

CSS Font, Text, and Image Properties – Detailed Guide

This document is a **beginner-friendly yet detailed guide** to the most useful **CSS font, text, image, and background image properties**. It is designed for **practice, revision, and real-world usage**.

1. Font Properties in CSS

Font properties control **how text looks** (typeface, size, weight, and style).

Common Font Properties (Definition & Usage)

Property	Definition	Usage Example	Common Values
<code>font-family</code>	Specifies the font type for text	<code>font-family: Arial, sans-serif;</code>	Arial, Roboto, serif, sans-serif
<code>font-size</code>	Controls the size of the text	<code>font-size: 18px;</code>	px, em, rem, %
<code>font-weight</code>	Defines thickness of text	<code>font-weight: bold;</code>	100–900, normal, bold
<code>font-style</code>	Changes text style	<code>font-style: italic;</code>	normal, italic, oblique
<code>font-variant</code>	Displays text in small caps	<code>font-variant: small-caps;</code>	normal, small-caps
<code>line-height</code>	Space between lines of text	<code>line-height: 1.6;</code>	number, px, %

Example

```
p {  
  font-family: 'Roboto', sans-serif;  
  font-size: 16px;  
  font-weight: 400;  
  line-height: 1.5;  
}
```

2. Text Properties in CSS

Text properties control **alignment, spacing, decoration, and color**.

Text Properties (Definition & Usage)

Property	Definition	Usage Example	Common Values
color	Sets text color	color: #333;	hex, rgb, color name
text-align	Aligns text	text-align: center;	left, right, center, justify
text-transform	Changes text case	text-transform: uppercase;	uppercase, lowercase, capitalize
text-decoration	Adds decoration to text	text-decoration: underline;	none, underline, line-through
letter-spacing	Space between letters	letter-spacing: 2px;	px, em
word-spacing	Space between words	word-spacing: 5px;	px, em
text-shadow	Adds shadow to text	text-shadow: 2px 2px 5px gray;	x y blur color
white-space	Controls text wrapping	white-space: nowrap;	normal, nowrap

Example

```
h1 {
  color: navy;
  text-align: center;
  text-transform: uppercase;
  letter-spacing: 2px;
}
```

3. Image Properties in CSS

Image properties control **size, shape, and visual effects**.

Image Properties (Definition & Usage)

Property	Definition	Usage Example	Common Values
width	Sets image width	width: 300px;	px, %, auto
height	Sets image height	height: auto;	px, %, auto
max-width	Prevents overflow	max-width: 100%;	%, px

Property	Definition	Usage Example	Common Values
border-radius	Rounds image corners	border-radius: 50%;	px, %
object-fit	Fits image inside box	object-fit: cover;	cover, contain
opacity	Controls transparency	opacity: 0.7;	0-1
box-shadow	Adds shadow	box-shadow: 0 4px 8px gray;	x y blur color

Example

```
img {
  width: 200px;
  border-radius: 10px;
  object-fit: cover;
}
```

4. Background Image Properties

Used to apply images to **containers (div, section, body)**.

Background Properties (Definition & Usage)

Property	Definition	Usage Example	Common Values
background-image	Sets background image	background-image: url('img.jpg');	url()
background-size	Controls image size	background-size: cover;	cover, contain
background-position	Positions image	background-position: center;	top, center
background-repeat	Repeats image	background-repeat: no-repeat;	repeat, no-repeat
background-attachment	Scroll behavior	background-attachment: fixed;	scroll, fixed

Example

```
.hero {
  background-image: url('banner.jpg');
```

```
background-size: cover;  
background-position: center;  
height: 300px;  
}
```

5. Real-World Practice Example

```
.card {  
  width: 300px;  
  padding: 20px;  
  font-family: Arial, sans-serif;  
  text-align: center;  
  box-shadow: 0 4px 10px rgba(0,0,0,0.2);  
}  
  
.card img {  
  width: 100%;  
  border-radius: 10px;  
}  
  
.card h2 {  
  text-transform: uppercase;  
  letter-spacing: 1px;  
}
```

6. How to Practice Effectively

- Change values and observe results in browser
- Combine font + text properties
- Practice responsive images using `%` and `max-width`
- Use `rem` instead of `px` for better scaling

7. CSS Units Explained (px, em, rem, %)

CSS units control **size and responsiveness**. Choosing the right unit is important for scalable layouts.

Common CSS Units (Two-Line Explanation)

Unit	Explanation	Example
<code>px</code>	Fixed unit that does not scale with screen or font size. Best for precise layouts.	<code>font-size: 16px;</code>

Unit	Explanation	Example
<code>em</code>	Relative to the parent element's font size. Scales with nesting.	<code>font-size: 1.5em;</code>
<code>rem</code>	Relative to the root (<code>html</code>) font size. Best for responsive design.	<code>font-size: 1rem;</code>
<code>%</code>	Relative to parent element size. Commonly used for width and height.	<code>width: 50%;</code>

Usage Tip

- Use `rem` for fonts
- Use `%` for layouts
- Avoid too much nesting with `em`

8. CSS Practice Exercises (With Answers)

Exercise 1: Font Styling

Task: Make text bold, 18px size, and use Arial.

Answer:

```
p {
  font-family: Arial, sans-serif;
  font-size: 18px;
  font-weight: bold;
}
```

Exercise 2: Text Alignment

Task: Center align text and make it uppercase.

Answer:

```
h1 {
  text-align: center;
  text-transform: uppercase;
}
```

Exercise 3: Responsive Image

Task: Make image responsive and rounded.

Answer:

```
img {  
  max-width: 100%;  
  border-radius: 10px;  
}
```

Exercise 4: Background Image

Task: Add a background image that covers the container.

Answer:

```
div {  
  background-image: url('img.jpg');  
  background-size: cover;  
  background-position: center;  
}
```

9. Interview-Oriented CSS Questions (Two-Line Answers)

1. What is the difference between `class` and `id`?

A class can be reused multiple times, while an id is unique and used once per page.

2. Difference between `display: none` and `visibility: hidden`?

`display: none` removes the element from layout, `visibility: hidden` keeps space.

3. What is `rem` and why is it preferred?

`rem` is relative to root font size and provides consistent scaling.

4. How do you make images responsive?

Use `max-width: 100%` and `height: auto`.

5. Difference between `inline` and `block` elements?

Inline elements don't start new lines; block elements do.

6. What does `object-fit: cover` do?

It fills the container while maintaining image aspect ratio.

End of Document

This guide now includes **units, exercises, and interview prep** for complete CSS fundamentals.