14 common CSS mistakes

javascriptmastery

1. Not using fallback fonts

```css
body {
    font-family: Roboto;
}
```

Every computer doesn't have every font that you would use.

Also if you have imported a font from a cdn like google fonts, sometimes for some reason it may not get downloaded.

So its always better to use common fallback fonts similar to them. for eg: sans-serif font.

```css
body {
    font-family: Roboto, Arial, sans-serif;
}
```

2. Using px for everything

```css
p {
    font-size: 20px;
    line-height: 28px;
    padding: 16px;
}
```

Using px everywhere is not relevant, If we use 'px' units, every user will see the same sized elements, and it's not good because every user has a different monitor screen sizes.

Instead of 'px' units, if we use relative units like 'rem', 'em' the elements will scale appropriately according to the monitor screen size.

```css
p {
    font-size: 1.3rem;
    line-height: 1.6;
    padding: 1em;
}
```

3. Complicating Selectors

```css
header .navigation ul.nav-links {
    list-style-type: none;
}
```

When you can use direct class for a particular element, you should not complicate by nesting with different selectors.

There can be also Specificity problems.

You can just use a class for simplicity & it will be also easy to read & understand.

```css
.nav-links {
    list-style-type: none;
}
```

4. Repeting the same code

```css
#home-section {
  background: #F5F5F5;
  color: #202020;
}

#about-section {
  background: #F5F5F5;
  color: #202020;
}
```

When there are same styles for multiple elements, don't repeat your code, just group selectors.

To group selectors, separate each selector with a comma.

It will be better to group the selectors, to minimize the code.

```css
#home-section, #about-section {
  background: #F5F5F5;
  color: #202020;
}
```

5. Using Color names

```css
.box {
  background-color: Teal;
}
```

You should avoid using color names because every browser don't display the same color.

for eg: The browser displays what it thinks Teal color is.

If you want the colors to function the same in all the browsers, you should use hex, rgb or hsl.

```css
.box {
  background-color: #008080;
}
```

6. z-index mistakes

```css
.modal-container {
  z-index: 123;
}


.modal {
  z-index: 1234567;
}
```

Many developers use a really high z-index value to put an element in front.

And it becomes difficult when you want to put another elements in front of others. The z-index value starts increasing much higher.

The solution is to use moderate values, so that it doesn't become difficult for long run.

```css
.modal-container {
  z-index: 1;
}


.modal {
  z-index: 2;
}
```

7. Not using Shorthand

```
.container {
  margin-top: 1em;
  margin-right: 2em;
  margin-bottom: 0;
  margin-left: 3em;
}
```

There are many properties where you can use both shorthand and longhand methods.

You should try to use shorthand properties in most of the cases

```
.container {
  margin: 1em 2em 0 3em;
}
```

08. Using !important more oftenly

09. Using inline CSS

10. Not organizing your CSS

11. Inconsistent naming

12. Using CSS's @import

13. Using IDs instead of classes

14. Using only one Stylesheet for everything

THANK YOU FOR YOUR ATTENTION

Subscribe for more Advanced Tutorials

JavaScript Mastery

link in bio