

ids

For years, developers have treated IDs as the way to say "that thing!" However, IDs have a downside: they are completely unique and therefore cannot be reused. You could potentially style every element on your page with an ID selector but you would lose a lot of the benefit of CSS in the process.

One of CSS's benefits is the ability to reuse style rules in multiple places. When you start out with an ID selector, you're automatically limiting that style to a single element. Suppose you have this:

```
#header a {  
  color: black;  
}
```

This style is only useful within the element with the ID of `header`. But now suppose that you want another section of the page to be styled the same way, you'll probably end up defining a new class that does the same thing, such as:

```
#header a,  
.callout a {  
  color: black;  
}
```

Once you've gotten to this point, you might as well just use the class and not mention the ID:

```
.callout a {  
  color: black;  
}
```

Eventually you will end up needing or wanting to reuse the style specified with the ID, and you'll end up defining a class for that purpose. By not using IDs from the start, you allow for the maximum level of reusability with your CSS.

Rule Details

Rule ID: `ids`

This rule is aimed at improving maintainability by flagging the use of IDs in selectors. Every instance of an ID will result in a warning.

The following patterns are considered warnings:

```
#mybox {  
  display: block;  
}  
  
.mybox #go {  
  color: red;  
}
```

Additional Reading

- [Don't use IDs in CSS selectors?](#)
- [Don't use ID selectors in CSS](#)
- [CSS Specificity Should Be \(Mostly\) Irrelevant](#)