

5 CSS MISTAKES YOU SHOULD AVOID



1

AVOID MAGIC NUMBERS

A magic number is a value that is used 'because it just works'.

Never use numbers just because they work. In this situation it's better to use **top: 100%**, which basically means 'all the way from the top'.



```
.dropdown-container { /* styles */}

.dropdown-menu {
  /* This is a magic number */
  margin-top: 47px;
}
```

✗ Don't do this



```
.dropdown-container { /* styles */}

.dropdown-menu {
  top: 100%;
}
```

✓ Do this

AVOID MIXING CONTAINER WITH CONTENT STYLES

Don't use location dependant styles on standalone components.

A component should look the same no matter where you put it. Instead use layout wrappers for specific use cases.

```
.form-input {  
  font-size: 14px;  
  padding: 4px 8px;  
  /*  
    🚫 Here we are mixing container with content 📌  
  */  
  margin-top: 20px;  
}
```

❌ Don't do this

```
.form-input-wrapper {  
  margin-top: 20px;  
}  
  
.form-input {  
  font-size: 14px;  
  padding: 4px 8px;  
}
```

✅ Do this

AVOID USING QUALIFIED SELECTORS

These are selectors who are needlessly prepended by an element. These are bad news because:

- ❌ They totally inhibit reusability on another element.
- ❌ They increase specificity.
- ❌ They increase browser workload (decreasing performance).

```
ul.nav {}  
a.button {}  
div.header {}
```

❌ Don't do this

```
.nav {}  
.button {}  
.header {}
```

✅ Do this

AVOID USING ABSOLUTE VALUES FOR LINE HEIGHT

Line heights should always be set relatively in order to make them more flexible. If you ever change the font-size of a h1, you want to know that your line-height will track it.

```
h1 {  
  font-size: 24px;  
  line-height: 32px;  
}  
  
.site-title {  
  font-size: 36px;  
  line-height: 48px;  
}
```

✗ Don't do this

```
h1 {  
  font-size: 24px;  
  line-height: 1.333;  
}  
  
.site-title {  
  font-size: 36px;  
}
```

✓ Do this

AVOID LOOSE CLASS NAMES

Loose class names are very bad for two main reasons:

- ❌ You can't predict its purpose from the class alone.
- ❌ It's so vague that it could very easily be redefined by another dev.



```
/* What does this classes mean?*/  
.card {}  
.modal {}  
.user {}
```



Don't do this



```
.user-post-card {}  
.confirmation-modal {}  
.user-avatar {}
```



Do this