



Developing Reusable Components

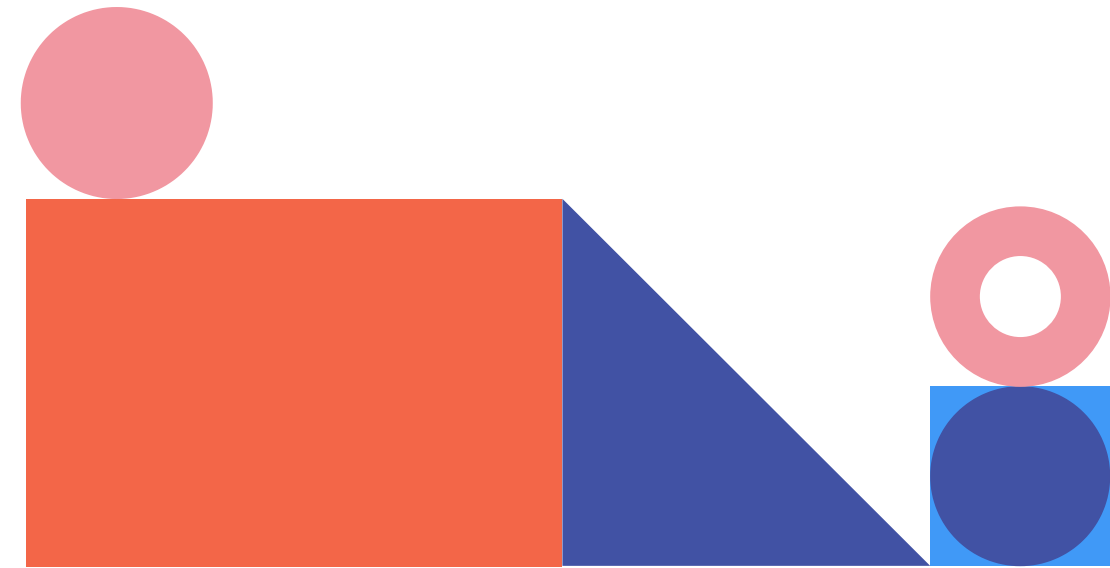
Let's take the components we just designed in Figma and build them using React styled components.

How we used to apply CSS

External CSS stylesheet with a link in the head HTML element

Within the HTML file in a `<style>` tag

Inline on the HTML elements `<h1 style="margin-right: 4px">`



CSS Specificity

1. Type selectors and pseudo-elements
2. Class selectors, attribute selectors, and pseudo-classes
3. ID selectors

```
/* Type selectors and pseudo-elements */  
h1 {  
  ...;  
}  
  
h1::before {  
  ...;  
}  
  
/* Class selectors, attribute selectors, and pseudo-classes */  
.cat {  
  ...;  
}  
  
input[type="radio"] {  
  ...;  
}  
  
h1:hover {  
  ...;  
}  
  
/* ID selectors */  
#cat {  
  ...;  
}
```

Problems with CSS

Styles slowly became decentralized and hard to remove or update.

A lack of knowledge about CSS specificity leads to !important.

```
li {  
  color: blue;  
}
```

Problems with CSS

Styles slowly became decentralized and hard to remove or update.

A lack of knowledge about CSS specificity leads to !important.

```
ul.shopping-cart > li {  
  color: blue;  
}  
  
li.shopping-cart-item {  
  color: blue !important;  
}  
  
.shopping-cart > .shopping-cart-item {  
  color: blue;  
}  
  
.shopping-cart-item {  
  color: blue;  
}
```

CSS naming architectures

CSS naming architectures were created to superficially scope CSS styling.

They're simply a naming convention but won't prevent leaked styling.

```
.shopping-cart__shopping-cart-item {  
  color: blue;  
}  
  
.shopping-cart__shopping-cart-item--out-of-stock {  
  color: red;  
}
```

CSS pre-processors

Provide styling capabilities such as nesting, mixins, and more which allow more control over how our CSS styles are applied.

```
.shopping-cart {  
  &__item {  
    color: blue;  
  
    &--selected {  
      color: red;  
    }  
  }  
}
```

CSS-in-JS

JavaScript is used to style our components. When the components are parsed, CSS is generated and attached to the DOM.

