# **INSTRUCTIONS**

* Answer as much as you can with your own knowledge
* Research any topics you are not familiar with
* Answer all questions in your own words. DO NOT copy-paste answers from other sources

# HTML5/CSS

**Estimated Time: 40 hour.**

1. What is ***doctype***?

It is the declaration that someone needs to put in the head of an HTML script for the machine to recognize what sort of document the script will be about.

1. What are the different positions in CSS?

The positions in CSS control the arrangement and location of elements on a web page. The way an element is placed is influenced by its position type, which can change its relationship to the parent container and other elements on the page.

* Static, is the default positioning, where elements are placed according to the normal flow of the web page and don't affect the placement of other elements.
* Relative, for which the elements are positioned in relation to their original position on the page. The "top," "bottom," "left," and "right" properties can be used to adjust their location.
* Absolute, in this position, lements are taken out of the normal flow and are positioned based on the closest positioned ancestor or the initial containing block if no ancestor is positioned.
* Fixed, here the elements are taken out of the normal flow and positioned relative to the window. They stay in the same place even when the page is scrolled.
* Sticky, the elements follow the normal flow of the page, but when a user scrolls, they become "fixed" and maintain their position as the user continues to scroll.

1. How would you create a menu in which each element takes the same portion of space from its container, and if you change its size, it still remains the same space for each one.

I would use flexbox for the parent element that will contain all the menu items, flexbox is the best, easiest and flexible tool for declaring items for menus in CSS; and for the children items, I will declare them all to be with the same flex rule, so that each one of them can have the same amount or portion of space to be used.

Also, I can think of display:grid as an alternative for the layout, and in the grid area where I need to create the menu, I would declare suitable values for margin, padding and border that each element can share as part of that grid (with a parent class similar for each, for example).

1. What cross browser issues have you run into and how did you deal with them?

The issues I have dealt with so far are mainly:

1. Sometimes my browser doesn’t display or show correctly the font family from the original CSS stylesheet, and thus changes it for another one.
2. Other times, the elements in the layout are displayed differently, with different sizes and in some cases, the margins and spaces around from one element to another change and are not shown the same way from one browser to another.
3. Overall, I see that most of the times, the desktop versions of the web pages don’t work correctly in mobile devices.

The way I have been learning to deal with them, is precisely check whether the feature being used for the issued elements is detected in the browser I am using at the moment. Fortunately, now these detectors also provide sometimes solutions or recommedations for replacing these features or giving fallbacks. And also, there are some generic practices someone needs to learn to use for overcome in the most majority of scenarios: for example, displaying flex elements is a good practice because it can be well supported even for browsers that cannot still run grid-like layed out elements.

1. What does the ***float*** property do?

The float property allows someone to treat an element as one which is floating in the page and move it and customize its style it in a rudimentary way, like for example when someone inserts a picture inside a program for processing text within lines and wants to move it around the document, or display it in one way or another in a web page, in this case.

* + How do you clear a floated element?

Clearing floated elements here means, a rule that restricts them to be “floating” or being shown in a certain position of the web page. The correct rule syntaxis is type down the property “clear- ...” and the possible values can be: “left”, “right”, “both”, and “none”.

1. How do you change the direction of HTML text?

In its style sheet instructions, you declare a property embedded to the HTML text named “text-orientation”, and it must be together one of these values: “upright”, “sideways”, “sideways-right”, among others. Default is “mixed”. For right-to-left writing, the property is just “direction”, and possible values are “rtl” or “ltr”.

1. Explain how the new HTML5 features work:
   * semantic tags

Semantic tags are tags that HTML5 introduced for actually working in parts of a website that can be related to the name of these tags: header, section, article, etc., so that the coder can make the input of text according to a predefined tag that needs to be used for it.

* + Attributes

Attributes are in-line internal declarations someone can input into a tag that can further describe what the tag will be showing like its content, style and inheritance, like classes and ids.

* + Connectivity

HTML5 provides a set of APIs (Application Programming Interfaces) that allow developers to create online applications that can be cached locally, receive real-time updates, communicate in real-time, and persist data even when offline. This is done through the use of the Application Cache API, Server-Sent Events API, WebSockets API, and Web Storage API.

* + work offline

By using some APIs, HTML5 enables web developers to create applications that can work offline and online, providing a seamless and consistent user experience regardless of the network connection.

* + Storage

HTML5 storage refers to the ability of web applications to save data on the user's device, even if they close their browser or move to a different website. There are two ways to store data in HTML5: local storage and session storage.

Local storage saves data on the user's device for a long period of time, even if they close their browser or turn off their device.

Session storage saves data only for the current session, meaning it is deleted once the user closes their browser or moves to a different website.

* + device access

It is the capability of web applications to access and reach to several hardware components in a user's device, such as the camera, microphone, accelerometer, GPS location, etc. This allows web applications to perform tasks that were previously only possible in native apps, like locate someone’s device, take photos, etc.

* + Styling

Generally speaking, the styling feature is the improved, enhanced use of CSS3 (latest version of CSS) along with HTML5, as the two of them are supportive between each other. The cascade style sheets are the tools that can help the HTML user to customize style, sizes, shapes, fonts, layouts, etc. about the HTML elements, and CSS3 can support the new semantic tags HTML5, as well as perform new features itself that previous versions did not have.

* + performance & integration

Performance and integration in HTML5 are about making web applications run smoothly and work well with other technologies and platforms. To improve performance, HTML5 uses new techniques such as Web Workers, Web Sockets, and local storage, so that web apps respond quickly and run smoothly. To make integration easier, HTML5 includes APIs that allow web apps to connect with other things like native apps, databases, and cloud services. This helps web apps work seamlessly with other things and provides a great user experience on different devices and platforms.

1. How can you generate a **public key** in html?

There is a [<keygen>](https://www.geeksforgeeks.org/html-keygen-tag/) tag in HTML. It generates an encryption key for passing encrypted data to a server. The purpose of the <keygen> element is to provide a secure way to authenticate users. Someone can submit it with a form tag, where an external user will input some text and thus, the page will associate to this so called key.

Actually, when a form is submitted then two keys are generated, private key and public key. The private key is stored locally, and the public key is sent to the server. The public key is used to generate a client certificate to authenticate the user for the future.

1. What are **<label>** elements used for?

The <label> element in HTML is used to provide a text description for form elements such as input, select, and textarea. The <label> element makes it easier for users to understand the purpose of form fields, and can also improve the accessibility of the form by providing additional information to screen reader users.

1. What is XHTML?

XHTML stands for eXtensible HyperText Markup Language, and it is a type of HTML language that is used to create web pages. XHTML is similar to HTML in many ways, but it has stricter rules for formatting and coding. It has rules for how the code should be written and structured, and it is designed to be more organized and easy to understand than other types of code. By using XHTML, web developers can create web pages that are well-structured and easy to maintain, which can improve the overall quality of the web pages they create.

1. Are there any optional tags?

Some tags are optional in HTML, according to the fact that actually typing them down in the HTML script or not will not end affecting the performance or appearance of the web page someone is creating with it.

For example, within the head section of the HTML script, the meta tag is optional, for it will not affect the display of the page. In the body section of the HTML, tags like footer or aside can also be optional if someone does not want to display such elements on it, and prefers to go by another way of showing the elements, even in a home or index page.

Also, we must mention there are some tags that can supply or replace the usage of another specific tag, and thus we can think of them also are optional, or in other words, replaceable.

1. When would you use a **<div>**, **<section>**, or **<article>** tag?
   * The div or division tag has a more general use, that can be used for a general division of a website. For example, if someone wants to divide each part of the website into its components, a good way to further mark it in the code is div. Also, if someone wants to specifically divide or separate one part out of another in the website, making a division tag is very useful.
   * The section tags are more specifically used for actually casting the parts that will contain information like images, articles, text and general contents in a website. This tag allows to differentiate them from the header, footer, and aside contents. Someone may want to declare the header section apart from the articles part, or the aside part. Sections are useful for these means, because this is the way to show in the code someone is passing from the “surroundings” of the site to the actual core, which contains all the entries relevant and diverse content within.
   * Article tags are precisely used for long entries of text, blocks and images that actually contain the information inside the sections. A section can contain various articles.
2. What are the different formats supported by **<audio>** and **<video>** tag components? Are there any advantages of one over the other?

The <audio> and <video> elements in HTML support a variety of audio and video file formats, including:

* <audio>: MP3, WAV, OGG
* <video>: MP4, WebM, OGG

Each of these formats has its own advantages and disadvantages, and the best format to use depends on the specific needs of your web page.

For example, MP3 is a widely used format for audio files, and is supported by most web browsers. It provides good audio quality, and is relatively small in size. However, MP3 is a proprietary format, and may not be suitable for use in some open-source projects.

OGG is a free, open-source format that provides good audio and video quality, and is supported by many web browsers. However, it is not supported by Internet Explorer or Safari, so you may need to provide alternative formats if you want your web pages to be accessible to all users.

WebM is a free, open-source format that provides high-quality video, and is supported by many web browsers. However, it is not supported by Internet Explorer or Safari, so you may need to provide alternative formats if you want your web pages to be accessible to all users.

Overall, the best format to use depends on the specific needs of your web page, and you may need to provide multiple formats to ensure that your pages are accessible to all users. When choosing a format, it is important to consider the quality of the audio or video, the file size, and the compatibility with different web browsers.

1. How can you apply CSS to only a part of the HTML document?

Depending on the case, for example if you want to only use certain style in only a part of the document and there is no other stylesheet or style instructions apart or ahead that will also modify it, you can input an in-line instruction into the element you want to edit or style out: <tag style=”......”></tag>.

In another case, there is a special feature or instruction that can be added inside each style rule, which is called: !important, that will prioritize this rule out of any before or after created.

Finally, for general means, if someone wants to apply certain CSS rule only to a certain part of the HTML document, the correct way to do it is using a class or and id ONLY for that part of the document, that will spread out the class or id among all the children.

1. What are the new features introduced by CSS3?

Media queries, animation, flexbox, background and borders.

* + Compare CSS2 and CSS3

CSS3 is a newer version of the CSS language and it has more tools and options compared to CSS2. With CSS3, it's easier to make websites look better and respond to different devices and screen sizes. In the past, CSS2 wasn't able to do as much, but now with CSS3, someone can do more complex and interesting things with your website designs. In short, CSS3 is a stronger and more flexible version of CSS and has many new features to help make website designs stand out, with more sophisticated and responsive web designs.

1. How can you highlight text in HTML?

The best way to do it is to mark it with a span tag. This is the best semantic way to highlight it out of the remaining of the text. Plus, the spanned text can also be manipulated within a stylesheet and distinguish it further among the others, like for example underlining it or putting it with a certain background color, according to someone’s will.

1. What is responsive design?

It is a generalized term for naming design someone can create and manipulate from a stylesheet that will actually behave and respond correctly to the end user. Like when for example, someone needs to change a media query, the font-size and other features, all in one call for the display or layout of the webpage. Creating responsive design allows the user to have a very smart HTML+CSS pair that will match to any requirement done by a browser or end user, according to the device this person is using for visiting the site. The website can adapt and adjust its layout and appearance based on the size and orientation of the device it is being viewed on, and provide a good user experience for visitors, no matter what device they are using.

1. What is progressive enhancement?

Progressive enhancement is an approach in web development that is carried along CSS, which makes the style of a website appealing, accessible and good looking for a rookie user, while also displaying a great experience for those who count on more experience on technology.

1. What is graceful degradation?

It is another approach that is used for web development that looks for optimal accessibility for old fashioned users, or those who have less time using browsers or being in touch with the looks or the appearance of websites, while keeping in good way to the new trends and to the new coming people that also surf or navigate throught this web pages, and may be actually keen on using this technology day by day as experts or frequent users.

1. Explain how CSS3 animations work.

Integrated CSS3 animations can be casted with the keyframe selector “**@keyframes name-of-the-animation-displayed-action{ from {...} to {...} }**”. The {...} properties can be predefined animated actions from a starting postion or beginning action up to an ending position/action to end. Properties available can be such as “transform” or “translate”, and the values specified must match each property, according to their documentation. Afterwards, in other part of the stylesheet, a class selector must be casted to show specifically what each animation can do, according to its keyframe previously defined selector, with such a property called “animation”. On the other hand, an ease-function that can work with a property named “animation-timing-function" and diverse, different values for each. CSS3 animations can be used for block and in-line elements.

1. Explain how transitions work

Transitions are one of the available rules that can be appliable to animate in a style sheet. They allow someone to smoothly change from one style to another over a specified duration of time. They add some sort of subtle animations and effects to website elements, and can be used to transition between different styles for animations.

To create a transition, first it is needed to specify the CSS property to animate, the duration of the transition, and any timing function desired to use to control the pace of the animation:

button {

transition: background-color 0.3s ease;

}

button:hover {

background-color: blue;

}

The "ease" timing function is used to control the pace of the animation, causing it to start and end slowly and speed up in the middle.

1. Explain the different CSS units of measurement.

* Ems: This is the width size of the letter em (“m”) in the font size of different text inputs.
* Rems: This is a best enhanced, responsive option for ems, that actually stands for the unit of measurement “em” of the root element in the HTML. Rems are justified correctly to the side ends and thus can be displayed well in a website, no matter the font size of each child element, because they start from the point of reference of the root element.
* Px: Pixels, these are super useful for predefined images that have a size indicated in pixels width times pixels length.
* Percentages (%). This unit corresponds to the percentage of the size of the parent element, its use is better when trying to specify or compare against the sizes from the parent container.
* Pts: Points, used in print media, equals to 1/72 of an inch.
* Vh, vw, vmin, vmax: These are viewport units, and are relative to the size of the viewport that can be left in any browser, no matter the device. There are four different viewport units: vw (viewport width), vh (viewport height), vmin (the smaller value of viewport width and height), and vmax (the larger value of viewport width and height). These units allow elements to be sized based on a percentage of the size of the viewport, making it easier to create responsive designs that adjust to different screen sizes and devices.
  + Which one do you prefer to use and why?

For text, I believe rems are the best ones, because of their root response feature.

For images only, pxs. For mixed text and images, I think it depends whether the image is shown in-line (in that case, I would use ems or rems), or on the other hand as just captioned images or elements aside text (maybe keeping pixels in that case is the best way).

Finally, if my general intention is to show or display different elements for different users or different devices, I would go for viewport units, and play with the available space left for render in the web page.

1. How do you capitalize using only CSS?

There is a special rule property for text someone can introduce in a stylesheet, called “text-transform”. If someone calls the value “capitalize” for this property, all the text inputs will be showed capitalized.

1. What are the possible values for the ***display*** rule and what do they do?

The allowed values are: “inline”, “block”, “in-line block”, “flex”, “in-line flex”, “grid”, “in-line grid” and “table”.

* Display:inline. Makes a tag element behave like a text line, thus it will not be able to change height, width, padding, margins, etc.
* Display: block. Similar to div and p tags, this rule will allow someone to do the tasks that were forbidden in the inline rule, changing in this way the settings and style of the shape and surroundings of the element.
* Display: inline-block. A hybrid of the previous two rules, will make an inline element behave like a block one, inheriting the capability of editing shape and surroundings.
* Display: flex. Similar to display:block, that improves the way interior tag elements are layed out, a rule that ends up making block elements to be very flexible on the way they are shown, displayed and layed out in a website.
* Display: flex-inline. The same as the rule before for flex, but now the flex elements can be treated as inline ones, too.
* Display: grid. This rule is the most advanced, recent tool that has been improved in the latest CSS, for displaying layouts. It is only needed to specify a parent class or parent rule that actually contains how the general layout of the page, or section of the page, will be displayed, and the CSS will automatically make the division in default blocks, that yet the user can edit and further display like he/she wants.
* Display: grid-inline. The same as display:grid, but for grid elements that can be or need to be treated as inline ones.
  + What is the difference between ***display:block*** and ***display:inline-block***?

The extra added word “inline” before block will act upon an element that is already a block, and will treat it as an inline text input, making the block element fit or be displayed along the size of a line of text.

1. What is a ***class*** and an ***id***?

Classes and id’s are different sort of attributes that can label tags and HTML elements in general, and supply a way to operate over them thanks to inheritance over parent and children tags: they can act inside the HTML itself, or also cast such class or id selectors from a stylesheet operating in the HTML script.

* + When would you use an id instead of a class?

Ids are very useful for ruling out any other HTML element for precise specification of a selector from a stylesheet, or also for labeling different sections, articles or parts in general inside a website operated from an HTML script. For example, id’s are more specific and faster for selectors to operate into the website, and thus can be used for unique cases in an HTML script, like when someone can make a jump inside a website anchored link. Or for stylesheet selectors to prioritize id’s over any other selector.

* + When would you use a class instead of an id?

Classes are the best attribute to operate over tags or HTML elements, because they define also a unique way to label each one of them like id’s, and can sum up to be a good, elegant semantic way to arrange selectors from a stylesheet to operate over a HTML script. Abusing the usage of id’s or too many classes for each element is very harmful for an HTML+CSS pair; instead, someone should use single-class semantics per element. In other words, each element should contain at most one class by attribute, and the

1. Which selector is faster, id or class?

An id is always faster for the interpreter of the HTML over a class, when both operate over as attributes for the element it is declared before.

1. How do you target a direct child element?

All someone needs to do is to call the selector of the parent element and thus, all the children beneath will be affected or targeted by that selector. According to the precision that is required, someone might add as well other selectors or create more classes for the child (children), and call them too in the sytlesheet, in case further specificity is needed.

1. Can you target a single, specific element with a particular class?

Yes, the way this can be accomplished for a certain specific child element, is that someone needs to declare as selector all the classes (parent and children) which it belongs to; and furthermore, specifying the tag before these classes in the selector is useful to distinguish them among other tags that might be sharing the same classes: “tag.class-parent.class-child { ..... } “.

* + When would this be useful?

For example, this is super useful when someone wants to target a single, specific child tag, that already has some previous parent selector acting above or before it.

1. What is the use of the ***data-\**** attribute?

It is used to store custom data on an HTML element. The value of the attribute can be any string and is not typically used to influence the styling or behavior of an element, because it will not be displayed to the user in the webpage, but instead might be useful for another interaction afterwards with another program, as reference to the person coding or when exported to another piece of code or location (JavaScript, for example).

1. What is the difference between ***display:none*** and ***visibility:hidden***?

The difference is that display: none completely removes an element from the page, including any space that it would have taken up, while visibility: hidden only hides the element, but the space it would have taken up remains, and the element just disappears, leaving the blank it was occupying displayed.

1. Does ***overflow: hidden*** create a new block formatting context?

Yes, this means that the element becomes a container for its children, and any positioning and floating of the children will be contained within the boundaries of the parent element. Additionally, the value of overflow: hidden can be used to hide any content that exceeds the dimensions of the container, which can be useful for controlling the appearance of a layout.

1. Is it possible to use percentages in border widths?

No, it is not possible to use percentages in border widths in CSS. Border widths must be specified in absolute units of length, with fixed values, like px, ems or rems.

1. Is it possible to use percentages in margins?

Yes, because it is possible to express the size of a margin according to the percentage of space and size displayed by the container.

1. How do you reset a CSS style?

Someone can cast a pre-prepared CSS script that will reset an existing one and thus replace it in case the code user is needed to do so. It will remove any browser-specific styling that might be applied. It can be typed down by the same user, all it needs is to override all the default styles that are being applied to the former HTML elements in the page. Also, some sheets or methods can be searched online for this override.

1. If you have a way of dividing an interface horizontally and vertically, could any layout be made?

Yes, it can be possible while the most suitable display option is used for the corresponding layout. In principle, any layout can be made, if the customized styles properties are not against each other, and if someone has a clear mind of what is the ending point to reach out with the layout.

1. How can you store data on the browser?

In HTML and CSS, someone can store data on the browser using web storage APIs, such as local storage and session storage. Local storage allows to store data on the client-side, even after the browser is closed, while session storage only lasts for the duration of a single browser session. Both local and session storage allow to store key-value pairs as needed for further or upcoming, external interactions with JavaScript.

Also, about web storage and client-side storage, there are ways to actually save data in the client (browser), like cookies. Web applications can store data locally within the user’s browser. During every server request data is stored in the form of cookies.

1. What debug tools are available for CSS?

There are several ones, but the most common, easy ones are the incorporated to the browser dev-tools, that can interact with the existing layout and display of the elements of the CSS acting on the web page. Also, there are some web extensions that can be useful as well. Finally, CodePen and any other tool in programming editors that debug can be used to check or verify on the fly what can be mistaken when editing a CSS sheet.

1. How do you make a mobile-first approach?

To implement a mobile-first approach in CSS, you can start by defining styles for smaller screens and then using media queries to add or adjust styles for larger screens.

1. Explain how the box-model works.

It is a defined method for CSS to create and show layouts in the web page. Someone can regard any element in CSS as an element contained into a box, and this concept can help to understand how stylizing such elements, and making modifications such as align, tabulate, separate, etc., can be accomplished. Each whole element considered in the box-model will be itself a bloc-like object.

* + What are the properties related to it?

The content of the box, which can be an in-line like object or not, and the surroundings of it will be three boxes partitioned into: padding box, border box and margin box.

1. What is flexbox? Have you used it?

A tool feature in the newest CSS that makes easier to create flexible and responsive layouts. Flexbox provides a more efficient way to lay out elements in a row or column, and can help reduce the need for complex grid systems.

Yes, flexbox allows to treat a group or collection of elements as flexible items per se, and display them with grace and flexibility along the webpage, as one requires typing down inside their rules in a style sheet.

1. What are media queries?

They are CSS techniques that enable to apply different styles to HTML documents based on the features of the device displaying it. A media query defines a set of conditions, such as screen size, screen resolution, and device type, that must be met in order for the associated styles to be applied.

* + What is the use of ***only***?

The "only" keyword helps to improve cross-browser compatibility by limiting the styles to specific devices and browsers that support the media type.

Example: “only screen and (min-width: 600px)" to apply styles only to screens with a minimum width of 600 pixels, and not to other types of devices such as print or mobile.

* + Does the ***screen*** keyword apply to the device’s physical screen or the browser’s viewport?

It applies to the device's physical screen. It specifies that the styles inside the media query will only be applied when the output device is a screen-type device such as a computer monitor or television. The viewport is the area of the browser window where the web page is visible.

1. What are **pseudo-classes** and **pseudo-elements**?

Pseudo-classes are a further conditioning way to apply a change of style when a selector matches a more specific rule for certain class, that now the stylesheet asks the pseudoclass to test it to the class elements and apply the rule of this new pseudoclass only when it becomes true.

On the other hand, a pseudoelement is yet another pseudo-selector that will be applied to the class elements and modify the layout of the elements as if we were adding a whole new HTML element applying it, rather than just re-stylizing the existing class element like in pseudoclasses.

1. Name a few pseudo-classes and what they are used for.

“hover” is used for applying style changes to a certain class element when the user of the webpage hovers the mouse over such element in the page.

Also, there is another pseudoclass named “focus” that is useful when the user of the webpage has given focus to this element, like for example if someone clicks or taps or uses the Tab Key onto it.

“first-child” is another pseudoclass that will apply the style changes of this pseudoclass rule only to the first child of the parenthood matching a class family of children. Other similar options are “last-child” (in this case the styling will be applied to the last-child appearing the web page of such family), “only-child” (for an only one, single, child element, and one only), “invalid” (very useful when trying to style an element that does not match a query or a form input or any other operation over the element done by the user of the webpage).

1. Explain how CSS shorthand syntax works for padding/margin.

It is a one single line syntax rule in CSS that someone can specify for the values of the padding and margin properties in each possible four sides of an element, so that the rules can be written in a compact form.

* + How do they behave when defining 1, 2, 3, or 4 values

The syntax of the allowed four possible values typed down is: top right bottom left, in clockwise direction:

padding: top right bottom left

margin: top right bottom left

It is not completely necessary to type them all four down. If only one value is typed down, it will automatically be applied to the four sides of the element. If two values appear, the first value will be applied to top and right sides, the second to the remaining two. If three appear, the first will be applied to the top, the second to the right and left sides, and the third to the bottom. Four will meet the complete extended rule and apply to each of the four sides.

1. How is an HTML5 form implemented?

Using a form tag, this predefined tag will automatically perform like the combination of an input tag for text input, a select tag and a textarea tag.

The most simple syntax is:

<form>

<input attribute=”Text 1” />

</form>

* + How is validation performed?

HTML supports built-in form validation through attributes, such like:

* required: This attribute is used to specify that an input field must be filled in before submitting the form.
* pattern: This attribute allows you to specify a regular expression pattern that input data must match.
* min and max: These attributes are used to specify the minimum and maximum values that input data must fall within.
* step: This attribute is used to specify the increment or decrement that input data must adhere to.
* type: This attribute is used to specify the type of input data, such as email, url, number, etc. HTML5 provides built-in validation for these types.
  + How are key-value pairs sent to the server?

By using a web form, where the key-value pairs are created based on the input provided by the user. The data is then sent to the server using the HTTP protocol and is processed on the server-side.

The key is the name of the form element, and the value is the user's input. For example, if a form has an input element with the name "email", and the user enters "[example@example.com](mailto:example@example.com)" into that input, the key-value pair would be "email: [example@example.com](mailto:example@example.com)". This data can then be processed on the server, either by a script or a back-end application, to perform tasks such as storing the data in a database or sending an email to the user.

* + How is a field declared mandatory?

by using the "required" attribute in the corresponding form element, such as the input, select, or textarea element. For example:

<input type="text" name="username" required>

This will ensure that the user must fill in the field before submitting the form.

* + How is a field set to be mandatory AND avoid having the browser validate?

Also by using the "required" attribute and further, to avoid browser validation, someone can use the "novalidate" attribute on the <form> element. This attribute specifies that the form should not be validated on submit, so the browser will not perform any built-in validation checks on the form.

1. Describe how to improve page load times when you have 20 js files, 20 css files, and 20 images.

Optimizing such a big task file by file can start with some minor operations in each file, that could take a long time to correct or improve in each as a whole. So as a general task, the first one could be combining the JS files together with the CSS ones in only one file, and improve then the load times.

Also, an excellent idea is storing each file in the browser cache, so that the future incoming operations do not need to the frequently used files to be loaded each time again and again.

If someone was about to optimize file by file, a good practice is to compress each one, or in the case of the images, optimize them so that they are not too heavy but keep the same resolution and appearance.

1. How can an inline style be overridden?

By using a more specific selector in a stylesheet apart, such like a class or an id, which are yet more specific than an inline style attribute.

1. Explain how to implement a carousel using CSS / CSS3

A carousel can be implemented using CSS3 and HTML by creating a container element that holds a series of items to be displayed as slides. The slides can be set to automatically rotate by using CSS3 transitions and animations. To achieve the desired sliding effect, it can be used the transform: translateX() property to change the position of the slides relative to the container. The speed and direction of the animation can be also controlled using the transition property.

1. How can the performance of a web page be measured?

The following metrics can be employed for such a task:

* Page load time: The time taken for the page to fully load and be displayed to the user.
* Time to first byte (TTFB): The time taken for the first byte of data to be received from the server.
* Render time: The time taken for the browser to render and display the content of the page.
* Resource loading time: The time taken for various resources such as images, scripts, and stylesheets to be loaded.
* Frame rate: The number of times a new visual frame is displayed per second.
* Memory usage: The amount of memory used by the page and its components.
* Number of requests: The number of requests made to the server to load the page and its components.

1. Describe different ways in which a web page can be optimized.

* Minifying HTML, CSS, and JavaScript files to reduce their file sizes and improve load times.
* Compressing images and other media to reduce their file sizes without sacrificing quality.
* Using CSS sprites to combine multiple images into a single file, reducing the number of HTTP requests made by the browser.
* Implementing responsive design to ensure the website looks good and functions properly on all devices and screen sizes.
* Using CSS media queries to apply different styles based on the characteristics of the device being used, such as its screen size or orientation.
* Choosing efficient selectors in your CSS and avoiding overly specific selectors that can slow down the rendering of the page.
* Using efficient CSS layout techniques, such as flexbox or grid, to achieve complex layouts.
* Using CSS preprocessors, such as SASS or LESS, to write more efficient and organized CSS code.
* Implementing lazy loading for images and other media, so that they only load when they are needed, improving the initial loading time of the page.

1. Is there a performance difference between the different selectors?

Yes, there is a subtle difference that actually becomes relevant in terms of speed and efficiency. Id selectors can become faster than class ones, and also those only operating over tag selectors.

* + How do you optimize CSS selectors?

It is recommended to choose selectors based on the best semantic and good to maintain approach, rather than discard others for performance reasons. For example, optimizing only the use of class selectors with parents and sometimes children when useful, and reduce the number of times id selectors are casted, or absolutely avoid them being used.

* + Which selector is the fastest?

Id selectors. They can be combined along classes (a class selector inside an id selector) and become yet faster than any other selector.

1. What are the different CSS filters you can use?

CSS filters allow you to apply various effects to elements in a web page, such as blurring, grayscaling, brightness, contrast, opacity, and more. Some of the different CSS filters are:

* blur()
* brightness()
* contrast()
* drop-shadow()
* grayscale()
* hue-rotate()
* invert()
* opacity()
* saturate()
* sepia()

These filters can be applied to an element using the "filter" property in CSS.

1. What is specificity?

It is a feature that CSS rules export to HTML elements when applying styles over some element(s), no matter the position in the webpage, but only the attributes and type of element or tag that concerns. With specificity, the code user can determine what and how many rules of style from a stylesheet will be applied to any element in the web page.

* + How do you calculate it?

The form of calculating it is first, locate the element(s) where the style rules are being applied at. Second, it must be determined which inheritance attributes this element has in the HTML script according to the styling invoked in the stylesheet. If this element has a parent selector acting upon it, this element will be styled out as the parent selector in the stylesheet commands, and later, the subsequent selector(s) for the children will begin to act too.

When the selectors for the parent and the children are located and determined, now it is the turn to check whether they are class selectors or id ones. Id selectors are more specific than class ones, so the specificity that will rule first overall would be the id parent selectors, then the id children, then the class parent and then class children. Afterwards, if any other class or id is shared, the specificity will be claimed upon what selectors appears more frequently in the attrbitues of the HTML element, then which appears last in the stylesheet. And finally, the subsequent remaining selectors will be applied, if any, in case they are not already covered by the existing ones.

1. How would you use sprites?

Sprites in CSS refer to a single image file that contains multiple smaller images, used to reduce the number of HTTP requests made by a web page. To use sprites in CSS, you would first create an image sprite, which is a single image file that contains all of the smaller images you want to use. Then, you would define the background-image property for each HTML element, and use the background-position “background-image” property to specify which portion of the sprite image should be displayed for that element.

1. How can you load CSS resources conditionally?

Using the media attribute in the link tag, you can specify a media type, such as screen or print, that the CSS should be applied to. This allows you to create different styles for different devices or screen sizes. For example:

<link rel="stylesheet" type="text/css" media="print" href="print.css">

1. What is gzipping? How is it used?

Gzipping is a method for compressing files to reduce the amount of data sent over the network. It works by compressing a file and sending the compressed version to the client, which then decompresses it and renders the content.

It can be used to reduce the size of the CSS files being sent to the browser, which can improve the performance of a website. This is because smaller files take less time to download and therefore can improve the load time of a web page. To enable gzipping, the server needs to be configured to support it and the browser must support decompression.

1. Mention any CSS framework that you have used.

I have not used them yet thoroughly, but I can mention Bootstrap.

1. How do you serve a page in multiple languages?

By using HTML Markup in the head of the HTML elements, such as the lang attribute, to specify the language of the content, or also someone can rule them out by using CSS Selectors to target specific language versions and style them differently.

Also, there are some further capabilities using side scipt versions, that can serve different versions of the same page but displayed in different languages, with tools like PHP and JavaScript.

1. Explain what standard and quirks mode are.

These are two modes that a web browser uses to display a web page. The mode is determined by the DOCTYPE declaration of the HTML document.

Standard mode is a mode where the browser follows the latest version of the World Wide Web Consortium (W3C) standards for HTML and CSS. In this mode, the browser adheres to strict layout and rendering rules, which ensure consistency across browsers. This mode is the best for cross-browser performance to be improved and show consistency among the majority of browsers and devices.

Quirks mode, on the other hand, is a mode where the browser uses an older engine to display a web page. This mode is used to handle web pages designed for older browsers that may not be in rule completely with the new standards. Quirks mode may also be used to display pages that were written in an inconsistent or incorrect manner.

1. Have you used **<meta>** tags? Explain if you have.

Yes, some meta tags are predefined in the shortcut script file of HTML5 I use in VS Code, and precisely, it is located inside the head section of the script. As mentioned in some responses above, these are completely optional or at least are not indispensable for creating an HTML5 web page. However, their usage can modify and target some functionalities that might become essential for the web page to be layed out or displayed correctly.

<meta charset="UTF-8"> is used for declaring the character sets that will be used throughout the HTML scripting of the webpage, in this case, is Unicode Typeset Font 8.

<meta http-equiv="X-UA-Compatible" content="IE=edge"> as I know is used for declaring the compatibility of the HTML script, it is ready tu use for Microsoft Edge browsers.

<meta name="viewport" content="width=device-width, initial-scale=1.0"> is used for declaring the contents display in the webpage, as of the viewport and its scaling.

1. Describe what happens when you type a URL into a browser

* The browser sends a request to the server associated with the domain of the URL.
* The server processes the request and returns a response, which includes the HTML content of the web page.
* The browser renders the HTML content into a visually appealing format, applying styles specified in the CSS associated with the HTML content.
* The browser requests any additional resources such as images, scripts, and stylesheets required to display the page.
* The browser continues to update the page as any dynamic interactions occur, such as user input or data updates.

This process is called the "HTTP Request/Response Cycle."

1. When sending form data, what is the difference between the GET and POST methods?

These are methods defined in HTML.

The GET method appends the form data to the URL in the form of query parameters. It's simple, fast, and can be cached, but it has limitations on the amount of data that can be sent, and it's not suitable for sensitive information since it is visible in the URL.

The POST method sends the form data in the body of an HTTP request. It's more secure than the GET method, it's not limited in the amount of data that can be sent, and it's not visible in the URL.

## **Bootstrap**

**Estimated Time: 3 hour.**

1. Explain what bootstrap is and how you make use of it.

It is a CSS framework that provides a wide range of style sheets for elements such as headings, buttons, forms, and tables, making it easy to create a good looking layout for someone to include in a web page. Bootstrap has pre-designed classes and plugins for creating layouts.

To use Bootstrap, it is only needed to include the Bootstrap CSS and JavaScript files in the desired main script in HTML.

* + Explain its key features such as the grid system, glyphs, responsive design, etc.

1. How would you plan creating a CSS framework similar to bootstrap?

Decide on the CSS preprocessor to be used, and the grid system to be implemented, so that the architecture and the structure is well defined. Also, test and ensure a good responsive design to be implemented, so that they layouts can be good looking and operative in different screen sizes. Then, there must be some rules and selectors well structured and defined that can be useful as documentation, along with minor instructions that can be readable and well understood. Finally, test it with a final beta version and keep improving it.

1. What is your opinion on bootstrap?

I think is a very useful tool for someone that has already learned basic CSS with flexbox and grid, that can become actually an excellent way to implement style to existing projects or also, learn how to implement new ones, which would be hard to create from scratch.

1. What version of bootstrap have you used?

The current latest version available, which is Bootstrap 5.

1. What is the difference between bootstrap 4 and 5?

Bootstrap 5 is the latest version of this front end web development framework. It no longer supports Internet Explorer and focuses on modern browsers. Bootstrap 5 has dropped its dependency on jQuery and now uses vanilla JavaScript.

It also has improved its utilities classes, which makes it easier to customize the styles of your components, also improved its grid system, typography and accessibility. Now, new version includes new UI components such as a custom file input, form control and a few new navigation components.

1. Give a comparison between **bootstrap** and **foundation**.

Bootstrap follows a more traditional design philosophy, using a 12-column grid system and pre-designed UI components. Foundation, on the other hand, uses a more flexible, mobile-first approach, with a custom responsive grid system and an emphasis on semantic HTML.

Foundation provides a smaller set of UI components and a more minimal set of default styles, which gives more control over the appearance of a website.

Performance: Both frameworks are lightweight and optimized for performance, but Foundation is generally considered to be lighter and faster than Bootstrap. However, Foundation has far less documentation than Bootstrap.

## **CSS Preprocessors**

**Estimated Time: 1 hour.**

1. What are CSS preprocessors?

CSS preprocessors are scripting languages that extend the capabilities of CSS and add more features to the language. These preprocessors allow to use variables, nested selectors, mixins, functions and mathematical operations in CSS.

1. Explain how LESS works; advantages & disadvantages.

It allows for variables, mixins, functions and operations to be used in stylesheets, making the styling process more efficient and organized.

Pros:

* It extends the capabilities of CSS.
* Reusability: LESS makes it possible to use variables and mixins, making it easier to reuse styles and reduce the amount of code.
* Organized code: LESS allows for better organization of code and structure, making it easier to maintain and update.
* Improved efficiency: LESS provides a more efficient way of writing CSS.

Cons:

* More complex: LESS can sometimes make code more complex, especially for developers who are not familiar with the syntax.
* Performance: LESS requires a preprocessing step to convert the LESS code into CSS, which can affect the overall performance of a website.
* Browser compatibility: Some older browsers may not support the CSS generated by LESS, which could cause styling issues.
  + What is a mixin?

A mixin is like a reusable template that can be used to apply multiple CSS styles to an HTML element. It makes it easier to reuse styles and reduce code repetition. Mixins can contain any number of CSS declarations, including variables, operations and even other mixins. To use a mixin, you simply include it in your stylesheet and call it by name, passing any required parameters. This helps to keep your CSS organized, maintainable and reduces the amount of code you have to write.

1. What are the reasons to use preprocessors?

The reason for using them is to make the styling process more organized, efficient, and maintainable. Additionally, it can also help to prevent having to write repetitive code, making the code easier to read and debug, since code can become now reusable. Also, some values in variables can be stored and they only need to be modified, instead of looking up for the instance or line of code of each single property involved.

## **Exercises**

**Total Estimated Time: 16 hour.**

1. Create a star-rating system with hover using only HTML/CSS. No javascript.
   * Use elements that are semantic compliant when possible.
   * Assume you have a sprite with 5 stars selected and 5 stars unselected.
   * The selection should persist

**Estimated Time: 8 hour.**

1. Create an image slider with the following markup:

| <div id='carousel'>  <ul>  <li>Slide1</li>  <li>Slide2</li>  <li>Slide3</li>  </ul>  </div> |
| --- |

* + Must have 500x300 dimensions
  + Only show one slide at a time
  + Create a transition animation

**Estimated Time: 4 hour.**

1. Write the necessary HTML and CSS to create a layout as follows (using HTML5 semantics):
   * The two content elements and side element must be next to each other
   * These three elements must be vertically aligned
   * Total sum of the 3 elements must be 100%
   * The footer should have a width of 100% and a height of 200px
   * Make the height of the 3 elements be 100% — the 200px of the footer
   * Have each letter of a word pop-up when hovered over.

| | **Header with navigation** | | | | --- | --- | --- | | **Content 1** | **Content 2** | **Side** | | **Footer** | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**Estimated Time: 4 hour.**

1. Explain how you would create a simple slider using CSS and HTML with values of 0 for min and 100 for max. No JS.
2. Create a layout displaying a navigation menu with at least 5 options
   * Provide a way to:
     + Display the menu on the left
     + Display the menu on the right
     + Display the menu on the top
     + hide/show submenus (drop-down)
     + Allow the user to switch between all positioning options
3. Create a CSS3 animation.
4. Create a 4x4 album grid with divs.
   * Make the HTML auto break the number of columns to 4
   * Make it responsive.
   * Consider what to do if a large number of albums are to be shown (100000+)
5. Create 3 divs and have them side-by-side.
   * Each div has text centered both horizontally and vertically.
   * Center a paragraph both horizontally and vertically inside a div
6. Create two elements, positioned side-by-side, each having ***width: 50%***.
   * Each element has additional padding and a border.
   * Make sure the parent does not overflow
   * Make sure the elements do not cause a break line.
7. Scenario issue: You are given a URL in which, when accessed, returns only a blank page.
   * How do you resolve the issue?