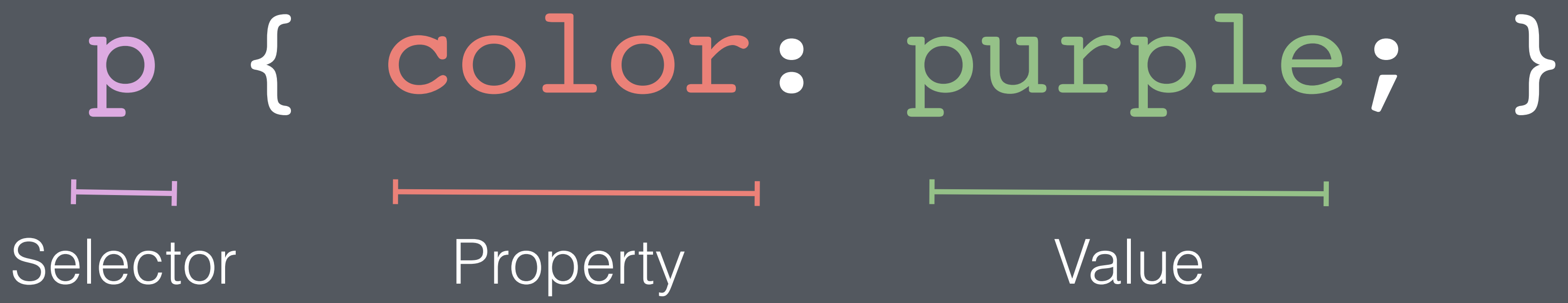
CSS

WHAT IS CSS?

- Stands for **C**ascading **S**tyle **S**heets
- Proposed in 1994 by Håkon Lie (“How Con Lee”)
- Stylesheet language that *describes* HTML markup
- Separates document content (HTML) from document styling (CSS)

ANATOMY OF CSS



```
p { color: purple; }
```

Selector

Property

Value

What styles look like

```
p {
  color: white;
  text-decoration: underline;
}
```

Which one is the selector, value, property?

SELECTORS

```
p {
  color: white;
  text-decoration: underline;
}
```

Selectors tell the browser where to apply the style.

SELECTORS

```
p, h2 {
   color: white;
   text-decoration: underline;
}
```

You can have two selectors together - mixing HTML elements if you like.

SELECTORS

```
p, h2, div, li {
   color: white;
   text-decoration: underline;
}
```

You can have an unlimited number of selectors together, just separate with a comma.

SELECTORS

```
div > p {
  color: white;
  text-decoration: underline;
}
```

The symbol > can be used to select a child element of another. This would select all p elements within divs.

CLASSES

Classes allow us to create groups of elements to style

```
<p class=“red”>Red paragraph.</p>
<p>Normal paragraph.</p>
<p class=“red”>Red paragraph.</p>
```

We can then select and style those elements differently

```
.red {
  color: red;
  text-decoration: underline;
}
```

IDS

IDs allow us to pick out a single element

```
<p>Normal paragraph.</p>
<p id=“blue”>Blue paragraph.</p>
<p>Normal paragraph.</p>
```

We can also select elements by ID in our CSS

```
#blue {
   color: blue;
   text-decoration: underline;
}
```

Remember!

CLASSES

Group elements with the same style

IDS

Are only used for one specific individual unique solitary element

SELECTORS

PROPERTIES

VALUES

CLASSES

IDS

COLORS

FONTS

DOM TREE

CASCADING

SPECIFICITY

LINKING CSS

YOUR TURN

FORK ON CODEPEN

Properties

```css
p {
  color: white;
  text-decoration: underline;
}
```

Properties are parts of the CSS spec that control style behavior. Most are logically named, but not all.

PROPERTIES

are

ALWAYS

followed by a colon and a

VALUE

then a semicolon

PROPERTIES + VALUES

```
p {
  color: white;
}
```

You must have at least one property / value pair per selector (otherwise nothing happens). They MUST end with a semicolon - ALWAYS!

PROPERTIES + VALUES

```
p {
  color: white;
  font-weight: bold;
  padding: 15px;
  text-decoration: underline;
}
```

You can have an unlimited amount of property/value pairs in a style declaration.

PROPERTIES + VALUES

ALWAYS
indent your properties under your selectors!

```
p {
   color: white;
   font-weight: bold;
}
```

GOOD!

```
p {
color: white;
font-weight: bold;
}
```

BAD!

There is a universe of CSS

PROPERTIES + VALUES

SOME IMPORTANT PROPERTIES

`background` - what's behind an elements
`border` - edge of an elements
`color` - sets color of text
`font` - controls font family, size, style, weight
`height` - how tall an element is
`margin` - space between two elements
`padding` - space inside of an element
`text-align` - which direction text lines up
`text-decoration` - underlines or crosses out text
`text-transform` - controls capitalization / uppercase

CSS COLORS

Colors can be specified using:

- Color keyword: red
- RGB value: rgb(255, 0, 0)
- Hex code: #FF0000

To color text use the color property

To color backgrounds use background-color

FONTS AND TEXT

font-family:

- Applies a particular font to the element
- Accepts multiple values.
- Always end with serif, sans-serif, or monospace

```
p {
  font-family: ‘Comic Sans MS’, helvetica, sans-serif;
}
```

FONTS AND TEXT

font-size

- The size of the text
- Different types of units of size

px:

  the pixel size of the text

em / %:

  the size relative to the parent element

rem:

  the size relative to the root element

FONTS AND TEXT

font-weight:

- Sets the “thickness” of the font
- Normal and **bold**; values 100-900

text-align:

- How the text is oriented within the containing block
- center, left, right, justify

Fonts and Text

`text-decoration:`

- Controls extra-textual visual emphasis (underlines)

`text-style:`

- Sets a “slant” to the font
- Normal and *italic*

QUICK STYLING TIPS

- Always put a new style on a new line

- Try to alphabetize your properties

- Use external stylesheets

- Use white space so people can read your code - think logically.

SELECTORS
PROPERTIES
VALUES
CLASSES
IDS
COLORS
FONTS
DOM TREE
CASCADING
SPECIFICITY
LINKING CSS

YOUR TURN

FORK ON CODEPEN

CASCADING

What does that mean?

Think of the DOM tree as a real tree

```
<main>
 <section>
  <div class=“a”>
   Content A
  </div>
  <div class=“b”>
   Content B
  </div>
 </section>
</main>
```

DOM tree

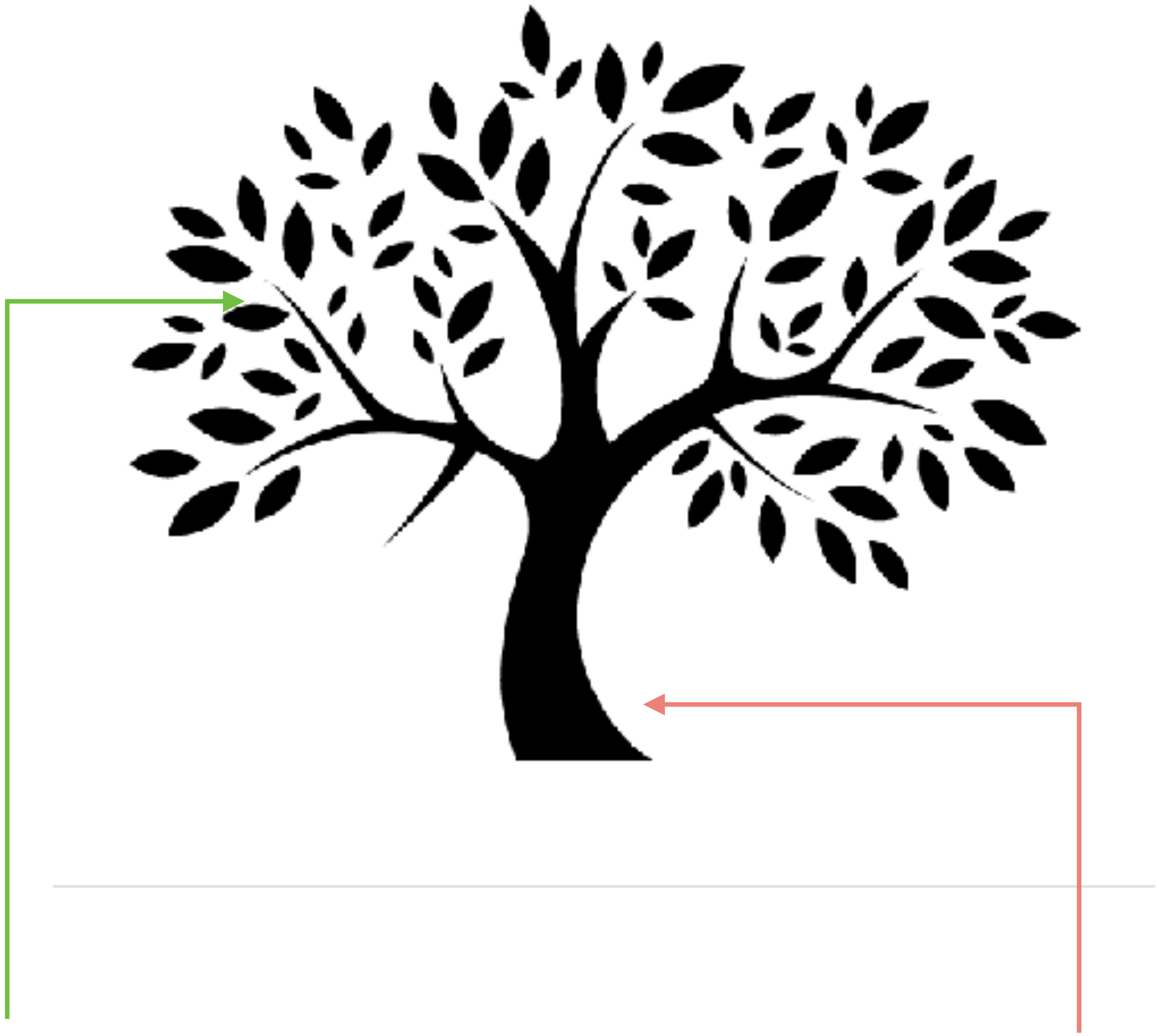
Real tree

Styles

CASCADE

through your document following basic ‘trunk’ elements down to the ‘leaves’

What if I apply my styles at each of these spots? What happens?



<div class=“a”>

<main>

The more…

GENERAL

your selector is (like `body`, `img`, or `p`) the more your style will

CASCADE

throughout your HTML document(s).

The more…

SPECIFIC

your selector is (like `#my-id, .my-class`) the more your style will

NOT CASCADE

throughout your HTML document(s).

ID SELECTORS

(like `#my-id`)

OVERRIDE

CLASS SELECTORS

(like `.my-class`)

OVERRIDE

GENERAL SELECTORS

(like `div, h2, body`)

NOTES TO REMEMBER

- If two selectors apply to the same element, the more specific selector wins

- When selectors are equally specific, the last rule is the one that counts

SELECTORS

PROPERTIES

VALUES

CLASSES

IDS

COLORS

FONTS

DOM TREE

CASCADING

SPECIFICITY

LINKING CSS

LINKING FILES

BASIC LINKAGES

For stylesheets (CSS):

```
<link rel="stylesheet" href="styles/main.css">
```

ABSOLUTE VS RELATIVE

Relative paths are missing the http:// stuff:

```
<link rel="stylesheet" href="styles/main.css">
```

Absolute paths have the http:// stuff. You'll generally see them with anchor tags, like this:

```
<a href="https://google.com">Google</a>
```

Absolute vs Relative

Relative paths can use dot notation to reference folders above them

```
<link rel="stylesheet" href="../styles/main.css">
```

SELECTORS
PROPERTIES
VALUES
CLASSES
IDS
COLORS
FONTS
DOM TREE
CASCADING
SPECIFICITY
LINKING CSS