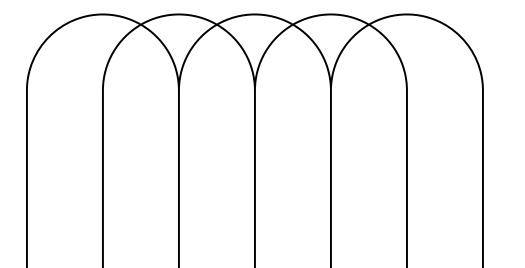
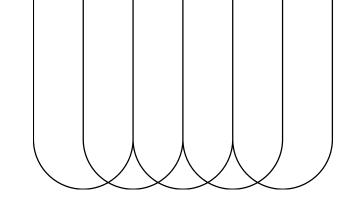




CSS Sizes, Units & Styling Images







CSS Sizes

CSS provides height and width properties to control size of elements, impacting layout, responsiveness, and overall design consistency.

Height & Width

The height and width properties control the size of an element. They can be set using different units depending on the desired level of flexibility and responsiveness.

• Using CSS Units (px, rem, em, %, vh, vw)

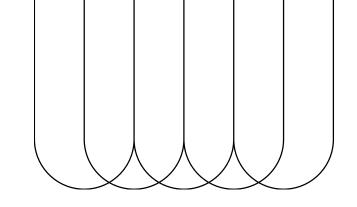
Example:

```
.box {
    width : 200px;
    height: 30vh;
}
```

• Using special values (auto, min-content, max-content, fit-content)

Example:

```
.box {
   width : fit-content;
   height: 30vh;
}
```





CSS UNITS

CSS uses a variety of units to specify the size of elements, their margins, padding, fonts, and more. These units can be broadly categorized into **absolute units** and **relative units**. Understanding these units is crucial for creating responsive and scalable web designs.

Absolute Units

Absolute units are fixed and do not change based on the viewport or parent elements. They are useful when a consistent size is required, but they may not be ideal for responsive designs.

px (Pixels): The most commonly used unit. It represents a single dot on the screen.

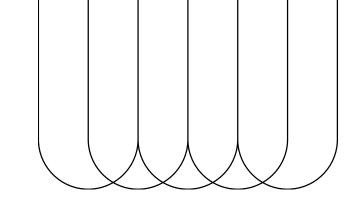
Examples:

p { font-size: 16px }

cm, mm & in: These units are useful for print styles but rarely used in web design.

Examples:

div { width: 10cm }





CSS UNITS

Absolute Units

Relative units adjust based on the context of their usage, making them more flexible and ideal for responsive designs.

% (Percentage): Relative to parent element

Examples:

div { width: 50% }

em: Relative to the font-size of the parent element.

Examples:

p span { font-size: 1.25em }

rem (Root em): Relative to the font-size of the root element (usually HTML).

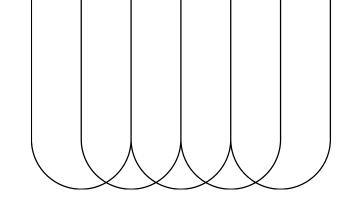
Examples:

html { font-size: 16px }
p {font-size: 2rem } /* 32px */

vw, vh: Relative to the viewport size.

Examples:

div { width: 50vw; } /* 50% of the viewport width */





CSS UNITS

Time Units

Time units in CSS are used primarily for animations and transitions.

ms & s: Specifies time in milliseconds (1s = 1000ms) and second.

Examples:

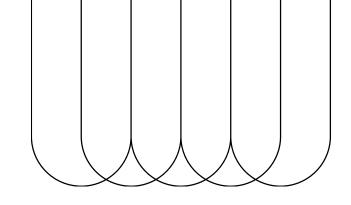
a { transition: all 400ms ease-in-out; }
div { animation-duration: 2s }

Choosing Right Units

Choosing the right unit depends on the context:

- Use **px** for precise control over elements.
- Use **em/rem** for scalable/responsive typography.
- Use **vw/vh** for full-screen elements.
- Use % for elements inside containers.
- Use **s/ms** for animations and transitions.

There are many more units that are used in css, you can explore them on your own on w3c documentation on <u>CSS</u> <u>Values and Units</u>





Styling Images

Rounded Images

```
img { border-radius: 10px; }
.img-circle{ border-radius: 50%; }
```

Thumbnail Images

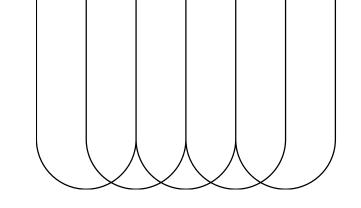
```
img { border: 1px solid #e0e0e0; padding: 5px; border-radius: 5px}
```

Responsive Images

```
.img-responsive { max-width: 100%; height: auto; }
```

Object Fit

```
.object-fit-cover { object-fit: cover; }
.object-fit-contain { object-fit: contain ;}
.object-fit-fill { object-fit: fill ;}
```





THANK YOU

PHONE NUMBER

(+91) 778 899 2897

WEBSITE

www.indixpert.com

