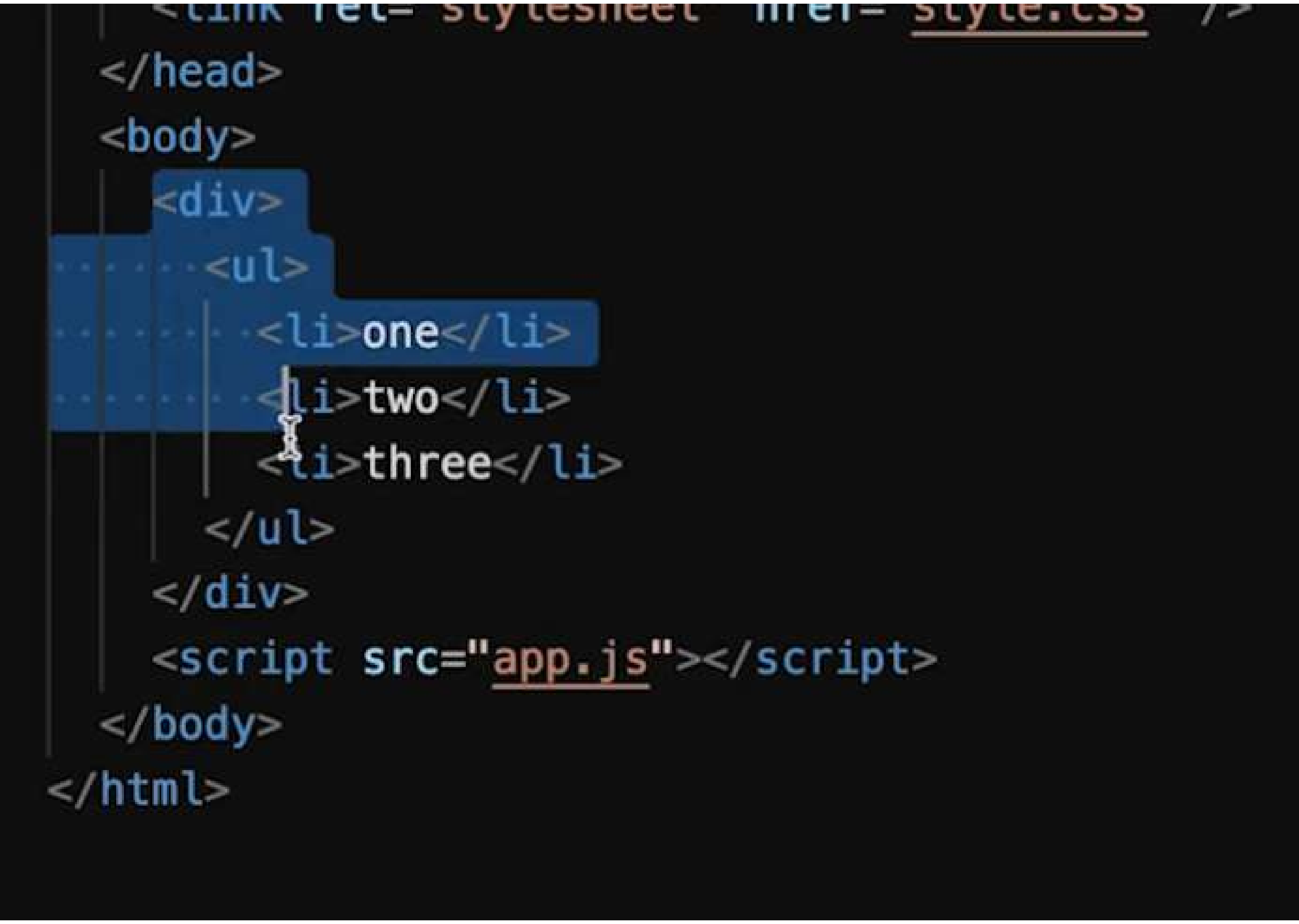
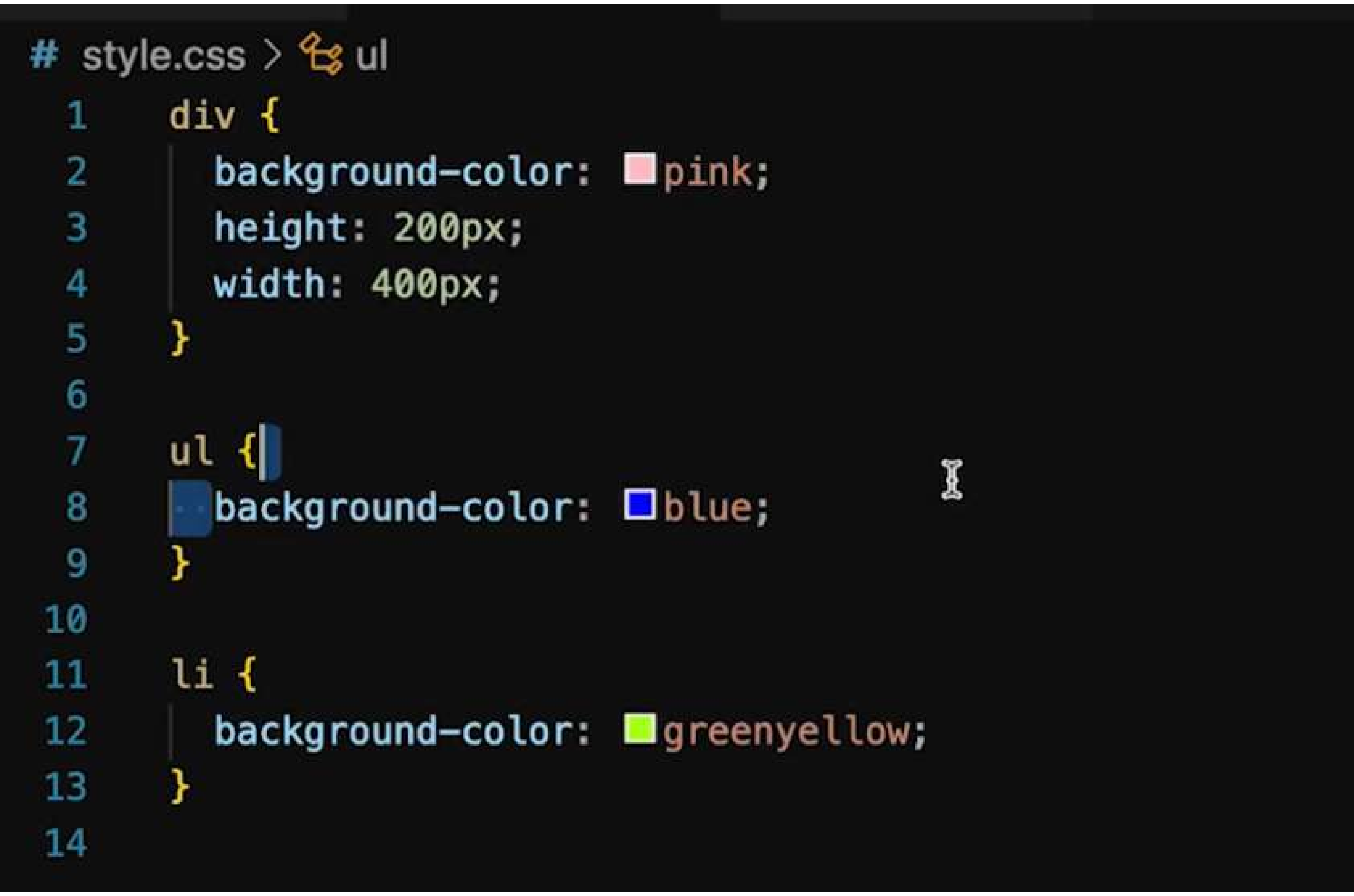
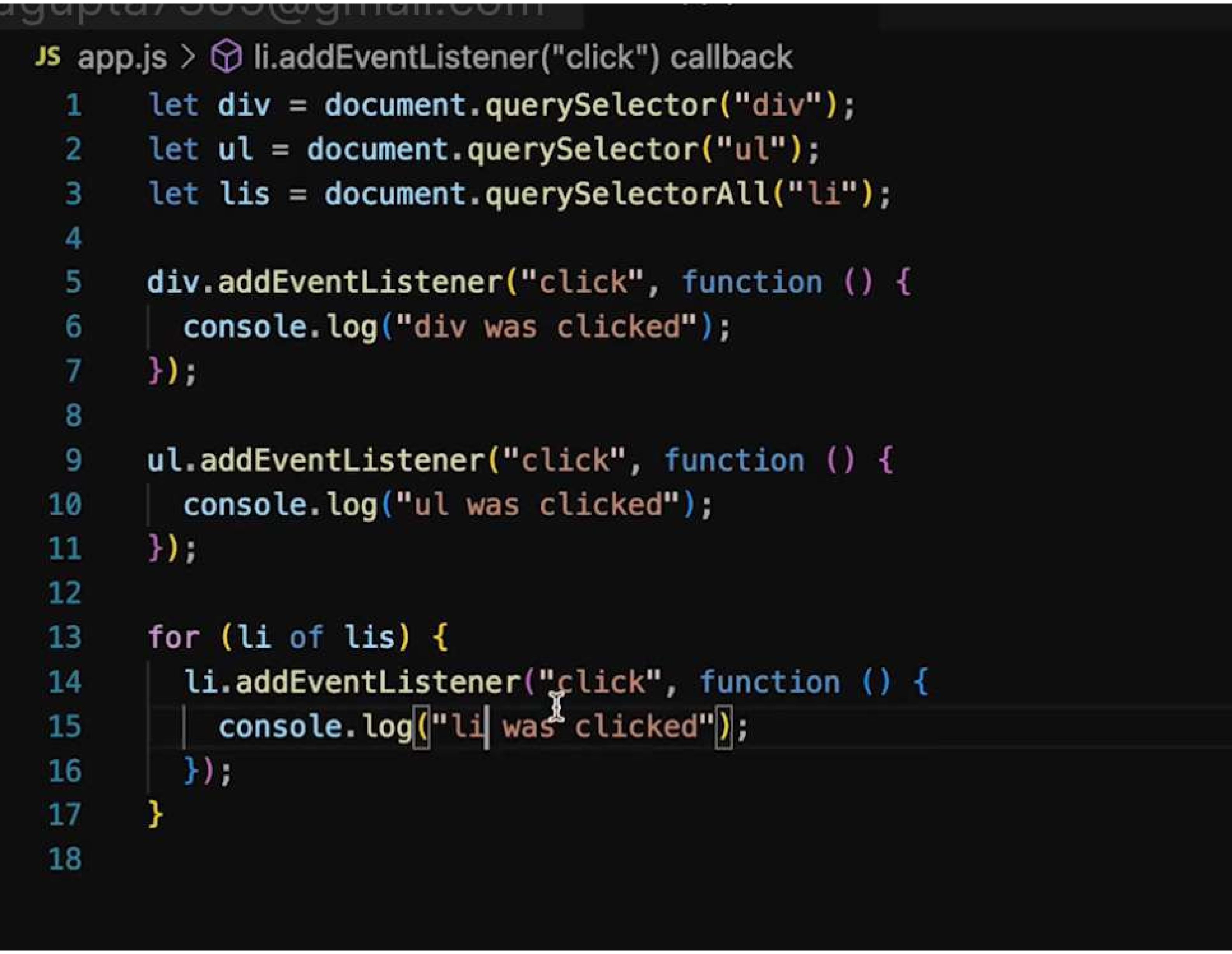
Event Bubbling



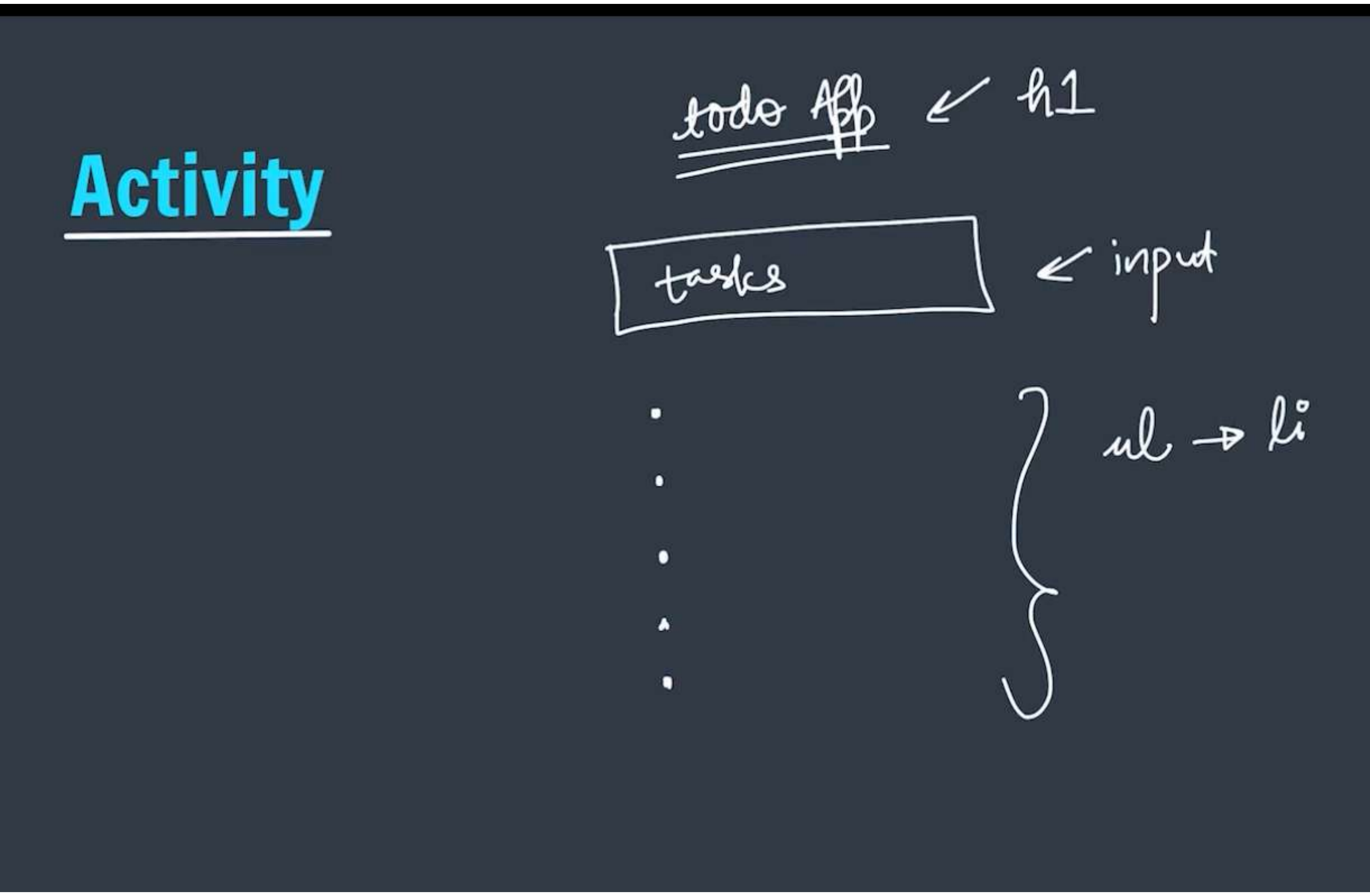
```
    <link rel="stylesheet" href="style.css" />
  </head>
  <body>
    <div>
      <ul>
        <li>one</li>
        <li>two</li>
        <li>three</li>
      </ul>
    </div>
    <script src="app.js"></script>
  </body>
</html>
```



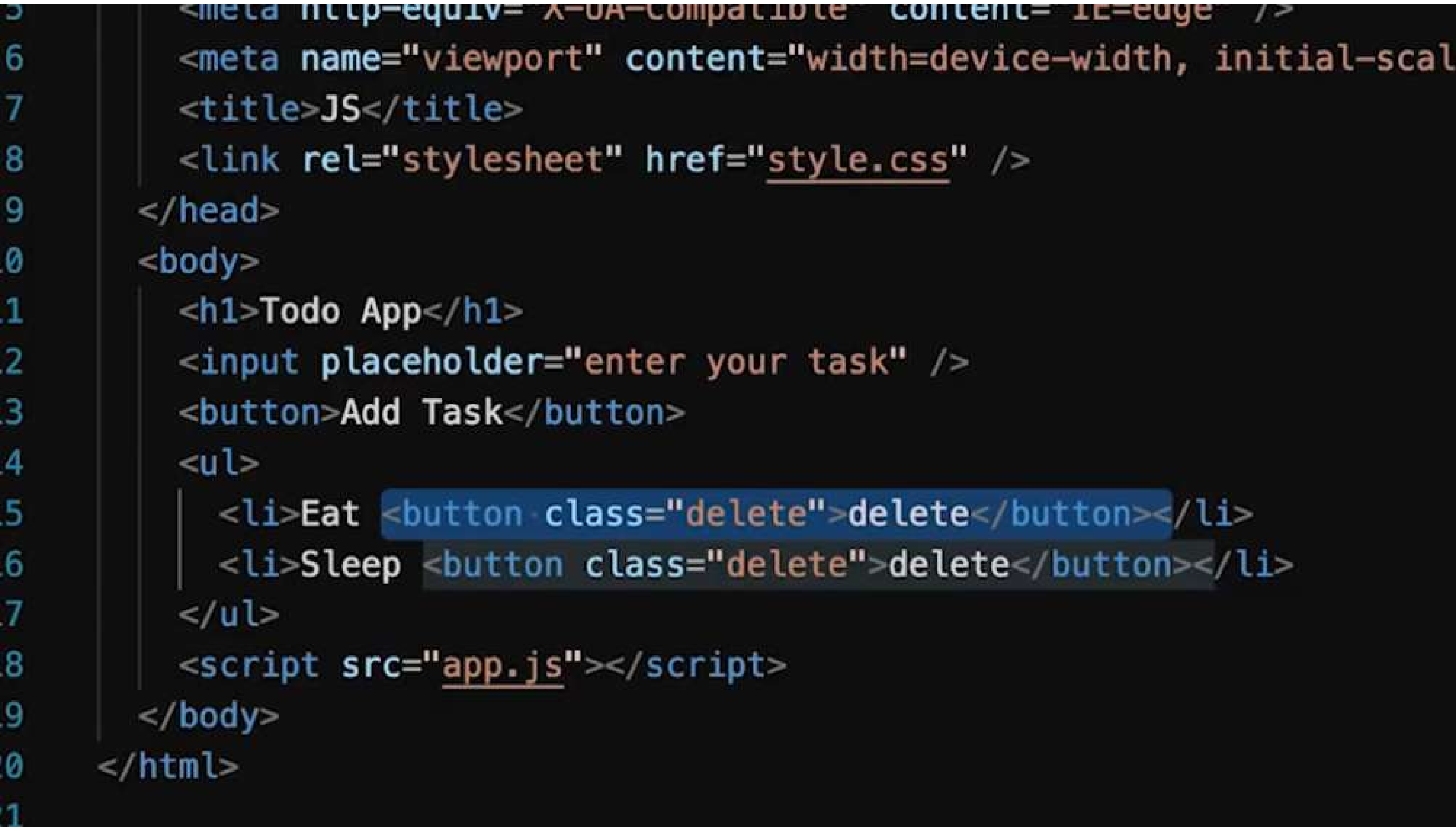
```css
div {
  background-color: pink;
  height: 200px;
  width: 400px;
}

ul {
  background-color: blue;
}

li {
  background-color: greenyellow;
}

```



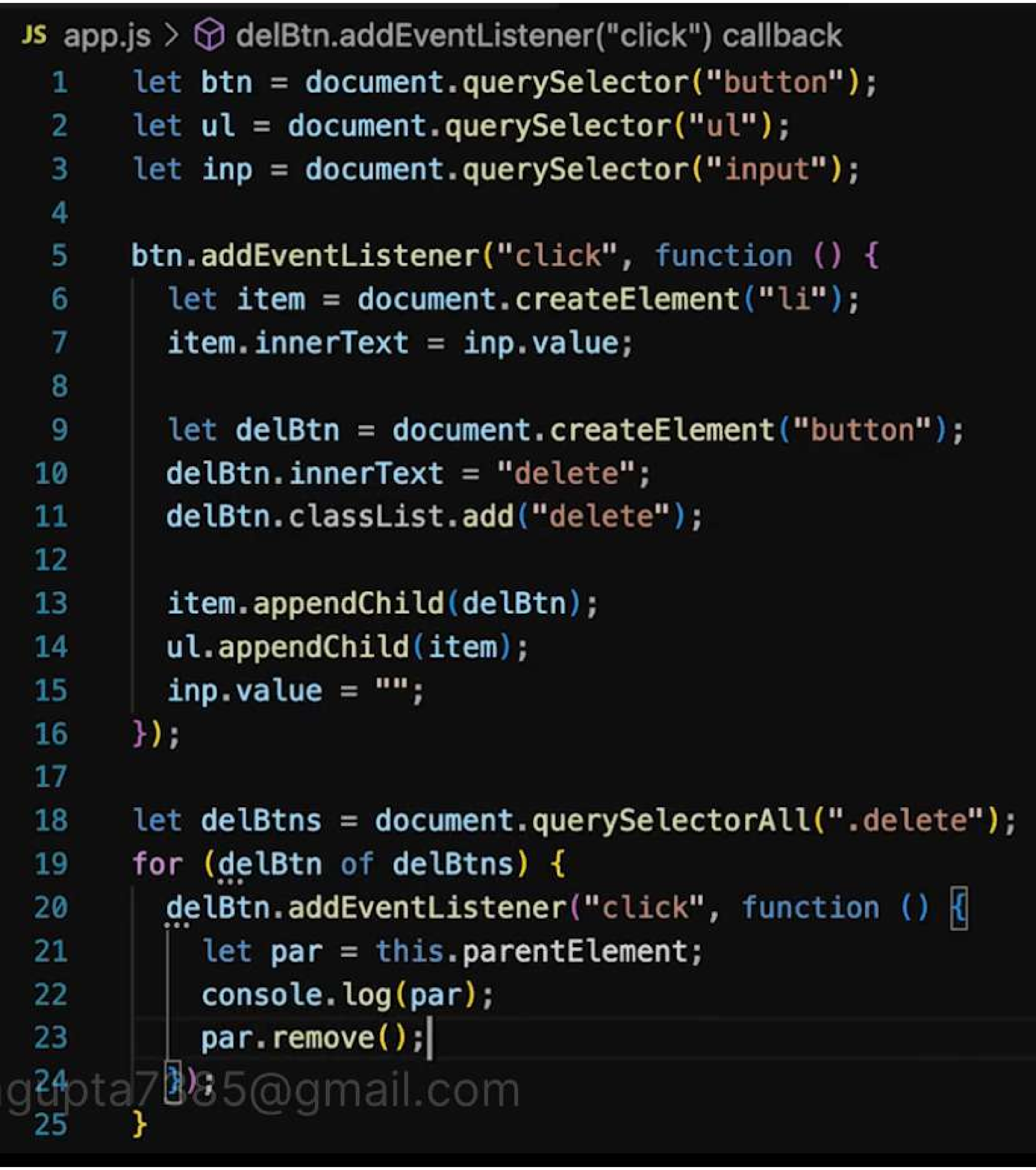
```javascript
let div = document.querySelector("div");
let ul = document.querySelector("ul");
let lis = document.querySelectorAll("li");

div.addEventListener("click", function () {
  console.log("div was clicked");
});

ul.addEventListener("click", function () {
  console.log("ul was clicked");
});

for (li of lis) {
  li.addEventListener("click", function () {
    console.log("li was clicked");
  });
}
```



Activity

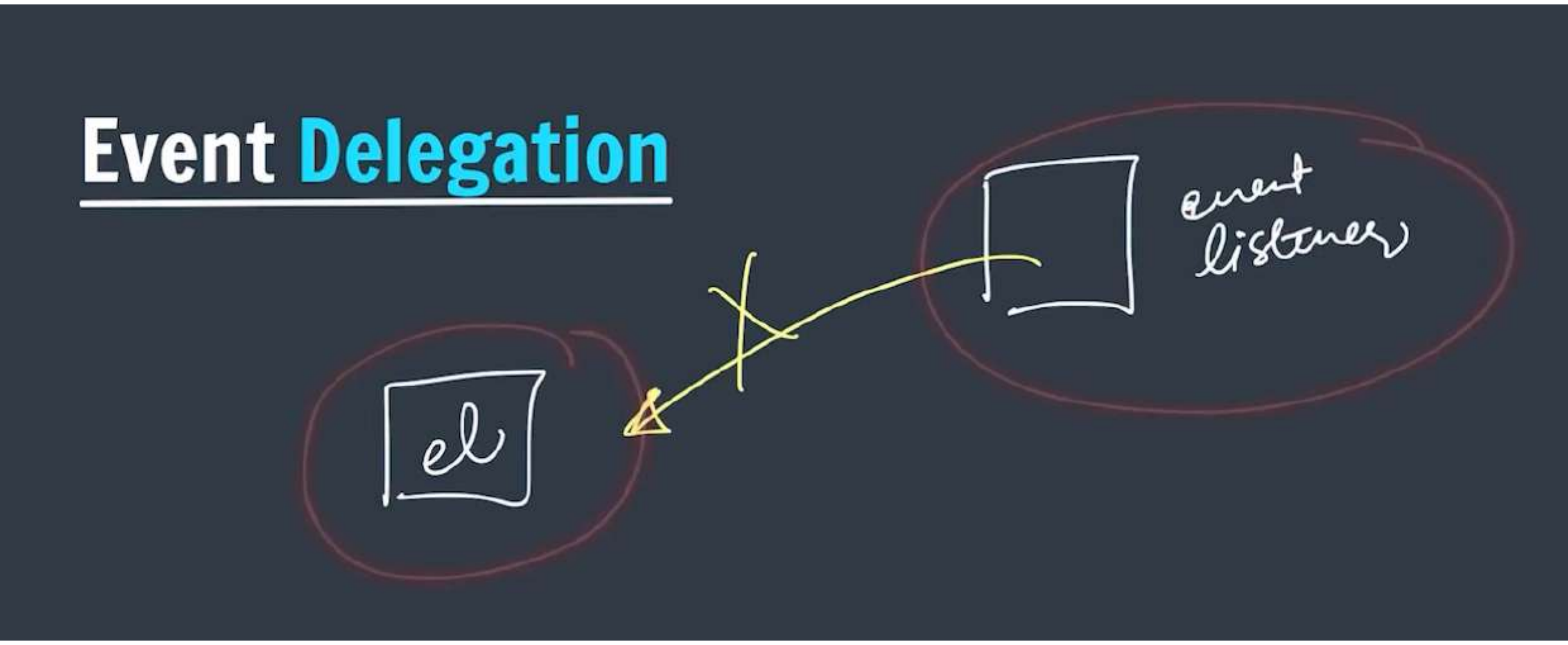


```
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scal
    <title>JS</title>
    <link rel="stylesheet" href="style.css" />
  </head>
  <body>
    <h1>Todo App</h1>
    <input placeholder="enter your task" />
    <button>Add Task</button>
    <ul>
      <li>Eat <button class="delete">delete</button></li>
      <li>Sleep <button class="delete">delete</button></li>
    </ul>
    <script src="app.js"></script>
  </body>
</html>
```

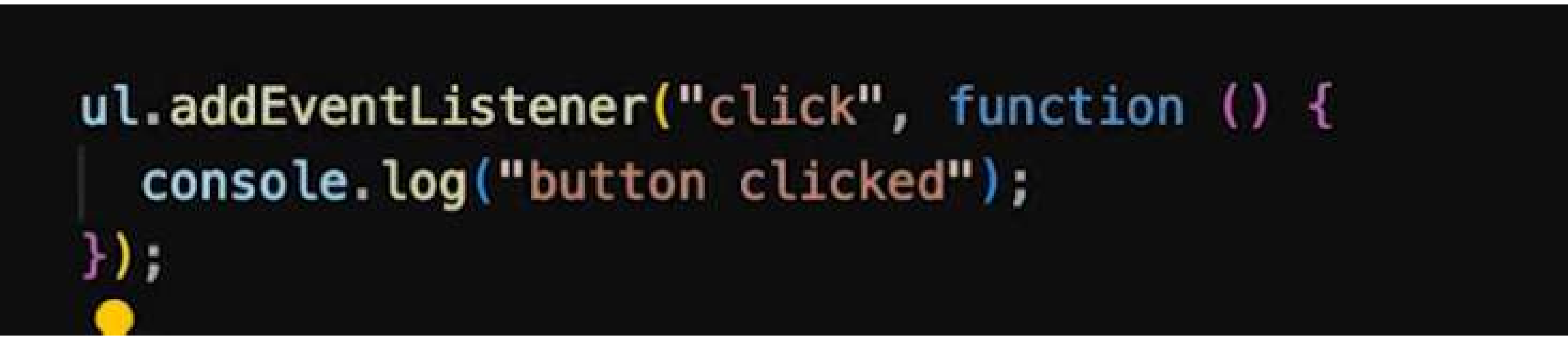


JS app.js > delBtn.addEventListener("click") callback

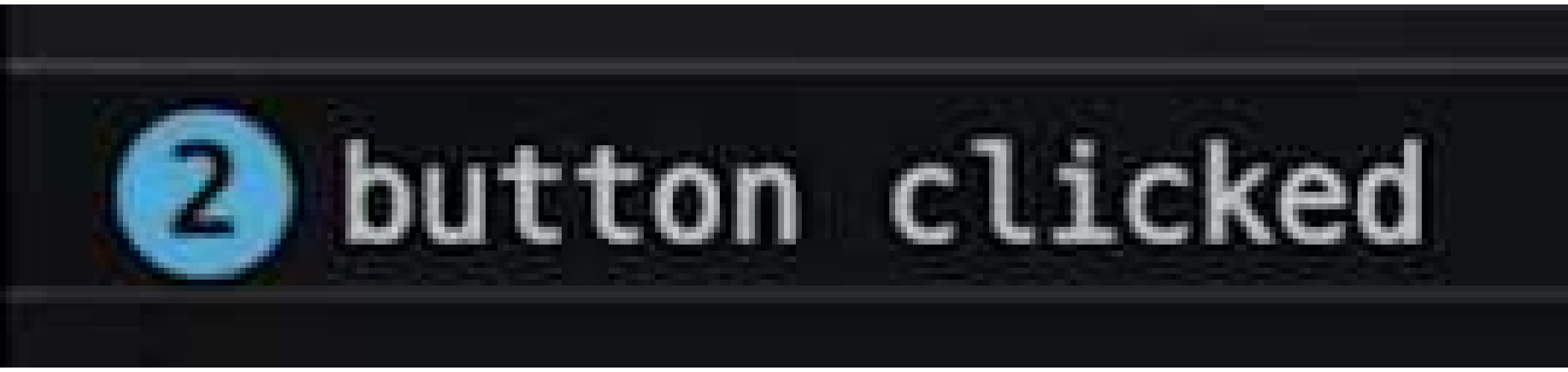
```
let btn = document.querySelector("button");
let ul = document.querySelector("ul");
let inp = document.querySelector("input");

btn.addEventListener("click", function () {
  let item = document.createElement("li");
  item.innerText = inp.value;

  let delBtn = document.createElement("button");
  delBtn.innerText = "delete";
  delBtn.classList.add("delete");

  item.appendChild(delBtn);
  ul.appendChild(item);
  inp.value = "";
});

let delBtns = document.querySelectorAll(".delete");
for (delBtn of delBtns) {
  delBtn.addEventListener("click", function () {
    let par = this.parentElement;
    console.log(par);
    par.remove();
  });
}
```



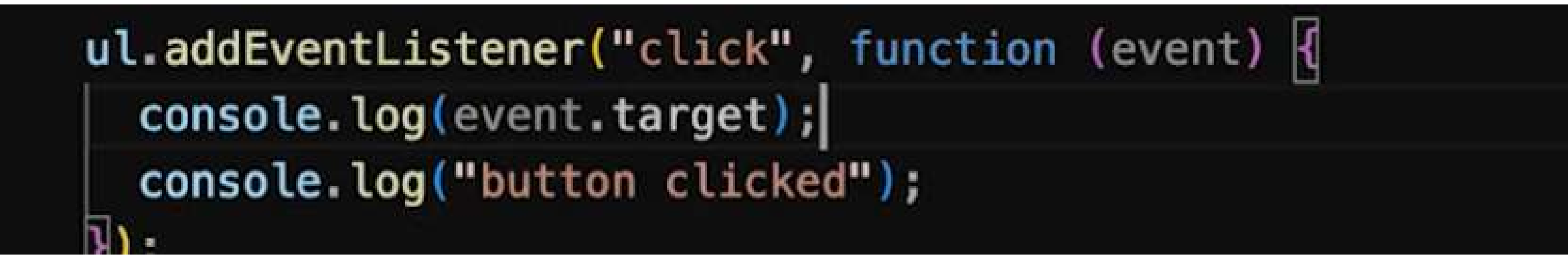
Event Delegation



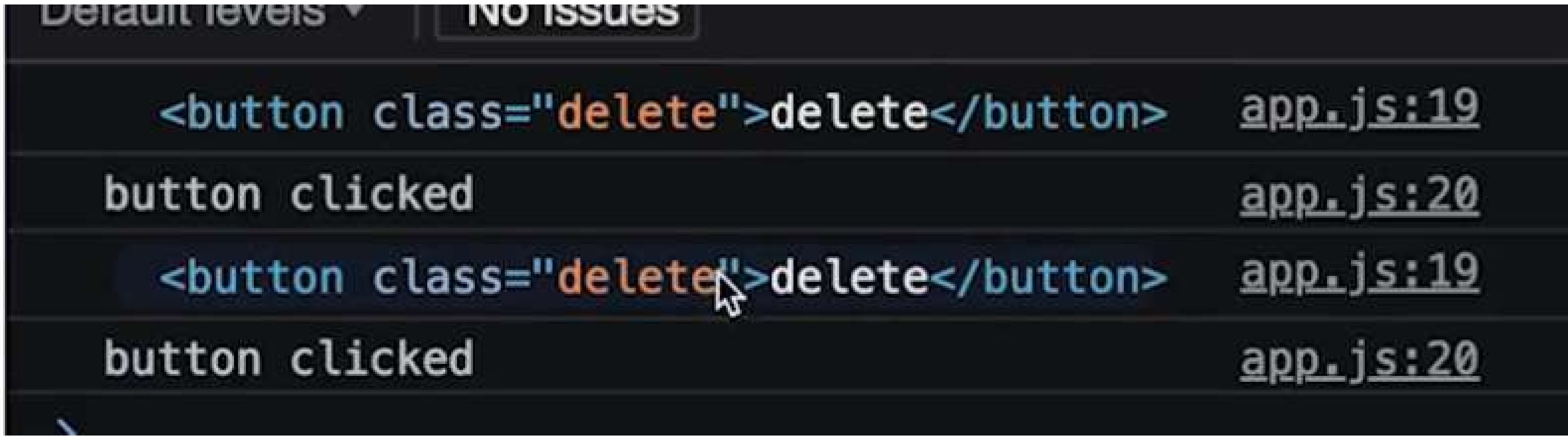
```
ul.addEventListener("click", function () {
  console.log("button clicked");
});
```



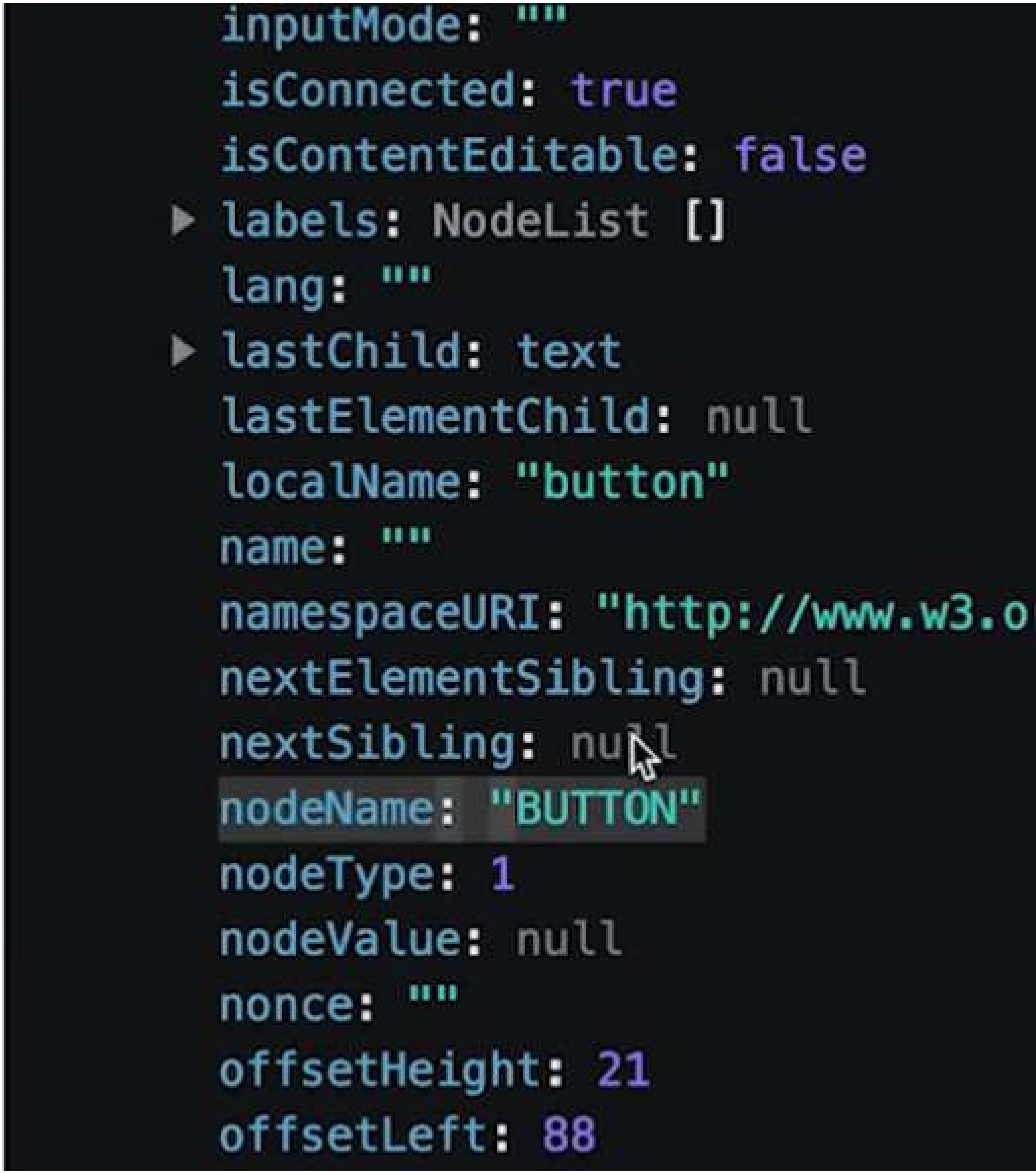
```
2 button clicked
```



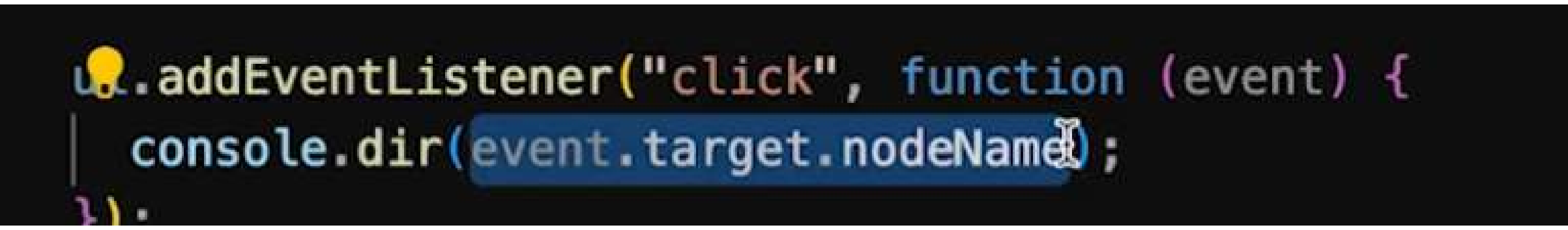
```
ul.addEventListener("click", function (event) {
  console.log(event.target);
  console.log("button clicked");
});
```



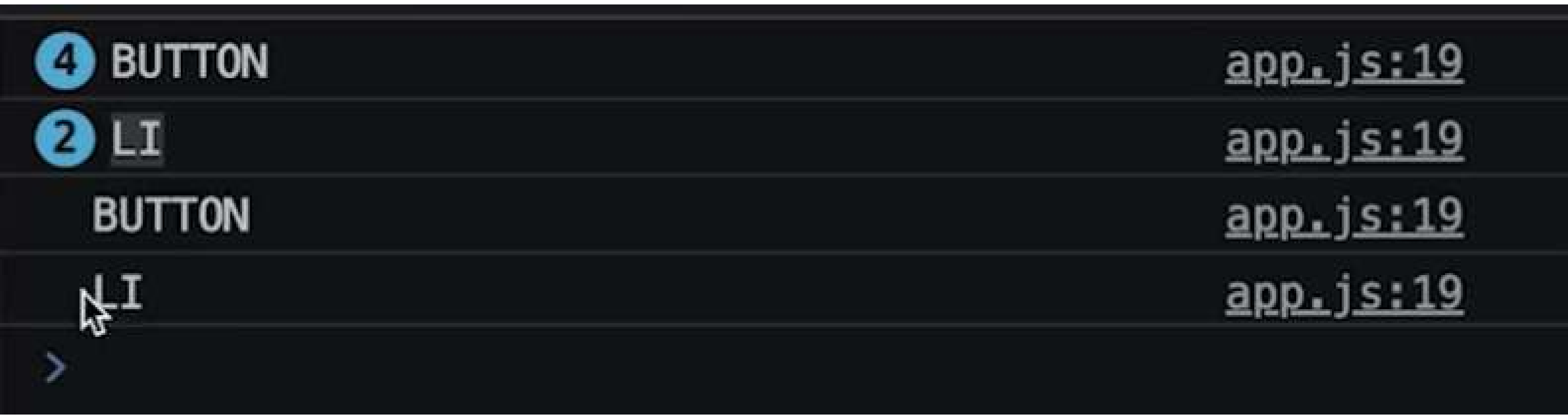
```
  <button class="delete">delete</button>    app.js:19
button clicked                              app.js:20
  <button class="delete">delete</button>    app.js:19
button clicked                              app.js:20
>
```



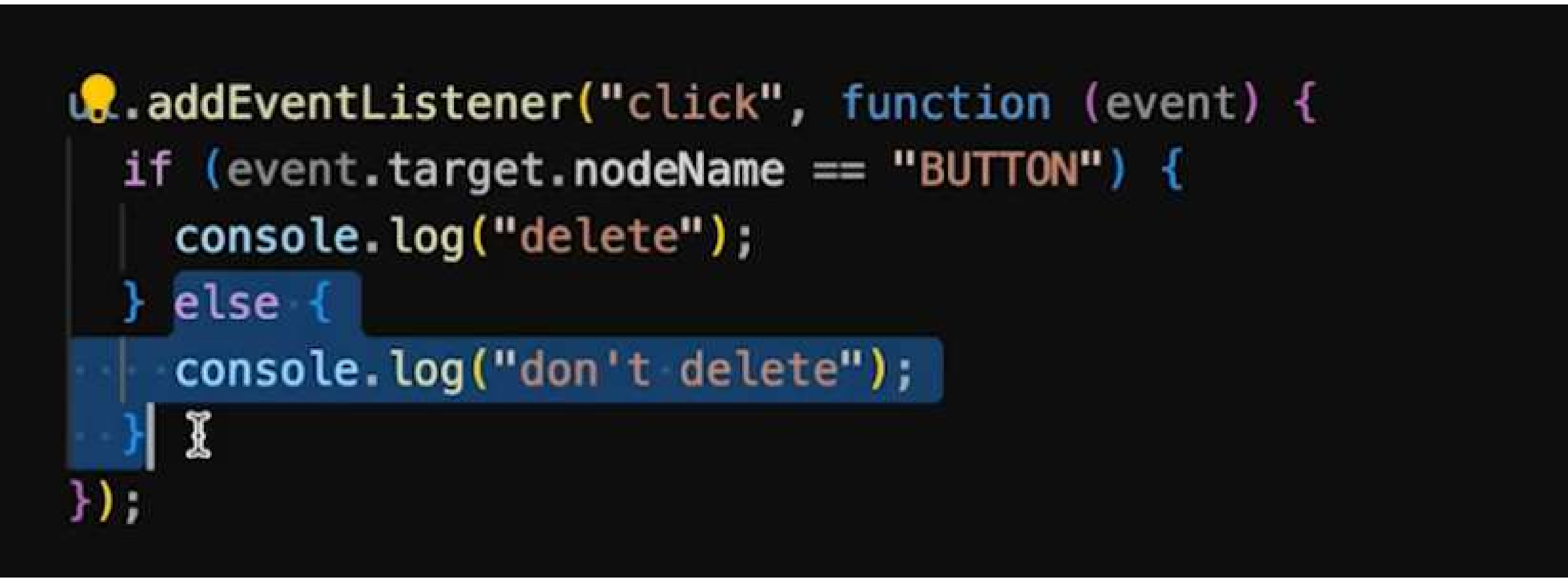
```
  inputMode: ""
  isConnected: true
  isContentEditable: false
▶ labels: NodeList []
  lang: ""
▶ lastChild: text
  lastElementChild: null
  localName: "button"
  name: ""
  namespaceURI: "http://www.w3.o
  nextElementSibling: null
  nextSibling: null
  nodeName: "BUTTON"
  nodeType: 1
  nodeValue: null
  nonce: ""
  offsetHeight: 21
  offsetLeft: 88
```



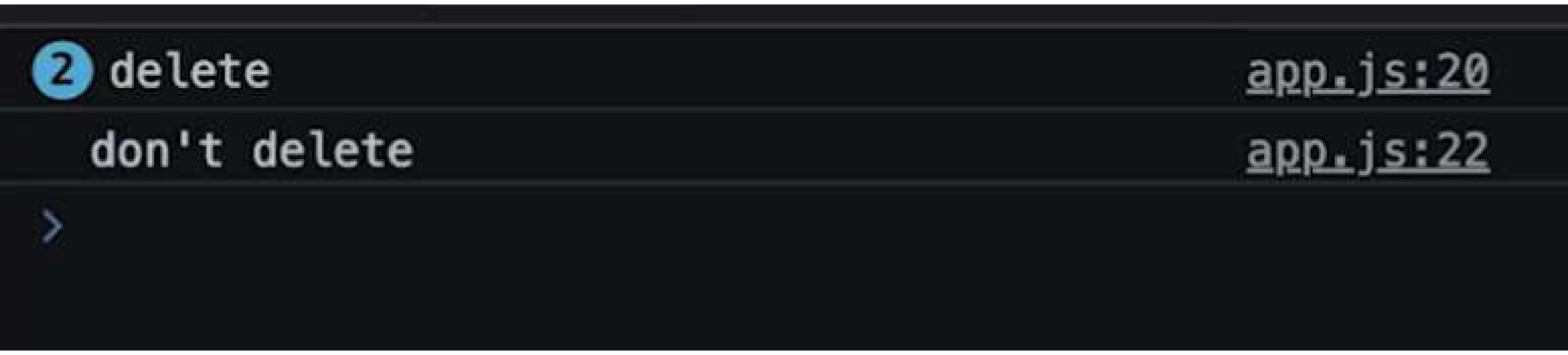
```
ul.addEventListener("click", function (event) {
  console.dir(event.target.nodeName);
});
```



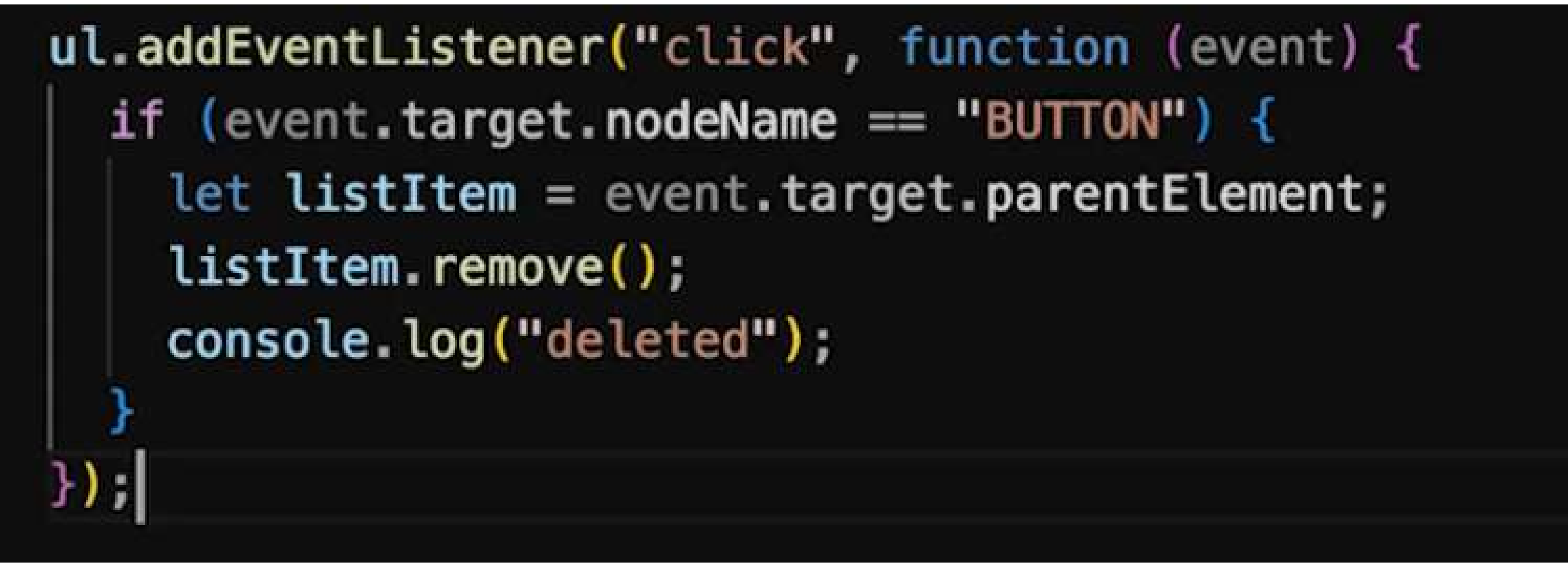
```
4 BUTTON                                      app.js:19
2 LI                                          app.js:19
  BUTTON                                      app.js:19
  LI                                          app.js:19
>
```



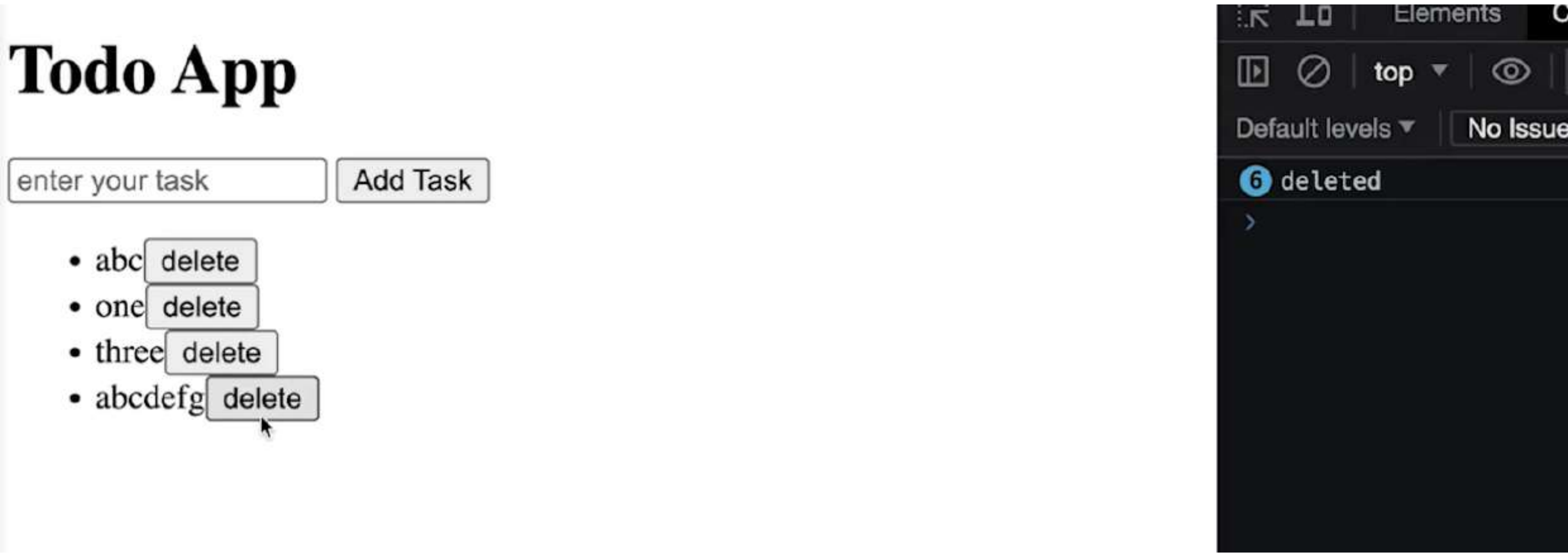
```
ul.addEventListener("click", function (event) {
  if (event.target.nodeName == "BUTTON") {
    console.log("delete");
  } else {
    console.log("don't delete");
  }
});
```



```
2 delete                                   app.js:20
  don't delete                             app.js:22
>
```

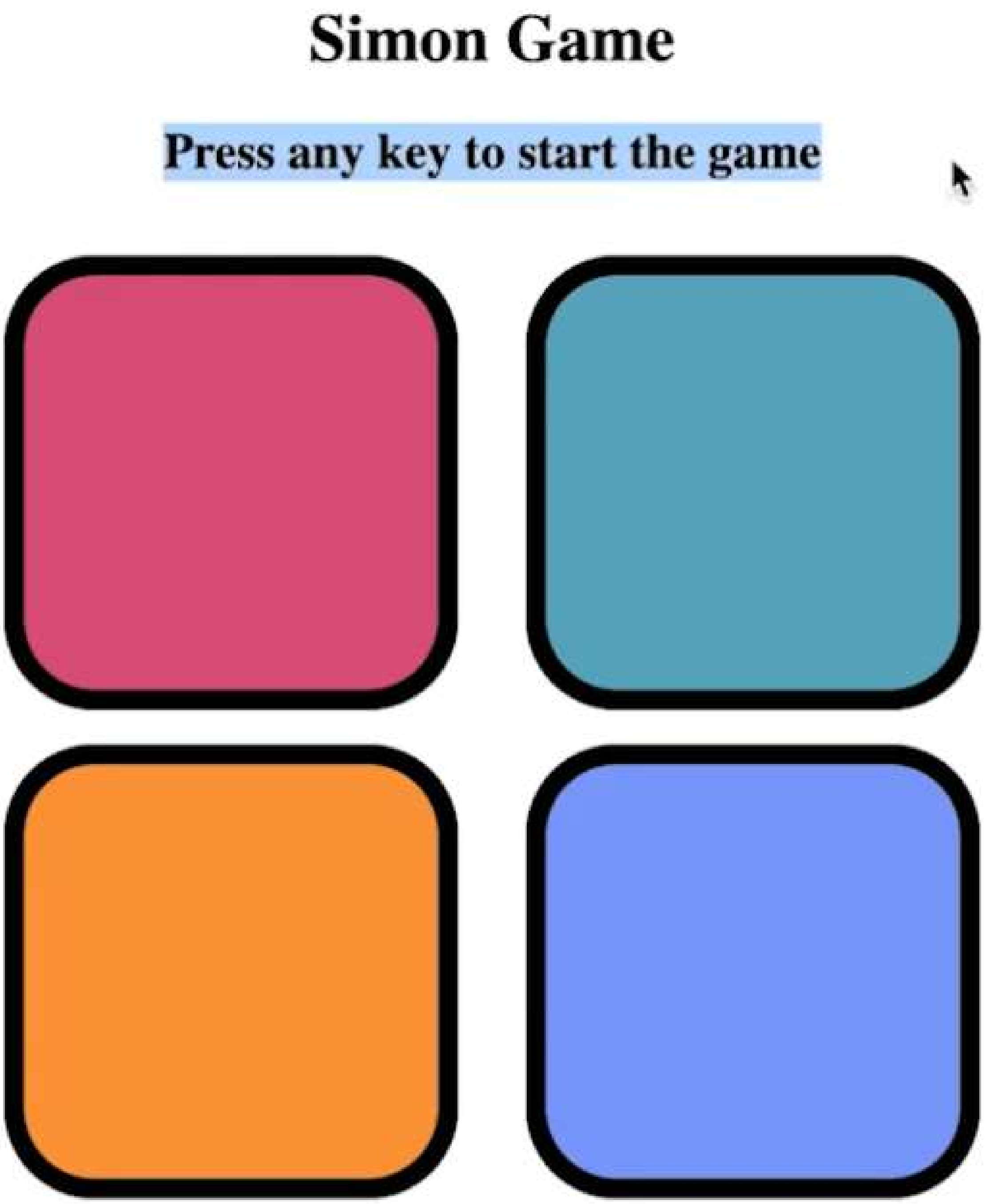


```
ul.addEventListener("click", function (event) {
  if (event.target.nodeName == "BUTTON") {
    let listItem = event.target.parentElement;
    listItem.remove();
    console.log("deleted");
  }
});
```



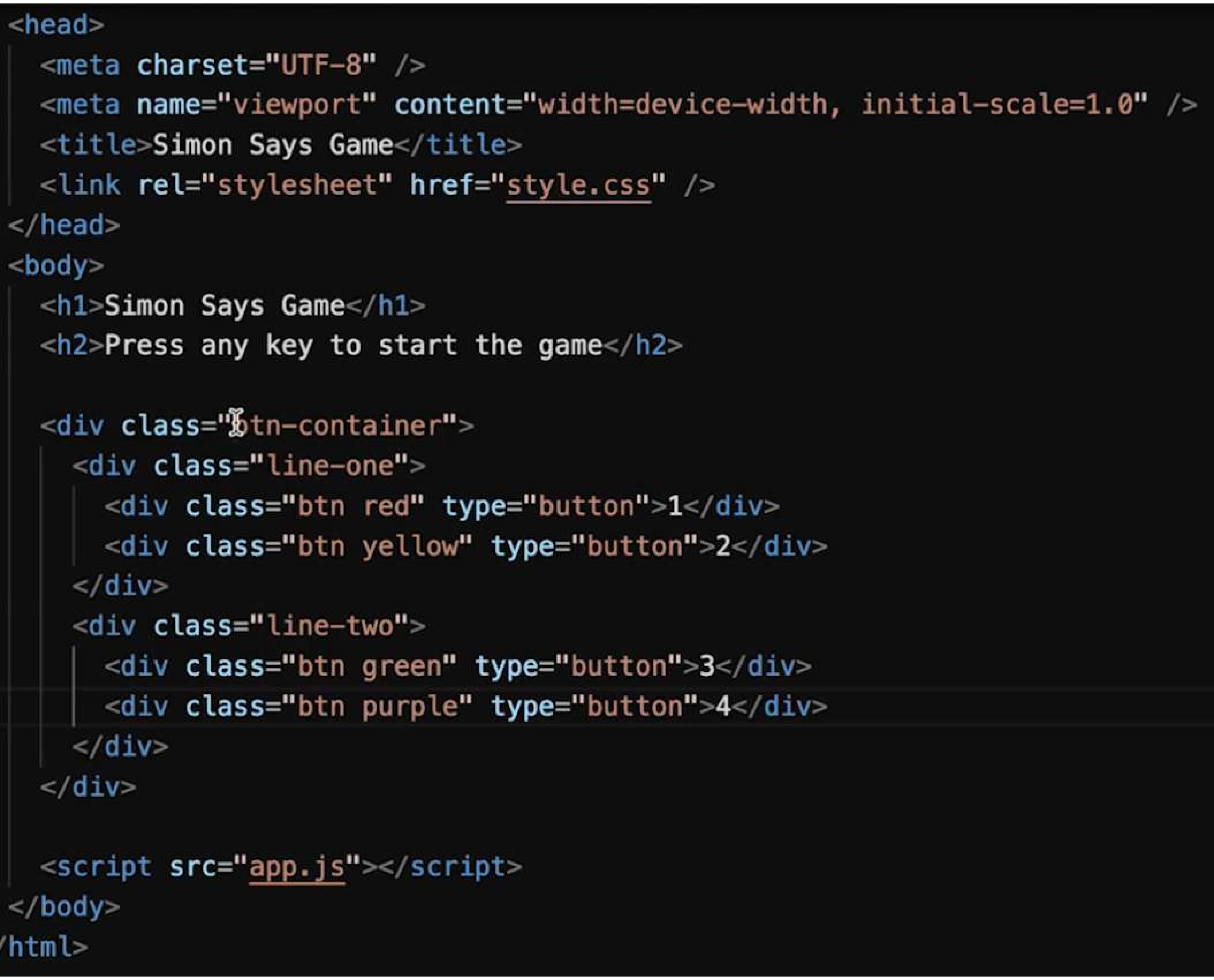
- abc
- one
- three
- abcdefg

```
6 deleted
```

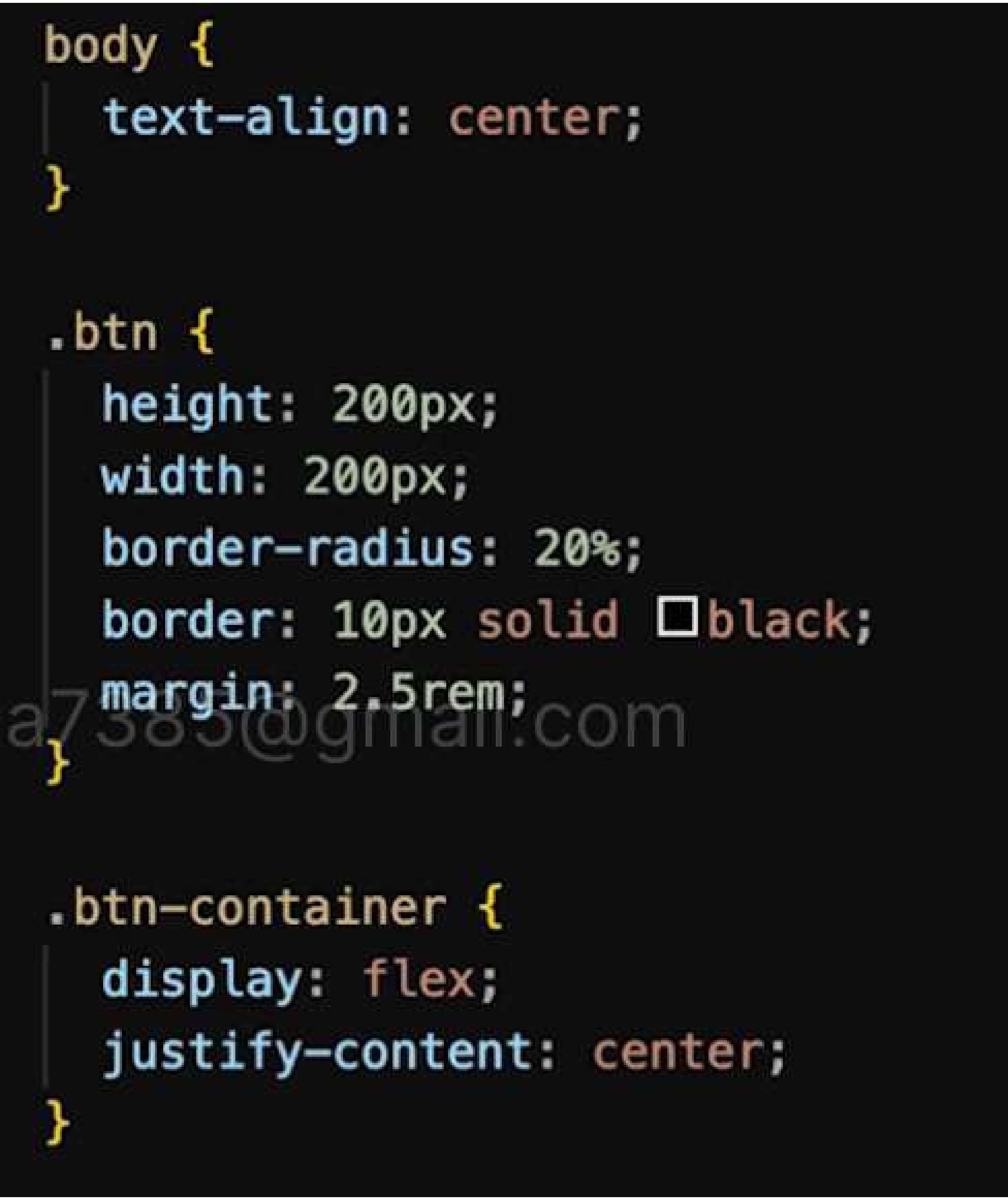


Simon Game

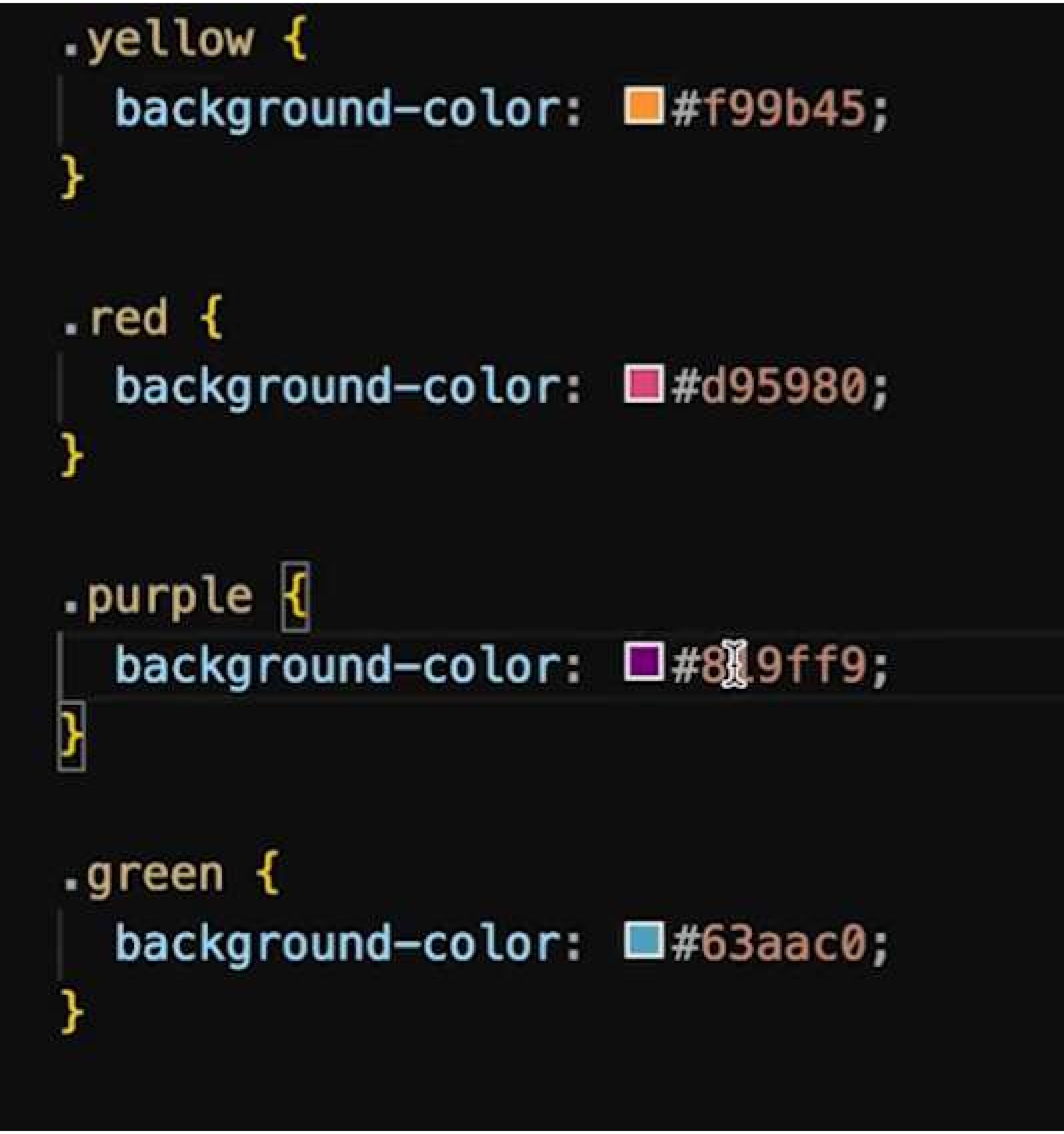
Press any key to start the game



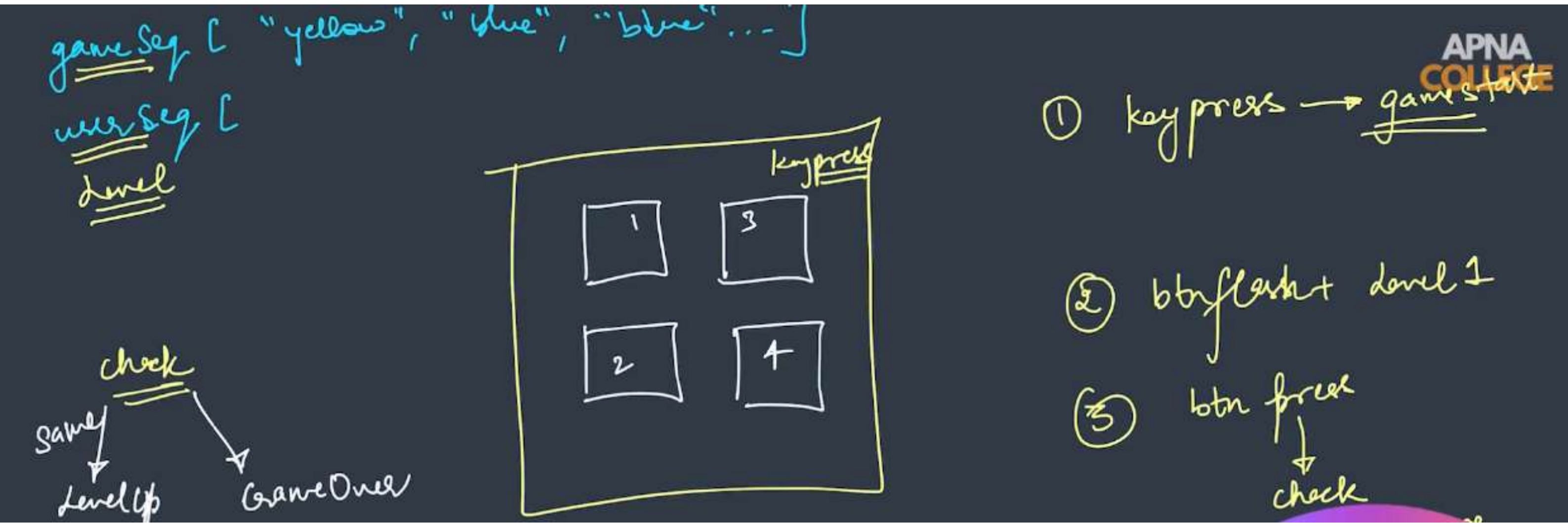
```
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Simon Says Game</title>
  <link rel="stylesheet" href="style.css" />
</head>
<body>
  <h1>Simon Says Game</h1>
  <h2>Press any key to start the game</h2>

  <div class="btn-container">
    <div class="line-one">
      <div class="btn red" type="button">1</div>
      <div class="btn yellow" type="button">2</div>
    </div>
    <div class="line-two">
      <div class="btn green" type="button">3</div>
      <div class="btn purple" type="button">4</div>
    </div>
  </div>

  <script src="app.js"></script>
</body>
/html>
```



```css
body {
  text-align: center;
}

.btn {
  height: 200px;
  width: 200px;
  border-radius: 20%;
  border: 10px solid black;
  margin: 2.5rem;
}

.btn-container {
  display: flex;
  justify-content: center;
}
```



```css
.yellow {
  background-color: #f99b45;
}

.red {
  background-color: #d95980;
}

.purple {
  background-color: #819ff9;
}

.green {
  background-color: #63aac0;
}
```


gameSeq ["yellow", "blue", "blue"...]
userSeq [
level
keypress
1
3
2
4
check
same
LevelUp
GameOver
① keypress → gamestart
② btnflash + level 1
③ btn press
check