**Information about some important properties gathered from various websites:**

**CSS text-rendering**

L

Text-rendering is a property that allows you to choose what to optimize when rendering text. It sends the information to the rendering engine about what to optimize for while rendering text.

It’s actually not a CSS property and is not defined in any CSS standard. It’s an SVG property but Gecko and Webkit browsers allow you to apply this property to the HTML elements. In doing so compromises among other aspects like speed, legibility(Clarity), and geometric precision.

optimizeLegibility: Browser prioritizes more legibility compared to rendering speed and geometric precision while drawing text.  It enables kerning and optional ligatures.

100vh represents the full height of the viewport, regardless of the content on the page. When you use 100vh for an element's height, it will take up the entire vertical height of the user's screen, and it won't be affected by the content or other elements on the page. 100% is a relative unit of measurement.

The background-attachment property sets whether a background image scrolls with the rest of the page, or is fixed.

|  |  |
| --- | --- |
| **Value** | **Description** |
| scroll | The background image will scroll with the page. This is default |
| fixed | The background image will not scroll with the page |
| local | The background image will scroll with the element's contents |
| initial | Sets this property to its default value. [Read about initial](https://www.w3schools.com/cssref/css_initial.php) |
| inherit | Inherits this property from its parent element. [Read about inherit](https://www.w3schools.com/cssref/css_inherit.php) |

The background-size property specifies the size of the background images.

|  |  |
| --- | --- |
| cover | Resize the background image to cover the entire container, even if it has to stretch the image or cut a little bit off one of the edges |

The background-position property sets the starting position of a background image.

The content property is used with the [::before](https://www.w3schools.com/cssref/sel_before.php) and [::after](https://www.w3schools.com/cssref/sel_after.php) pseudo-elements, to insert generated content.

Ex:

ul {  
  list-style: none; /\* Remove HTML bullets \*/  
  padding: 0;  
  margin: 0;  
}  
  
li {  
  padding-left: 16px;  
}  
  
li::before {  
  content: "•"; /\* Insert content that looks like bullets \*/  
  padding-right: 8px;  
  color: blue; /\* Or a color you prefer \*/  
}

The <figure> tag specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.

While the content of the <figure> element is related to the main flow, its position is independent of the main flow, and if removed it should not affect the flow of the document.

The visibility property specifies whether or not an element is visible.

**Tip:** Hidden elements take up space on the page. Use the [display](https://www.w3schools.com/cssref/pr_class_display.php) property to both hide and remove an element from the document layout!

The clear property controls the flow next to floated elements.

The clear property specifies what should happen with the element that is next to a floating element.

The <blockquote> tag specifies a section that is quoted from another source.

Browsers usually indent <blockquote> elements (look at example below to see how to remove the indentation).

The <cite> tag defines the title of a creative work (e.g. a book, a poem, a song, a movie, a painting, a sculpture, etc.).

**Note:** A person's name is not the title of a work.

The text in the <cite> element usually renders in *italic*.

The vertical-align property sets the vertical alignment of an element.

|  |  |
| --- | --- |
| middle | The element is placed in the middle of the parent element |

In HTML, the content '\201C' can be used to display the left double quotation mark by using its Unicode entity.

# Adding Animations on Scroll with HTML, CSS and AOS.js

AOS.js (Animation on Scroll) is an animation-providing JavaScript library that makes it easier to add tons of animations by simply changing the name of the class in the **div** tag in which you want to add an animation. While there are different animation JavaScript libraries out there, AOS.js is probably the simplest of them.

The first category of animations that we will explore are the fading one's. Before we do that, we first need to make sure that **aos.css** and **aos.js** are available to us in our code, and we can get them through the help of CDN links.

You just need to paste the below code snippet at the end of the <head> tag in your HTML code.

<link rel="stylesheet" href="https://unpkg.com/aos@next/dist/aos.css" />

The above code snippet will help in getting the **css** file and in order to get the **js** file, we need to paste the CDN code snippet shown below at the end of the body tag in our HTML code.

<script src="https://unpkg.com/aos@next/dist/aos.js"></script>

<script>

AOS.init();

</script>

Once we added both the above code snippets in our HTML code, we are ready to make use of AOS in our code.

|  |  |  |  |
| --- | --- | --- | --- |
| *data-aos-duration* | \*Duration of animation (ms) | 600 | 400 |

## **Fade Animation Using AOS.js**

The Fade animation simulates the fading behaviour and in total, there are 8 different animations that can be achieved with it. These are −

* fade-up
* fade-down
* fade-left
* fade-right
* fade-up-left
* fade-up-right
* fade-down-left
* fade-down-left

The below code snippet is the only part in which we will make changes for all the above fade animations.

<div id="main">

<div data-aos="fade-up">Something up!</div>

</div>

## **Flip Animation Using AOS.js**

Flip animation simulates the flipping behaviour and in total there are 4 different animations that can be achieved with it. These are −

* flip-up
* flip-down
* flip-left
* flip-right

## **Zoom Animation Using AOS.js**

The Zoom animation simulates the zooming behaviour and in total there are 8 different animations that can be achieved with it. These are −

* zoom-in
* zoom-in-up
* zoom-in-down
* zoom-in-left
* zoom-in-right
* zoom-out
* zoom-out-up
* zoom-out-down
* zoom-out-left
* zoom-out-right