 500;  
21 }
```



# Regla @font-face

# Propiedades de fuente

## @font-face

Es una regla que **permite incorporar tipografías a nuestro proyecto desde una ruta, incluso en nuestro propio entorno de trabajo.**

```
@font-face { src: url(..../fuentes/BodoniFLF-BoldItalic.ttf);  
    font-style: italic; font-weight: bold;  
    font-family: miFuente;}
```

Siempre que trabajamos con un **recurso externo** a nuestra hoja de estilo debemos trabajar con el **protocolo url** y luego **indicar la ruta**.

## Generar variantes

Las tipografías tienen variantes, y **debemos explicitarlas** para poder trabajar, por ejemplo la **tipografía anterior consta de las siguientes variantes:**

A	BodoniFLF-Bold
A	<i>BodoniFLF-BoldItalic</i>
A	<i>BodoniFLF-Italic</i>
A	BodoniFLF-Roman

# @font-face

## Uso de variantes

Cada variante que nos interese utilizar debe estar mencionada de la siguiente forma:

```
✓ @font-face { src: url(..../fuentes/BodoniFLF-BoldItalic.ttf);
    font-style: italic; font-weight: bold;
    font-family: miFuente; }

✓ @font-face { src: url(..../fuentes/BodoniFLF-Bold.ttf);
    font-weight: 400;
    font-family: miFuente; }
```

Luego, aquel elemento que utilice esas variantes tomará ese recurso específico, por ejemplo:

```
p {font-family: miFuente; font-style: italic; font-weight:bold; }
```



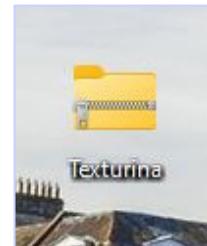
# @font-face

## Uso de tipografías variables

Las tipografías también pueden ser variables cuándo trabajamos con **Import**. Si bien hay numerosas páginas para trabajar, vamos a tomar la tipografía Texturina y descargarla desde la página de **Google fonts** de modo de simplificar el ejemplo y compresión del mismo.

Al momento de descargarlo en general este baja en formato **comprimido o zip**.

Al descomprimirlo encontramos si es el caso dos ficheros, uno con todos los ejes y variables posibles y otro con lo mismo más itálica, dado que siempre se trabaja aparte.

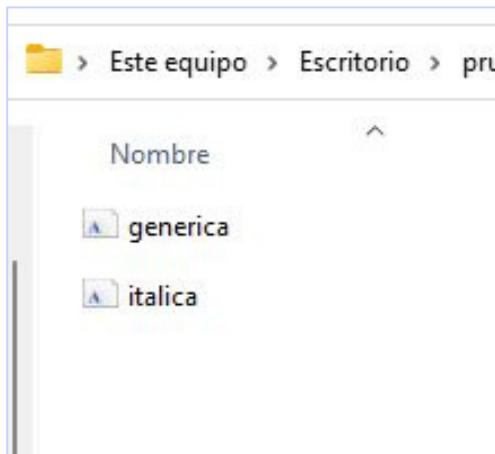


Texturina-Italic-VariableFont\_opsz,wght

Texturina-VariableFont\_opsz,wght

Para simplificar el ejemplo hemos sobreescrito el nombre de origen de los archivos o ficheros de la siguiente manera.

28

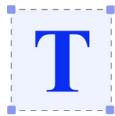


T



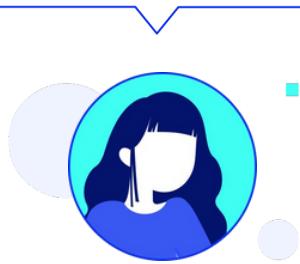
## Importación

Ahora debemos trabajar con la importación pero especificando el rango, luego a su vez según el diseño implementarla pero en este caso a través de la propiedad **font-variation-settings**.



```
index.html x # estilos.css x
css > # estilos.css > body
1 1 <@font-face {
2   font-family: miTipografia;
3   src: url(..tipografias/generica.ttf);
4   font-weight: 100 900;
5   font-optical-sizing: auto;
6
7 }
8
9
10 <body {
11
12   font-family: miTipografia;
13   font-variation-settings: 'wght' 100;
14
15 }
16
17 <h1 {
18
19   font-variation-settings: 'wght' 900;
20 }
```

El **resultado** del ejemplo anterior será el siguiente.



## Tipografía con @font-face

  Lorem ipsum, dolor sit amet consectetur adipisicing elit. Quasi non necessitatibus molestiae beatae delectus id numquam, eos inventore exercitationem pariatur sunt reiciendis porro voluptatibus ab quaerat saepe maiores illo quo?

# Tipografías Tamaño



# Propiedades de fuente

## font-size

Es una **propiedad que hace referencia al tamaño de la tipografía**.

Los valores nos servirán incluso para otras propiedades como **width, height o cualquier propiedad que involucre medidas**.

Se pueden utilizar **medidas de longitud relativas**, por ejemplo:

- rem
- vh
- em
- vw
- px
- vmin
- %
- vmax

**Nota:** sugerimos revisar la presentación medidas de longitud antes de continuar leyendo.

# Tipografías fluidas

Muchas veces tenemos tamaños de pantalla intermedios donde los **breakpoints** utilizados generan saltos donde la tipografía no logra el tamaño adecuado. Corregir esos problemas, genera mucho trabajo o múltiples**breakpoints** innecesarios en un proceso donde la idea es simplificar todo al máximo.

## **vh y vw**

Una alternativa es utilizar, entonces, **valores vinculados al ancho o alto de pantalla**, por ejemplo, **vh** o **vw**. De esta forma, nuestra **tipografía toma el tamaño de forma fluida**.

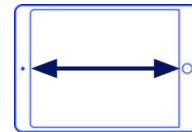
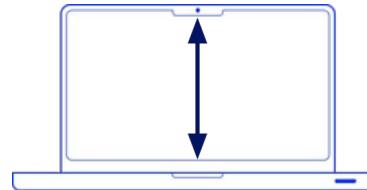
## vmin y vmax

Existen otras medidas vinculadas al cuerpo tipográfico en referencia a los tamaños de pantalla o *viewports*: **vmin** y **vmax** se utilizan para controlar el texto.

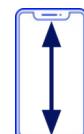
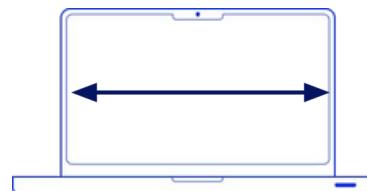
● **vmin es el radio más pequeño**, es decir,

cuando la altura del *viewport* es menor al ancho, 1vmin es igual a 1vh (recordemos que un 1vh es 1% del alto total del *viewport*). Si el ancho es menor que la altura, 1vmin será igual a un 1vw (1% del ancho total del *viewport*).

● **vmax sería lo opuesto: juega con los máximos.** Un ejemplo sería si el *viewport* es más alto que el ancho, 1vmax es igual a 1vh.



**vmin** utiliza el dato más pequeño entre la anchura o la altura de la ventana.



**vmax** utiliza el dato más grande entre la anchura o la altura de la ventana.

# Revisión

- Repasar los conceptos vistos de **familia de fuentes**.
- Revisar los conceptos de **tamaños de fuente**.
- Implementar estas nuevas propiedades.
- Ver todos los videos y materiales necesarios antes de continuar.

**Nota:** recomendamos seguir la guía de un diseñador con conocimiento tipográfico para la elección correcta de la familia tipográfica dependiendo el medio, pero si quieres conocer un poco más sobre tipografías y cómo usarlas, puedes visitar:

- [Cómo escoger la tipografía correcta en 5 pasos](#)
- [¿Cómo usar la tipografía en diseño web correctamente?](#)
- [Tipografía para Front Ends - Elena Ramírez \(LSE\)](#)

# Revisión

- Repasar los conceptos vistos de **familia de fuentes**.
- Revisar los conceptos de **tamaños de fuente**.
- Implementar estas nuevas propiedades.
- Ver todos los videos y materiales necesarios antes de continuar.



¡Sigamos  
trabajando!

