

Lesson:

Height, width, min & max height, min & max width, box-sizing



Topics Covered

- Text and font properties in CSS
- Line-height
- white-space
- Text-overflow
- The importance of typography in CSS
- Introduction to Google Fonts and how to use them with examples

Font and Text properties in CSS

font-family: This CSS property is used to provide a comma-separated list of font families. It sets the font face for the text content of an element. This property can hold multiple font names as a fallback system, i.e., if one font is unsupported in the browser, then others can be used. The different **font-family** is used for making attractive web pages.

Note: If the font name is more than one word, it must be in quotation marks, like: "Times New Roman". There are two types of font-family names in CSS, which are defined below

family-name: It is the name of the font-family such as **Courier, Arial, Times**, etc.

generic-family: It is the name of the generic family that includes five categories, which are "**serif**", "**sans-serif**", "**cursive**", "**fantasy**", and "**monospace**". It **should be placed at last in the list of the font family names**.

syntax

```
font-family: family-name|generic-family|initial|inherit;
```

Let's see the values of the font-family property.

family-name/generic-family: It is the list of font-family names and generic family names.

Initial: It is used to set the property to its default value.

Inherit: It is used to inherit the property from its parent element.

style.css:

```
h1 {
  font-family: Arial, sans-serif;
}
```

Browser output:

font-family

font-size: The font-size property in CSS is used to specify the height and size of the font. It affects the size of the text of an element. Its default value is medium and can be applied to every element.

The value of the property includes “xx-small”, “x-small”, “small”, “medium”, “large”, “x-large”, and “xx-large”.

The font-size can be relative or absolute.

We will see all the absolute and relative in upcomin section called “Introduction to CSS unit”.
For now lets see one example using most commonly used css unit “px”

style.css

```
h1 {  
    font-size: 36px;  
}
```

Browser Output-

font-size

When we set the size of text with pixels, then it provides us the full control over the size of the text.

font-weight: This property is used for setting the thickness and boldness of the font. It is used to define the weight of the text. The available weight depends on the font-family, which is used by the browser.

The value of the property includes **normal, lighter, bolder, bold, number, inherit, initial, unset;**

style.css

```
h1 {  
    font-weight: lighter  
}
```

Browser Output-

font-weight

font-style: This property specifies the style of the font, such as italic or normal.

Style.css

```
p {  
    font-style: italic;  
}
```

Browser output

font-style

text-decoration: This property specifies the decoration of the text, such as underlining or line-through.

style.css

```
.overline{  
    text-decoration: overline;  
}  
.lineThrough {  
    text-decoration: line-through;  
}  
.underline {  
    text-decoration: underline;  
}  
.underline overline {  
    text-decoration: underline overline;  
}
```

Browser Output:-

text-decoration: overline;

~~**text-decoration: line-through;**~~

text-decoration: underline;

text-decoration: underline overline;

text-align: The **text-align** property in CSS is used for the alignment of text. This **CSS** property is used to set the horizontal alignment of a table-cell box or the block element.

The value of the property includes **justify, center, right, left, initial, and inherit**.

The value **justify** stretches the content of the element to display the equal width of every line and generally used in magazines and newspapers.

style.css

```
.center {
    text-align: center;
}

.left {
    text-align: left;
}

.right {
    text-align: right;
}

.justify {
    text-align: justify;
}
```

Browser Output:-

text-align: center;

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,

text-align: left;

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,

text-align: right;

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,

text-align: justify;

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,

Line height

The **CSS line-height property** is used to define the minimal height of line boxes within the element. It sets the differences between two lines of your content. It defines the amount of space above and below inline elements. It allows you to set the height of a line of independently from the font size.

There are some property values that are used with **CSS** line-height property.

normal: This is a **default value**, it specifies a normal line-height

number: It specifies a number that is multiplied by the current font size to set the line height.

length: it is used to set the line height in px, pt, cm, etc.

Percentage %: It specifies the line height in percent of the current font.

Example:

html.css

```
<h2>line-height: normal (default);</h2>
<div class="normal">
    This is a paragraph with a standard line-height.<br />
    The standard line height in most browsers is about 110% to 120%.
</div>

<h2>line-height: 1.6 (recommended);</h2>
<div class="number">
    This is a paragraph with the recommended line-height.<br />
    The line height is here set to 1.6. This is a unitless value;<br />
    meaning that the line height will be relative to the font size.
</div>

<h2>line-height: 10px;</h2>
<div class="length">
    This is a paragraph with a smaller line-height.<br />
    The line height is here set to 10px.
</div>

<h2>line-height: 200%;</h2>
<div class="percentage">
    This is a paragraph with a bigger line-height.<br />
    The line height is here set to 200%.
</div>
```

Style.css

```
.normal {  
    line-height: normal;  
}  
.number {  
    line-height: 3.4;  
}  
.length {  
    line-height: 10px;  
}  
  
.percentage {  
    line-height: 50%;  
}
```

Browser output:

line-height: normal (default):

This is a paragraph with a standard line-height.

The standard line height in most browsers is about 110% to 120%.

line-height: 1.6 (recommended):

This is a paragraph with the recommended line-height.

The line height is here set to 1.6. This is a unitless value;

meaning that the line height will be relative to the font size.

line-height: 10px:

This is a paragraph with a smaller line-height.

The line height is here set to 10px.

line-height: 200%:

This is a paragraph with a bigger line-height.

The line height is here set to 200%.

white-space

The whitespace CSS property is used to specify how content in an element should be displayed. It is used to handle white spaces inside an element.

There are some property values that are used with CSS white-space property that you should know.

normal: This is a default value. In this, the text is wrapped when necessary, and sequences of white space will collapse into a single whitespace.

nowrap: Sequences of white spaces will collapse into a single whitespace. In this value, the text will never wrap to the next line and only break when
 tag is used.

pre: Whitespace is preserved by the browser. It acts like HTML <pre> tag. The text will only wrap on line breaks.

Examples

Index.html

```
<h2>white-space: normal:</h2>
<p class="narmal">
    Lorem ipsum, dolor sit amet consectetur adipisicing elit. Fugiat
    perspiciatis sint voluptatibus blanditiis incident, eaque quisquam optio
    voluptate, excepturi adipisci enim officiis quae, quasi mollitia iusto
    similius quidem consectetur quia.
</p>
<h2>white-space: nowrap:</h2>
<p class="nowrap">
    Lorem ipsum, dolor sit amet consectetur adipisicing elit. Fugiat
    perspiciatis sint voluptatibus blanditiis incident, eaque quisquam optio
    voluptate, excepturi adipisci enim officiis quae, quasi mollitia iusto
    similius quidem consectetur quia.
</p>
<h2>white-space: pre:</h2>
<p class="pre">
    Lorem ipsum, dolor sit amet consectetur adipisicing elit. Fugiat
    perspiciatis sint voluptatibus blanditiis incident, eaque quisquam optio
    voluptate, excepturi adipisci enim officiis quae, quasi mollitia iusto
    similius quidem consectetur quia.
</p>
```

style.css

```
.normal {
    white-space: normal;
}
.nowrap {
    white-space: nowrap;
}
.pre {
    white-space: pre;
}
```

Browser output:

white-space: normal;

 Lorem ipsum, dolor sit amet consectetur adipisicing elit. Fugiat perspiciatis sint voluptatibus blanditiis incident, eaque quisquam optio voluptate, excepturi adipisci enim officiis quae, quasi mollitia iusto similique quidem consectetur quia.

white-space: nowrap;

 Lorem ipsum, dolor sit amet consectetur adipisicing elit. Fugiat perspiciatis si

white-space: pre;

 Lorem ipsum, dolor sit amet consectetur adipisicing elit. Fugiat perspiciatis sint voluptatibus blanditiis incident, eaque quisquam optio voluptate, excepturi adipisci enim officiis quae, quasi mollitia iusto similique quidem consectetur quia.

Text-overflow

This property specifies the representation of overflowed text, which is not visible to the user. It signals the user about the content that is not visible. This property helps us to decide whether the text should be clipped, show some dots (ellipsis), or display a custom string.

This property does not work on its own. We have to use **white-space: nowrap;** and **overflow: hidden;** with this property.

Note: The overflow property will be discussed in the upcoming section.

Syntax-

```
text-overflow : clip  
text-overflow : ellipsis
```

Clip : The **default value** of the text-overflow property is a **clip**, **which clips any text that overflows its container**. This means that any text that extends beyond the boundaries of the container will be hidden from view, and may be truncated (cut off) in the middle of a character.

ellipsis: This value displays an ellipsis or three dots (...) to display the trimmed text. It is displayed in the area, reducing the amount of text.

Examples:

Index.html

```
<h2>text-overflow: clip;</h2>
<div class="clip">
  Lorem ipsum dolor, sit amet consectetur adipisicing elit. Sint, ipsum?
</div>
<h2>text-overflow: ellipsis;</h2>
<div class="ellipsis">
  Lorem ipsum, dolor sit amet consectetur adipisicing elit. Et, dolore?
</div>
```

style.css:

```
div {
  height: 30px;
  width: 250px;
  border: 2px solid black;
}
.clip {
  white-space: nowrap;
  overflow: hidden;
  text-overflow: clip;
}
.ellipsis {
  white-space: nowrap;
  overflow: hidden;
  text-overflow: ellipsis;
}
```

Browser output:

text-overflow: clip;

Lorem ipsum dolor, sit amet consectetur

text-overflow: ellipsis;

Lorem ipsum, dolor sit amet consec...

The importance of typography in CSS

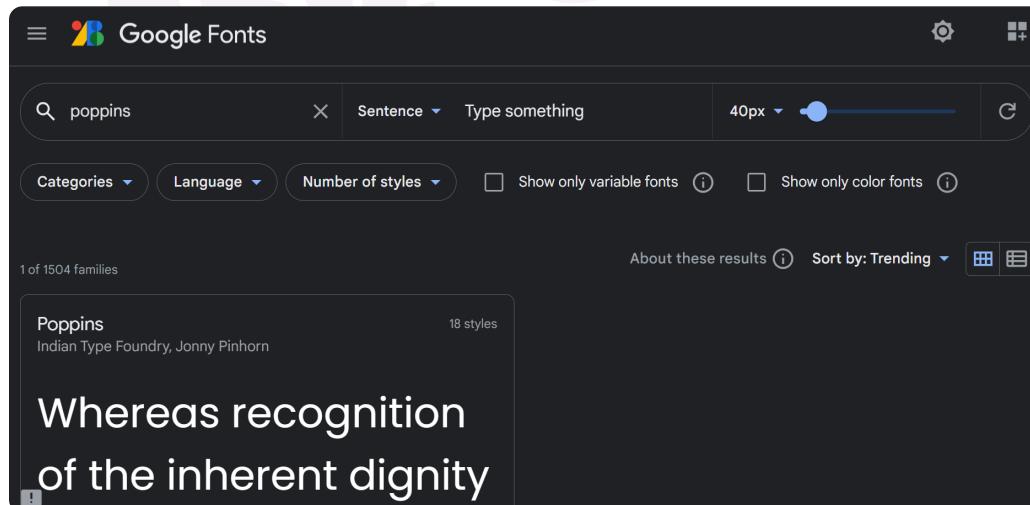
Typography plays a vital role in determining the look and feel of a website. Here are some of the reasons why typography is important in CSS:

- **Legibility:** ensures that the text on a website is **easy to read** and **understand**. It takes into consideration factors such as **font size**, **font weight**, **line spacing**, and **letter spacing** to make sure that the **text is clear or clear enough to read**.
- **Readability:** Good typography enhances the readability of a website by ensuring that the **text is organised, consistent**, and **easy to follow**.
- **Branding:** helps establish a website's brand identity by using fonts that are unique and recognizable. By choosing fonts that align with the website's brand, typography can help **reinforce its message** and create a **memorable user experience**.
- **Aesthetics:** Typography is an important aspect of web design aesthetics. It can make a website look **more professional, polished**, and **visually appealing**.
- **visual hierarchy:** helps to create a visual hierarchy on a website by using **different font sizes, weights**, and **styles to guide the user's attention**.

Introduction to Google Fonts and how to use them with examples

Google Fonts is a free service offered by Google that provides a collection of web fonts that can be **easily embedded into a website**. These fonts are optimized for web use and can be easily incorporated into CSS code using a few lines of code. Here's how to use Google Fonts in your CSS code:

Go to the Google Fonts website [Link](#) and browse the available fonts. Once you've found a font you want to use, select it.



The screenshot shows the Google Fonts interface. A search bar at the top contains the text "poppins". Below the search bar are several filter options: "Sentence" dropdown set to "Type something", a font size slider set to "40px", and checkboxes for "Show only variable fonts" and "Show only color fonts", both of which are unchecked. Further down, there are buttons for "Categories", "Language", and "Number of styles". At the bottom of the search area, it says "1 of 1504 families". The main content area displays the "Poppins" font family, which is described as having "18 styles" and being created by "Indian Type Foundry, Jonny Pinhorn". A preview window shows a sentence in the Poppins font: "Whereas recognition of the inherent dignity".

Once you've chosen a font, you will see a collection of the same fonts with different styles. Add the font to your collection, and choose how you want to embed it by adding a link in HTML or importing the fonts in a CSS file.

Now let's use Google Fonts.
Before applying Google Fonts



google fonts

First way to apply Google Fonts:

Copy the link and add it to your HTML file in <head>, and add a CSS rule in style.css .

To embed a font, copy the code into the <head> of your html

<link> @import

```
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Poppins:wght@200&display=swap" rel="stylesheet">
```

CSS rules to specify families

```
font-family: 'Poppins', sans-serif;
```

index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="preconnect" href="https://fonts.googleapis.com">
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
    <link href="https://fonts.googleapis.com/css2?family=Poppins:wght@200&display=swap" rel="stylesheet">
    <title>Document</title>
  </head>
  <body>
    <h1>google fonts</h1>
  </body>
</html>
```

Style.css

```
h1{  
font-family: "Poppins", sans-serif;  
}
```

Output after applying font

google
fonts

Second way:

Applying Google Fonts by importing the fonts

To embed a font, copy the code into the
<head> of your html

- <link> @import

```
<style>  
@import url('https://fonts.googleapis.com/css2?family=Poppins:wght@200&display=swap');  
</style>
```



CSS rules to specify families

```
font-family: 'Poppins', sans-serif;
```



index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
  </head>
  <body>
    <h1>google fonts</h1>
  </body>
</html>
```

Style.css

```
// Importing Google Fonts

@import url("https://fonts.googleapis.com/css2?
family=Poppins:wght@200&display=swap");

h1 {
  font-family: "Poppins", sans-serif;
}
```

Output after applying font

google
fonts