SVELTE • STYLING

Scoped styles

ON THIS PAGE

Svelte components can include a `<style>` element containing CSS that belongs to the component. This CSS is *scoped* by default, meaning that styles will not apply to any elements on the page outside the component in question.

This works by adding a class to affected elements, which is based on a hash of the component styles (e.g. `svelte-123xyz`).

```
<style>
  p {
    /* this will only affect <p> elements in this component */
    color: burlywood;
  }
</style>
```

Specificity

Each scoped selector receives a specificity increase of 0-1-0, as a result of the scoping class (e.g. `.svelte-123xyz`) being added to the selector. This means that (for example) a `p` selector defined in a component will take precedence over a `p` selector defined in a global stylesheet, even if the global stylesheet is loaded later.

In some cases, the scoping class must be added to a selector multiple times, but after the first occurrence it is added with `:where(.svelte-xyz123)` in order to not increase specificity further.

If a component defines `@keyframes` , the name is scoped to the component using the same hashing approach. Any `animation` rules in the component will be similarly adjusted:

```
<style>
  .bouncy {
    animation: bounce 10s;
  }

  /* these keyframes are only accessible inside this component */
  @keyframes bounce {
    /* ... *.
  }
</style>
```

Edit this page on GitHub

PREVIOUS
style:

NEXT
Global styles