



# .svelte files

ON THIS PAGE



Components are the building blocks of Svelte applications. They are written into `.svelte` files, using a superset of HTML.

All three sections — `script`, `styles` and `markup` — are optional.

MyComponent.svelte

```
<script module>
  // module-level logic goes here
  // (you will rarely use this)
</script>

<script>
  // instance-level logic goes here
</script>

<!-- markup (zero or more items) goes here -->

<style>
  /* styles go here */
</style>
```

## <script>

A `<script>` block contains JavaScript (or TypeScript, when adding the `lang="ts"` attribute) that runs when a component instance is created. Variables declared (or imported) at the top level can be referenced in the component's markup.

A `<script>` tag with a `module` attribute runs once when the module first evaluates, rather than for each component instance. Variables declared in this block can be referenced elsewhere in the component, but not vice versa.

```
<script module>
  let total = 0;
</script>

<script>
  total += 1;
  console.log(`instantiated ${total} times`);
</script>
```

You can `export` bindings from this block, and they will become exports of the compiled module. You cannot `export default`, since the default export is the component itself.

If you are using TypeScript and import such exports from a `module` block into a `.ts` file, make sure to have your editor setup so that TypeScript knows about them. This is the case for our VS Code extension and the IntelliJ plugin, but in other cases you might need to setup our [TypeScript editor plugin](#).

Legacy mode

show all

## `<style>`

CSS inside a `<style>` block will be scoped to that component.

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

 [Edit this page on GitHub](#)

PREVIOUS

[Getting started](#)

NEXT

[.svelte.js and .svelte.ts files](#)