Swift in the Browser with ElementaryUI

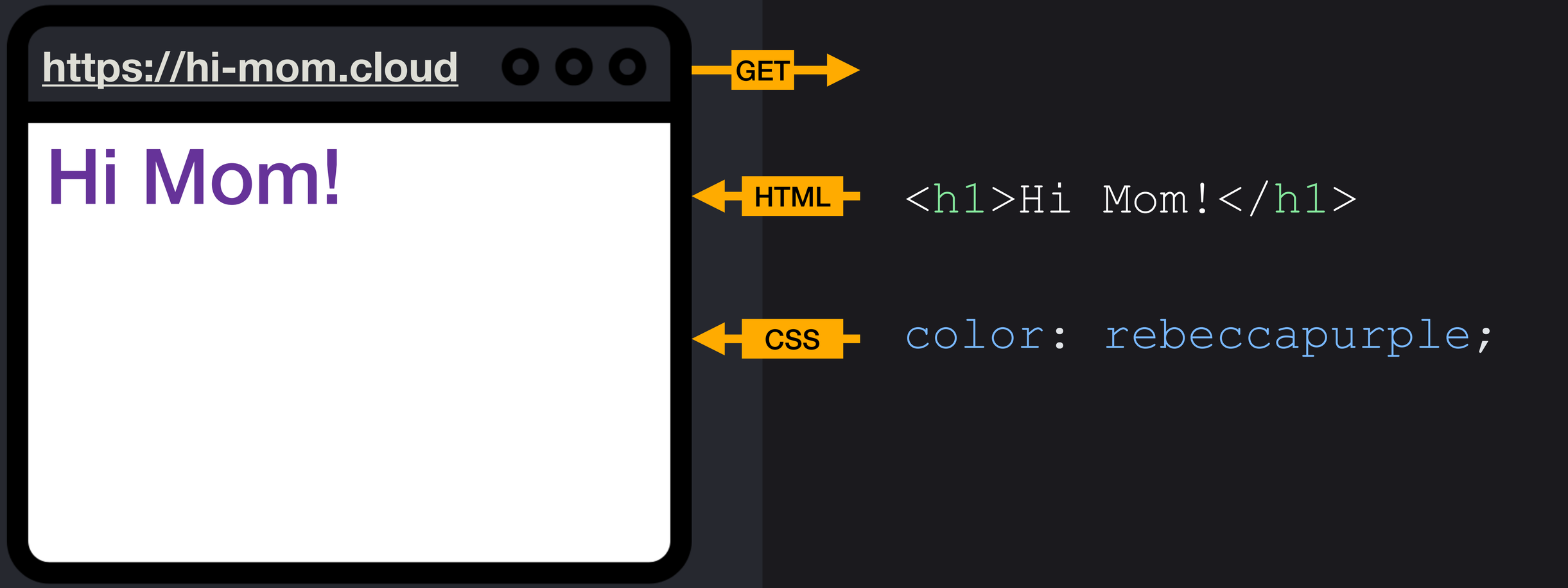
Simon Leeb

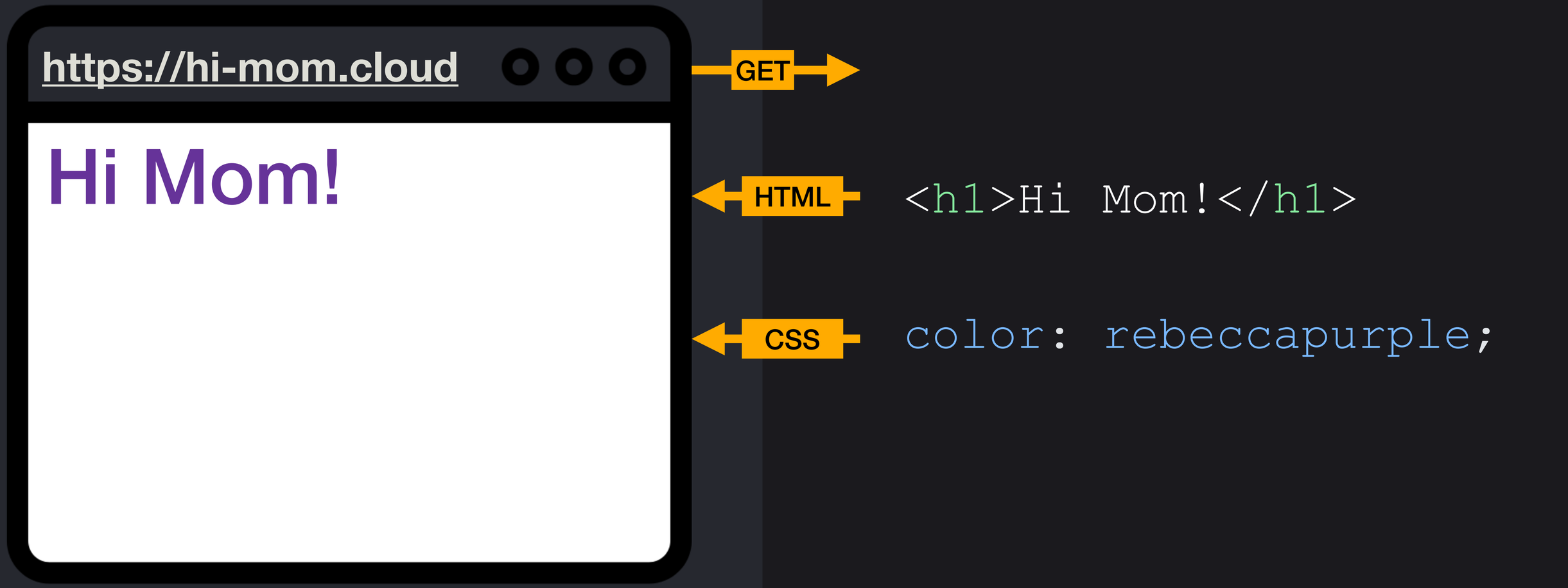

HTML


HTML
+CSS

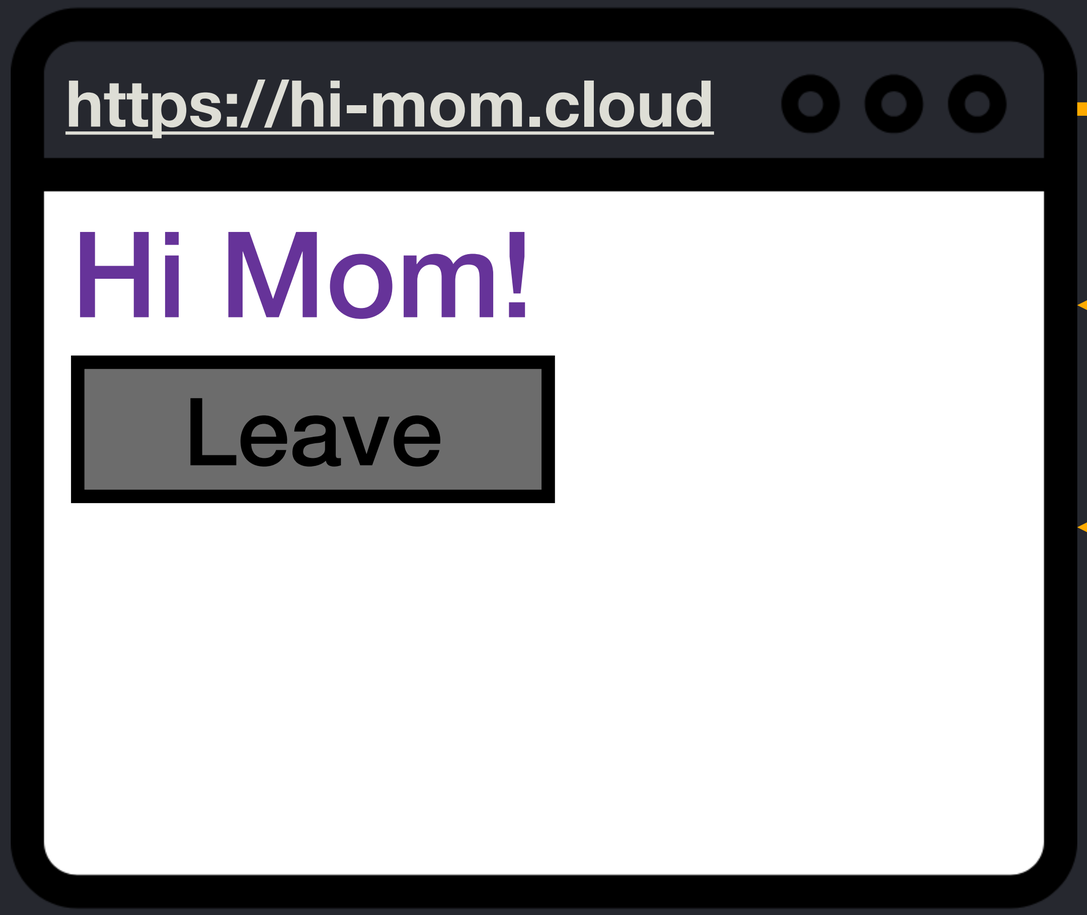
Browser

Server


Browser
Server
https://hi-mom.cloud
Hi Mom!
GET
HTML
<h1>Hi Mom!</h1>
CSS
color: rebeccapurple;


Browser
Server
https://hi-mom.cloud
Hi Mom!
GET
HTML
<h1>Hi Mom!</h1>
CSS
color: rebeccapurple;

Browser



Server

GET

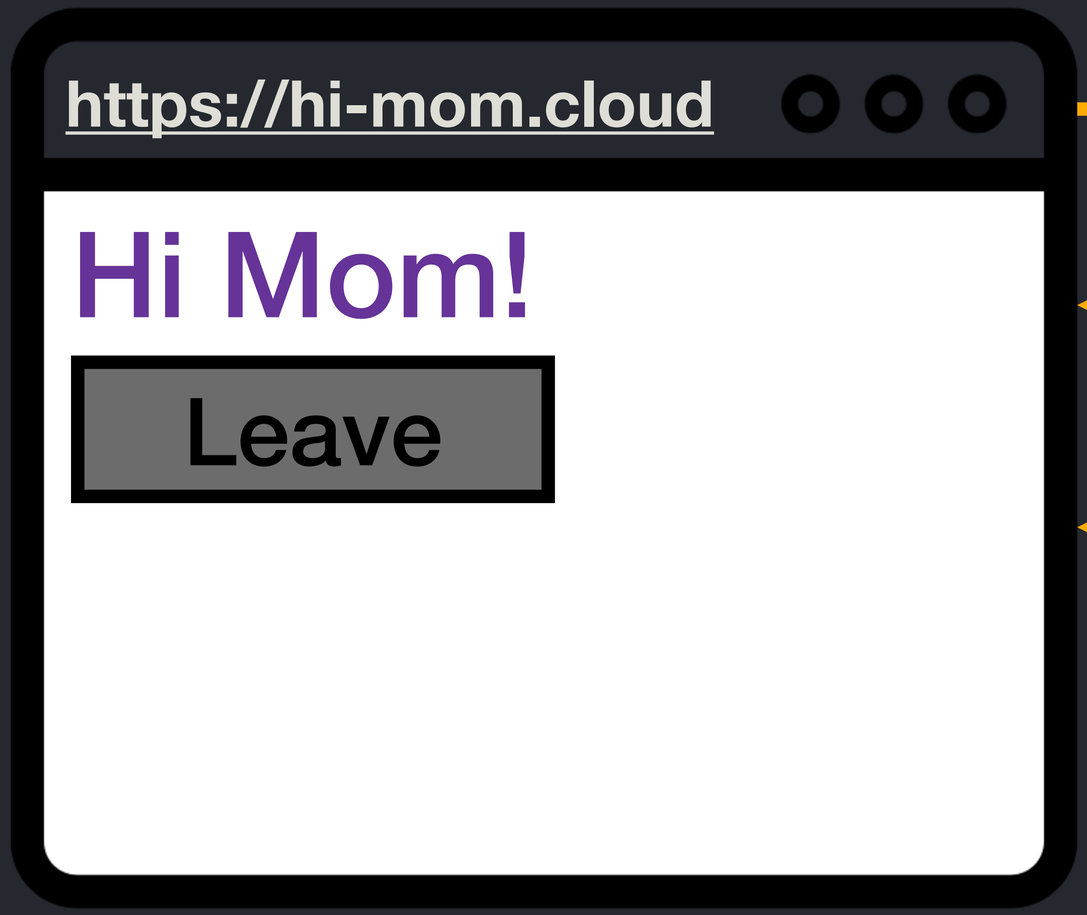
HTML

```
<h1>Hi Mom!</h1>
<button>Leave</button>
```

CSS

```
color: rebeccapurple;
```

Browser



Server

GET

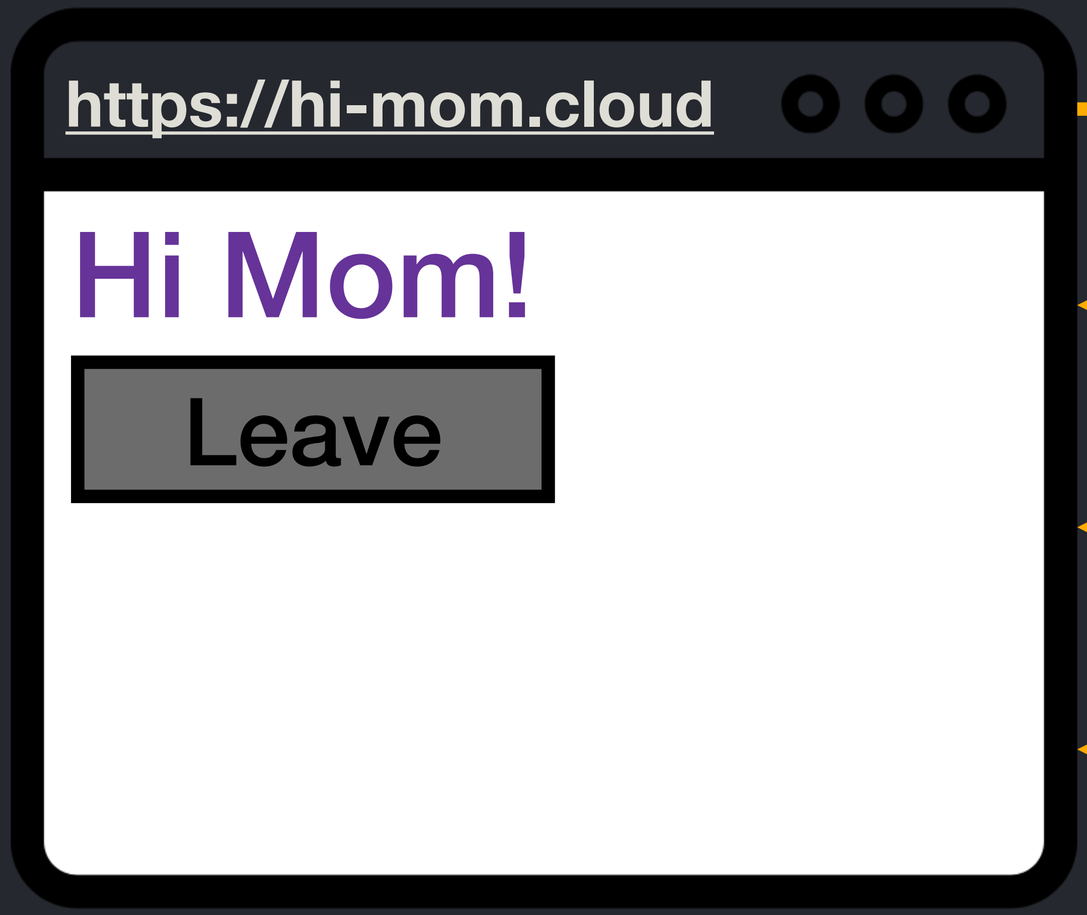
HTML

```
<h1>Hi Mom!</h1>
<button>Leave</button>
```

CSS

```
color: rebeccapurple;
```

Browser



Server

GET

HTML

```
<h1>Hi Mom!</h1>
<button>Leave</button>
```

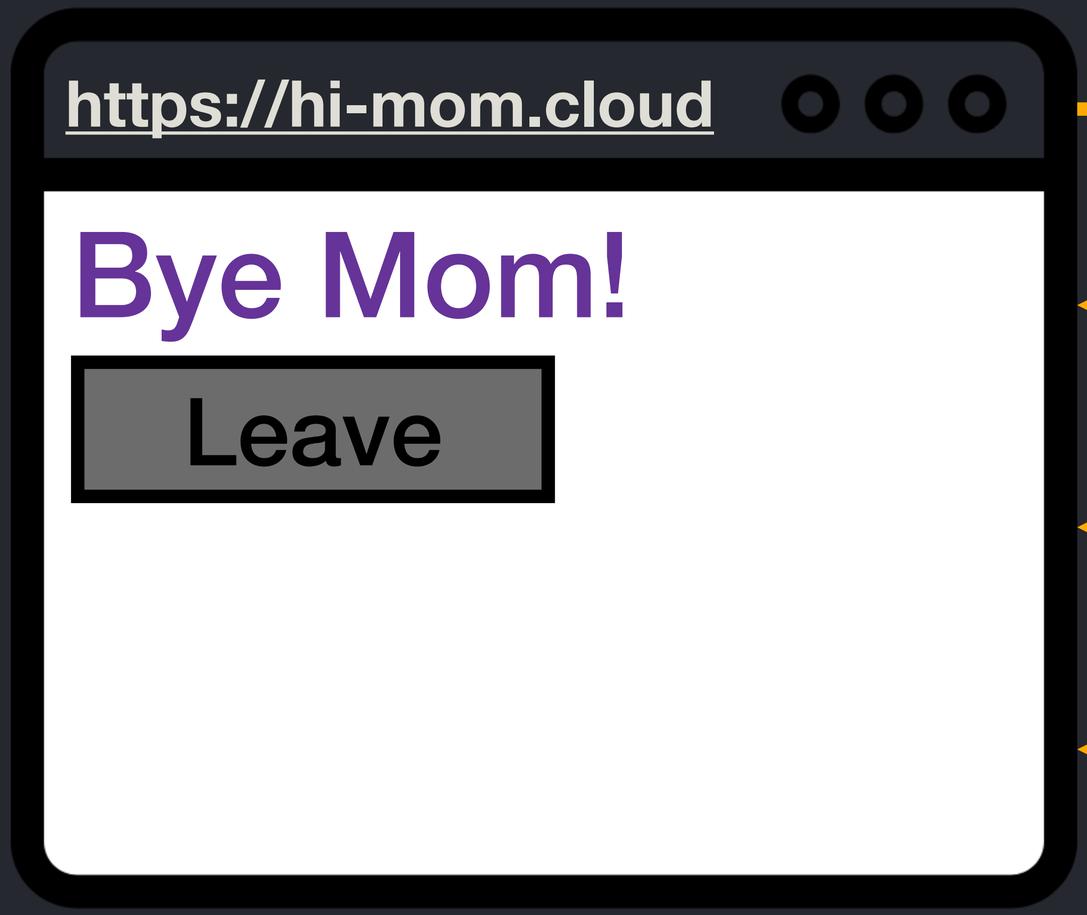
CSS

```
color: rebeccapurple;
```

JS

```
onclick="this
  .previousElementSibling
  .textContent = 'Bye Mom!'
"
```

Browser



Server

GET

HTML

```
<h1>Hi Mom!</h1>
<button>Leave</button>
```

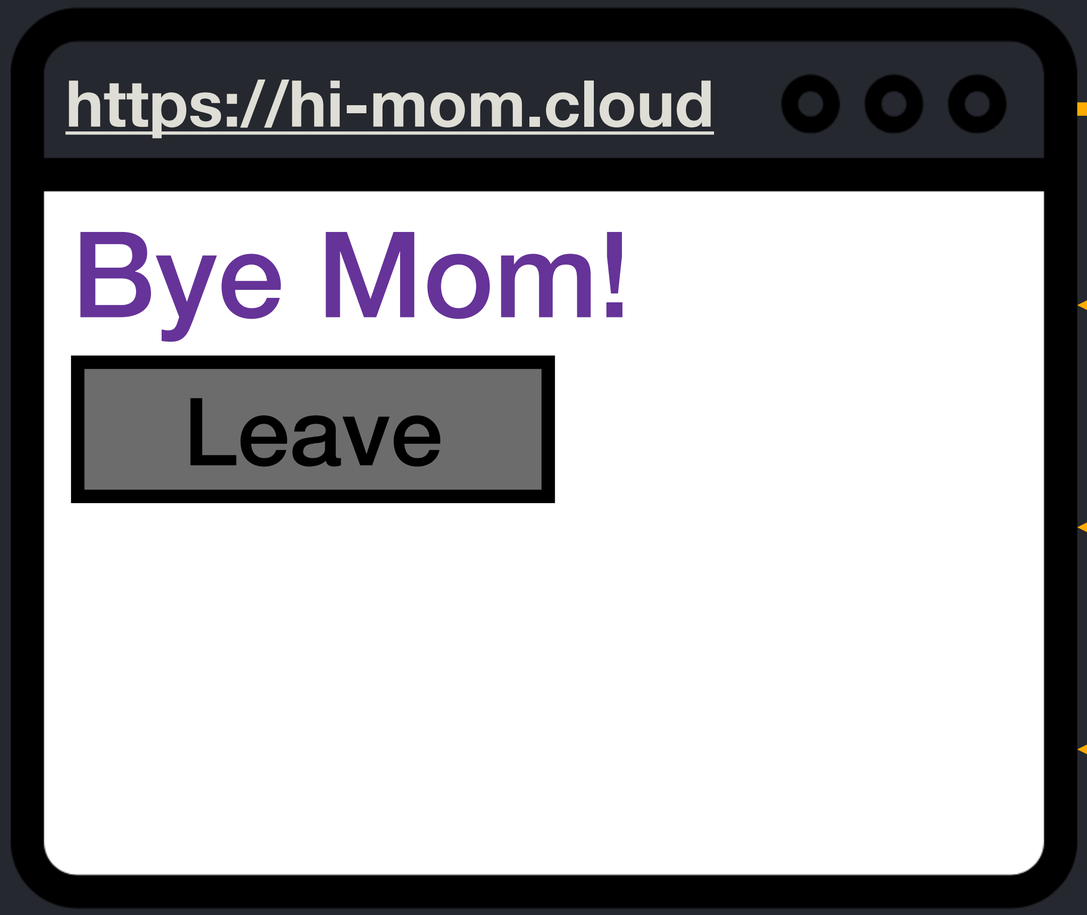
CSS

```
color: rebeccapurple;
```

JS

```
onclick="this
  .previousElementSibling
  .textContent = 'Bye Mom!'
"
```

Browser



Server

GET

HTML

```
<h1>Hi Mom!</h1>
<button>Leave</button>
```

CSS

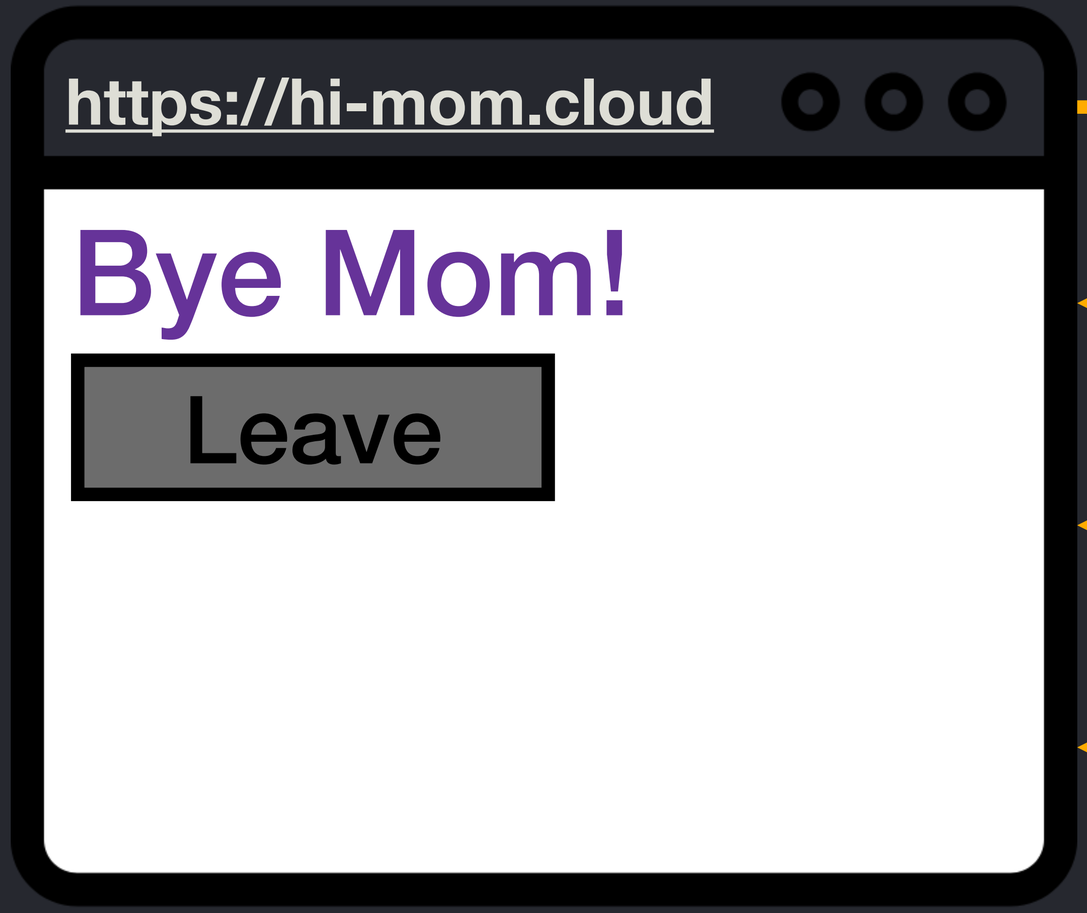
```
color: rebeccapurple;
```

JS

```
onclick="this
  .previousElementSibling
  .textContent = 'Bye Mom!'
"
```

Browser

Server



GET

HTML

```
<h1>Hi Mom!</h1>
<button>Leave</button>
```

CSS

```
color: rebeccapurple;
```

JS

```
onclick="this
  .previousElementSibling
  .textContent = 'Bye Mom!'
"
```

Server

```
import Elementary

struct MomView: HTML {
    var name = "Mom"

    var body: some HTML {
        h1 { "Hi \(name)!" }
        button(.on(.click, onClickJS)) { "Leave" }
    }

    private var onClickJS: String {
        "this.previousElementSibling.textContent = 'Bye \(name)!'"
    }
}
```

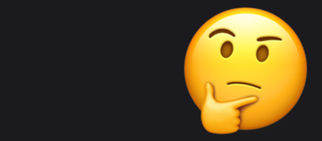
Server

```
import Elementary

struct MomView: HTML {
    var name = "Mom"

    var body: some HTML {
        h1 { "Hi \(name)!" }
        button(.on(.click, onClickJS)) { "Leave" }
    }

    private var onClickJS: String {
        "this.previousElementSibling.textContent = 'Bye \(name)!'"
    }
}
```

Server

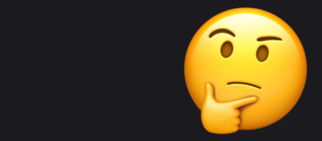
```
import Elementary

struct MomView: HTML {
    var name = "Mom"

    var body: some HTML {
        h1 { "Hi \(name)!" }
        button(.on(.click, onClickJS)) { "Leave" }
    }

    private var onClickJS: String {
        "this.previousElementSibling.textContent = 'Bye \(name)!'"
    }
}
```

Server



```
import Elementary

struct MomView: HTML {
    var name = "Mom"

    var isLeaving = false

    var body: some HTML {
        h1 { isLeaving ? "Bye \(name)!" : "Hi \(name)!" }
        button { "Leave" }
            .onClick { isLeaving = true }
    }
}
```

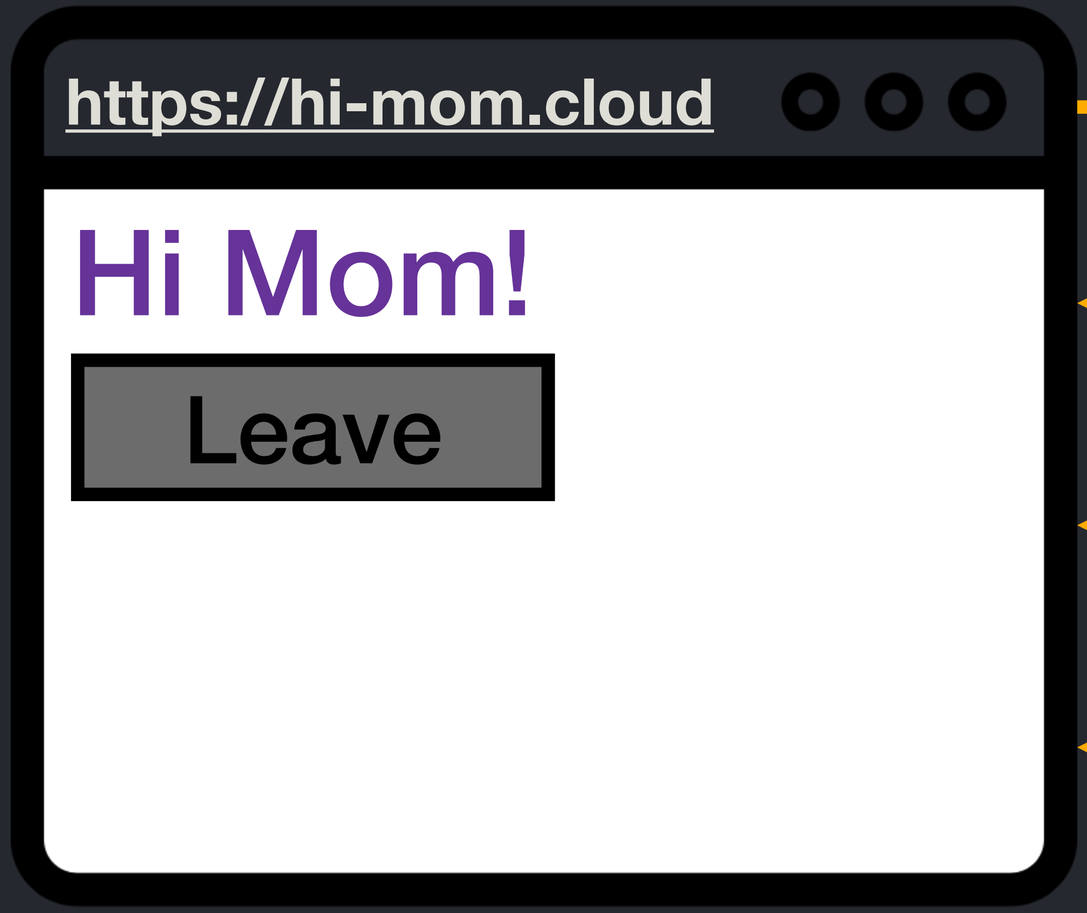
Server



```
import Elementary

struct MomView: HTML {
    var name = "Mom"

    @State var isLeaving = false

    var body: some HTML {
        h1 { isLeaving ? "Bye \(name)!" : "Hi \(name)!" }
        button { "Leave" }
            .onClick { isLeaving = true }
    }
}
```

Browser

Server



GET

HTML

```
<h1>Hi Mom!</h1>
<button>Leave</button>
```

CSS

```
color: rebeccapurple;
```

JS

```
onclick="this
  .previousElementSibling
  .textContent = 'Bye Mom!'
"
```

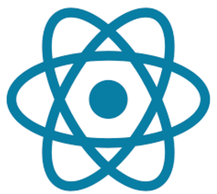
Browser

Server

https://hi-mom.cloud

Hi Mom!

Leave



GET

HTML

```
<h1>Hi Mom!</h1>
<button>Leave</button>
```

CSS

```
color: rebeccapurple;
```

JS

```
onclick="this
  .previousElementSibling
  .textContent = 'Bye Mom!'
"
```

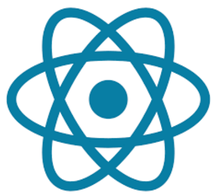
Browser

Server

https://hi-mom.cloud

Hi Mom!

Leave



GET

HTML

```
<h1>Hi Mom!</h1>
<button>Leave</button>
```

CSS

```
color: rebeccapurple;
```

JS

```
onclick="this
  .previousElementSibling
  .textContent = 'Bye Mom!'
"
```

Browser

Server

GET

JS

```
<h1>Hi Mom!</h1>
<button>Leave</button>
```

JS

```
color: rebeccapurple;
```

JS

```
onclick="this
  .previousElementSibling
  .textContent = 'Bye Mom!'
"
```

Browser

Server

GET

JS

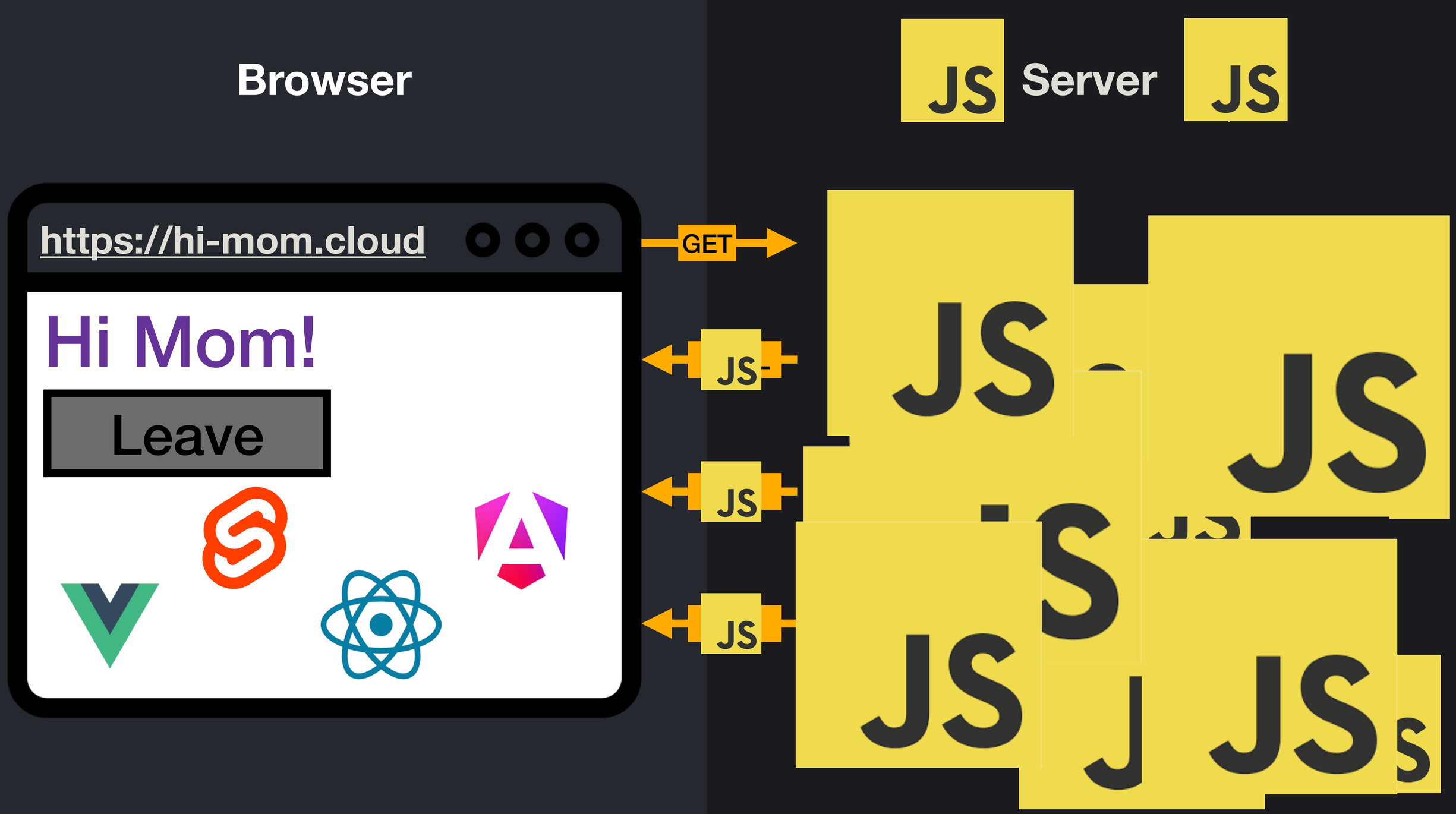
```
<h1>Hi Mom!</h1>
<button>Leave</button>
```

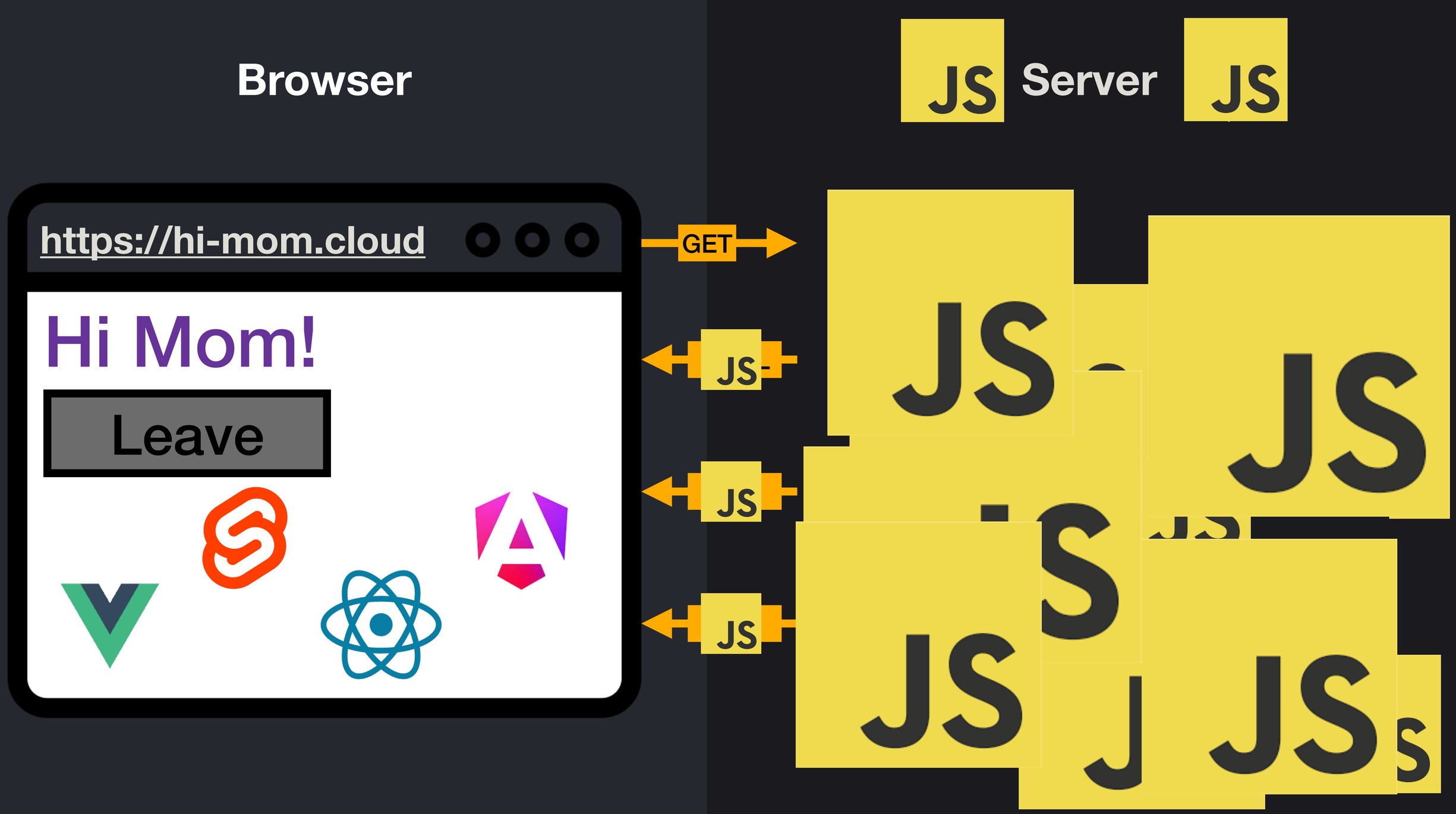
JS

```
color: rebeccapurple;
```

JS

```
onclick="this
  .previousElementSibling
  .textContent = 'Bye Mom!'
"
```


Browser
JS
Server
JS
https://hi-mom.cloud
GET
Hi Mom!
Leave
JS
JS
JS


Browser
JS
Server
JS
https://hi-mom.cloud
GET
Hi Mom!
Leave
JS
JS
JS

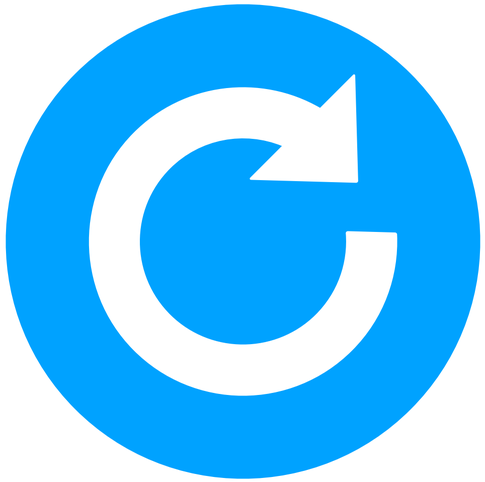
Use the right tool for the job

Uncaught TypeError:

Cannot read properties of undefined (reading 'nextSlide')

Uncaught TypeError:

nextSlide is not a function



Use the right tool for the job

Use the right tool for the job

An error occurred:
[object Object]

An error occurred:
[object Object]

Use the right tool for the job

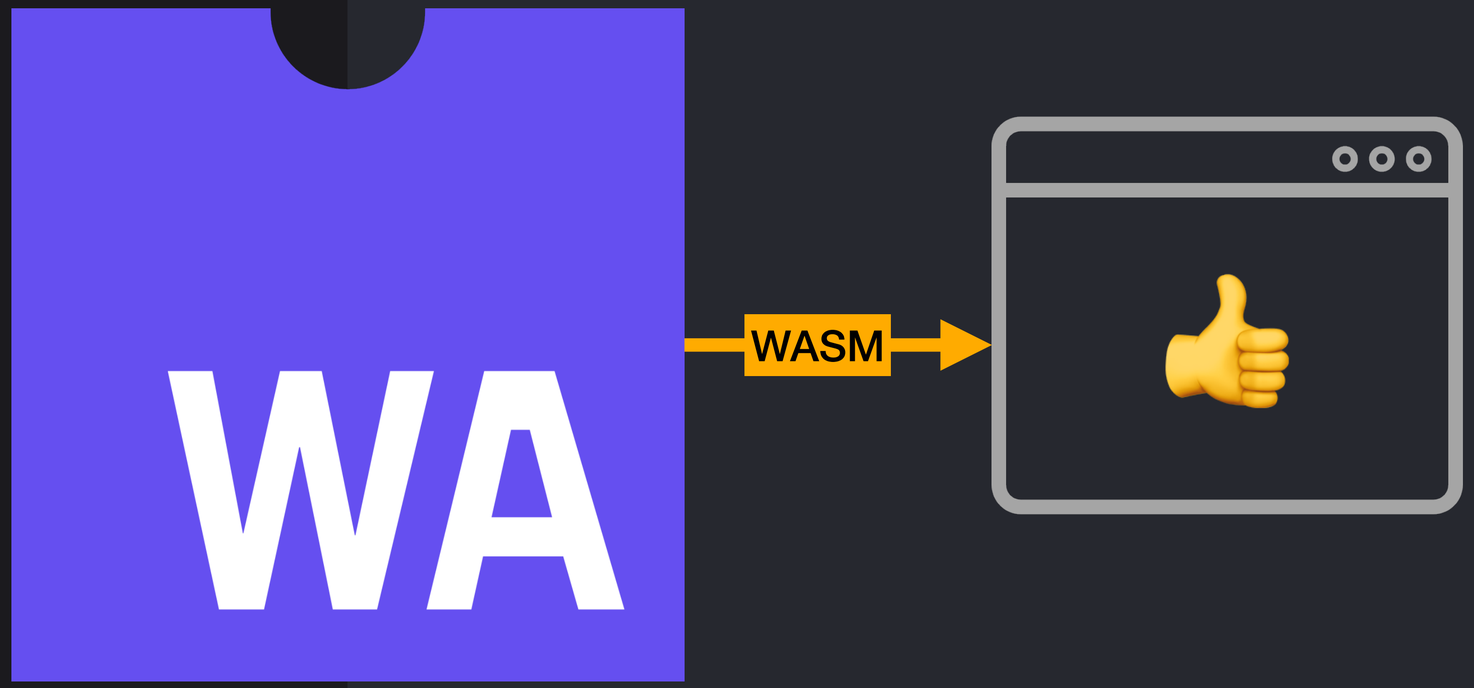
Use the right tool for the job

Use the right ~~tool~~ Swift for the job

Use the right ~~tool~~ Swift for the job

WebAssembly

WebAssembly



Swift SDK for WebAssembly

Swift SDK for WebAssembly

Swift SDK for WebAssembly

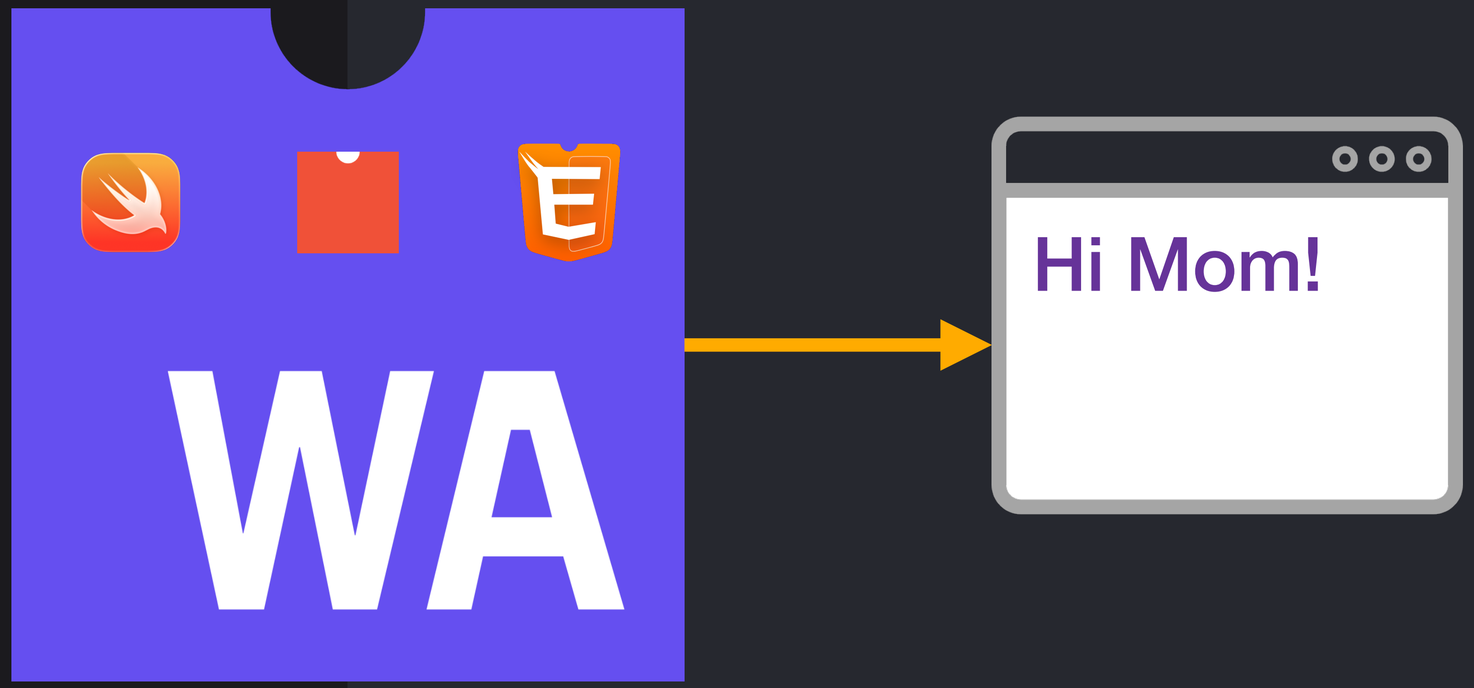
Swift SDK for WebAssembly

Swift SDK for WebAssembly

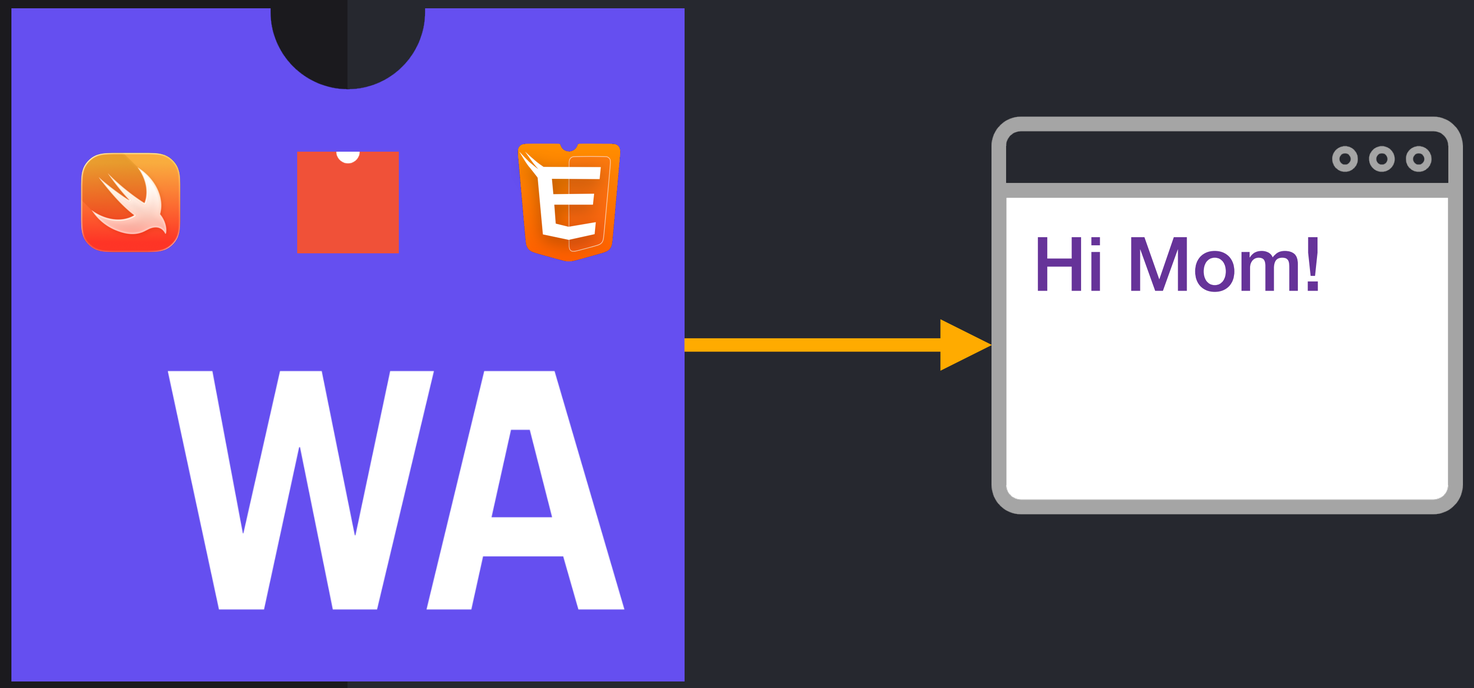
Swift SDK for WebAssembly

Swift SDK for WebAssembly

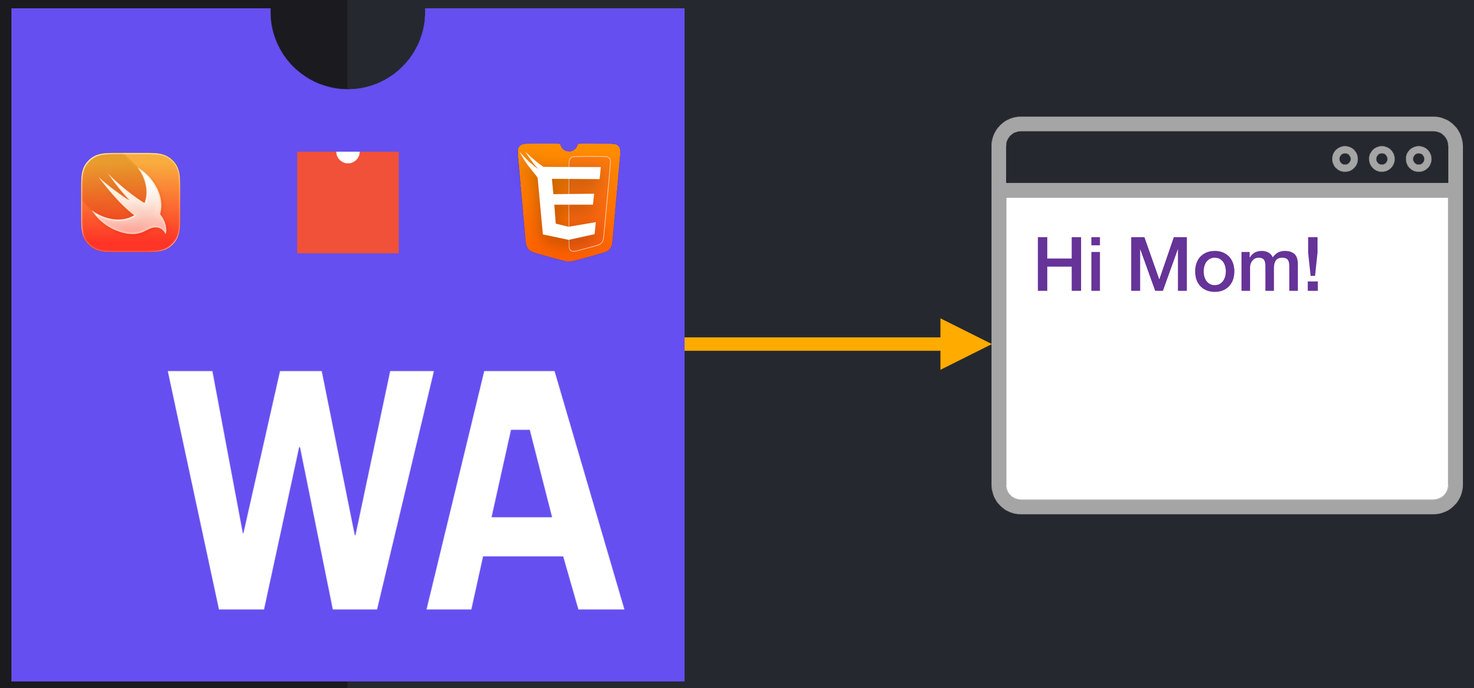
Swift SDK for WebAssembly



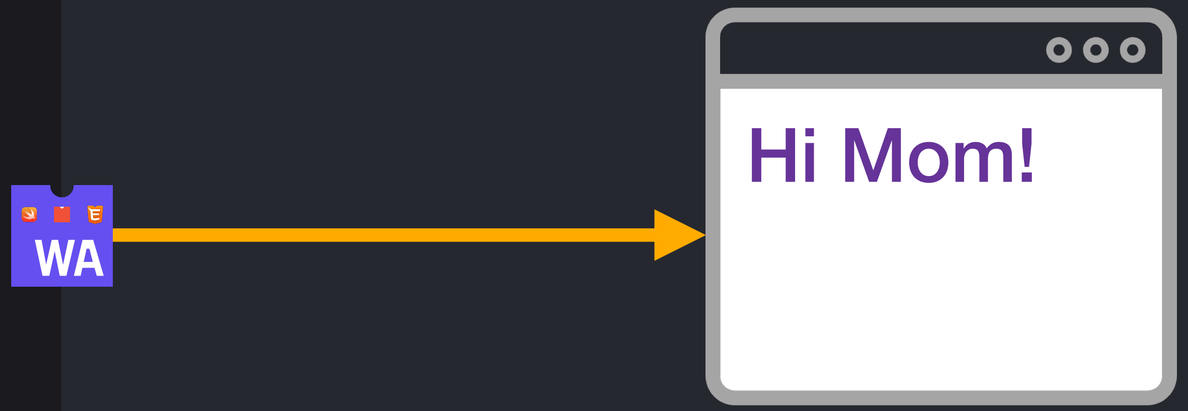
Embedded Swift SDK for WebAssembly



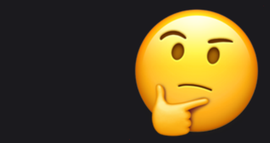
Embedded Swift SDK for WebAssembly



Embedded Swift SDK for WebAssembly



```
import Elementary


struct MomView: HTML {
    var name = "Mom"

    @State var isLeaving = false

    var body: some HTML {
        h1 { isLeaving ? "Bye \(name)!" : "Hi \(name)!" }
        button { "Leave" }
            .onClick { isLeaving = true }
    }
}
```



```
import ElementaryUI

@View
struct MomView: HTML {
    var name = "Mom"

    @State var isLeaving = false

    var body: some HTML {
        h1 { isLeaving ? "Bye \(name)!" : "Hi \(name)!" }
        button { "Leave" }
            .onClick { isLeaving = true }
    }
}
```

Browser

```
import ElementaryUI

@View
struct MomView {
    var name = "Mom"

    @State var isLeaving = false

    var body: some HTML {
        h1 { isLeaving ? "Bye \(name)!" : "Hi \(name)!" }
        button { "Leave" }
            .onClick { isLeaving = true }
    }
}
```

Demo Time

The Road to 1.0

Fetch API

@FocusState

Router

OpenAPI

SSG + Islands

WebAssembly Reference Types

Navigation

Web Storage API

SSR + Hydration

JavaScriptKit 1.0

SVG

PhaseAnimator

Web Components

BridgeJS

The to 1.0

Fetch API

@Foc

Router

OpenAPI

dIslands

WebAssembly Refe

Navigation

SSR + Hydratio

PI

JavaScriptKit 1.0

PhaseAnimator

Bridge

Web Components

Swift in the Browser

Swift in the Browser

Let's make it happen!

https://elementary.codes

Swift in the Browser

Let's make it happen!

https://elementary.codes