**CSS**

Cascading Style Sheets is used for giving design to the webpage or changing the layout.

CSS is a language that describes the style of an HTML document.

Style sheets enable you to build consistent, transportable, and well-defined style templates.

**SGML (Standard Generalized Markup Language) is the origin of CSS.**

**Advantages of using CSS**

* **CSS saves time** − You can write CSS once and then **reuse** same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.
* **Pages load faster** − If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all.
* **Easy maintenance** − To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.
* **Multiple Device Compatibility**
* **Global web standards** − Now HTML attributes are being deprecated and it is being recommended to use CSS. So its a good idea to start using CSS in all the HTML pages to make them compatible to future browsers.
* **Offline Browsing** − CSS can store web applications locally with the help of an offline cache. Using of this, we can view offline websites. The cache also ensures faster loading and better overall performance of the website.
* **Platform Independence** − The Script offer consistent platform independence and can support latest browsers as well.

**CSS selectors (**<https://www.w3schools.com/cssref/css_selectors.asp>**)**

For selecting every element

universal selector “\*{ }”

**Descendant Selector:**

Suppose you want to apply a style rule to a particular element only when it lies inside a particular element.

A descendant selector in CSS is any selector with white space between two selectors without a combinator. Descendant means anywhere nested within it in the DOM tree. Could be a direct child, could be five levels deep, it is still a descendant.

Ex. ul li { }

header h2 { }

Can you make a class selector particular to an element type?

h1.black {

color: #000000;

}

What is a child selector?

body > p {

color: #000000;

}

This rule will render all the paragraphs in black if they are direct child of <body> element.

**What is an attribute selector?**

You can also apply styles to HTML elements with particular attributes.

input[type = "text"]{

color: #000000;

}

How to select all paragraph elements with a lang attribute?

p[lang] : Selects all paragraph elements with a lang attribute.

How to select all paragraph elements whose lang attribute has a value of exactly "fr"?

p[lang="fr"]

How to select all paragraph elements whose lang attribute contains values that are exactly "en", or begin with "en-"?

p[lang|="en"]

What is the purpose of em measurement unit?

em − A relative measurement for the height of a font in em spaces. Because an em unit is equivalent to the size of a given font, if you assign a font to 12pt, each "em" unit would be 12pt; thus, 2em would be 24pt.

What is the purpose of pt measurement unit?

pt − Defines a measurement in points. A point is defined as 1/72nd of an inch.

What are browser safe colors?

There is the list of 216 colors which are supposed to be most safe and computer independent colors. These colors vary from hexa code 000000 to FFFFFF. These colors are safe to use because they ensure that all computers would display the colors correctly when running a 256 color palette.

Which property is used to control the scrolling of an image in the background?

The background-attachment

Which property is used to change the face of a font?

The font-family

**Which property is used to create a small-caps effect?**

The font-variant property is used to create a small-caps effect.

The font-style property is used to make a font italic or oblique.

**Which property is used to set the color of a text?**

The color property is used to set the color of a text.

**Which property is used to set the text direction?**

The direction property is used to set the text direction.

**Which property is used to indent the text of a paragraph?**

The text-indent

**Which property is used to capitalize text or convert text to uppercase or lowercase letters?**

The text-transform

Which property is used to control the flow and formatting of text?

The white-space

Which property of a hyperlink signifies unvisited hyperlinks?

The :link signifies unvisited hyperlinks.

Which property of a hyperlink signifies visited hyperlinks?

The :visited signifies visited hyperlinks.

Which property of a hyperlink signifies an element on which the user is currently clicking?

The :active

**Which property of a table specifies whether the browser should control the appearance of the adjacent borders?**

The **border-collapse** specifies whether the browser should control the appearance of the adjacent borders that touch each other or whether each cell should maintain its style.

**Which property of a table specifies the width that should appear between table cells?**

The border-spacing specifies the width that should appear between table cells.

**Which property of a table specifies whether the border should be shown if a cell is empty?**

The empty-cells

**Which property allows you to control the shape or appearance of the marker of a list?**

The list-style-type

Which property specifies whether a long point that wraps to a second line should align with the first line or start underneath the start of the marker of a list?

The list-style-position specifies

Which property specifies an image rather than a bullet point or number for the marker of a list?

The list-style-image

Which value of cursor property changes the cursor based on context area it is over?

auto

Can you set an image to be shown as cursor?

Yes! set the url as the source of a cursor image file.

**What is a sprite? How is it applied using CSS? What is the benefit?**

- A image sprite is a collection of images put into one single image.

- Using css positioning you can show and hide different parts of the sprite depending on what you need.

**What is the purpose of the z-index**

The z-index helps specify the stack order of positioned elements that may overlap one another. The z-index default value is zero, and can take on either a positive or negative number.

**disadvantages of External Style Sheets**

* The disadvantages are that it may affect loading time in some situations. It may also not be practical if there are not enough styling conditions to justify an external sheet.

**What is a Box Model?**

Every Element on a Page is a Rectangular Box and may have Width, Height, Padding, Borders, and Margins. Every section of the box model relates to a CSS property: width, height, padding, border, and margin.

**What are Pseudo-classes?**

( <https://www.w3schools.com/css/css_pseudo_classes.asp>)

A pseudo-class is used to define a special state of an element.

selector:pseudo-class {  
    property:value;  
}

**z-index:**

The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.

**Enlist Disadvantages of External Style Sheets:**  
1. It requires more data to download to import style information for each HTML document from a relative .CSS file.  
2. Rendering an HTML document is not possible if the Style Sheet is not loaded properly.  
3. Not practical for small style definitions.

**Explain Image Sprites in CSS.**  
When multiple images or a set of images is combined into a single image, it is known as an Image Sprite. It is better to use Sprite Images as loading every image on a Web page takes time and this reduces the page load time

**A CSS rule-set consists of a selector and a declaration block.**



**What are the advantages of External Style Sheets?**

* You can create classes for reusing it in many documents.
* By using it, you can control the styles of multiple documents from one file.
* Easily transportable.

**What is RWD?**

RWD stands for Responsive Web Design. This technique is used to display the designed page perfectly on every screen size and device. For example: Mobile, Tablet, desktop, laptop etc. You don't need to create a different page for each device.

<https://www.w3schools.com/cssref/css3_pr_mediaquery.asp>

**What is the float property of CSS?**

The CSS float property is used to move the image to the right or left along with the texts to be wrapped around it.

**What is tweening?**

It is the process of generating intermediate frames between two images.

It gives the impression that the first image has smoothly evolved into the second one.

It is an important method used in all types of animations.

In CSS3, Transforms (matrix, translate, rotate, scale etc.) module can be used to achieve tweening.

**ICONS:**

The simplest way to add an icon to your HTML page, is with an icon library, such as Font Awesome.

Add the name of the specified icon class to any inline HTML element (like <i> or <span>).

**CSS2 AND CSS3 DIFFERENCE**

The biggest difference between CSS2 and CSS3 is that CSS3 has been split up into different sections, called modules. ... Because each of the modules is being worked on individually, we have a much wider range of browser support for CSS3 modules.

**LIMITATIONS OF CSS**

* Pseudo-class not controlled by dynamic behavior.
* Pseudo-class not affect outside the parent element.

**Demerits of Embedded Style Sheets:**

* Multiple documents cannot be controlled.

**Can more than one declaration be added in CSS?**

Yes, it can be achieved by using a semicolon.

**CONTEXTUAL SELECTOR:**

A contextual selector matches when an element is an arbitrary descendent of some ancestor element (i.e., it may be any generation below the ancestor element). A contextual selector is made up of two or more selectors separated by white space.

Ex. h1 em { color: blue }

**GROUPING SELECTOR:**

When element selectors share the same declarations, they may be grouped into comma-separated lists.

Ex.

h1 { font-family: helvetica }

h2 { font-family: helvetica }

is equivalent to:

h1, h2, h3 { font-family: helvetica }

**NESTING:**

Specifying a selector within a selector is called nesting.

Ex. h1 em { color: blue }

**Graceful degradation** is the ability of a computer, machine, electronic system or network to maintain limited functionality even when a large portion of it has been destroyed or rendered inoperative. The purpose of graceful degradation is to prevent catastrophic failure.

* Animation, transform, font, flex, position
* <https://www.w3.org/TR/css-variables-1/>

**AT-RULE**:

An at-rule is a CSS statement that instructs CSS how to behave. They begin with an at sign, '@' followed by an identifier and includes everything up to the next semicolon.

* @charset — Defines the character set used by the style sheet.
* @import — Tells the CSS engine to include an external style sheet.
* @namespace — Tells the CSS engine that all its content must be considered prefixed with an XML namespace

**Why is @import only at the top?**

@import is preferred only at the top, to avoid any overriding rules.

**It's true, they are both - or more precisely, they are "inline block" elements. This means that they flow inline like text, but also have a width and height like block elements.**

**What is progressive enhancement?**

**Progressive enhancement** is a strategy for web design that emphasizes core webpage content first. This strategy then progressively adds more nuanced and technically rigorous layers of presentation and features on top of the content as the end-user's browser/internet connection allow.

**@import:**

Imported style sheet permits you to import the files which are external or combine one style sheet with another. Import function gives the provision to combine many elements or functionality into one.

Ex. @import url(<http://www.xyz.css>);

Import the "mobstyle.css" style sheet ONLY if the media is screen and the viewport is maximum 768 pixels:

@import "mobstyle.css" screen and (max-width: 768px);

Import the "printstyle.css" style sheet ONLY if the media is print:

@import "printstyle.css" print;

**INLINE VS INLINE-BLOCK VS BLOCK**

* Compared to display: inline, the major difference is that display: inline-block allows to set a width and height on the element.
* Also, with display: inline-block, the top and bottom margins/paddings are respected, but with display: inline they are not.
* Compared to display: block, the major difference is that display: inline-block does not add a line-break after the element, so the element can sit next to other elements.

**Important Selectors:**

* div, p - Selects all <div> elements and all <p> elements
* div p - Selects all <p> elements that are anywhere inside a <div> element
* div > p - Selects all <p> elements where the immediate parent is a <div> element
* div + p - Selects the first <p> elements that are placed immediately after a <div> element
* div ~ p - Selects all <p> elements that are anywhere preceded by a <div> element

**CONTENT:**

The content property is used with the ::before and ::after pseudo-elements, to insert generated content.

Ex. a::after {  
    content: " (" attr(href) ")";  
}

**COUNTER-INCREMENT: (**<https://www.w3schools.com/cssref/pr_gen_counter-increment.asp>**)**

The counter-increment property increases or decreases the value of one or more CSS counters.

The counter-increment property is usually used together with the counter-reset property and the content property.

**MEDIA QUERIES:**

* **CSS2 Introduced Media Types**

The @media rule, introduced in CSS2, made it possible to define different style rules for different media types.

Examples: You could have one set of style rules for computer screens, one for printers, one for handheld devices, one for television-type devices, and so on.

* CSS3 Introduced Media Queries

Media queries in CSS3 extended the CSS2 media types idea: Instead of looking for a type of device, they look at the capability of the device.

Media queries can be used to check many things, such as:

* + width and height of the viewport
  + width and height of the device
  + orientation (is the tablet/phone in landscape or portrait mode?)
  + resolution
* @media not|only *mediatype*and(*expressions*) { *CSS-Code;*}
* Media query types:
  + All: Used for all media type devices
  + Print: Used for printers
  + Screen: Used for computer screens, tablets, smart-phones etc.
  + Speech: Used for screenreaders that "reads" the page out loud

**CSS PRE-PROCESSORS:**

A preprocessor is an abstraction layer built on top of CSS. They are CSS preprocessors. They are an abstraction layer on top of CSS. They are a special syntax/language that compile down into CSS. They make managing CSS easier, with things like variables and mixins to handle vendor prefixes (among other things).

**There are many ways to express units of length within CSS, but these are just some of the more common ones.**

* cm: centimeters
* em: a relative unit of measurement based on the size of a font
* in: inches
* mm: millimeters
* pc: pica, a unit of length equivalent to 12 points, or 1/6 of an inch
* pt: 1/72 of an inch
* px: a device-specific relative measurement equivalent to a certain number of pixels on a display

**What are CSS vendor prefixes? Can you name some of the more common ones that you’re familiar with?**

<https://developer.mozilla.org/en-US/docs/Glossary/Vendor_Prefix>

Depending on your project, you might be looking for a CSS developer who can take advantage of experimental non-standard features that are only available on certain platforms. Vendor prefixes are extensions to CSS standards that can be added to these features to prevent incompatibilities from arising when the standard is extended. CSS vendor prefixes for some common platforms are listed below.

* -webkit-: Android, Chrome, iOS, and Safari
* -moz-: Mozilla Firefox
* -ms-: Internet Explorer
* -o-: Opera

**How would you select all the PDF links in the code block below with a single line of code?**

<p><a href="default.asp" target="\_blank">This is a link</a></p>

<p><a href="mydocument.pdf" target="\_blank">This is a PDF</a></p>

<p><a href="default.asp" target="\_blank">This is a link</a></p>

<p><a href="mydocument.pdf" target="\_blank">This is a PDF</a></p>

Ans. a[href$=“.pdf”]

**CSS FLEXBOX: (**<https://www.w3schools.com/css/css3_flexbox.asp>**)**

The Flexible Box Layout Module, makes it easier to design flexible responsive layout structure without using float or positioning.

**What is The Viewport?**

The viewport is the user's visible area of a web page. The viewport varies with the device, and will be smaller on a mobile phone than on a computer screen.

**Setting The Viewport:**

HTML5 introduced a method to let web designers take control over the viewport, through the <meta> tag.

You should include the following <meta> viewport element in all your web pages:

<meta name="viewport" content="width=device-width, initial-scale=1.0">

A <meta> viewport element gives the browser instructions on how to control the page's dimensions and scaling.

The width=device-width part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).

The initial-scale=1.0 part sets the initial zoom level when the page is first loaded by the browser.

**If a new element place after a block element(width:50%) then also rest of the width is maintained by margin AND INLINE ELEMENT STARTS FROM A NEW line.**

**Ex**

<body>

<p style=”width:50%”>Display links as block elements:</p>

<a href="/html/default.asp" target="\_blank">HTML</a>

<a href="/css/default.asp" target="\_blank">CSS</a>

<a href="/js/default.asp" target="\_blank">JavaScript</a>

</body>

**Position: Default value of Position: static**There are five different position values:

* static
* relative
* fixed
* absolute
* sticky

**Static**: HTML elements are positioned static by default. Static positioned elements are not affected by the top, bottom, left, and right properties. An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page

**Relative:** An element with position: relative; is positioned relative to its normal position.

**Absolute:** An element with position: absolute; is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed).

Here positioned ancestor doesn’t include static, If an absolute positioned element has no positioned ancestors, it uses the document body, and moves along with page scrolling.

**Fixed**: An element with position: fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled. The top, right, bottom, and left properties are used to position the element.

**Sticky:** An element with position: sticky; is positioned based on the user's scroll position. A sticky element toggles between relative and fixed, depending on the scroll position. It is positioned relative until a given offset position is met in the viewport - then it "sticks" in place (like position:fixed).

* **Var declaration in css:**

<https://www.w3schools.com/css/css3_variables.asp>

<https://www.w3schools.com/css/css3_variables_javascript.asp>

* **Specificity:**

<https://www.w3schools.com/css/css_specificity.asp>

* **What is !important?**

The !important rule in CSS is used to add more importance to a property/value than normal. In fact, if you use the !important rule, it will override ALL previous styling rules even those with high specificity.

* **Box-sizing:**

<https://www.w3schools.com/css/css3_box-sizing.asp>

* <https://css-tricks.com/the-difference-between-nth-child-and-nth-of-type/>
* **Make a div in center:**

// .wrapper {

//   display: flex;

//   justify-content: center;

//   align-items: center;

//   width: 100%;

//   height: 100%;

//   .center {

//     width: 100px;

//     height: 100px;

//     background-color: aquamarine;

//   }

// }

// .wrapper {

//   position: relative;

//   width: 100%;

//   height: 100%;

//   .center {

//     position: absolute;

//     top: calc(50% - 50px);

//     left: calc(50% - 50px);

//     width: 100px;

//     height: 100px;

//     background-color: aquamarine;

//   }

// }

// .wrapper {

//   position: relative;

//   width: 100%;

//   height: 100%;

//   .center {

//     position: absolute;

//     left: 50%;

//     top: 50%;

//     transform: translate(-50%, -50%);

//     width: 100px;

//     height: 100px;

//     background-color: aquamarine;

//   }

// }

**IMPORTANT LINKS**

* <https://css-tricks.com/interview-questions-css/>
* <https://www.toptal.com/css/interview-questions>