Inspecting CSS in the Browser

Introduction

Welcome to the lesson about the browser inspector. In this video, I would like to show you how powerful your browser is and what you can do to quickly find and target elements you want to edit. Getting familiar with this tool is inevitable, and I think it's

almost impossible to learn how to write proper CSS without learning how to inspect your website.

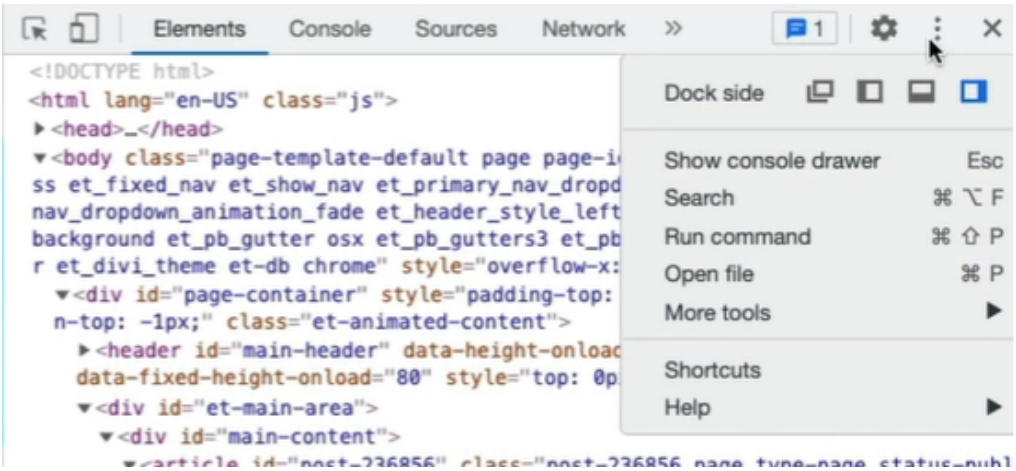
I will use Chrome Inspector as Chrome is probably the most popular browser, but many developers like to use Firefox, and Safari has a website inspector tool, as well. They all look similar, so you should be able to do the same thing, no matter the browser you're using. And these are very powerful and robust tools, but I will only focus on how you can use them when working with CSS.

Enabling the Chrome Inspector

As I mentioned, a browser inspector, otherwise known as Dev Tools, is essential for customizing and troubleshooting your CSS. You don't need to install anything to use it because it automatically comes with your browser.

And one way to open Dev Tools is with a keyboard shortcut F12 on Windows or Command+Option+I on the Mac. The other option is to simply right click anywhere in the browser window and select "Inspect".

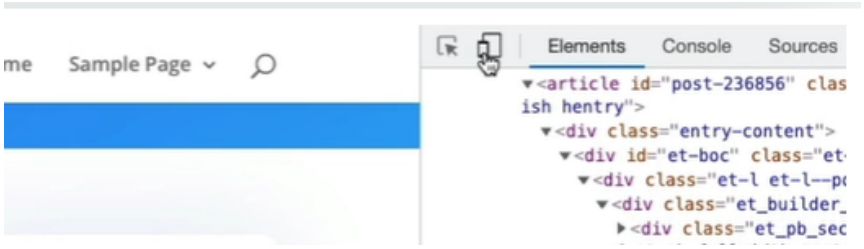
First, let's look at the dashboard. In the top right corner, you'll see this three dots icon. It gives you the option to orient the main Dev Tools dashboard to the left, bottom, right, or to a separate panel. I like to keep it on the right side, but go ahead and choose your preferred location that matches the screen you're working on.



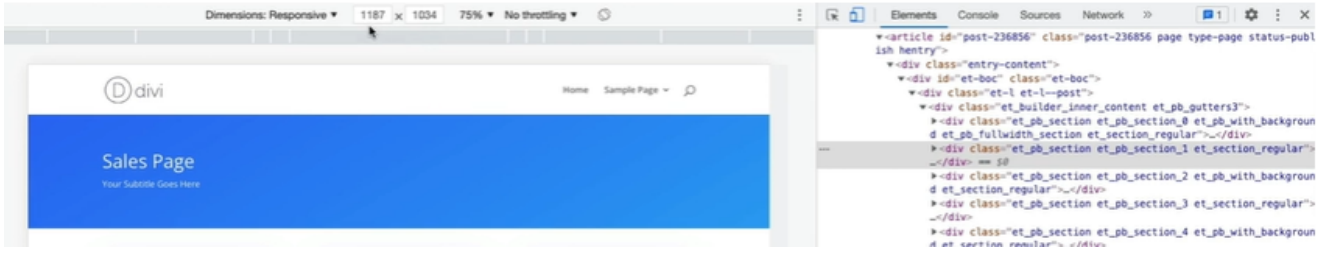
And once enabled, you will see your website on the left, and on the right here, the first panel, which is active is the "Elements" section, which shows the HTML structure here at the top. And along with the elements panel, the "Styles" panel is shown below, and here you'll see all the CSS code of different website elements.

Responsive Mode & Selecting Elements

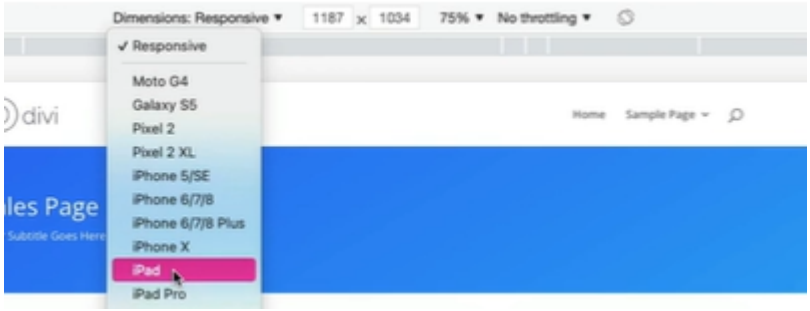
So on that main preview here, you can activate the responsive mode using this icon with mobile devices.



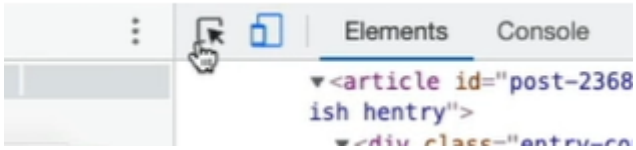
Once enabled, it will allow you to change the size of the preview. It shows you exactly what the current size of the viewport is here at the top.



And you can also choose to test some predefined device sizes, which may be helpful to check what part of your site is still above the fold on different screens.



Next to the devices icon there is another very useful mode which you can enable.

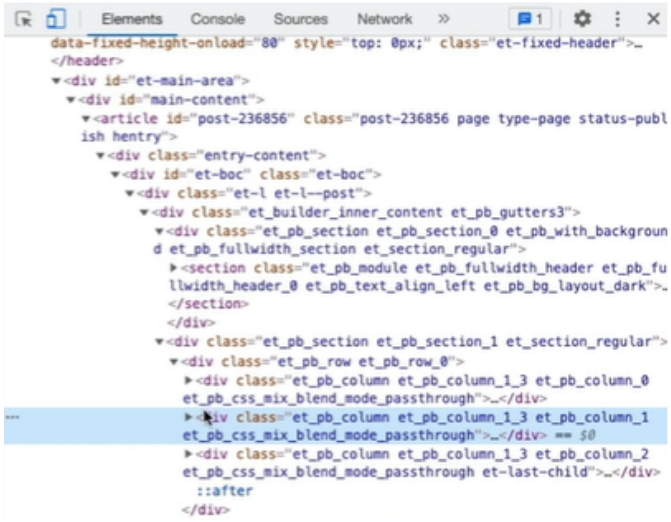


Once you click this arrow in the square, you will be able to hover and select the different elements on your site. And once selected, you will see that the HTML tag for that part of the page gets selected, too - in the Elements area.

HTML Preview

And here in the HTML source, you can clearly see the structure. You can use these little triangles to open and close different containers, and you can use that to see the parent-children relationship, which is important to see clearly when defining your CSS selectors.

And please note that it's not that everything which is above the selected element is its parent container. If HTML tags are on the same level, such as these two, for example, these are sibling elements, even though this may be above that - it's not its parent.



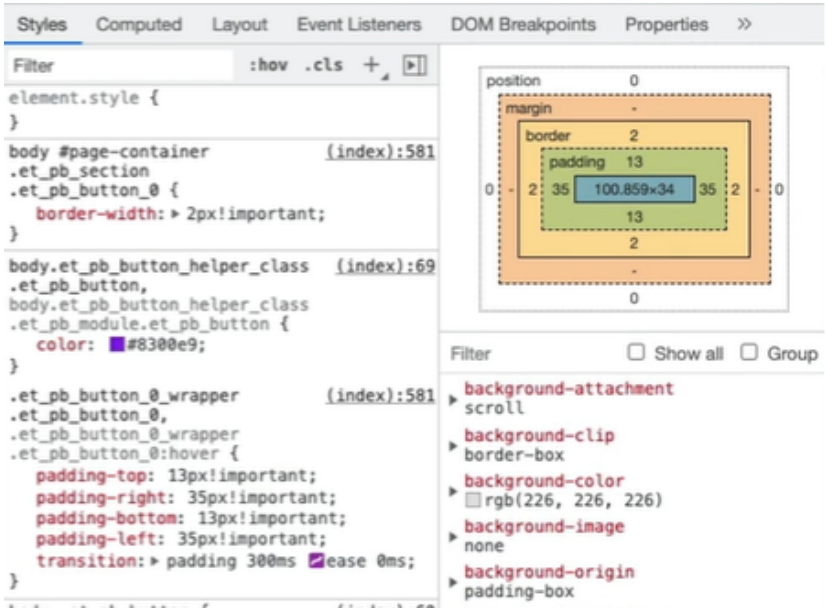
To see the parent, we need to look a bit to the left, notice the indent. You can also edit that HTML structure: add your attributes, basically test different scenarios.

But please note: everything you are doing inside of the Inspector is just a preview. You are not making any actual changes to your site. You cannot break anything. Once you refresh the page - it will go back to the original state.

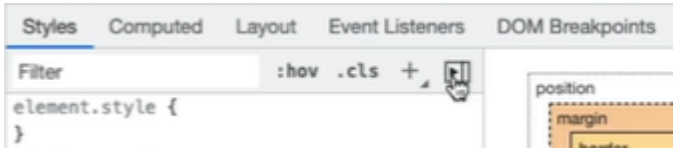
So, that is the HTML structure. You can select different tags here and it will get selected on the preview. Editing HTML is not something you would usually do, but you can definitely use the Inspector to edit CSS.

CSS Styles Panels

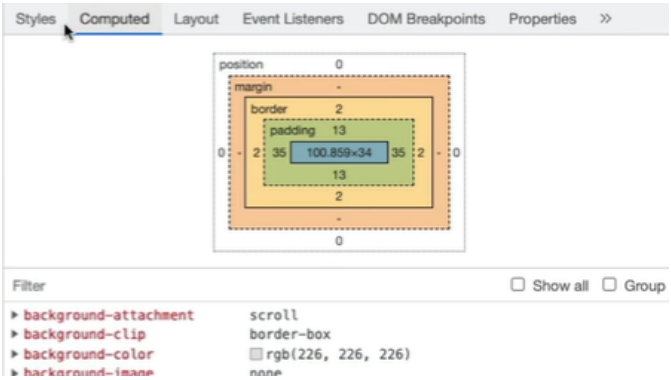
So let's have a closer look at the panels below. This is divided in two parts. We have "Styles" on the left and on the right: it shows the "Computed" section.



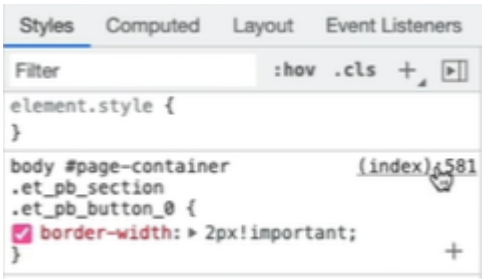
If you don't see that panel, you can toggle it on and off with this icon.



And if your screen isn't big enough, you can keep it closed. And then see it by clicking the link here.



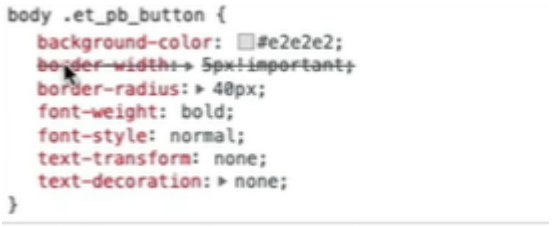
And first: Styles. Once our element is selected, this panel will show every CSS ruleset which is affecting that element in any way, all CSS code which is targeting that selected part, that selected element. We will see the target that is being used, the CSS declarations for that ruleset, and also the source of that code.



It can show you a specific CSS file or if it says index, that means that the code is inline in your page source, which is very likely with the new performance update. It also shows you the line number where exactly the CSS is located.

Crossed-out CSS properties

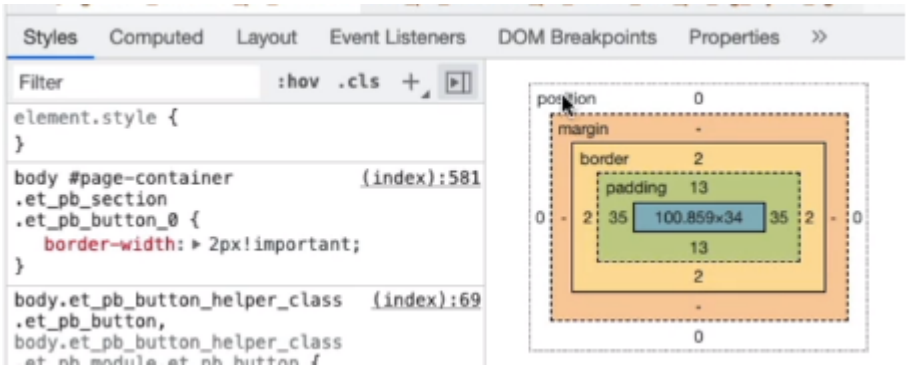
Now if the declaration, the CSS property and value is crossed out, it means that there is a different CSS ruleset using the same CSS property, and it's overwriting it.



The crossed out CSS doesn't work. Now, before I tell you what you can actually do with that, let's have a look at the right side.

CSS Box Model

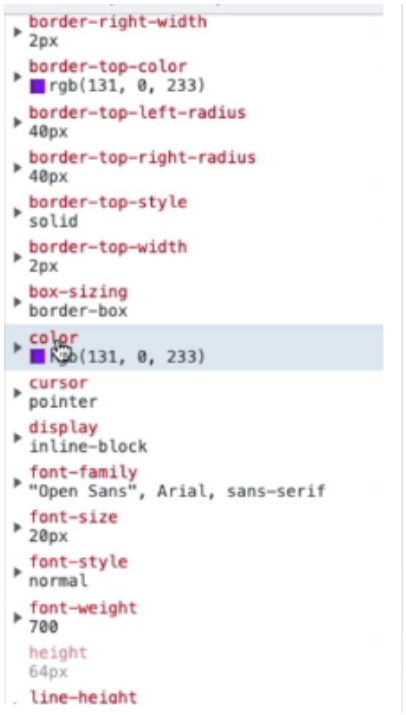
At the top we can see the CSS box model.



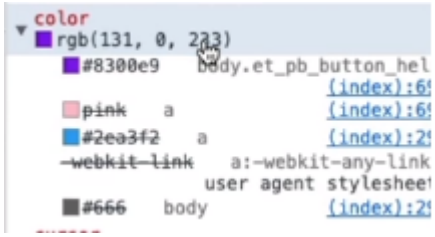
The sizing and spacing of every HTML element is represented with different colors. Blue is the element itself. Green represents the padding. Yellow shows us the size of the border, and orange - the margins. So getting familiar with the box model and the colors here gives us a quick overview of what space exactly our selected elements take. You can see it while hovering over or just when an element is selected. And below the box model, our computed styles. These are extremely important.

Computed Styles

Here we can see every CSS property and the value which is actually being used by the browser. So if you'd like to know what the color of that link text is, for example.



I can see the color property, and it will show me the value that the browser is actually using. And what's even better, if we click this little triangle here, it will show me where that CSS is coming from and if there are any additional CSS rulesets which are trying to change that color and maybe, you know, are overwritten. So this is great for troubleshooting.

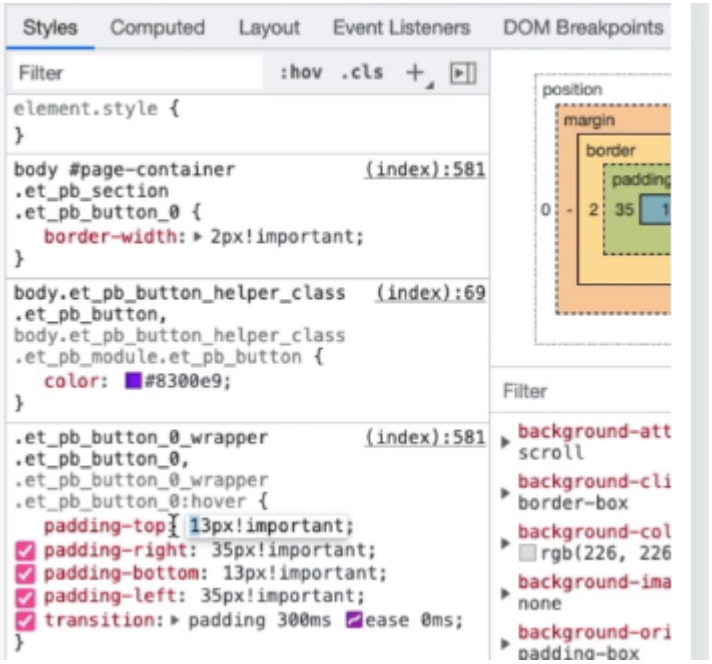


For example: you are trying to modify a property and if it doesn't seem to work. You can quickly check what CSS ruleset is overwriting your code, if you need to modify your selector, or if you're, you know, if your code is even listed here, maybe your selector is incorrect.

Right? So instead of going through that list of styles, we can quickly see any property that we want to check right here. Now let's see how we can modify the CSS here.

Editing the CSS

So you can edit what is already here, changing the values, checking and unchecking different properties to see what it would look like if that CSS weren't here or if it were different.

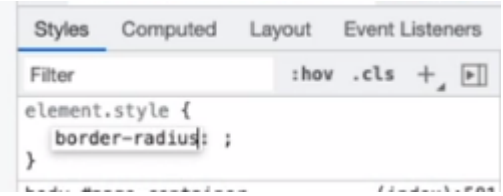


But in practice, that is not very helpful unless you are editing your own code and looking for a way to change it, then, sure, go ahead and edit your own code this way. But if that CSS is coming from Divi or a plugin, or Guttenberg, basically anywhere where you can't really control it... there's no point checking how the site would look if that CSS were different.

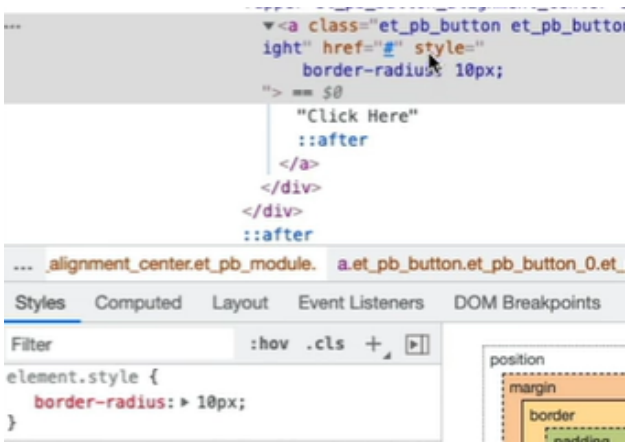
So, if you want to change the appearance of something, you can't just edit the existing code, but you need to write your own CSS to see how you can overwrite what's already there. And there are two different ways to add CSS here.

Adding inline CSS

First, at the very top here we have "elements.style".



You can simply click here and start typing the CSS property, and it will apply to the element that is currently selected. You will notice that it actually creates an inline style attribute within the HTML tag and adds your code there.

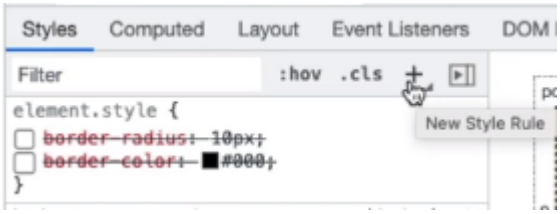


If you remember from the lesson about specificity, that inline CSS will have a great advantage over any code that is added somewhere inside the code. That means that this is not a very practical approach because in most cases, if you manage to define properties and get some results here in the preview, which is what you want - you won't be able to copy that code and place it inline on your HTML like you see here.

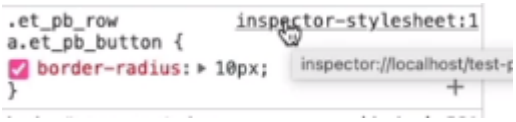
Because in most cases, we are working on a predefined HTML structure. We cannot modify it in this way. All we can do is add our own CSS code, not inline in the style attribute, but in a CSS file in a child theme, for example, or in a <style> tag inline on the page using Theme Options, but not on an element itself.

Adding custom CSS rulesets

So: to add our own code, we need to use the plus icon here.



It gives you the option to define your CSS selector. By default, it will list all the classes the selected element uses, but you can go ahead and define your own selector. That is the only way to truly test if your CSS will work. Once you define a selector and add some declarations, you will notice that it created that inspector stylesheet for you.

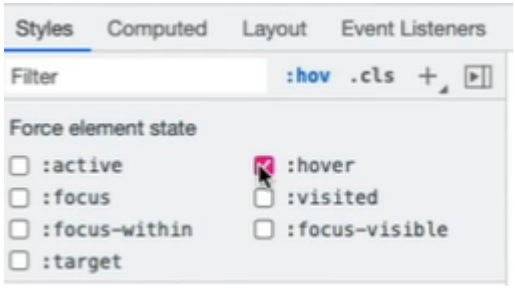


That is a temporary stylesheet, which will hold all your CSS modifications. So you can go ahead and edit multiple elements and then open that, and copy the CSS into Divi. Just remember, do not refresh your page before you copy the code, because it will be gone.

It's just a temporary preview.

Forcing :hover state

Another useful thing to know is that you can force a hover state on an element. Click on :hov next to the Filter bar in the Styles tab to access this menu.

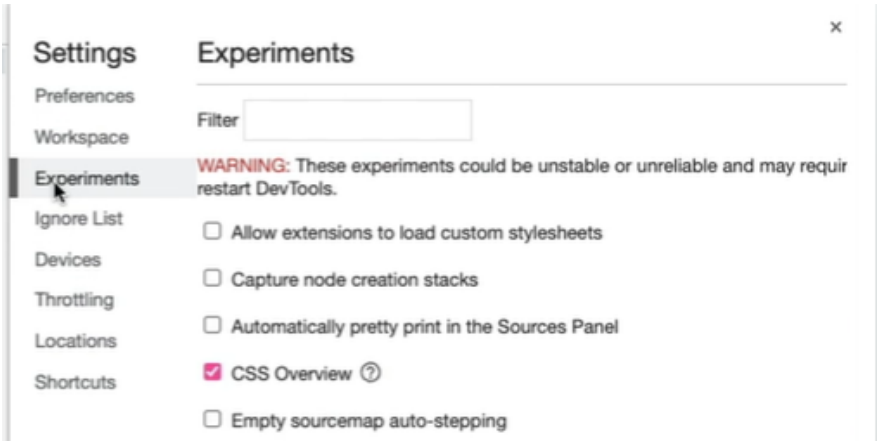


So when working with links or sub menus, you can trigger the :hover state on any element. And then you will see the CSS that is affecting that :hover state.

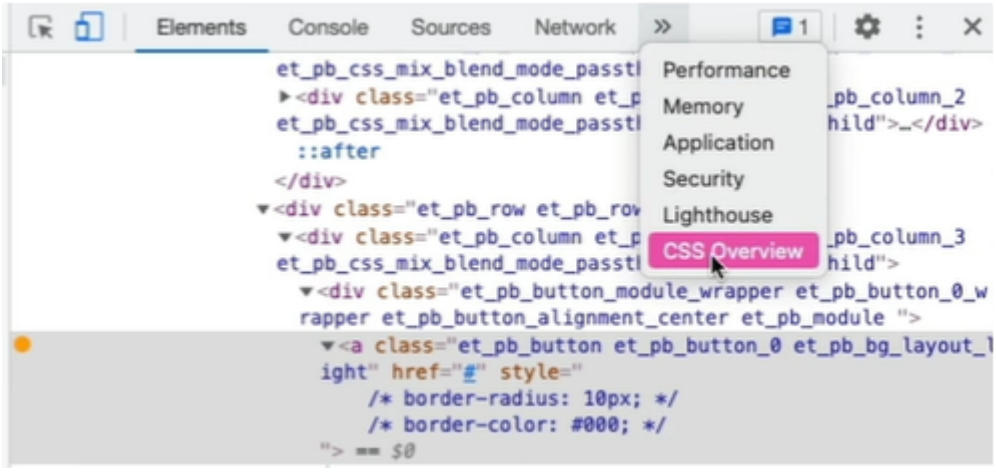
Enabling the CSS Overview

And one final tip or maybe not a tip, but just a feature I discovered recently inside the Browser Inspector, which is actually a beta feature.

So if we go to Settings (by clicking on the gear icon) and Experiments, we can enable the CSS Overview (it's disabled by default). I have it enabled here. And once you enable it, it will ask you to reload your page so it can load again with the overview.

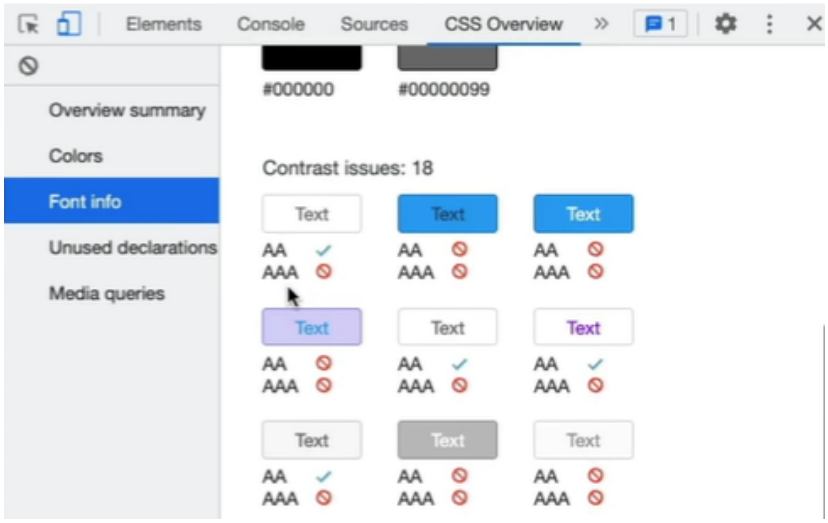


And now you have an additional section here, CSS overview.



And if you click capture overview, it will show you all the colors, all the fonts and all the font sizes you are using on the page. So it's just a fun and nice way to kind of check how

consistent you are in your design, but like I said: it's an experimental feature, but it does display some useful information you might find helpful.



Final thoughts

So I hope this lesson will give you the confidence and better understanding of this wonderful tool. And I want to encourage you to go and inspect, try to change, edit, tweak different things. See how your changes affect your website so that you can have a better understanding of how your website functions. That is very important if you want to get better in writing and be confident when it comes to using custom CSS. And remember, you cannot break anything in there.

Action Items

- ☐ Inspect your website! Go through all the different Dev Tools settings, change and tweak the CSS. Don't be afraid to edit and check different elements.
- ☐ Enable the CSS Overview experimental feature and check how well your website is doing and if you've been consistent with your color palettes and typography.