



# CSS SELECTORS

A SUPER IMPORTANT PART OF CSS!



# CSS RULES


Everything you do in CSS follows this basic pattern:



```
selector {  
  property: value;  
}
```

# CSS RULES

Everything you do in CSS follows this basic pattern:

A hand with the index finger pointing towards the left side of a code editor window. The window has a pink title bar with two small circles and a horizontal line. The code inside is a CSS rule: selector { property: value; }

```
selector {  
  property: value;  
}
```

# CSS RULES

Make all `<h1>` elements purple



```
h1 {  
  color: purple;  
}
```

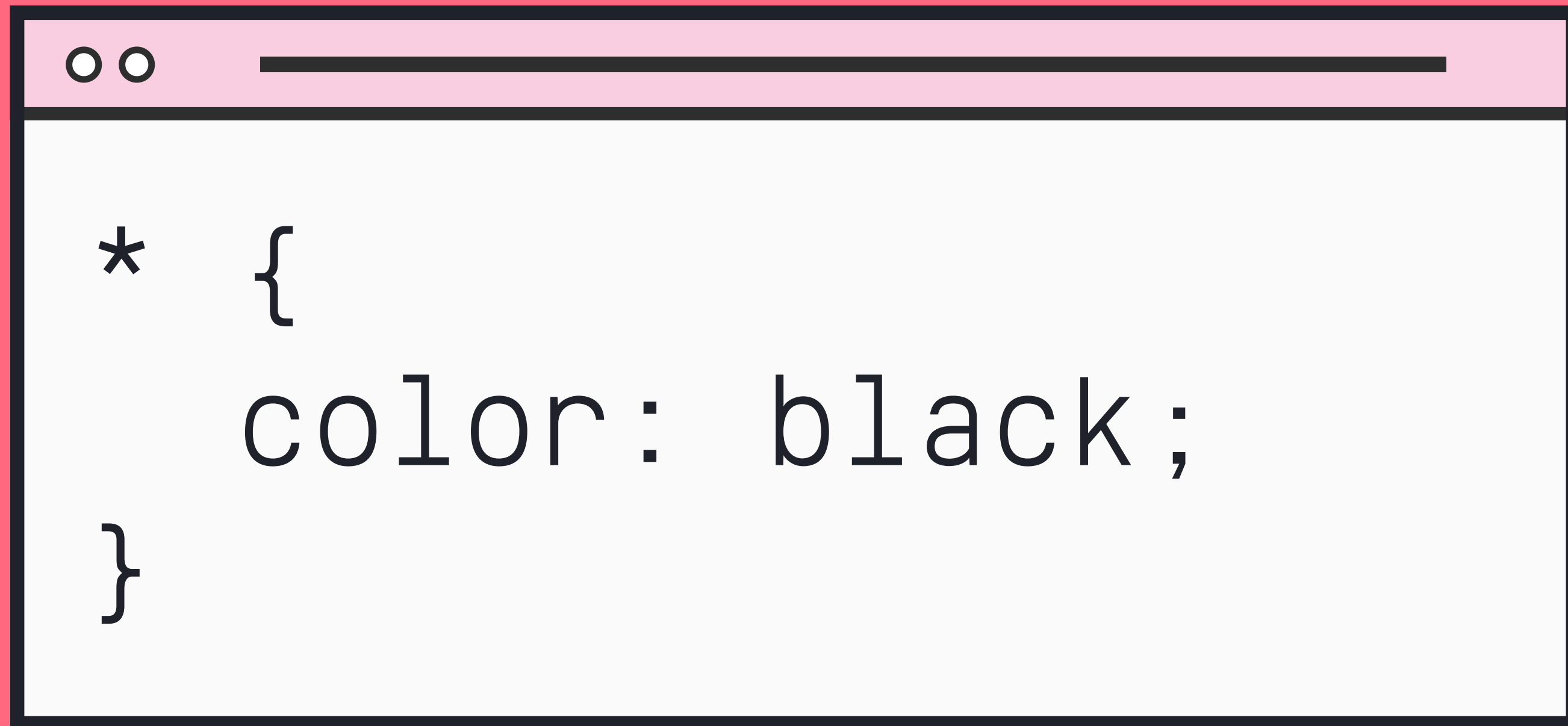
# FANCIER!

Select every other text input  
and give it a red border:

```
input[type="text"]:nth-of-type(2n){  
  border:2px solid red;  
}
```

# UNIVERSAL SELECTOR

*Select everything!*



```
* {  
  color: black;  
}
```

# ELEMENT SELECTOR

*Select all images*

```
img {  
  width: 100px;  
  height: 200px;  
}
```

# SELECTOR LIST

*Select all h1's and h2's*




```
h1, h2 {  
  color: magenta;  
}
```



# CLASS SELECTOR

*Select elements with class of 'complete'*



```
.complete {  
    color: green;  
}
```

# ID SELECTOR

*Select the element with id of 'logout'*

```
#logout {  
  color: orange;  
  height: 200px;  
}
```

# DESCENDANT SELECTOR

*Select all <a>'s that are nested inside an <li>*

```
li a {  
    color: teal;  
}
```

# ADJACENT SELECTOR

*Select only the paragraphs that are immediately preceded by an <h1>*

```
h1 + p {  
  color: red;  
}
```

# DIRECT CHILD

Select only the `<li>`'s that are direct children of a `<div>` element



```
div > li {  
  color: white;  
}
```

# ATTRIBUTE SELECTOR

*Select all input elements where the type attribute is set to "text"*

```
input[type="text"] {  
    width: 300px;  
    color: yellow;  
}
```

# PSEUDO CLASSES

keyword added to a selector that specifies a special state of the selected element(s)

- :active
- :checked
- :first
- :first-child
- :hover
- :not()
- :nth-child()
- :nth-of-type()

# PSEUDO ELEMENTS

Keyword added to a selector that lets you style a particular part of selected element(s)

- `::after`
- `::before`
- `::first-letter`
- `::first-line`
- `::selection`



What happens when  
conflicting styles  
target the same  
elements?

# THE CASCADE

The order your styles are declared in and linked to matters!

```
● ● ●  
  
h1 {  
  color: red;  
}  
h1 {  
  color: purple;  
}
```

Purple wins!

# SPECIFICITY

Specificity is how the browser decides which rules to apply when multiple rules could apply to the same element.

It is a measure of how specific a given selector is. The more specific selector "wins"

# SPECIFICITY

```
p {  
  color: yellow;  
}
```

Element Selector

```
section p {  
  color: teal;  
}
```

Element Selector  
+ Element Selector



**ID**



**CLASS**



**ELEMENT**

```
section p {  
  color: teal;  
}
```

0



ID Selectors

0



Class,  
Attribute, &  
Pseudo-Class  
Selectors

2



Element and  
Pseudo-Element  
Selectors

```
#submit {  
  color: olive;  
}
```

1

---

ID Selectors

0

---

Class,  
Attribute, &  
Pseudo-Class  
Selectors

0

---

Element and  
Pseudo-Element  
Selectors



```
nav a.active {  
  color: orange;  
}
```

0



ID Selectors

1



Class,  
Attribute, &  
Pseudo-Class  
Selectors

2



Element and  
Pseudo-Element  
Selectors



# INLINE STYLES

---

Inline Styles

---

ID Selectors

---

Class,  
Attribute, &  
Pseudo-Class  
Selectors

---

Element and  
Pseudo-Element  
Selectors

# INHERITANCE

