

Getting Started with Variables

Why variables are important and how to get started with them.

The Need for Variables

If you're new to variables in preprocessors or native CSS, here are a few reasons why variables are important.

- **More readable code**

Without saying much you can quickly tell how readable and more maintainable variables make any code base.

- **Ease of change across large documents**

If you have all constants saved in a separate file , then you do not have to jump through thousands lines of code when you want make a change to a variable. It becomes super-easy. Just change it in one place.

- **Spot typos faster**

It is pain to search through lines of codes trying to spot an error. It is more annoying if this is an error due to typos. They are difficult to spot, and the good use of variables eliminates the hassles.

To this end, readability and maintainability are the big wins. Thanks to CSS variables, now we can have these with native CSS too.

Defining CSS variables

Let me start with something you're already familiar with i.e variables in Javascript.

Assuming you write Javascript, a simple variable may be declared like so:

```
var amAwesome;
```

and then you assign it some value like so:

```
amAwesome = "awesome string"
```

In CSS, a CSS variable is any “property” whose name begins with two dashes.

```
/*can you spot the variable here? */  
.block {  
  color: #8cacea;  
  --color: blue  
}
```



Scoping CSS Variables

There's one more thing to point your attention to.

Remember that In Javascript, variables have a scope. They may either have a `global` or `local` scope.

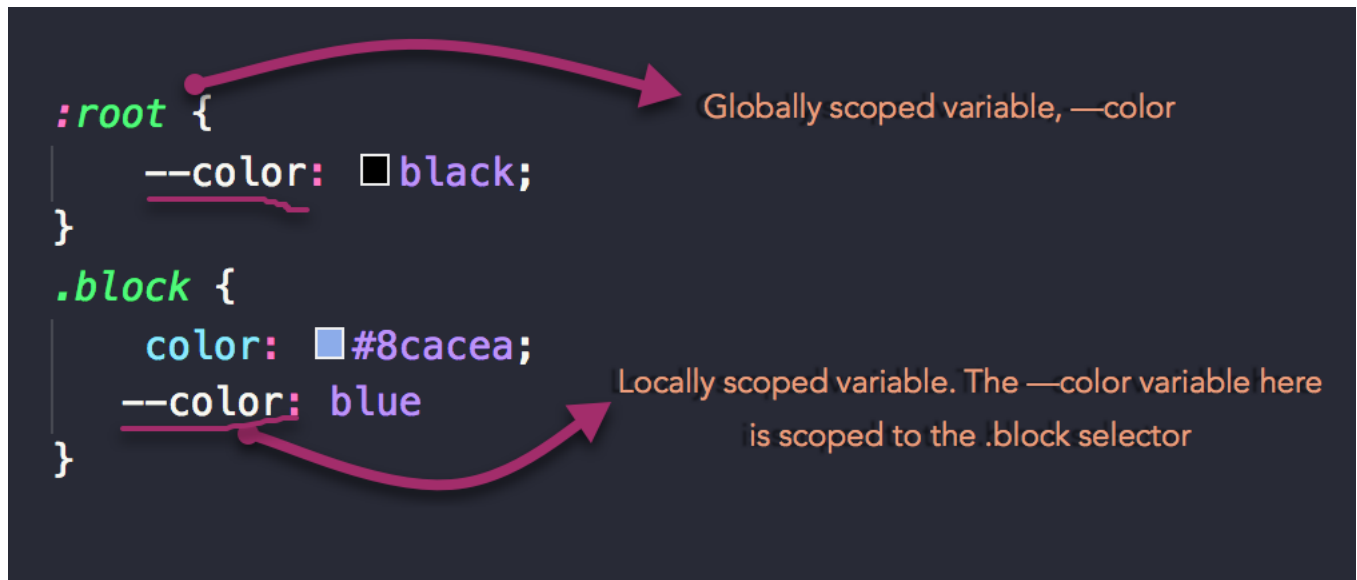
The same may be said of CSS variables.

Consider the example below:

```
:root {  
  --main-color: red  
}
```

The `:root` selector allows you to target the highest-level element in the DOM, or document tree.

So, variables declared in this way are kind of scoped to the global scope Got that?



Example 1

Assuming you wanted to create a CSS variable that stored the primary color of a themed site. How would you go about this?

1. You create the scoped selector. Use `:root` for a 'global' variable

```
:root {  
  
}
```

2. Define the variable

```
:root {  
  --primary-color: red  
}
```

Remember, a CSS variable is any “property” whose name begins with two dashes e.g `--color`

That was simple.