

CSS Essentials

Debugging



- 1. DevTools**
- 2. Inspecting the applied CSS**
- 3. Edit value**
- 4. Adding property**
- 5. Understand box model**
- 6. Solving specificity problem**

Section 1

DevTools

1. DevTools

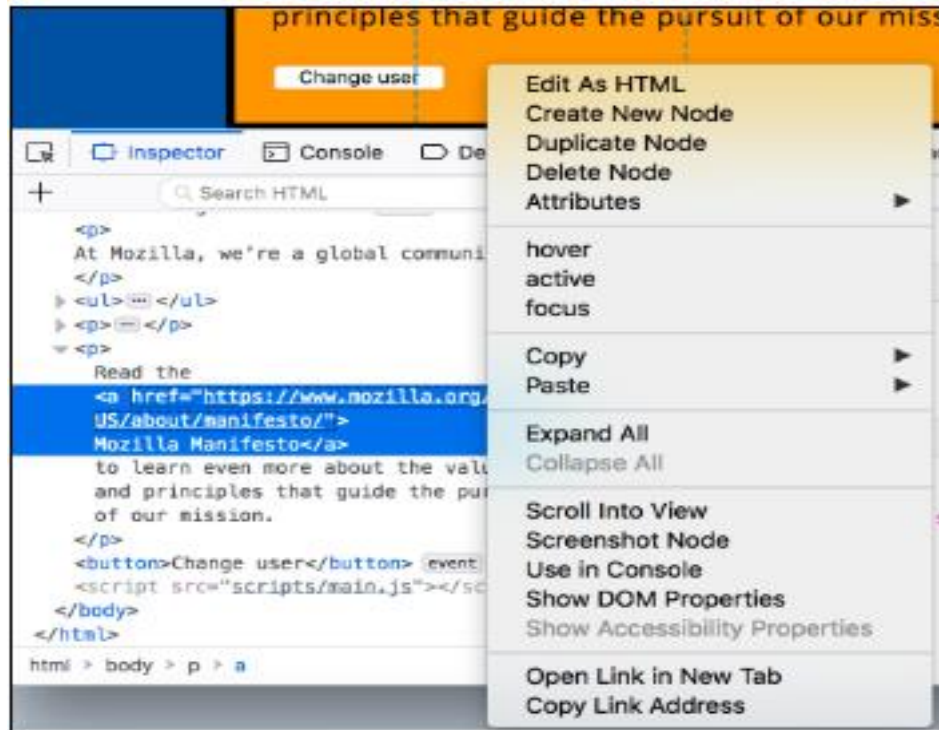
- Every modern web browser includes a powerful suite of **developer tools**. These tools do a range of things, from inspecting currently-loaded HTML, CSS and JavaScript to showing which assets the page has requested and how long they took to load
- The **devtools** live inside your browser in a subwindow that looks roughly like this, depending on what browser you are using

Section 2

Inspecting the applied CSS

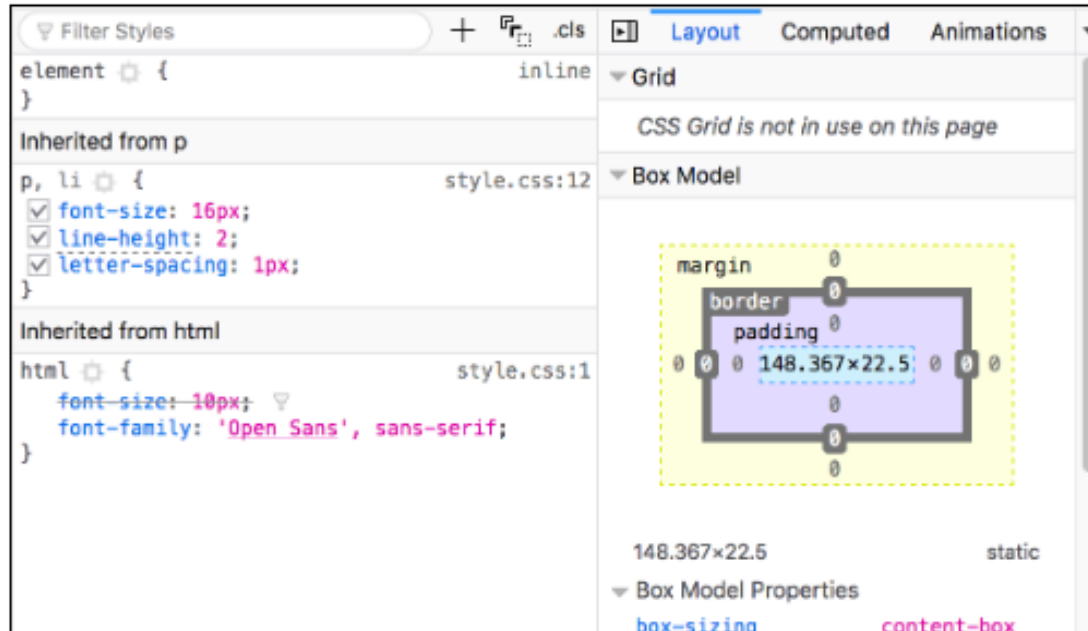
2. Inspecting the applied CSS

➤ Exploring the DOM inspector



2. Inspecting the applied CSS

- **Exploring the CSS editor:** By default, the CSS editor displays the CSS rules applied to the currently selected element



2. Inspecting the applied CSS

➤ Exploring the CSS editor

- The rules applied to the current element are shown in order of most-to-least-specific.
- Click the checkboxes next to each declaration to see what would happen if you removed the declaration.
- Click the little arrow next to each shorthand property to show the property's longhand equivalents.
- Click a property name or value to bring up a text box, where you can key in a new value to get a live preview of a style change.

2. Inspecting the applied CSS

➤ Exploring the CSS editor

- Next to each rule is the file name and line number the rule is defined in. Clicking that rule causes the dev tools to jump to show it in its own view, where it can generally be edited and saved.
- You can also click the closing curly brace of any rule to bring up a text box on a new line, where you can write a completely new declaration for your page

2. Inspecting the applied CSS

➤ Exploring the CSS editor – Clickable tab

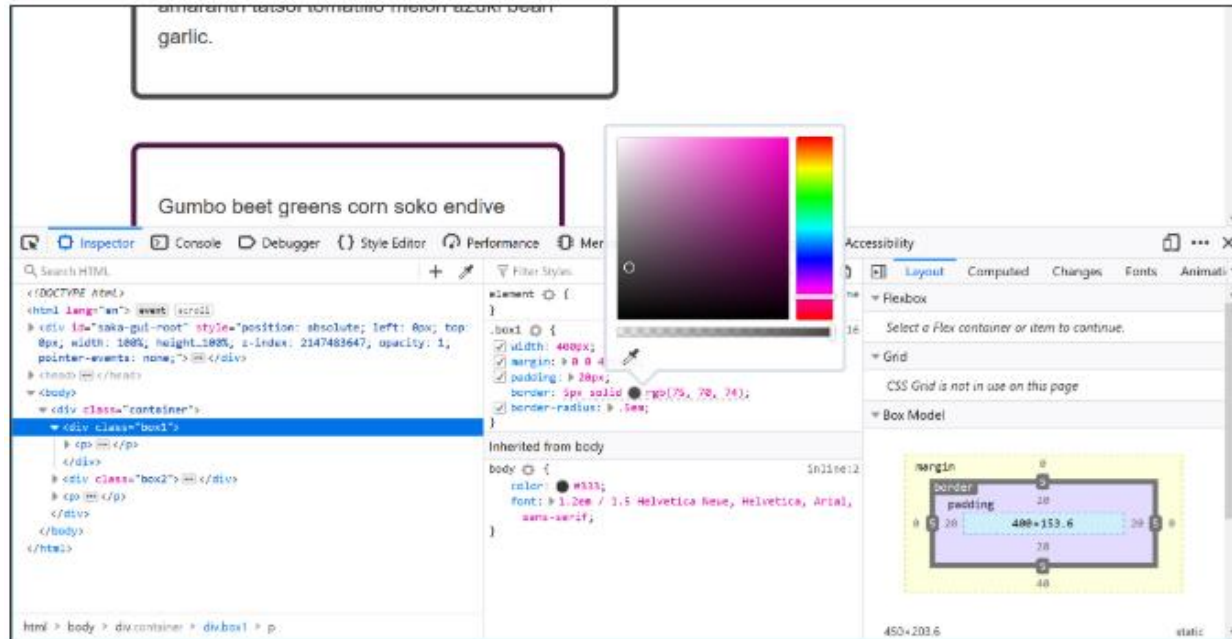
- **Computed:** This shows the computed styles for the currently selected element (the final, normalized values that the browser applies).
- **Layout:** In Firefox, this area includes two sections:
- **Box Model:** Represents visually the current element's box model, so you can see at a glance what padding, border and margin is applied to it, and how big its content is.
- **Grid:** If the page you are inspecting uses CSS Grid, this section allows you to view the grid details.
- **Fonts:** In Firefox, the Fonts tab shows the fonts applied to the current element.

Section 3

Editing value

3. Editing value

- In addition to turning properties on and off, you can **edit** their values, DevTools can save you a lot of time editing a stylesheet and reloading the page.

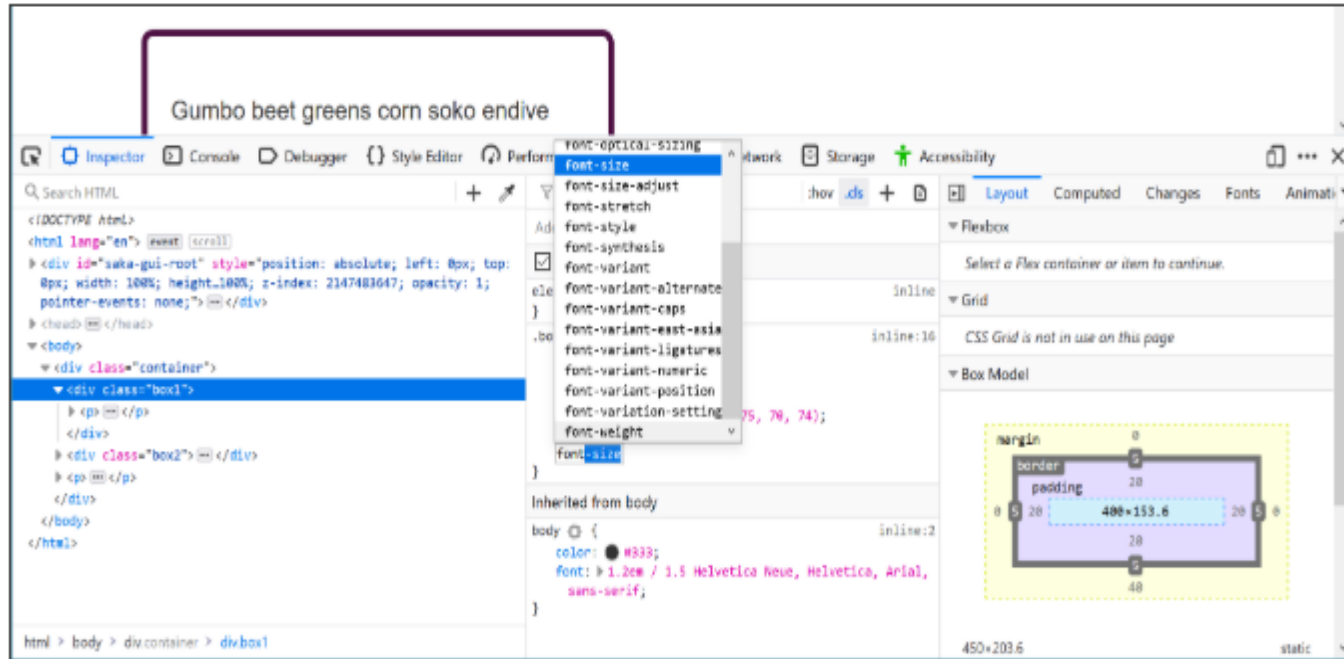


Section 4

Adding a new property

4. Adding a new property

- You can add properties using the DevTools. You can try this out in DevTools before adding it to your CSS file.

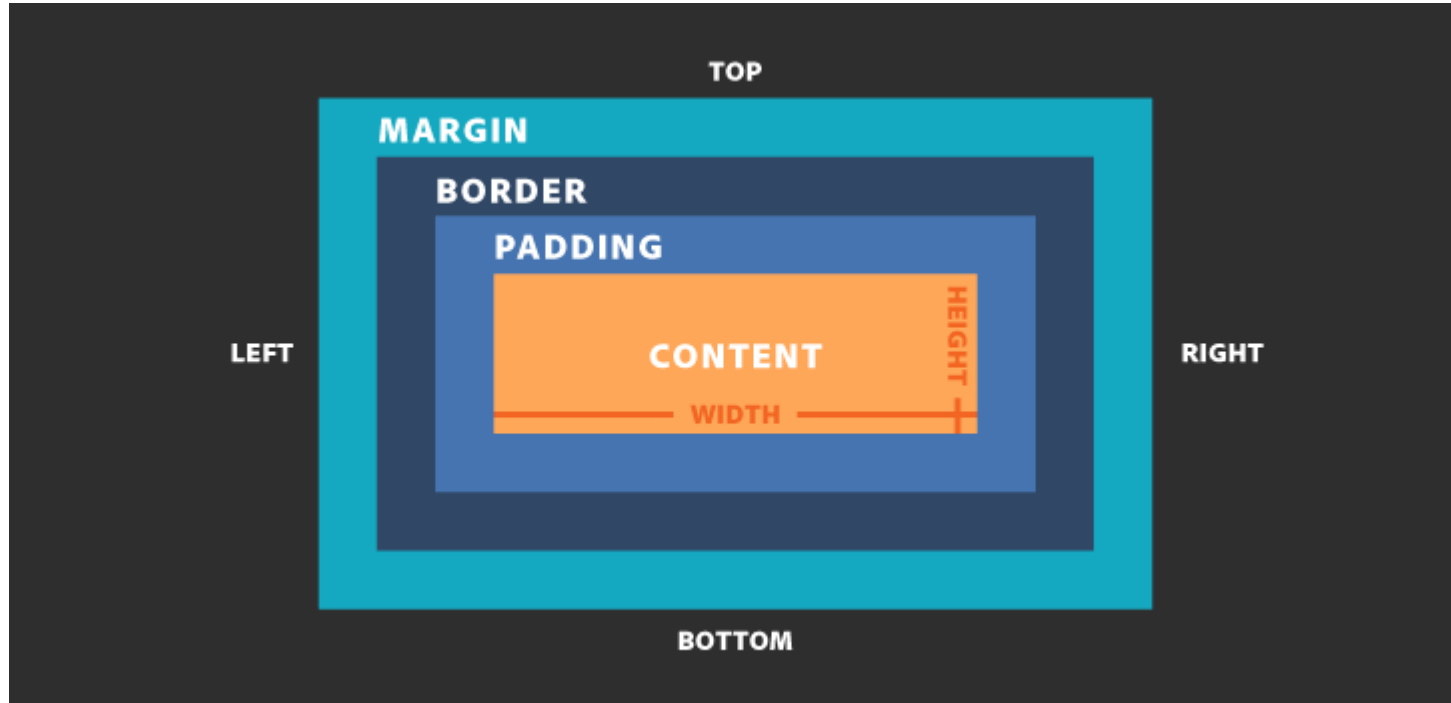


Section 5

Understanding box model

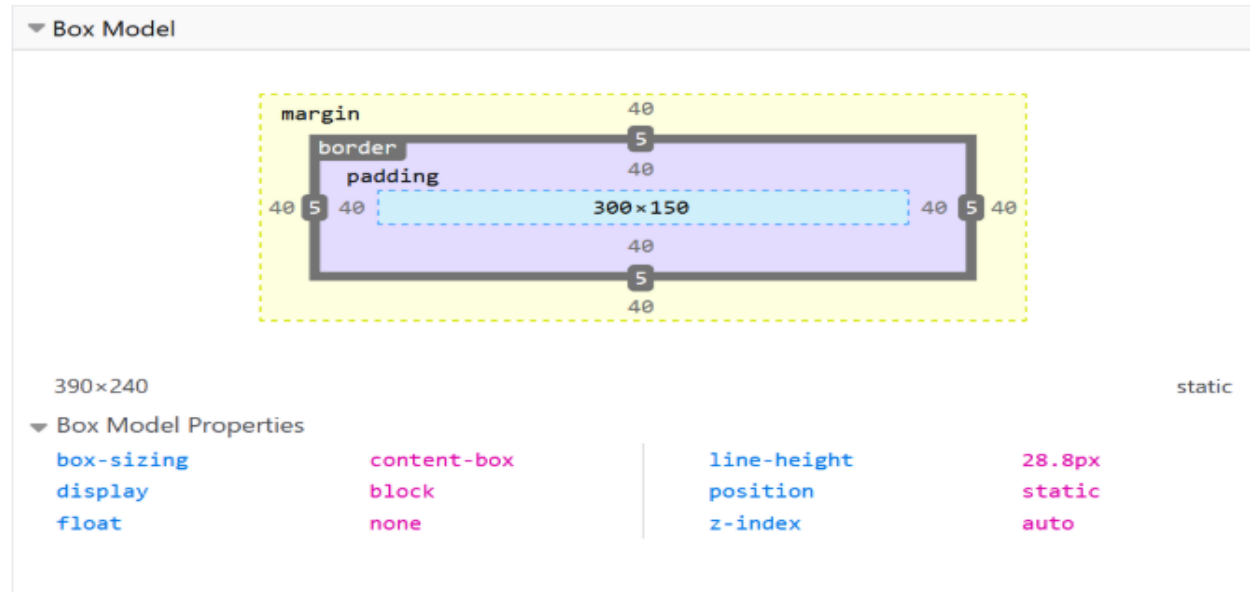
5. Understanding box model

➤ Box model



5. Understanding box model

- If you inspect an element in browser **DevTools**, you can see the size of the element plus its margin, padding, and border. Inspecting an element in this way is a great way to find out if your box is really the size you think it is

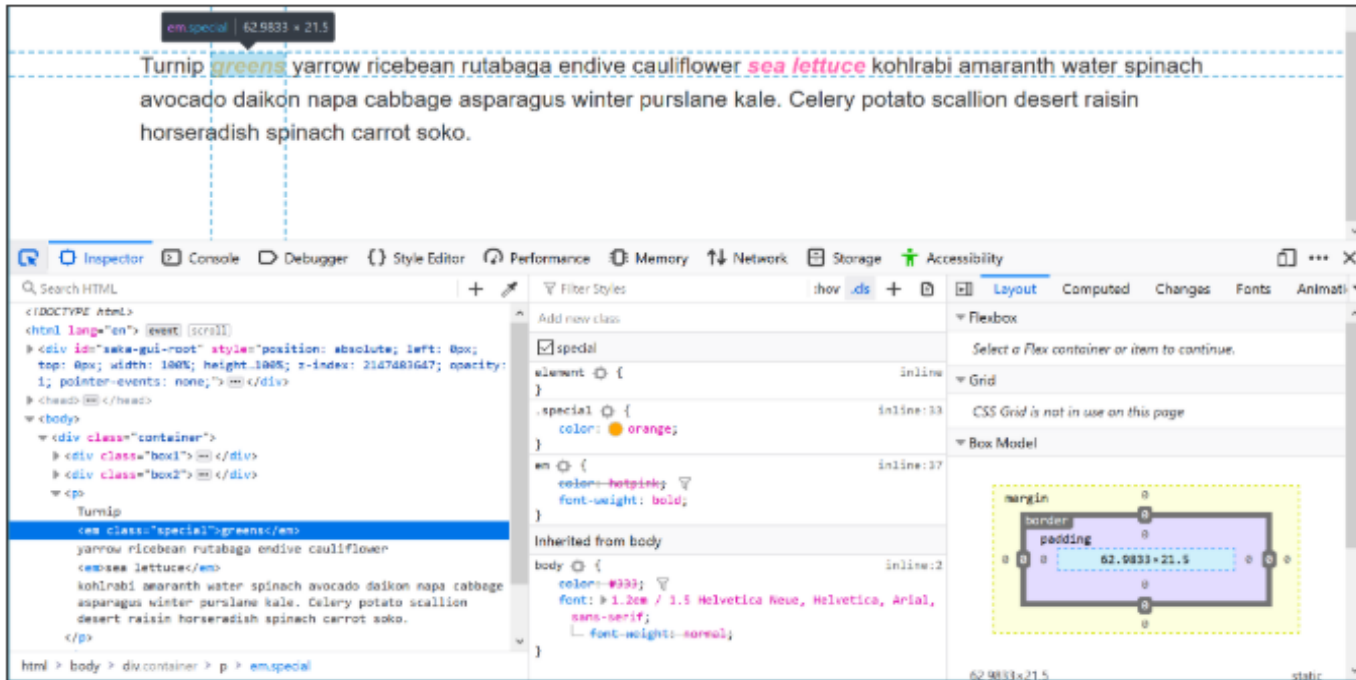


Section 6

Solving specificity problem

6. Solving specificity problem

- We can solve specificity problem with DevTool by inspecting element to see which style is applied to element.



Section 7

Debugging problems in CSS

7. Debugging problems in CSS

➤ **Some tips for debugging problems in CSS**

- Take a step back from the problem
- Do you have valid HTML and CSS?
- Is the property and value supported by the browser you are testing in?
- Is something else overriding your CSS?
- Make a reduced test case of the problem

Thank you

