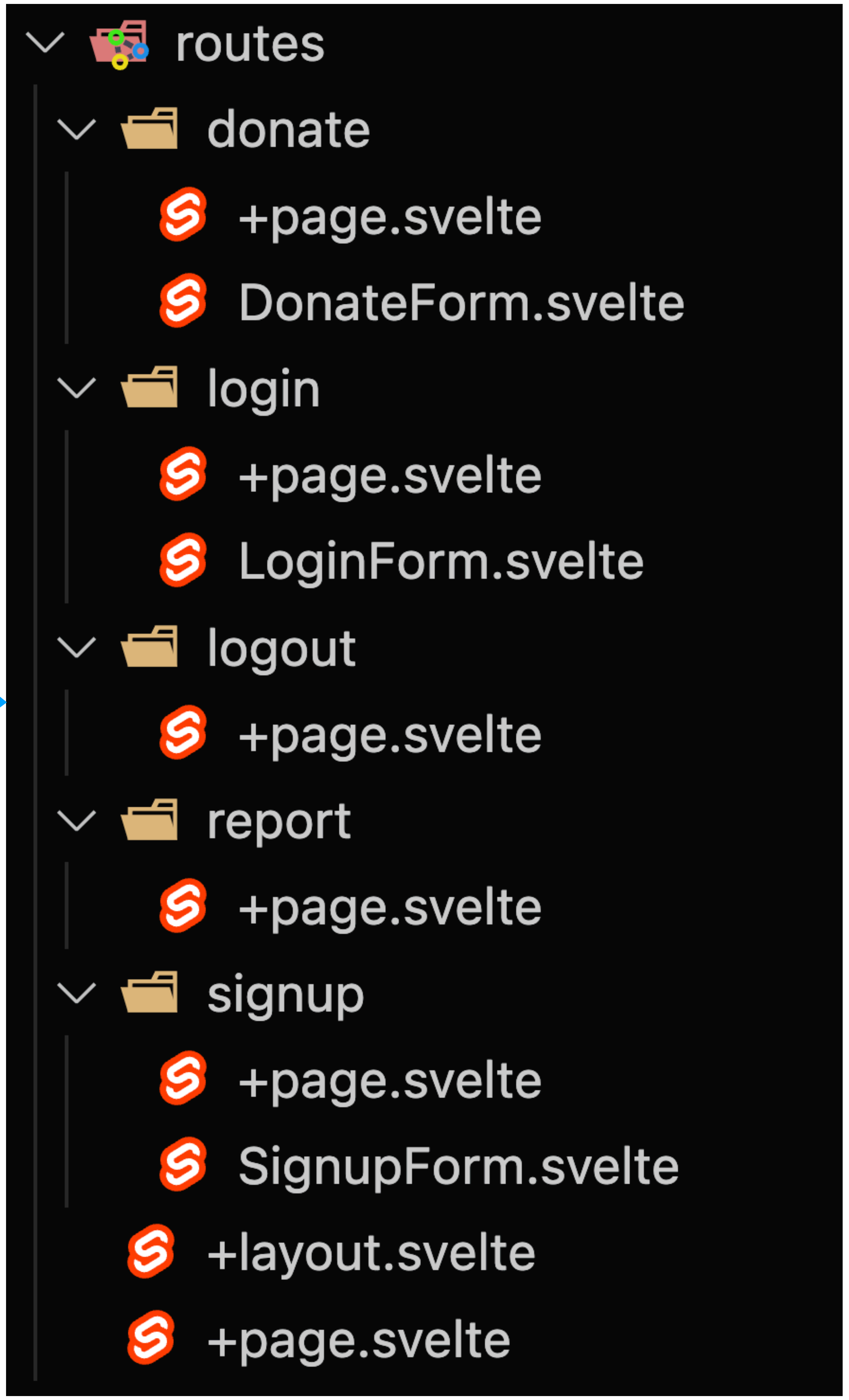
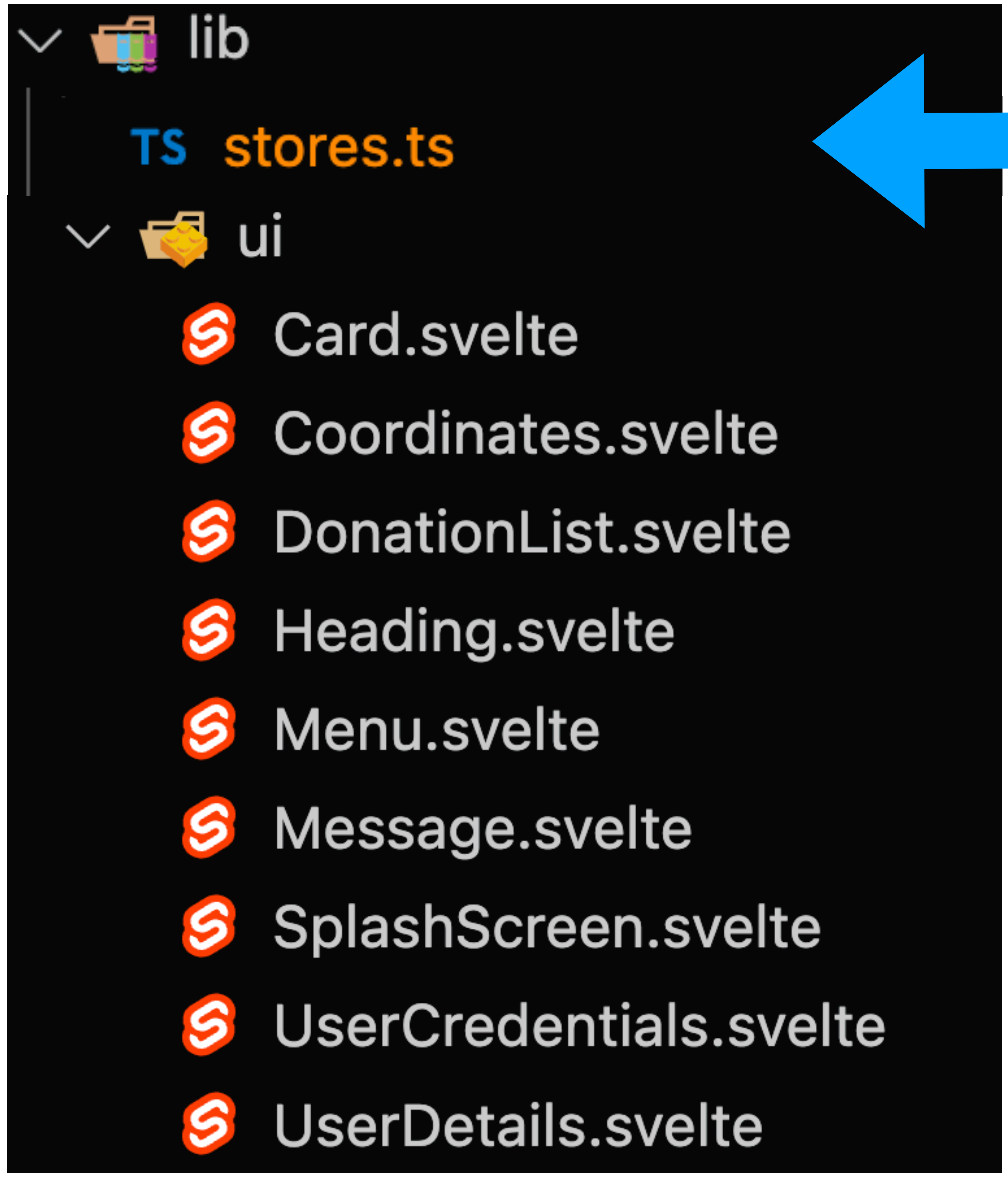
Pages & Components

Creating and assembling Svelte Components

SvelteKit Web Application


```
routes
  donate
    +page.svelte
    DonateForm.svelte
  login
    +page.svelte
    LoginForm.svelte
  logout
    +page.svelte
  report
    +page.svelte
  signup
    +page.svelte
    SignupForm.svelte
  +layout.svelte
  +page.svelte
```

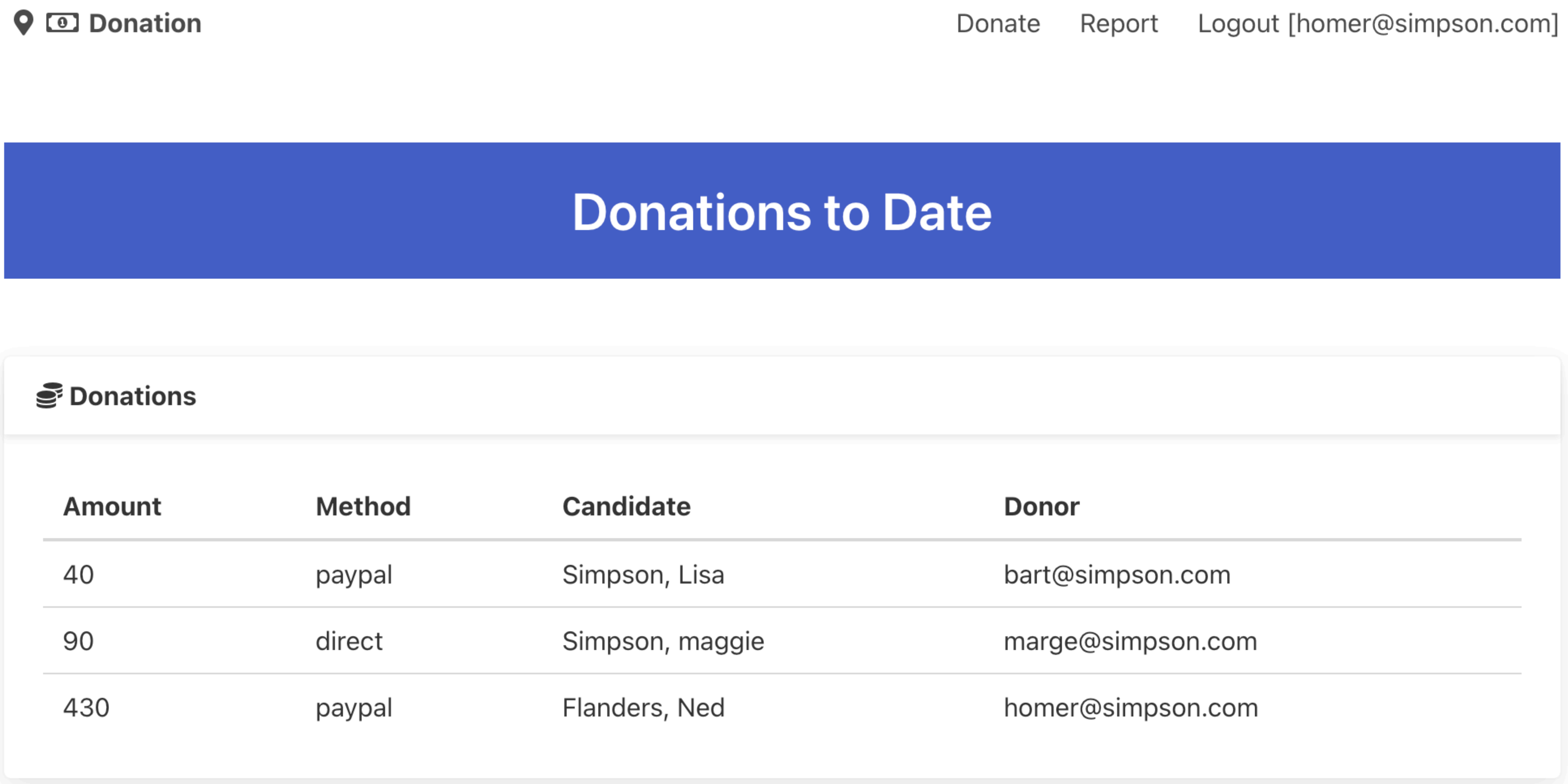
Routes/Views


```
lib
  stores.ts
  ui
    Card.svelte
    Coordinates.svelte
    DonationList.svelte
    Heading.svelte
    Menu.svelte
    Message.svelte
    SplashScreen.svelte
    UserCredentials.svelte
    UserDetails.svelte
```

Stores/Shared state

Reusable UI Components

Report Route



Donations to Date

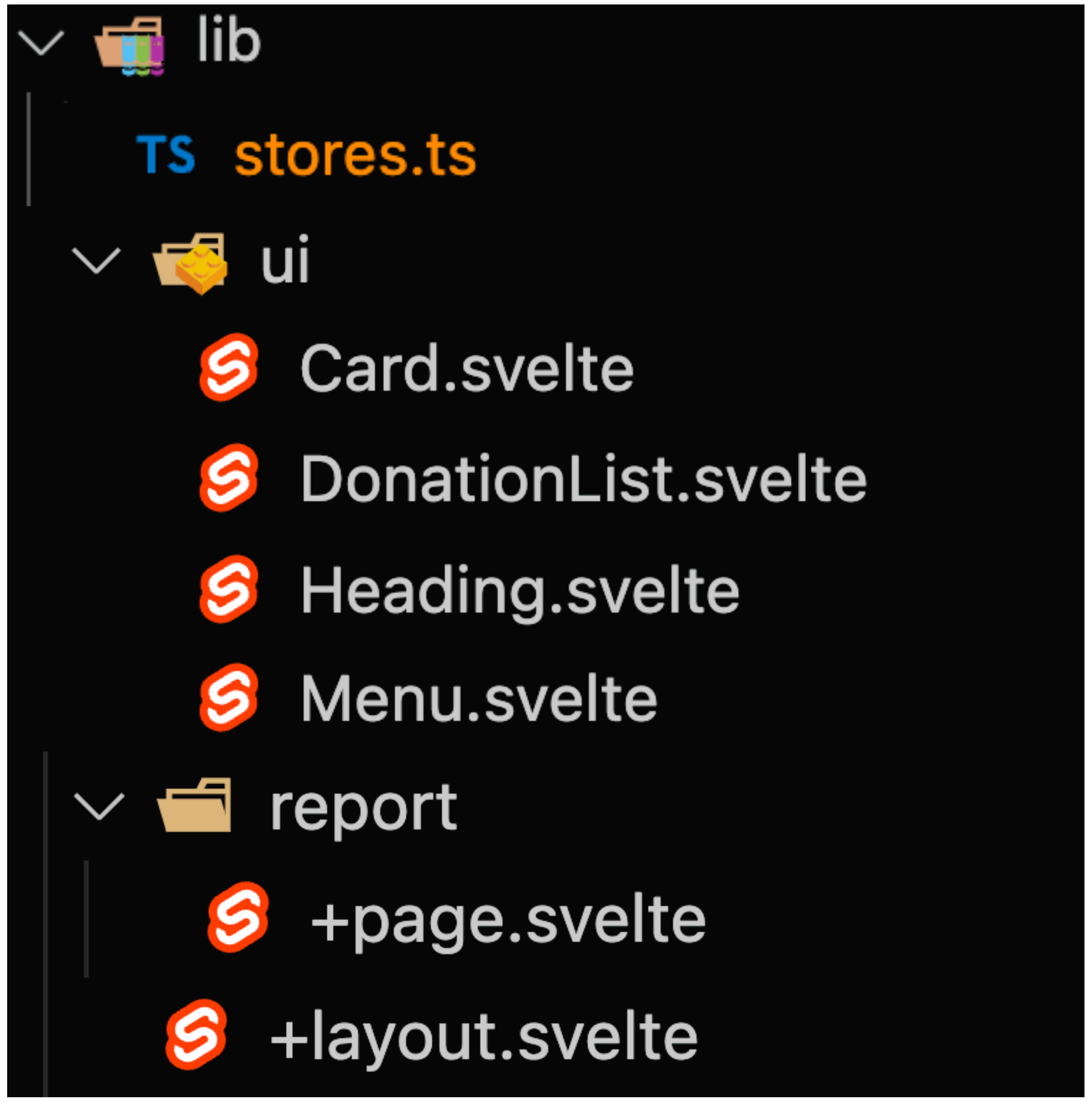
Donations

Amount	Method	Candidate	Donor
40	paypal	Simpson, Lisa	bart@simpson.com
90	direct	Simpson, maggie	marge@simpson.com
430	paypal	Flanders, Ned	homer@simpson.com

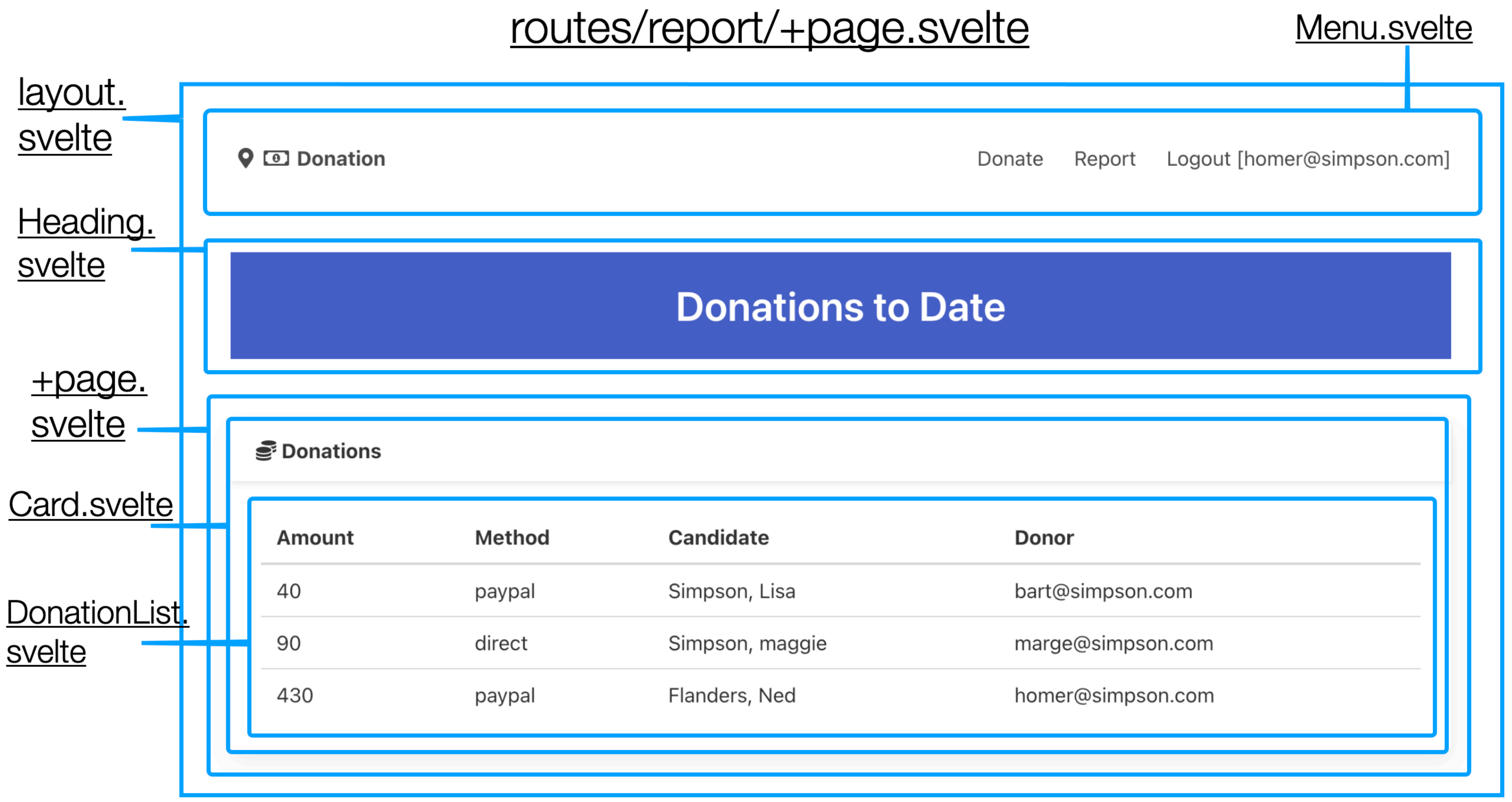
routes/report/+page.svelte

Menu.svelte
layout.svelte
Heading.svelte
+page.svelte
Card.svelte
DonationList.svelte
Donation
Donate
Report
Logout [homer@simpson.com]
Donations to Date
Donations

Amount	Method	Candidate	Donor
40	paypal	Simpson, Lisa	bart@simpson.com
90	direct	Simpson, maggie	marge@simpson.com
430	paypal	Flanders, Ned	homer@simpson.com



```
lib
  TS stores.ts
  ui
    Card.svelte
    DonationList.svelte
    Heading.svelte
    Menu.svelte
report
  +page.svelte
+layout.svelte
```



Amount	Method	Candidate	Donor
40	paypal	Simpson, Lisa	bart@simpson.com
90	direct	Simpson, maggie	marge@simpson.com
430	paypal	Flanders, Ned	homer@simpson.com

routes/report/+page.svelte

Menu.svelte
layout.svelte
Heading.svelte
+page.svelte
Card.svelte
DonationList.svelte

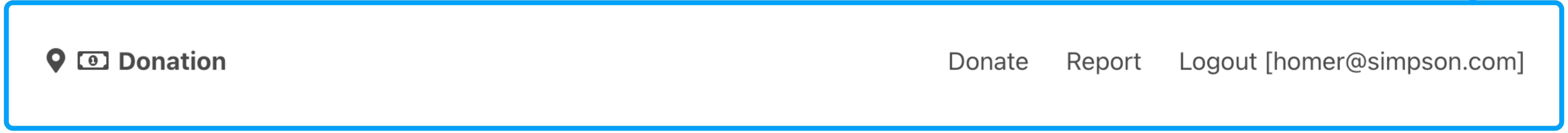
Donation   Donate   Report   Logout [homer@simpson.com]

Donations to Date

Donations

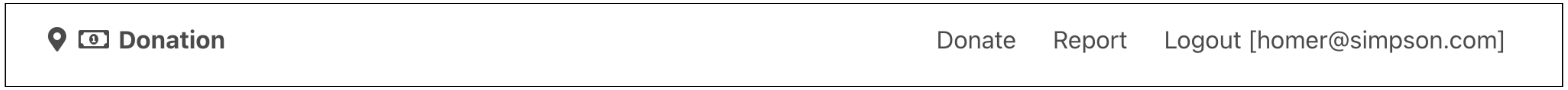
Amount	Method	Candidate	Donor
40	paypal	Simpson, Lisa	bart@simpson.com
90	direct	Simpson, maggie	marge@simpson.com
430	paypal	Flanders, Ned	homer@simpson.com

Menu.svelte



Donation

Donate Report Logout [homer@simpson.com]



Menu.svelte

Import “currentSession” store

```
<script lang="ts">
  import { currentSession } from "$lib/stores";
</script>

<nav class="navbar is-full-width">
  <div class="container">
    <div class="navbar-brand">
      <a class="navbar-item" href="/dashboard">
        <span class="icon"> <i class="fas fa-map-marker-alt"></i></span><span class="icon mr-1">
          <i class="far fa-money-bill-alt"></i></span><span><strong>Donation</strong>
          </span>
      </a>
    </div>
    <div id="navbarMenu" class="navbar-menu">
      <div class="navbar-end">
        <a class="navbar-item" href="/donate"> Donate </a>
        <a class="navbar-item" href="/report"> Report </a>
        <a class="navbar-item" href="/logout"> Logout [{$currentSession}]</a>
      </div>
      <div></div>
    </div>
  </div>
</nav>
```

Place current value of store

stores.ts

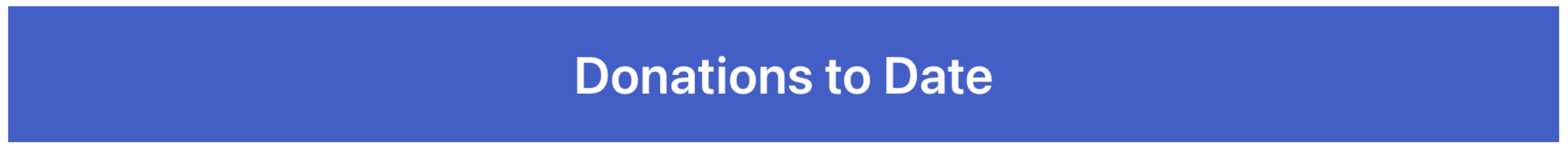
```
export const currentSession = writable<string>();
```

routes/report/+page.svelte

Menu.svelte
layout.svelte
Heading.svelte
+page.svelte
Card.svelte
DonationList.svelte
Donation
Donate
Report
Logout [homer@simpson.com]
Donations to Date
Donations

Amount	Method	Candidate	Donor
40	paypal	Simpson, Lisa	bart@simpson.com
90	direct	Simpson, maggie	marge@simpson.com
430	paypal	Flanders, Ned	homer@simpson.com

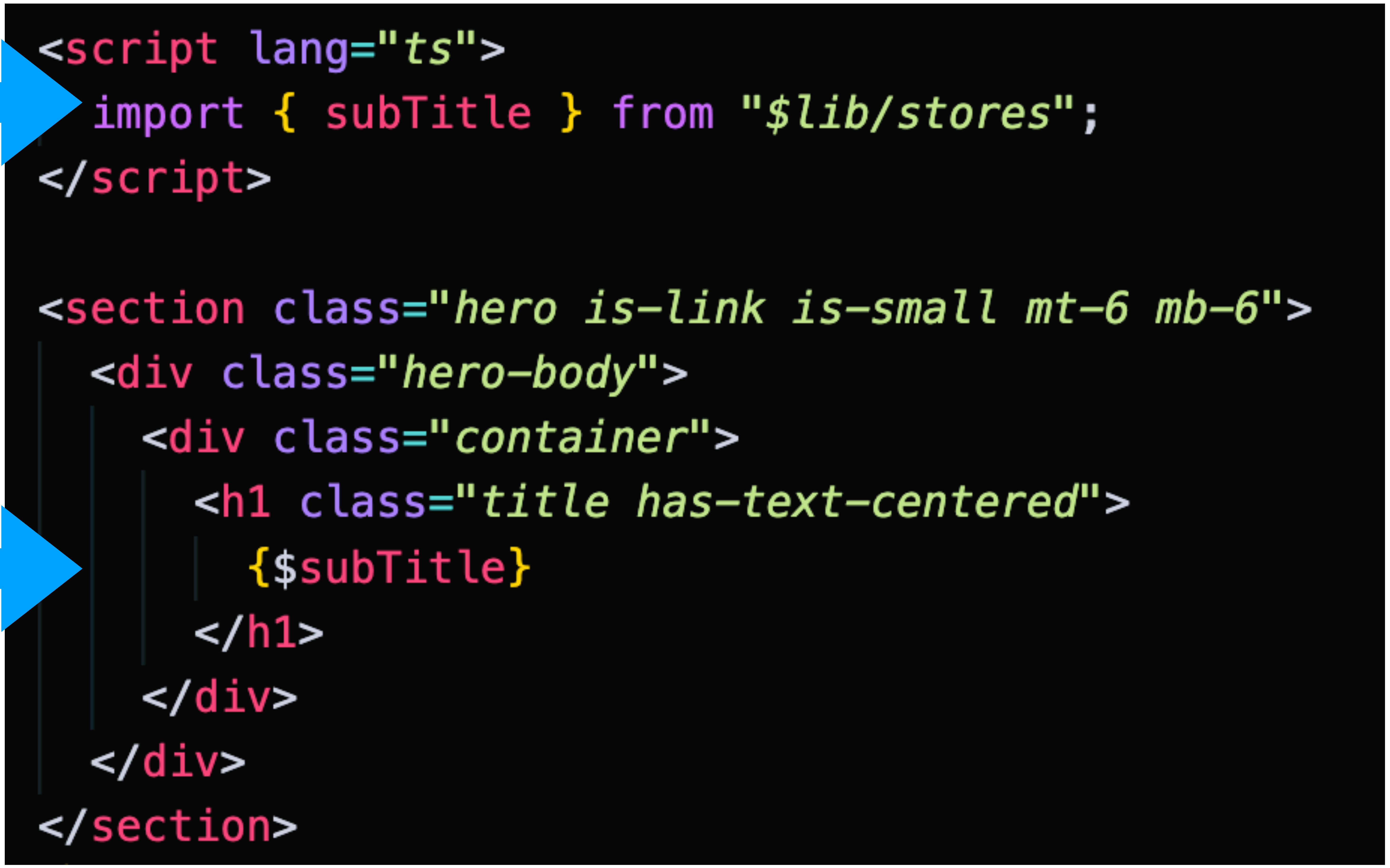
Heading.
svelte



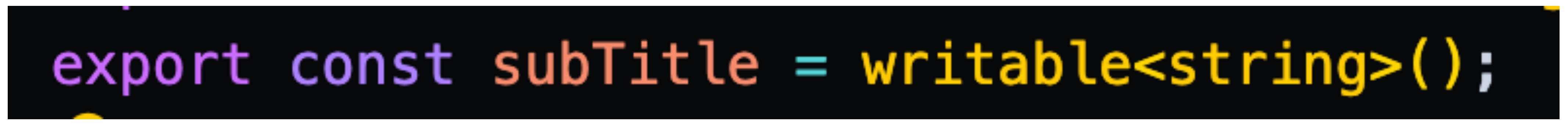
Heading.svelte

Import “subTitle” store

Place current value of store


```
<script lang="ts">
  import { subTitle } from "$lib/stores";
</script>

<section class="hero is-link is-small mt-6 mb-6">
  <div class="hero-body">
    <div class="container">
      <h1 class="title has-text-centered">
        {$subTitle}
      </h1>
    </div>
  </div>
</section>
```

stores.ts

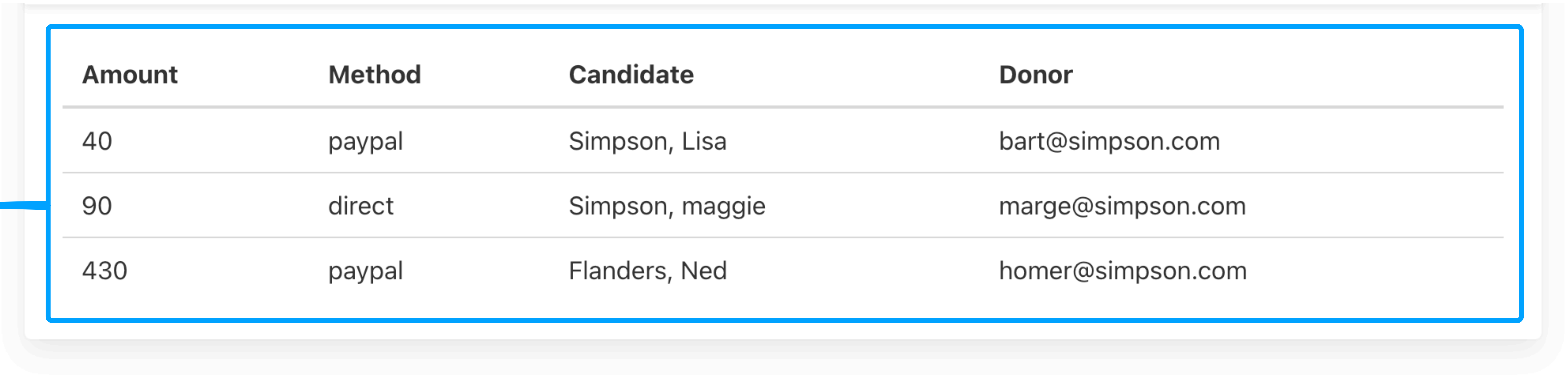

```
export const subTitle = writable<string>();
```

routes/report/+page.svelte

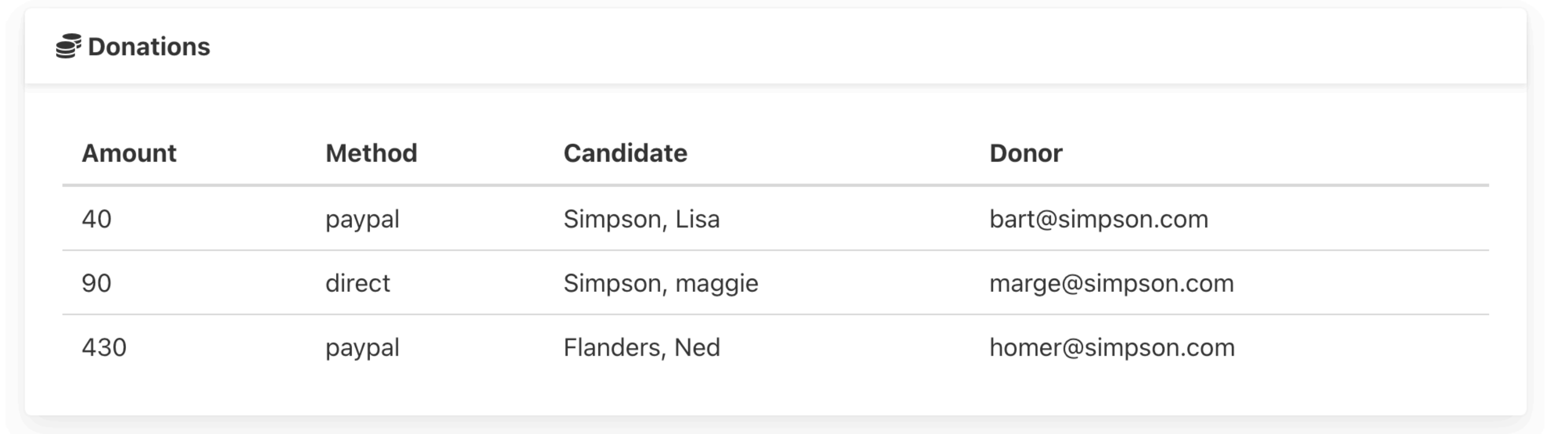
Menu.svelte
layout.svelte
Heading.svelte
+page.svelte
Card.svelte
DonationList.svelte
Donation
Donate
Report
Logout [homer@simpson.com]
Donations to Date
Donations

Amount	Method	Candidate	Donor
40	paypal	Simpson, Lisa	bart@simpson.com
90	direct	Simpson, maggie	marge@simpson.com
430	paypal	Flanders, Ned	homer@simpson.com

DonationList.
svelte



Amount	Method	Candidate	Donor
40	paypal	Simpson, Lisa	bart@simpson.com
90	direct	Simpson, maggie	marge@simpson.com
430	paypal	Flanders, Ned	homer@simpson.com


Donations

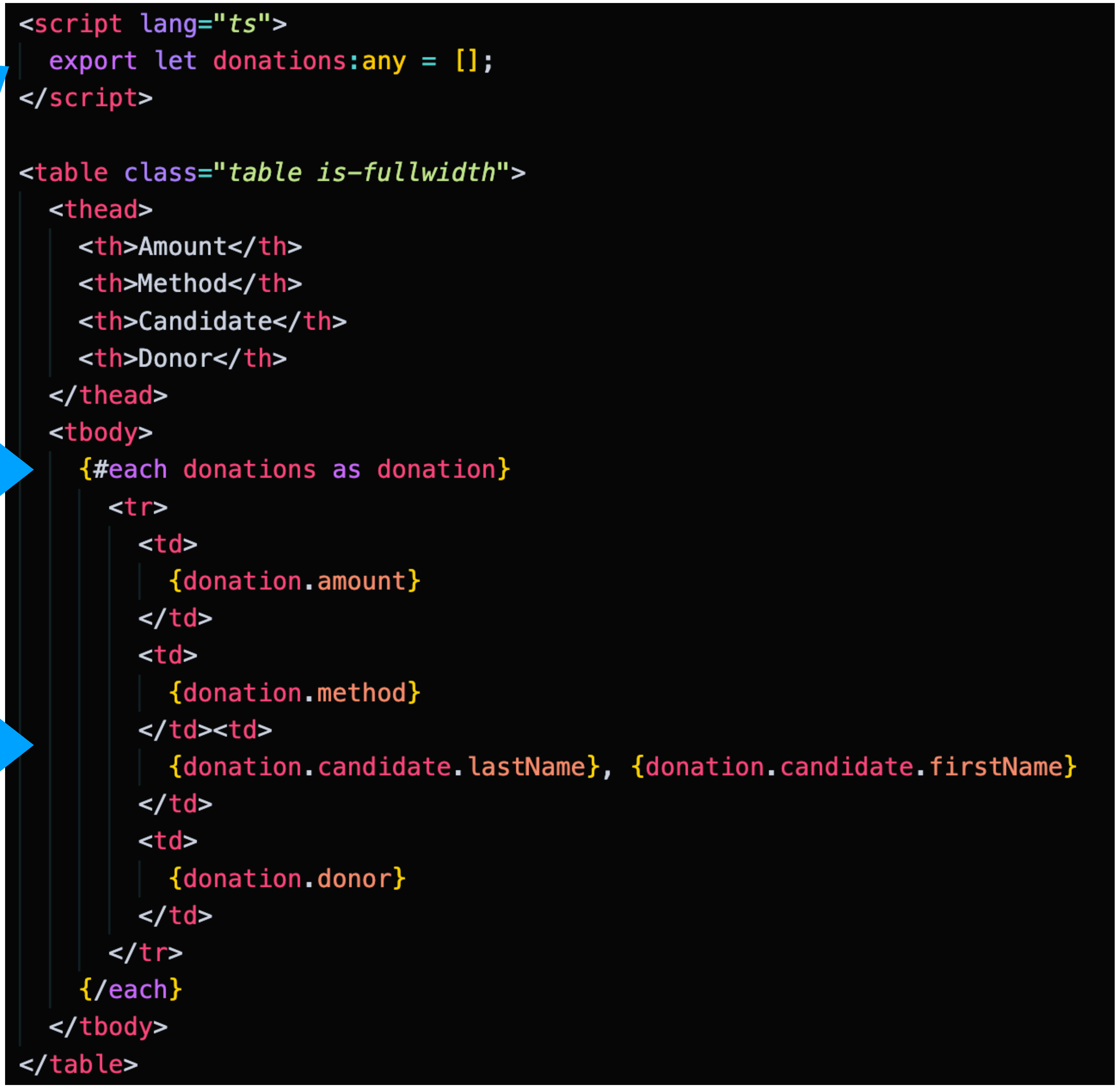
Amount	Method	Candidate	Donor
40	paypal	Simpson, Lisa	bart@simpson.com
90	direct	Simpson, maggie	marge@simpson.com
430	paypal	Flanders, Ned	homer@simpson.com

DonationList.svelte

Expect an array of donations to be passed to the component

Iterate through each donation in the array

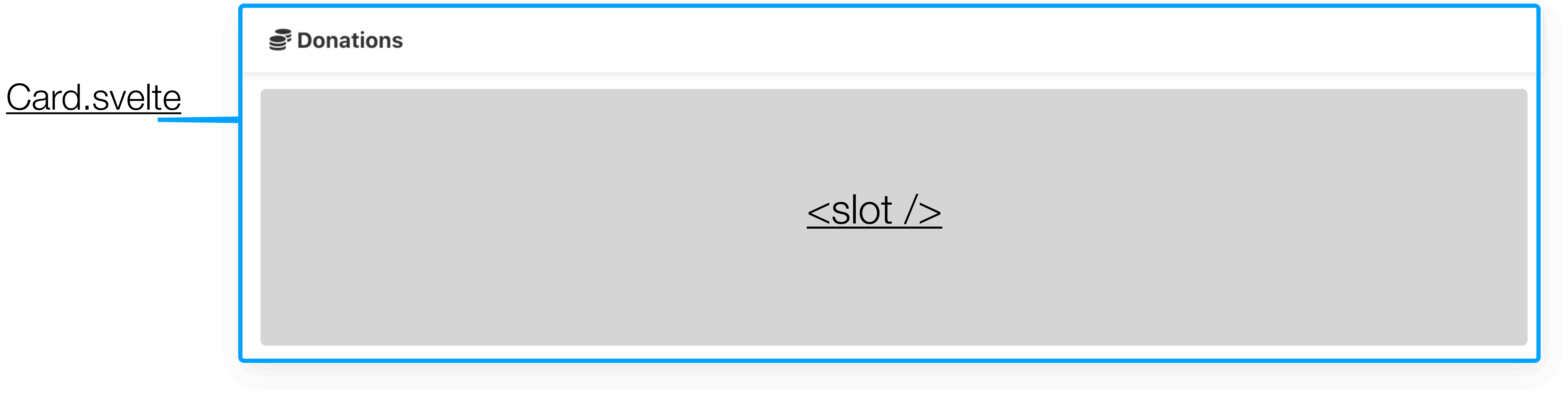
Display each row of the table

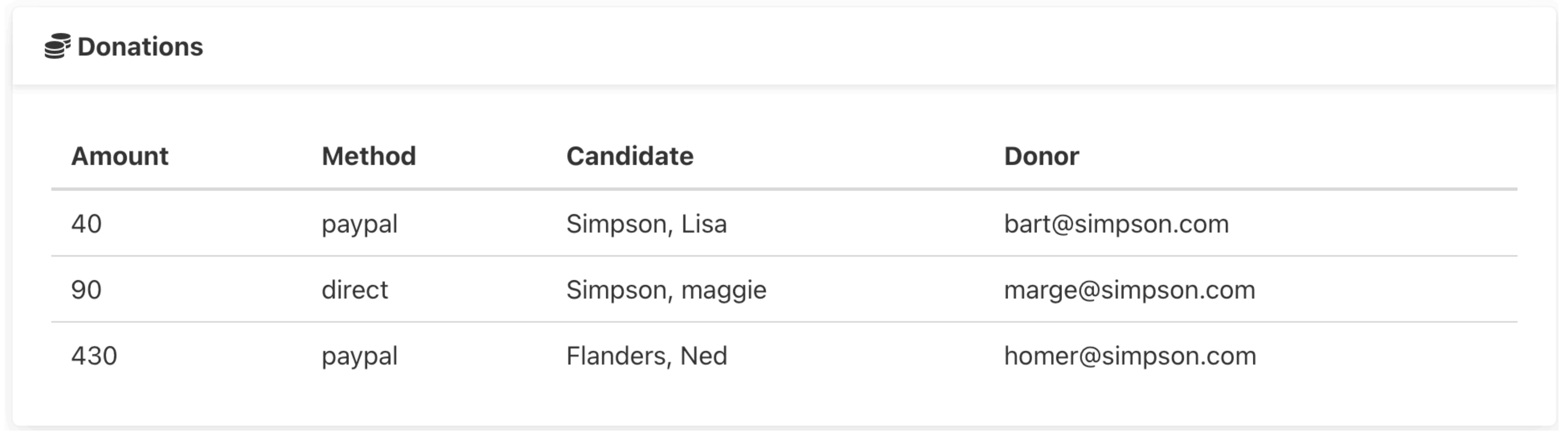

```
<script lang="ts">
  export let donations:any = [];
</script>

<table class="table is-fullwidth">
  <thead>
    <th>Amount</th>
    <th>Method</th>
    <th>Candidate</th>
    <th>Donor</th>
  </thead>
  <tbody>
    {#each donations as donation}
      <tr>
        <td>
          {donation.amount}
        </td>
        <td>
          {donation.method}
        </td><td>
          {donation.candidate.lastName}, {donation.candidate.firstName}
        </td>
        <td>
          {donation.donor}
        </td>
      </tr>
    {/each}
  </tbody>
</table>
```

routes/report/+page.svelte

Menu.svelte
layout.svelte
Heading.svelte
+page.svelte
Card.svelte
DonationList.svelte
Donation
Donate
Report
Logout [homer@simpson.com]
Donations to Date
Donations

Amount	Method	Candidate	Donor
40	paypal	Simpson, Lisa	bart@simpson.com
90	direct	Simpson, maggie	marge@simpson.com
430	paypal	Flanders, Ned	homer@simpson.com


Card.svelte
Donations
<slot />

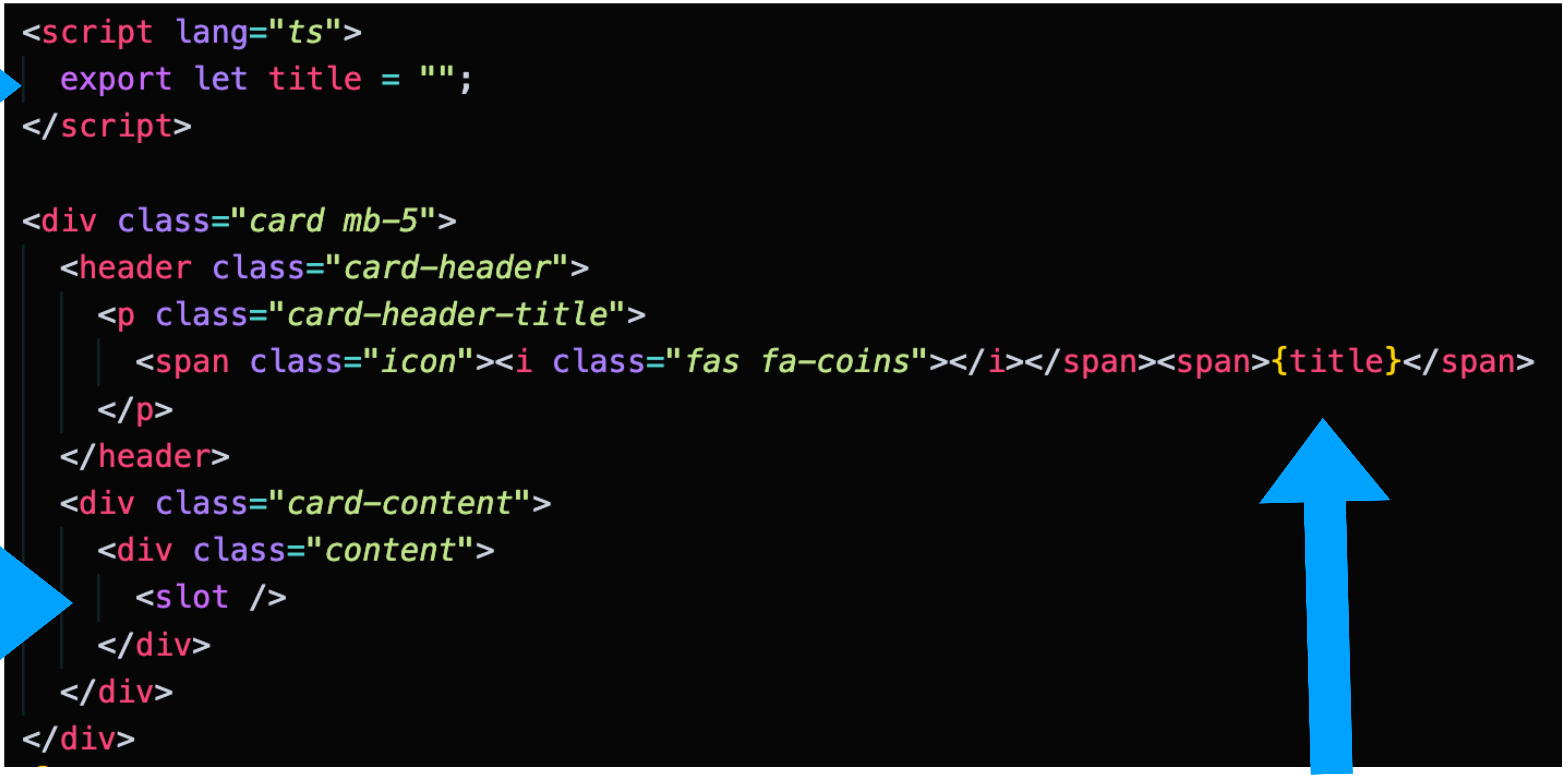


Donations

Amount	Method	Candidate	Donor
40	paypal	Simpson, Lisa	bart@simpson.com
90	direct	Simpson, maggie	marge@simpson.com
430	paypal	Flanders, Ned	homer@simpson.com

Card.svelte

Expect 'title' property to be based to component



```
<script lang="ts">
  export let title = "";
</script>

<div class="card mb-5">
  <header class="card-header">
    <p class="card-header-title">
      <span class="icon"><i class="fas fa-coins"></i></span><span>{title}</span>
    </p>
  </header>
  <div class="card-content">
    <div class="content">
      <slot />
    </div>
  </div>
</div>
```

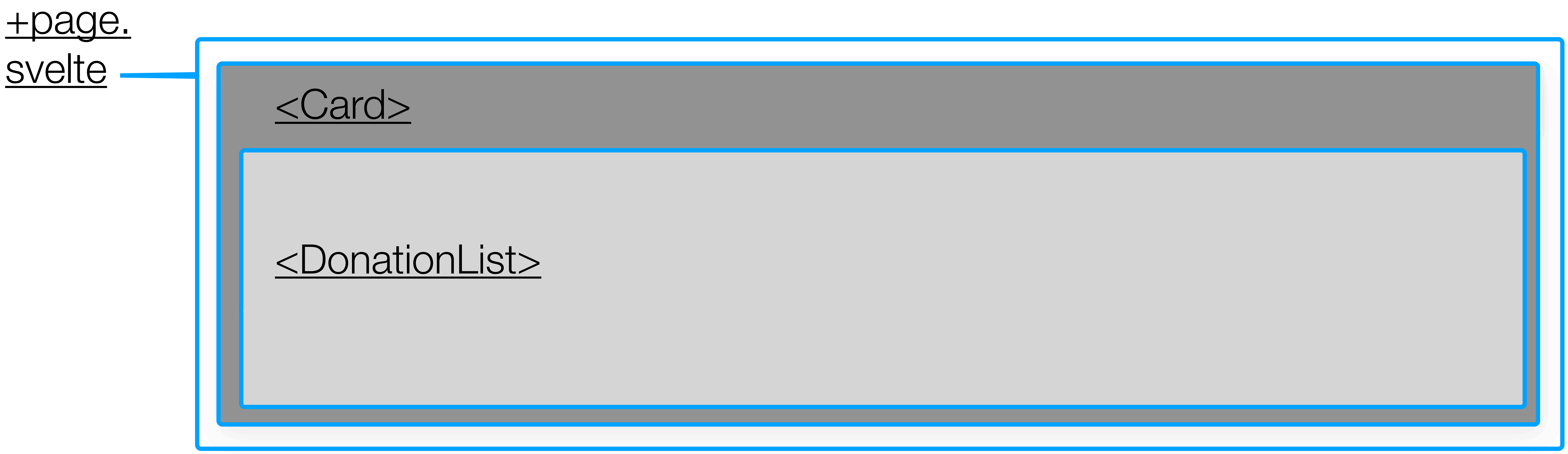
The Slot: Provided by the component that creates this Card component

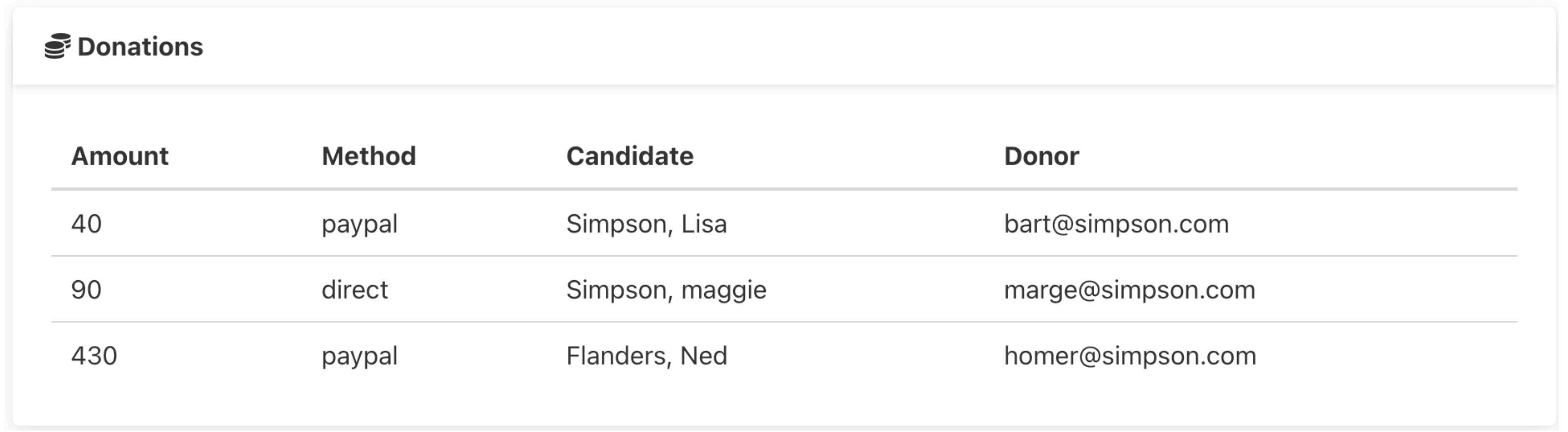
Place 'title' in element

routes/report/+page.svelte

Menu.svelte
layout.svelte
Heading.svelte
+page.svelte
Card.svelte
DonationList.svelte
Donation
Donate
Report
Logout [homer@simpson.com]
Donations to Date
Donations

Amount	Method	Candidate	Donor
40	paypal	Simpson, Lisa	bart@simpson.com
90	direct	Simpson, maggie	marge@simpson.com
430	paypal	Flanders, Ned	homer@simpson.com


+page.
svelte
<Card>
<DonationList>



Donations

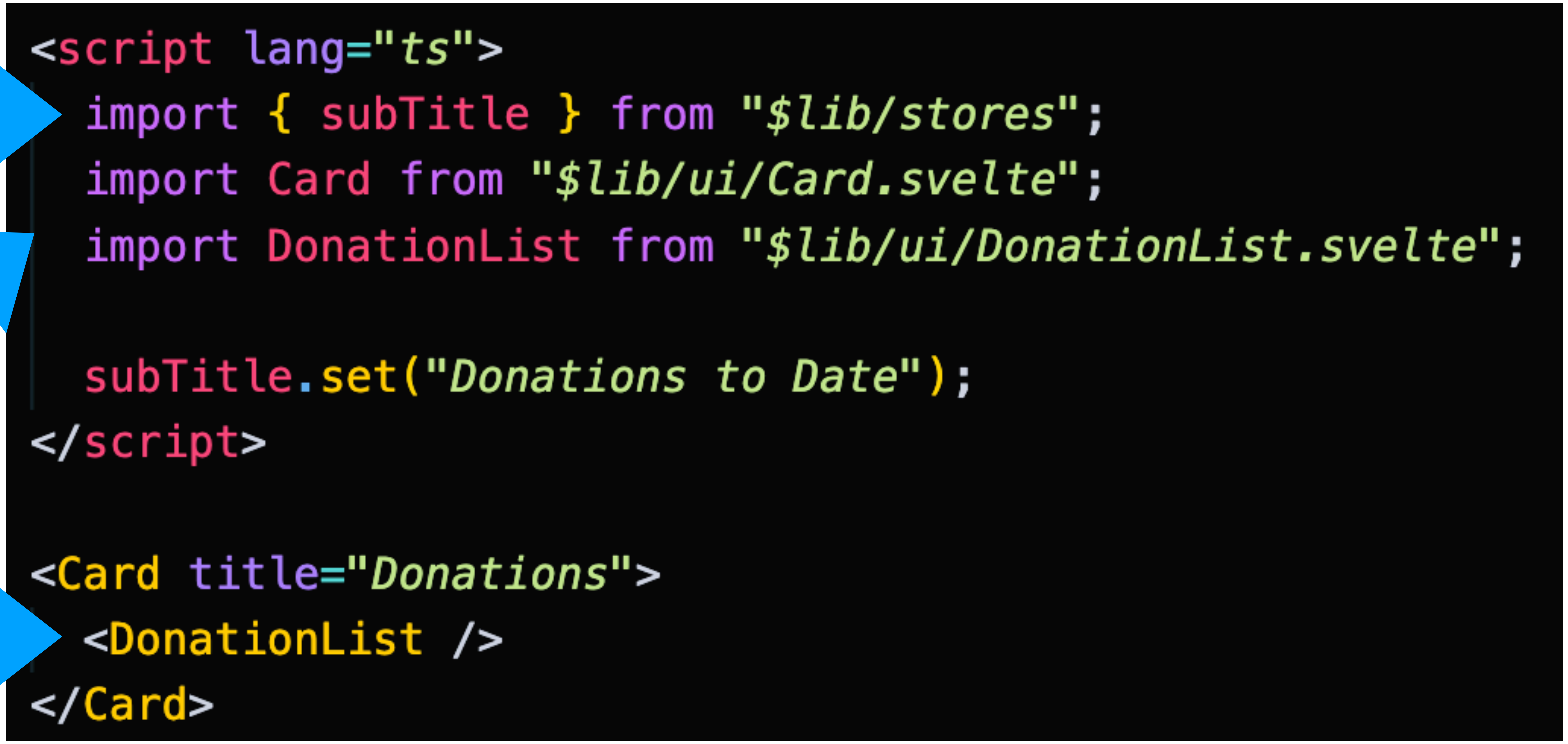
Amount	Method	Candidate	Donor
40	paypal	Simpson, Lisa	bart@simpson.com
90	direct	Simpson, maggie	marge@simpson.com
430	paypal	Flanders, Ned	homer@simpson.com

+page.svelte

Import subTitle store - so we can set its value

Import Card and DonationList components

The Slot: provide the component to be 'slotted' into the card body



```
<script lang="ts">
  import { subTitle } from "$lib/stores";
  import Card from "$lib/ui/Card.svelte";
  import DonationList from "$lib/ui/DonationList.svelte";

  subTitle.set("Donations to Date");
</script>

<Card title="Donations">
  <DonationList />
</Card>
```

routes/report/+page.svelte

Menu.svelte
layout.svelte
Heading.svelte
+page.svelte
Card.svelte
DonationList.svelte
Donation
Donate
Report
Logout [homer@simpson.com]
Donations to Date
Donations

Amount	Method	Candidate	Donor
40	paypal	Simpson, Lisa	bart@simpson.com
90	direct	Simpson, maggie	marge@simpson.com
430	paypal	Flanders, Ned	homer@simpson.com

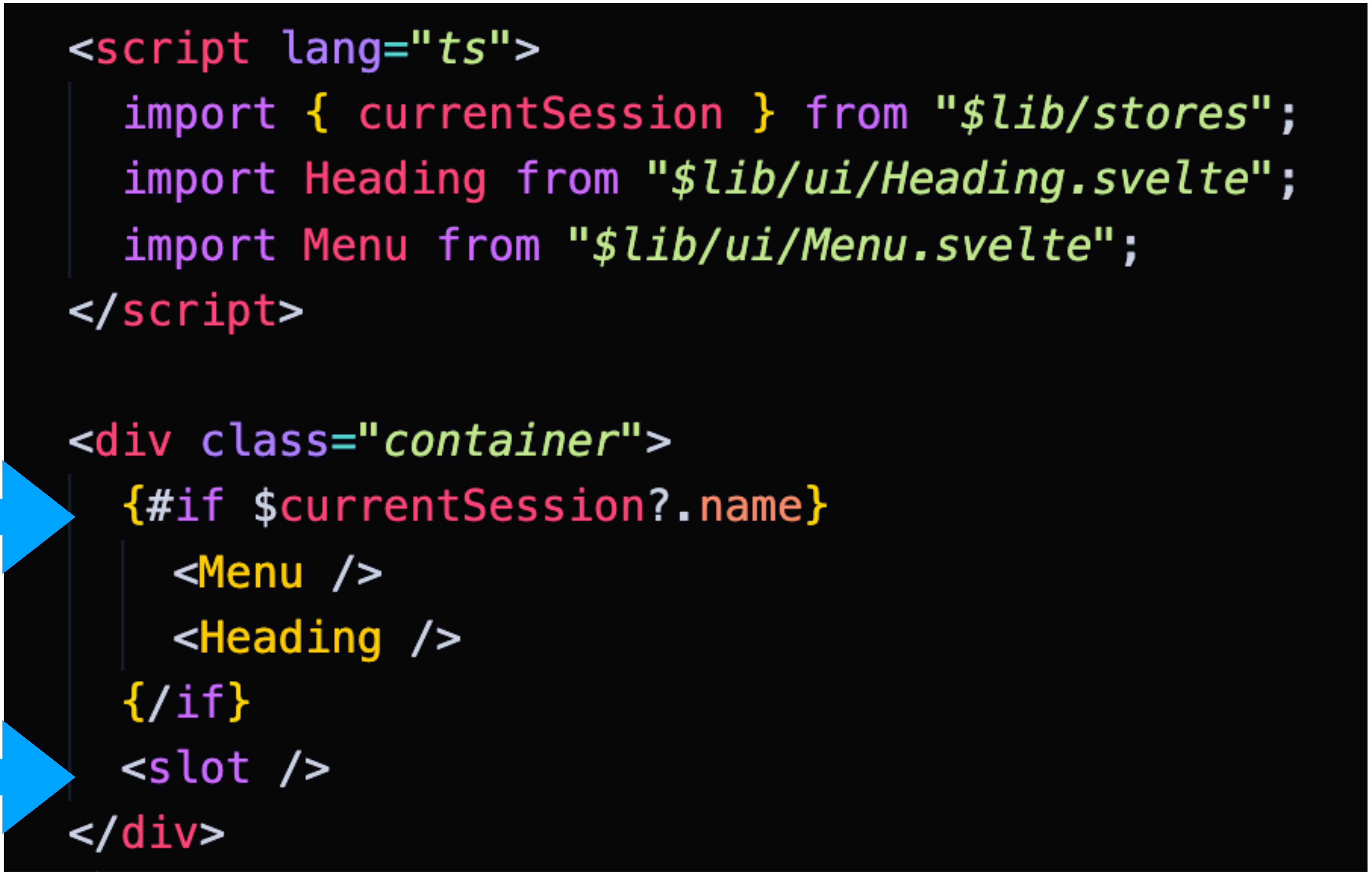
layout.
svelte

+layout.svelte

- Plays a similar role to layout.hbs in Handlebars

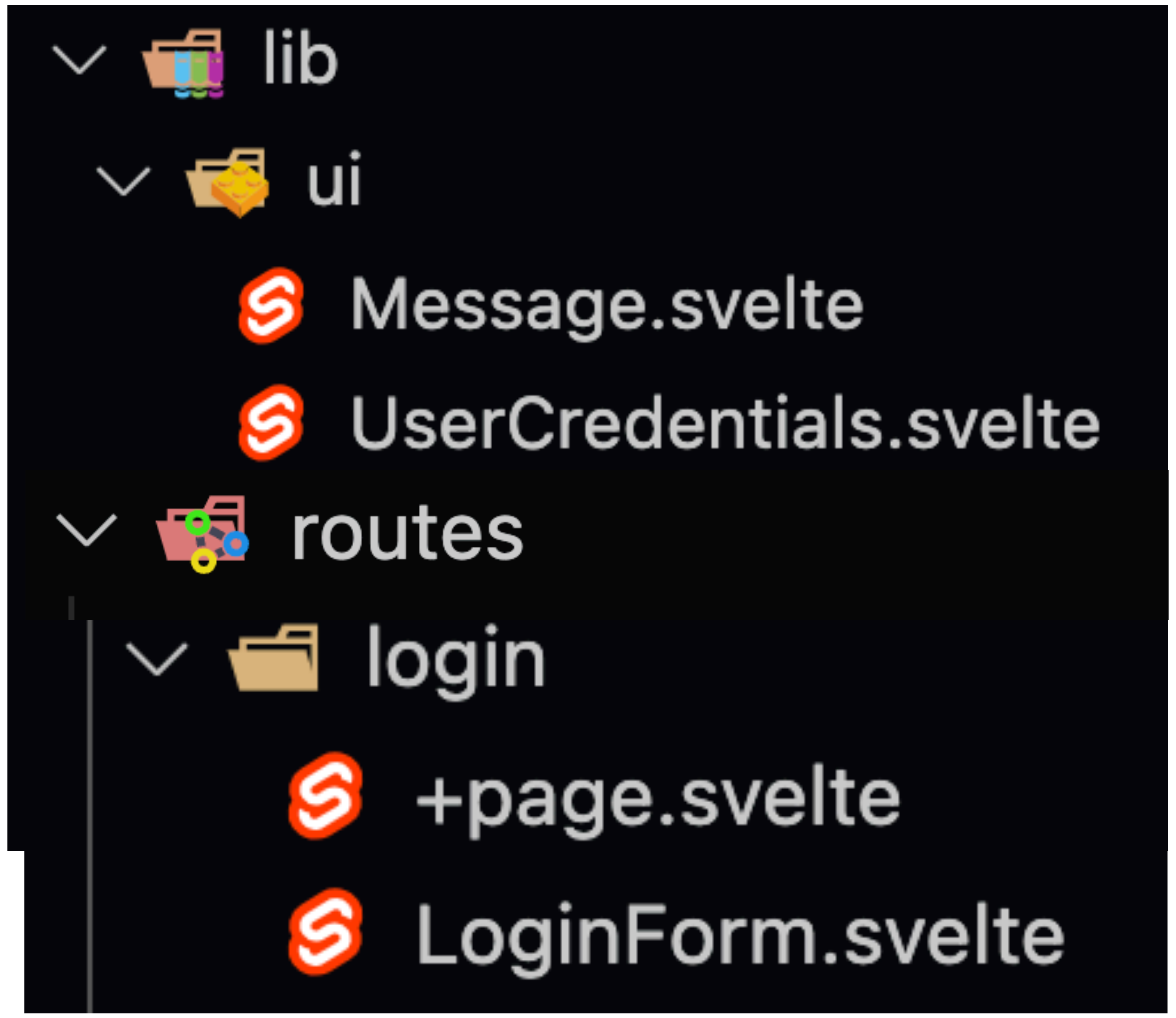
- Display Menu + Heading if we have a session defined

- All page content based on this layout

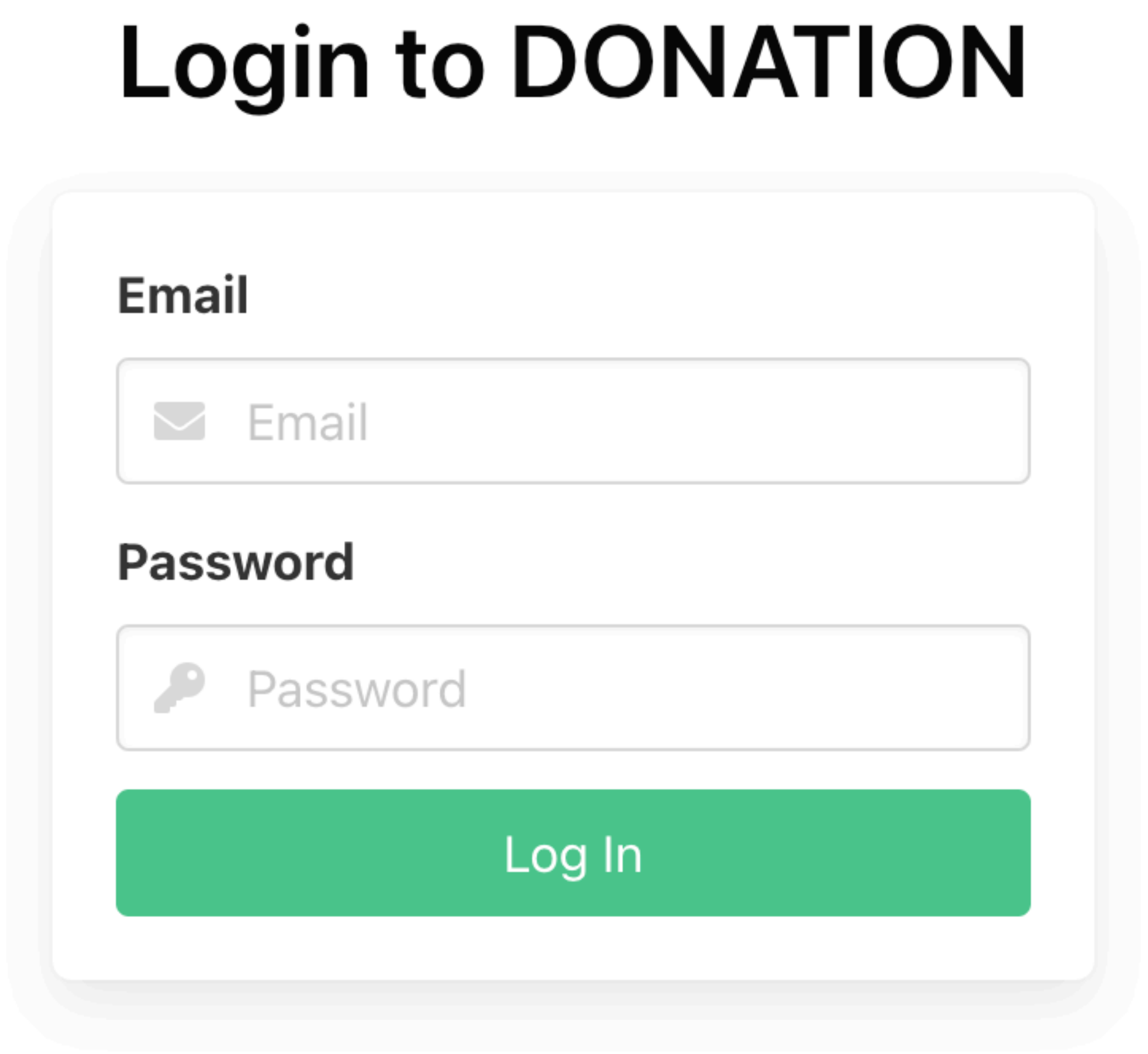


```
<script lang="ts">
  import { currentSession } from "$lib/stores";
  import Heading from "$lib/ui/Heading.svelte";
  import Menu from "$lib/ui/Menu.svelte";
</script>

<div class="container">
  {#if $currentSession?.name}
    <Menu />
    <Heading />
  {/if}
  <slot />
</div>
```

Login Route



```
lib
  ui
    Message.svelte
    UserCredentials.svelte
routes
  login
    +page.svelte
    LoginForm.svelte
```



Login to DONATION

Email

Email

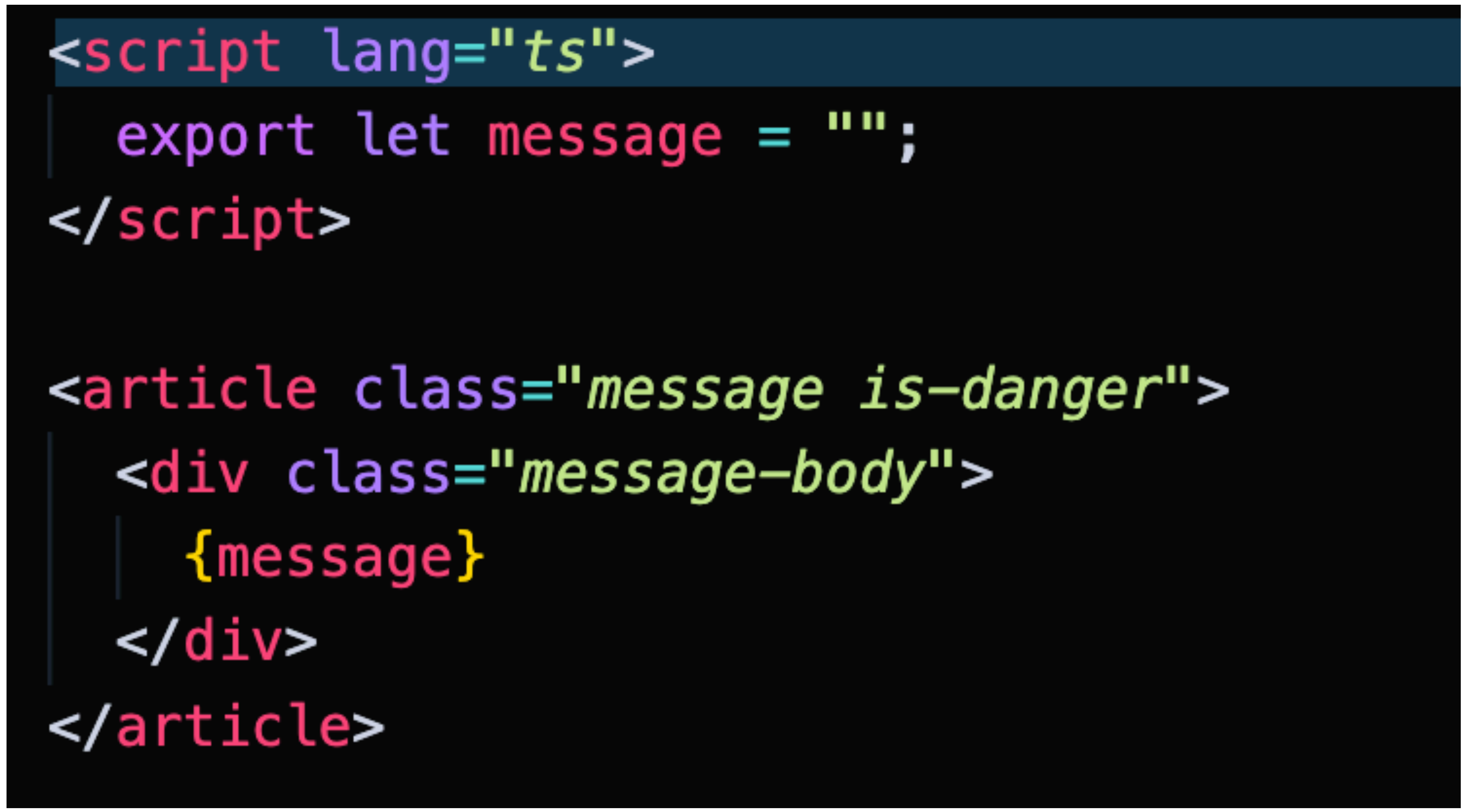
Password

Password

Log In

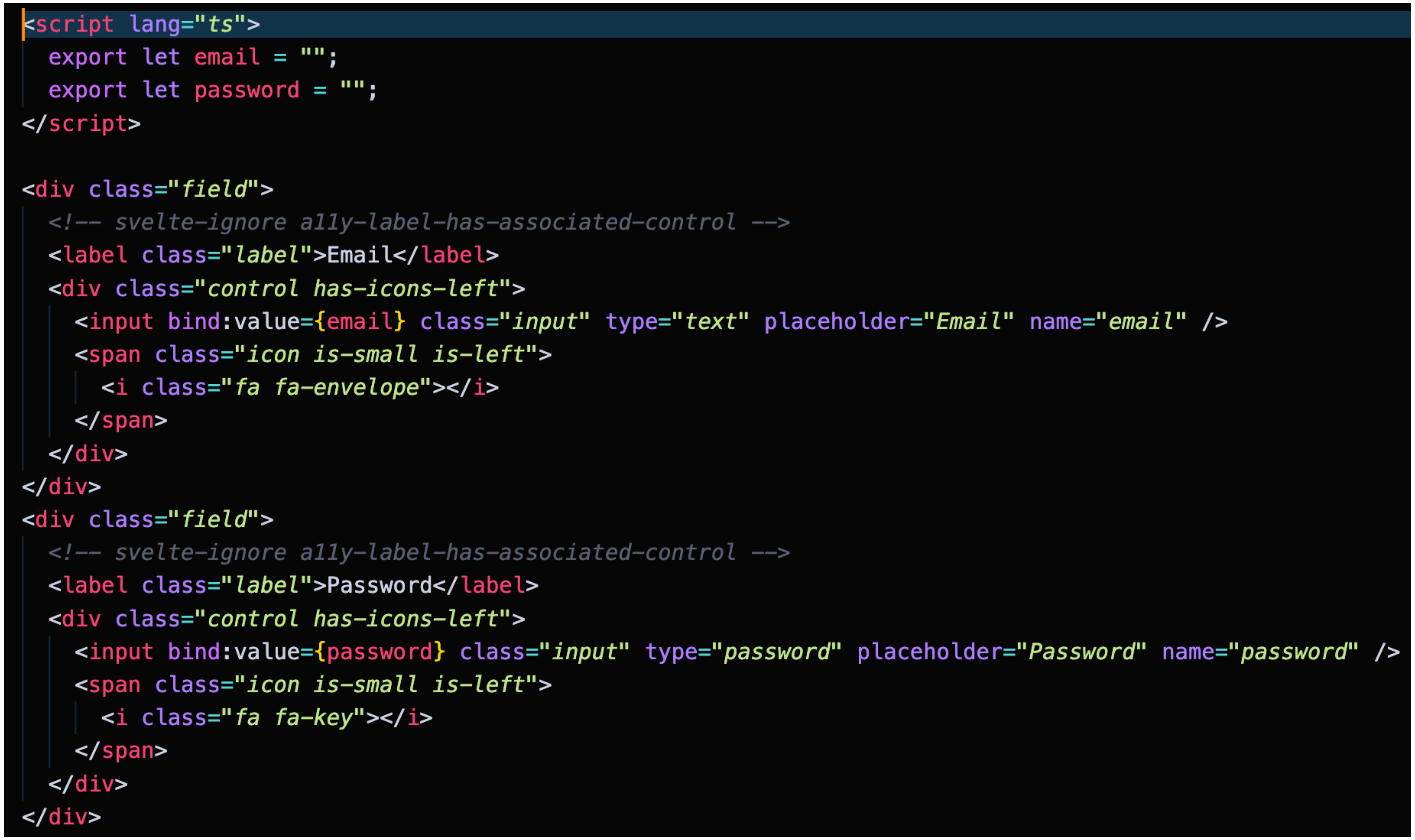
Message.svelte

- Simple message box tone displayed if authentication fails

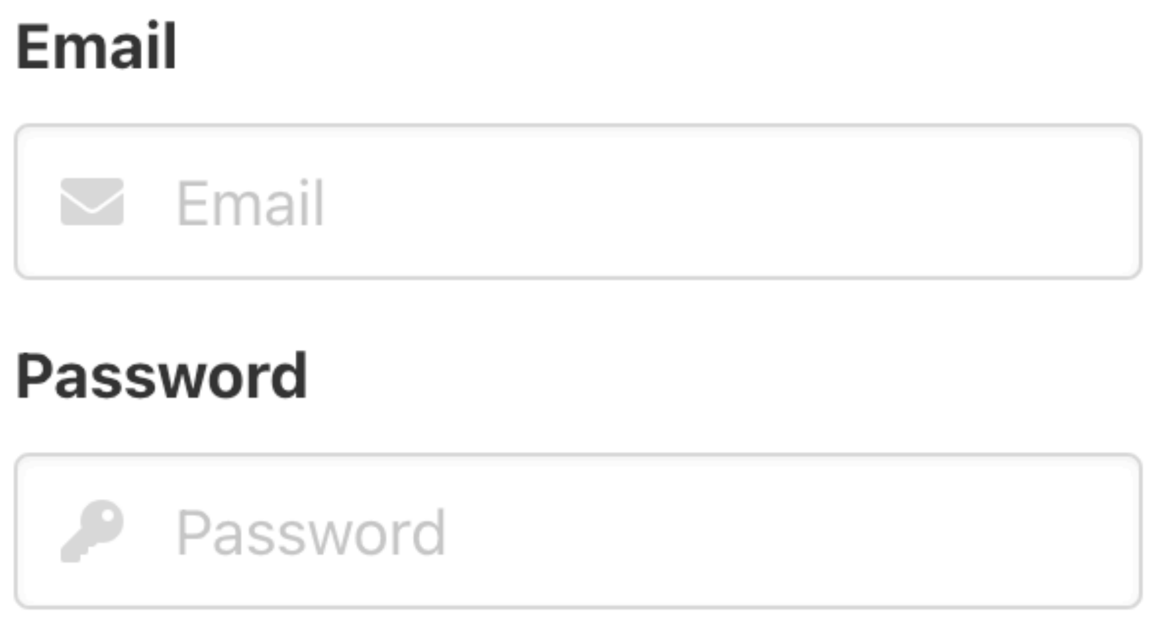


```
<script lang="ts">
  export let message = "";
</script>

<article class="message is-danger">
  <div class="message-body">
    {message}
  </div>
</article>
```

UserCredentials.svelte



```
<script lang="ts">
  export let email = "";
  export let password = "";
</script>

<div class="field">
  <!-- svelte-ignore a11y-label-has-associated-control -->
  <label class="label">Email</label>
  <div class="control has-icons-left">
    <input bind:value={email} class="input" type="text" placeholder="Email" name="email" />
    <span class="icon is-small is-left">
      <i class="fa fa-envelope"></i>
    </span>
  </div>
</div>
<div class="field">
  <!-- svelte-ignore a11y-label-has-associated-control -->
  <label class="label">Password</label>
  <div class="control has-icons-left">
    <input bind:value={password} class="input" type="password" placeholder="Password" name="password" />
    <span class="icon is-small is-left">
      <i class="fa fa-key"></i>
    </span>
  </div>
</div>
```

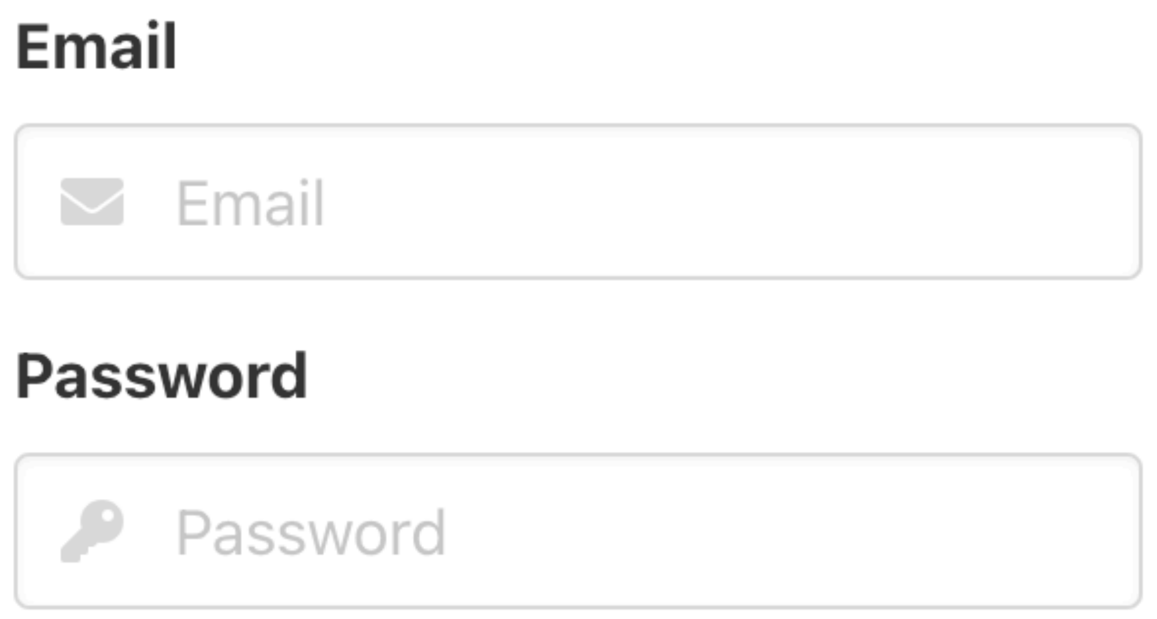


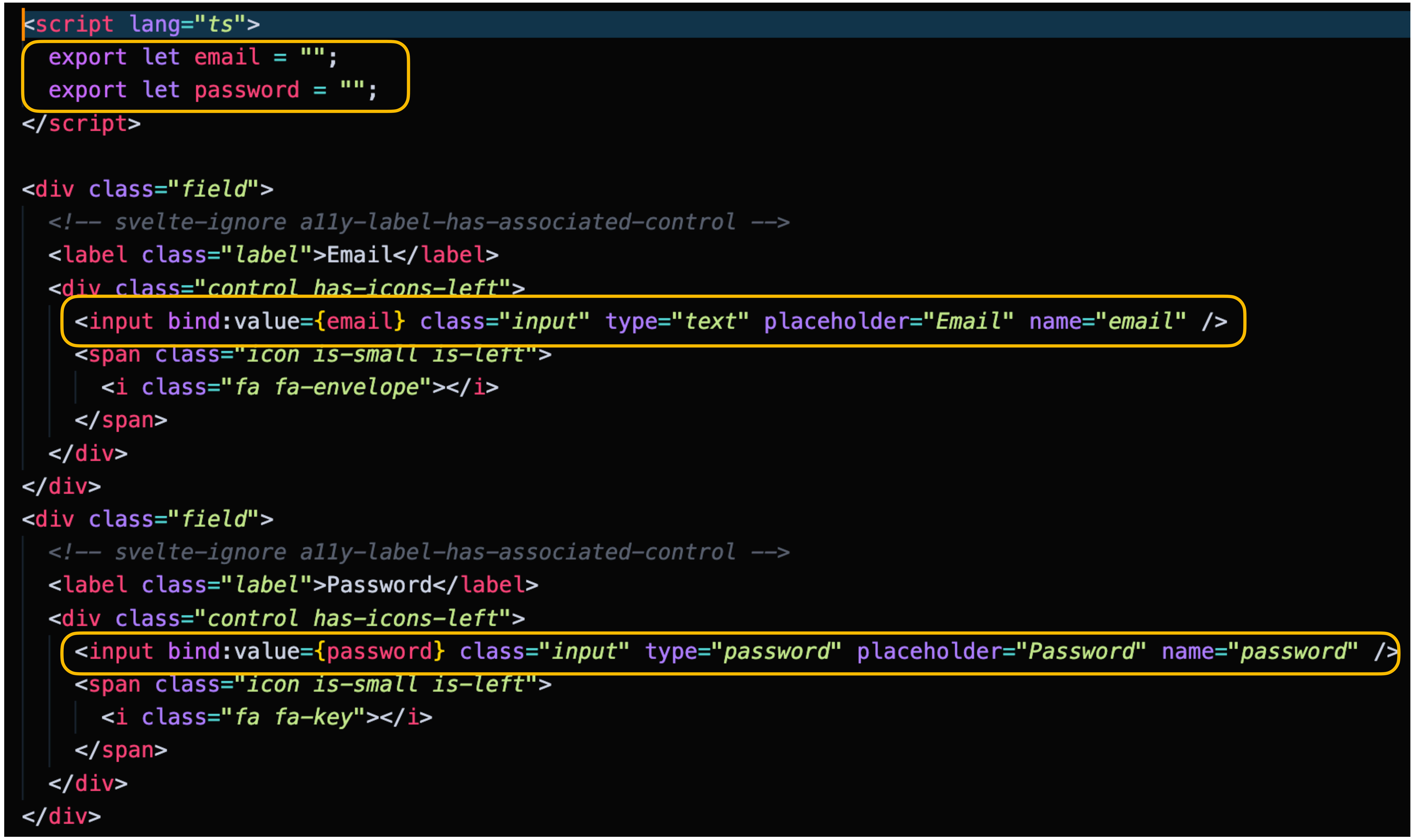
UserCredentials.svelte

- Display user credentials text fields

- Input controls values bound to js variables

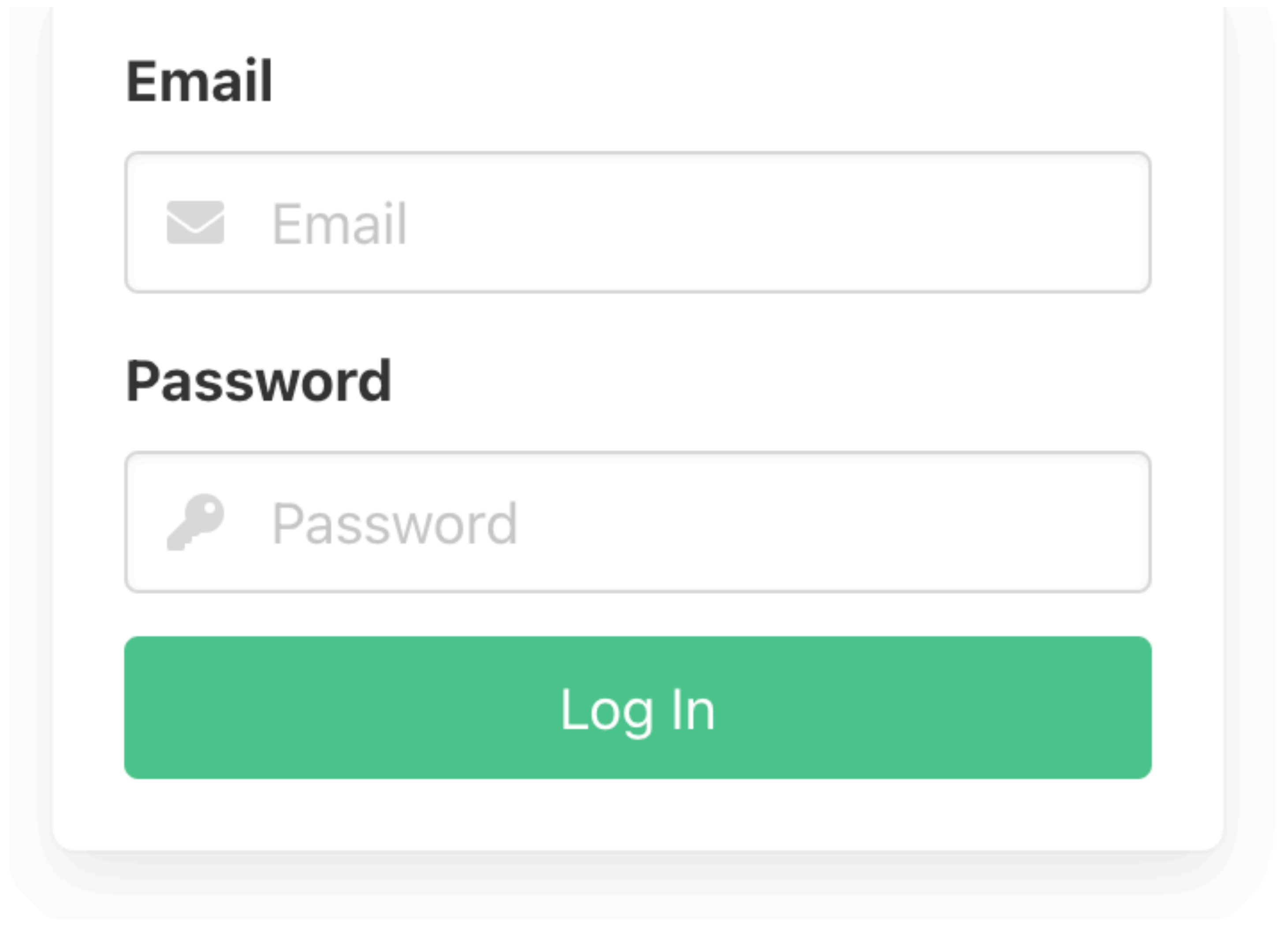
- Variables available as props to parent component

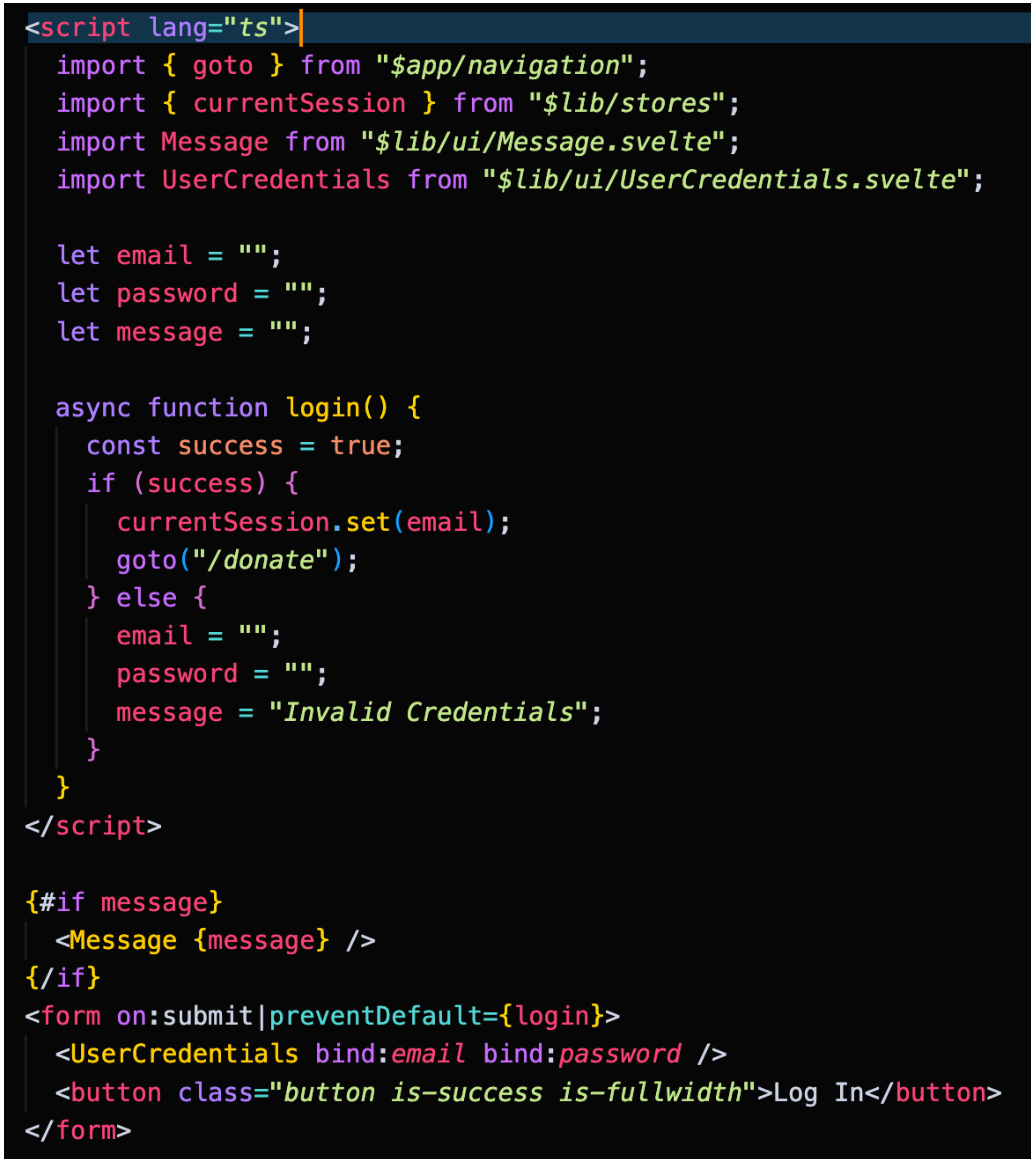




```
<script lang="ts">
  export let email = "";
  export let password = "";
</script>

<div class="field">
  <!-- svelte-ignore a11y-label-has-associated-control -->
  <label class="label">Email</label>
  <div class="control has-icons-left">
    <input bind:value={email} class="input" type="text" placeholder="Email" name="email" />
    <span class="icon is-small is-left">
      <i class="fa fa-envelope"></i>
    </span>
  </div>
</div>
<div class="field">
  <!-- svelte-ignore a11y-label-has-associated-control -->
  <label class="label">Password</label>
  <div class="control has-icons-left">
    <input bind:value={password} class="input" type="password" placeholder="Password" name="password" />
    <span class="icon is-small is-left">
      <i class="fa fa-key"></i>
    </span>
  </div>
</div>
```

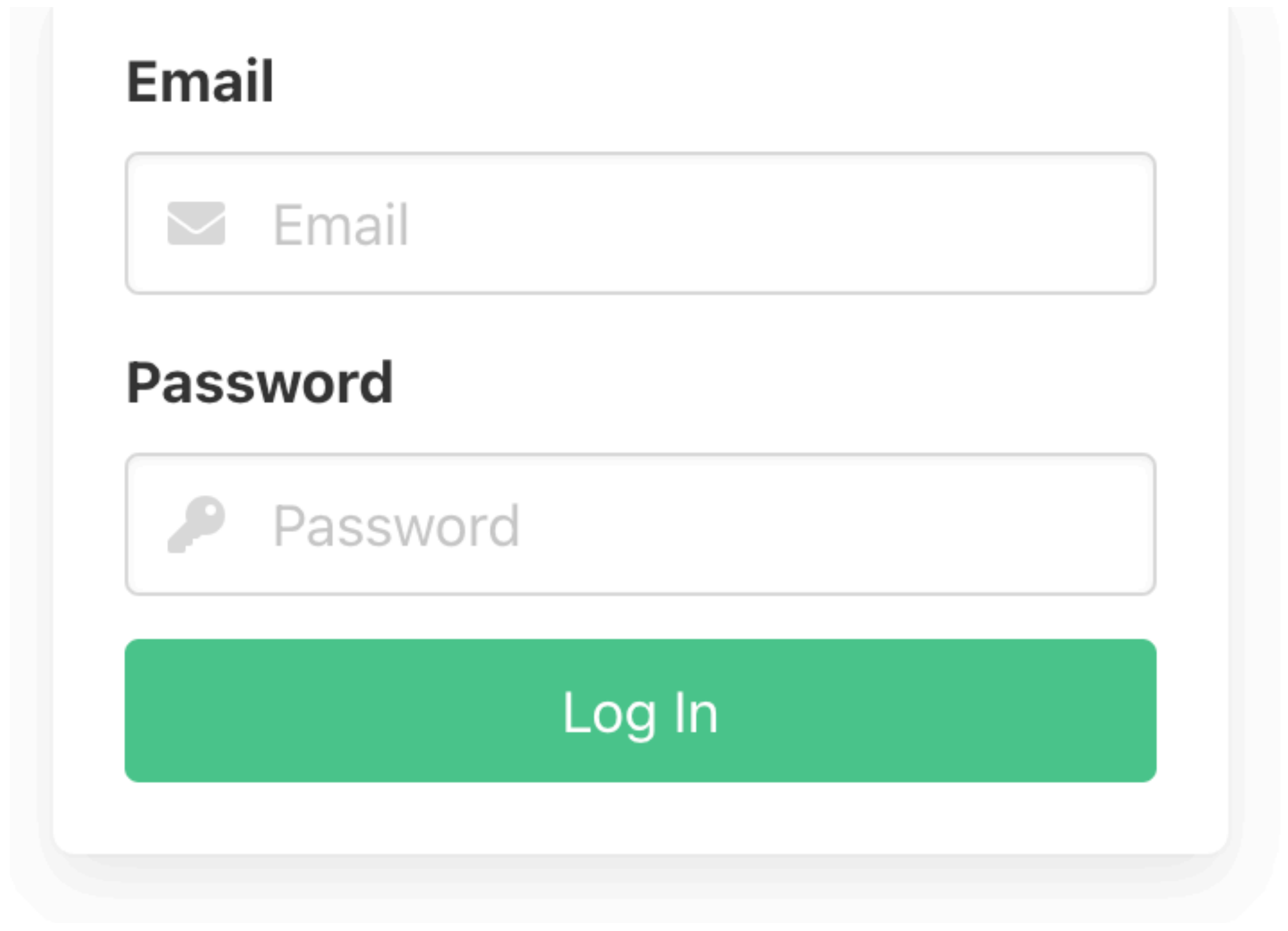
LoginForm.svelte





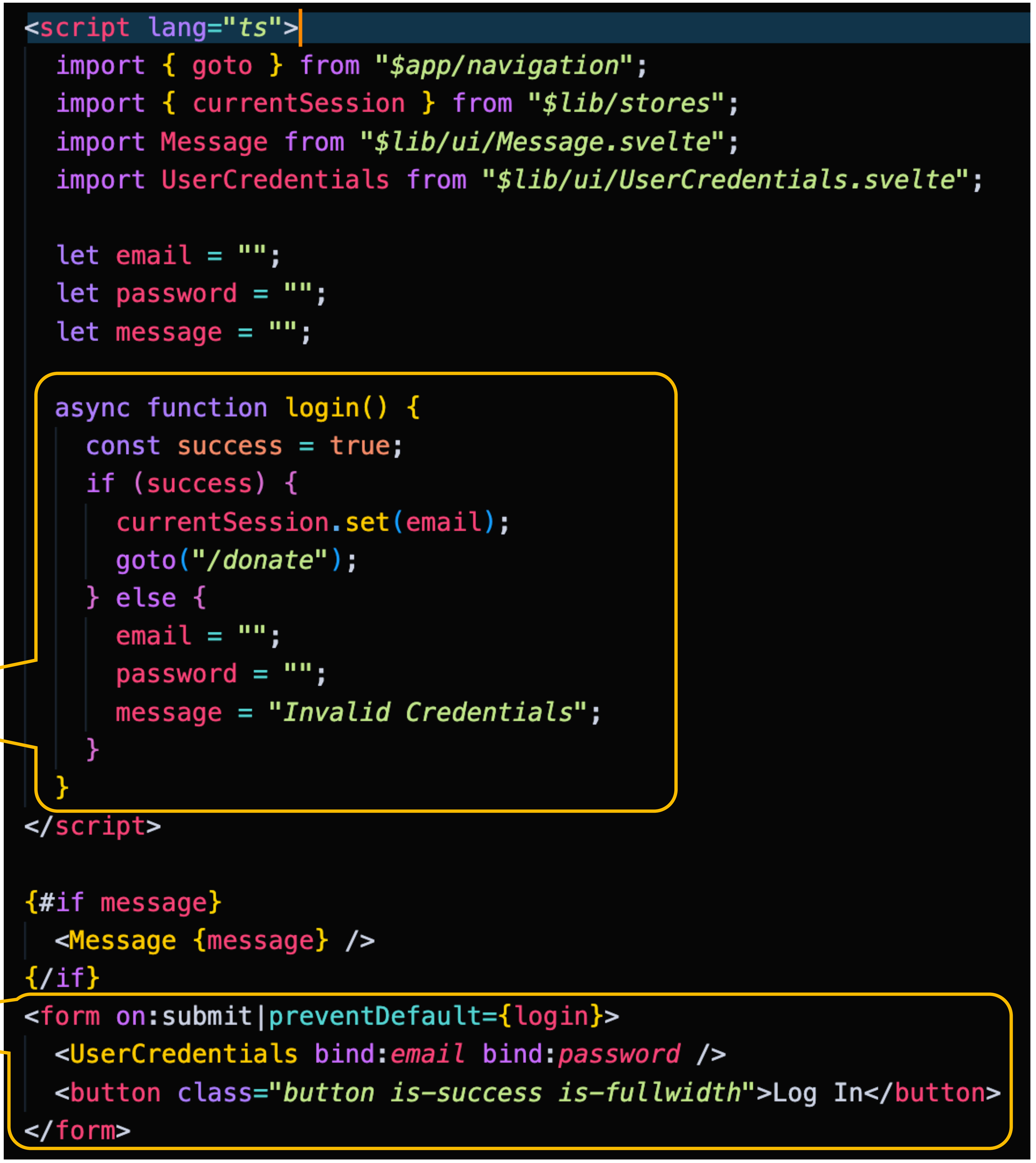
```
<script lang="ts">
  import { goto } from "$app/navigation";
  import { currentSession } from "$lib/stores";
  import Message from "$lib/ui/Message.svelte";
  import UserCredentials from "$lib/ui/UserCredentials.svelte";

  let email = "";
  let password = "";
  let message = "";

  async function login() {
    const success = true;
    if (success) {
      currentSession.set(email);
      goto("/donate");
    } else {
      email = "";
      password = "";
      message = "Invalid Credentials";
    }
  }
</script>

{#if message}
  <Message {message} />
{/if}
<form on:submit|preventDefault={login}>
  <UserCredentials bind:email bind:password />
  <button class="button is-success is-fullwidth">Log In</button>
</form>
```

LoginForm.svelte



- Login Event Handler: invoked when button pressed

- Variables 'email' & 'password' bound as UserCredentials props

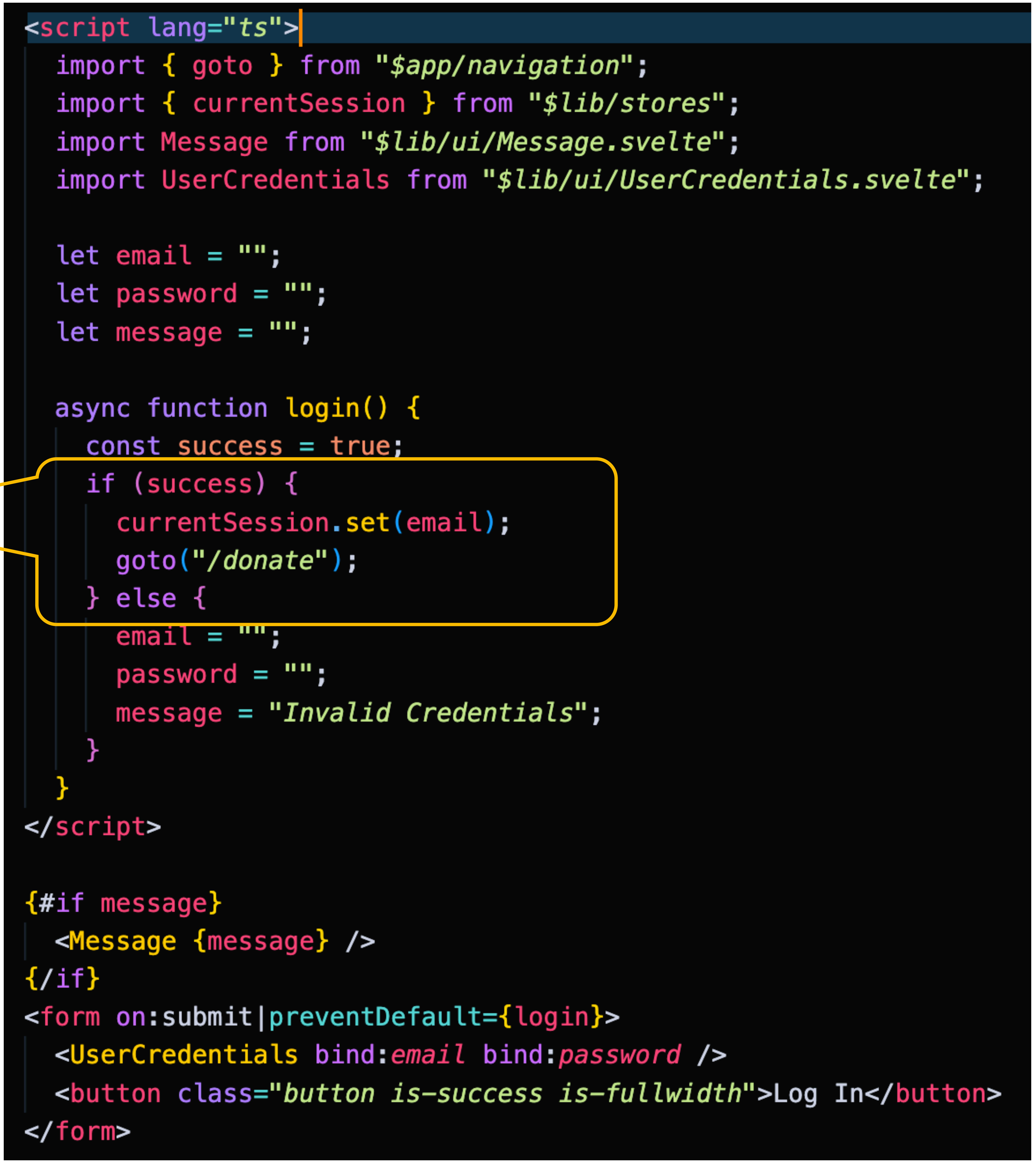


```
<script lang="ts">
  import { goto } from "$app/navigation";
  import { currentSession } from "$lib/stores";
  import Message from "$lib/ui/Message.svelte";
  import UserCredentials from "$lib/ui/UserCredentials.svelte";

  let email = "";
  let password = "";
  let message = "";

  async function login() {
    const success = true;
    if (success) {
      currentSession.set(email);
      goto("/donate");
    } else {
      email = "";
      password = "";
      message = "Invalid Credentials";
    }
  }
</script>

{#if message}
  <Message {message} />
{/if}
<form on:submit|preventDefault={login}>
  <UserCredentials bind:email bind:password />
  <button class="button is-success is-fullwidth">Log In</button>
</form>
```

LoginForm.svelte

- Hard code ‘success’ to true

- Goto forces SvelteKit router to load “/donate” route



```
<script lang="ts">
  import { goto } from "$app/navigation";
  import { currentSession } from "$lib/stores";
  import Message from "$lib/ui/Message.svelte";
  import UserCredentials from "$lib/ui/UserCredentials.svelte";

  let email = "";
  let password = "";
  let message = "";

  async function login() {
    const success = true;
    if (success) {
      currentSession.set(email);
      goto("/donate");
    } else {
      email = "";
      password = "";
      message = "Invalid Credentials";
    }
  }
</script>

{#if message}
  <Message {message} />
{/if}
<form on:submit|preventDefault={login}>
  <UserCredentials bind:email bind:password />
  <button class="button is-success is-fullwidth">Log In</button>
</form>
```

Pages & Components

Creating and assembling Svelte Components