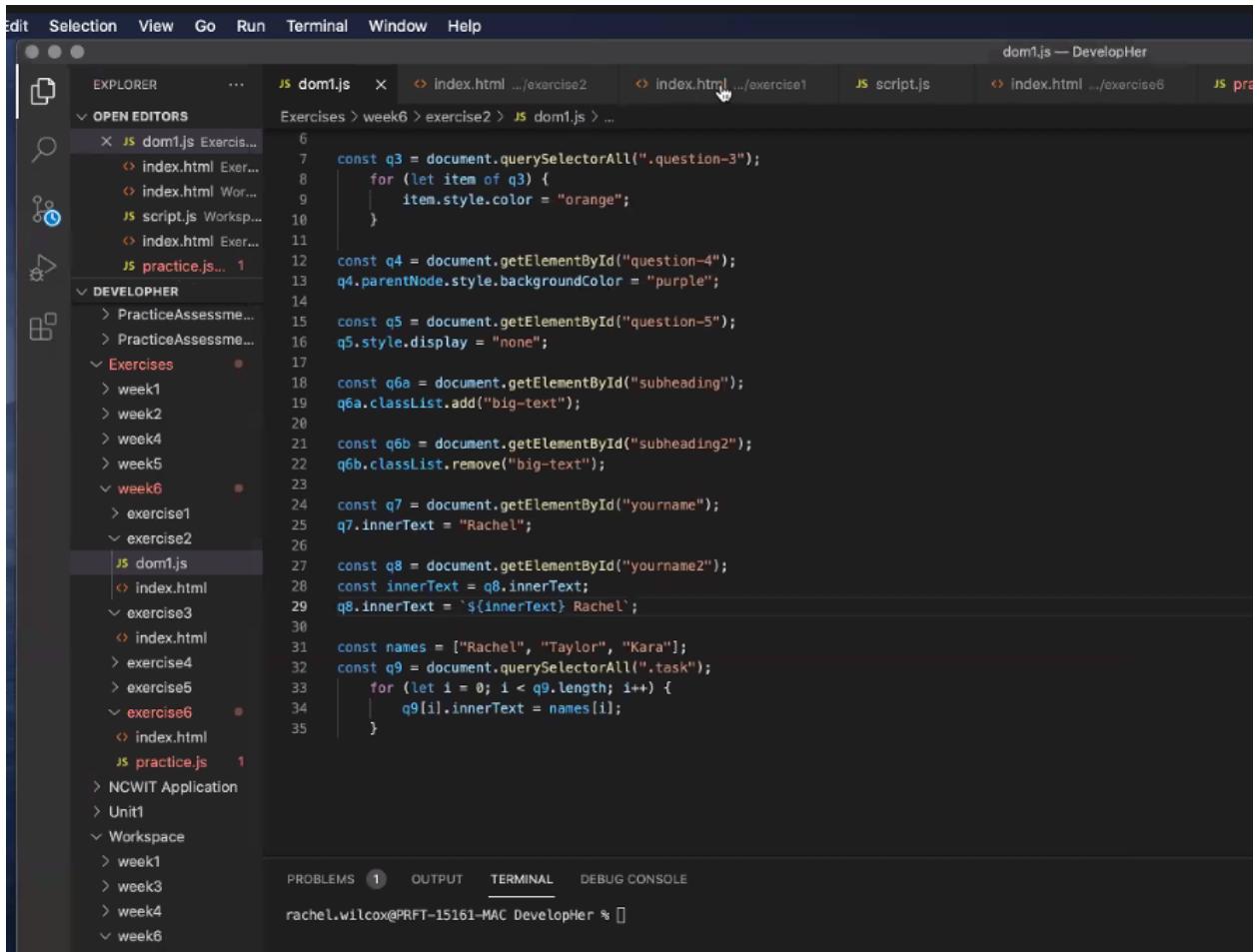


30 July 2021

This is Rachel



The screenshot shows a code editor interface with a dark theme. The top menu bar includes 'Edit', 'Selection', 'View', 'Go', 'Run', 'Terminal', 'Window', and 'Help'. The title bar indicates the file is 'dom1.js — DevelopHer'. The left sidebar has sections for 'EXPLORER', 'OPEN EDITORS', and 'DEVELOPER'. Under 'OPEN EDITORS', there are files like 'dom1.js', 'index.html', 'script.js', and 'practice.js'. Under 'DEVELOPER', there are sections for 'PracticeAssessme...', 'Exercises', and 'week6'. 'week6' contains sub-folders 'exercise1' and 'exercise2', with 'dom1.js' selected in 'exercise2'. The main editor area displays the following JavaScript code:

```
6  const q3 = document.querySelectorAll(".question-3");
7  for (let item of q3) {
8      item.style.color = "orange";
9  }
10
11 const q4 = document.getElementById("question-4");
12 q4.parentNode.style.backgroundColor = "purple";
13
14 const q5 = document.getElementById("question-5");
15 q5.style.display = "none";
16
17 const q6a = document.getElementById("subheading");
18 q6a.classList.add("big-text");
19
20 const q6b = document.getElementById("subheading2");
21 q6b.classList.remove("big-text");
22
23 const q7 = document.getElementById("yourname");
24 q7.innerText = "Rachel";
25
26 const q8 = document.getElementById("yourname2");
27 const innerText = q8.innerText;
28 q8.innerText = `${innerText} Rachel`;
29
30 const names = ["Rachel", "Taylor", "Kara"];
31 const q9 = document.querySelectorAll(".task");
32 for (let i = 0; i < q9.length; i++) {
33     q9[i].innerText = names[i];
34 }
35 }
```

At the bottom, there are tabs for 'PROBLEMS', 'OUTPUT', 'TERMINAL', and 'DEBUG CONSOLE'. The terminal tab shows the command 'rachel.wilcox@PRFT-15161-MAC: ~ %'.

Google javascript string interpolation linke 31,

A screenshot of a code editor showing a file named `practice2.js`. The code contains the following JavaScript:

```
1
2
3  const q1 = document.getElementById('alert-button');
4
5  function showAlert() {
6      alert('Hello!');
7  }
8
9  q1.addEventListener('click', () => {
10 })
11 }
```

Line 9 is an arrow function

A screenshot of a code editor showing a file named `practice2.js`. The code has been modified to include two event listeners:

```
1
2
3  const q1 = document.getElementById('alert-button');
4
5  function showAlert() {
6      alert('Hello!');
7  }
8
9  q1.addEventListener('click', showAlert);
10 q2.addEventListener('click', showAlert);
```

This are from class exercise from 7/28/2021

#6

A screenshot of a code editor window titled "practice2.js - gc_shife_2021_1". The window shows the following code:

```
1
2
3 const q5 = document.getElementById('alert-button');
4
5 q5.addEventListener('click', () => {
6     alert('Hello!')
7 });
8
9
10 const q6 = document.getElementById('add-tv-show-button');
11
12 q6.addEventListener('click', () => {
13     const li = document.createElement('li');
14     li.innerText = 'The West Wing';
15
16     const ul = document.getElementById('tv-shows');
17
18     ul.append(li);
19
20 });
21
22
23 });
24
```

A screenshot of a code editor window showing the continuation of the "practice2.js" file:

```
21     ul.appendChild(li);
22
23 });
24
25
26
27 const q7 = document.getElementById('question-7-remove-button');
28
29 q7.addEventListener('click', () => {
30
31     document.getElementById('comment-7').remove();
32
33 });
34
35
36
37 const q8 = document.getElementById('question-8-remove-button');
38
39 q8.addEventListener('click', () => [
40
41     q8.parentNode.remove();
42
43 ]);
```

A screenshot of a code editor showing a file named `practice2.js`. The code contains several event listeners and some DOM manipulation logic. A context menu is open over the line `q8.addEventListener('click')`, with the option `q8.addEventListener('click') callback` highlighted. The code editor interface includes tabs for other files like `dom2.html`, `practice.html`, `practice.css`, and `mouseover.html`. The code itself is as follows:

```
16     li.innerText = 'The West Wing';
17
18     const ul = document.getElementById('tv-shows');
19
20     ul.appendChild(li);
21
22 });
23
24
25
26
27 const q7 = document.getElementById('question-7-remove-button');
28
29 q7.addEventListener('click', () => {
30
31     document.getElementById('comment-7').remove();
32
33 });
34
35
36
37 const q8 = document.getElementById('question-8-remove-button');
38
39 q8.addEventListener('click', (event) => {
40
41     event.target.parentNode.remove();
42
43 });
```

A screenshot of a code editor window titled "practice2.js — gc_ahfe_2021_1". The menu bar includes "File", "View", "Go", "Run", "Terminal", "Window", and "Help". The left sidebar shows a file tree with various files like "dom2.html", "practice.html", "practice2.js", "practice.css", "mouseover.html", and "mouseover.css". The main editor area contains the following JavaScript code:

```
const q7 = document.getElementById('question-7-remove-button');
q7.addEventListener('click', () => {
  document.getElementById('comment-7').remove();
});

const q8 = document.getElementById('question-8-remove-button');
q8.addEventListener('click', (event) => {
  event.target.parentNode.remove();
});

q8.addEventListener('click', () => {
  q8.parentNode.remove();
});
```

#9

```
});

const q8 = document.getElementById('question-8-remove-button');

function removeElement(event) {
    console.log(event);
}

q8.addEventListener('click', removeElement);
q7.addEventListener('click', removeElement);

function highlight(event) {
    event.target.style.backgroundColor = 'yellow';
    event.target.style.color = 'white';
}

const q9 = document.querySelector('.highlight');

q9.addEventListener('mouseover', highlight);
```

```
practise2.js > ...
9
0  function removeElement(event) {
1      console.log(event);
2  }
3
4
5  q8.addEventListener('click', removeElement);
6  q7.addEventListener('click', removeElement);
7
8
9  function highlight(event) {
0      event.target.style.backgroundColor = 'yellow';
1      event.target.style.color = 'white';
2  }
3
4  function unhighlight(event) {
5      event.target.style.backgroundColor = 'white';
6      event.target.style.color = 'black';
7  }
8
9
0  const q9 = document.querySelector('.highlight');
1
2  q9.addEventListener('mouseover', highlight);
3  q9.addEventListener('mouseleave', unhighlight);
```

#10

```
1 Practice2.js > Q1 forEach() callback
 2
 3
 4
 5     console.log(event);
 6
 7
 8
 9
10     q8.addEventListener('click', removeElement);
11     q7.addEventListener('click', removeElement);
12
13
14
15     function highlight(event) {
16         event.target.style.backgroundColor = 'yellow';
17         event.target.style.color = 'white';
18     }
19
20
21     function unhighlight(event) {
22         event.target.style.backgroundColor = 'white';
23         event.target.style.color = 'black';
24     }
25
26
27
28     const all = document.querySelectorAll('.highlight');
29
30
31     all.forEach(p => {
32         p.addEventListener('mouseover', highlight);
33         p.addEventListener('mouseleave', unhighlight);
34     });
35
36
37
38
```

#11

```
53
54     function unhighlight(event) {
55         event.target.style.backgroundColor = 'white';
56         event.target.style.color = 'black';
57     }
58
59
60     const all = document.querySelectorAll('.highlight');
61
62     all.forEach(p => {
63         p.addEventListener('mouseover', highlight);
64         p.addEventListener('mouseleave', unhighlight);
65     });
66
67
68
69
70     const q12 = document.getElementById('say-hello');
71
72     q12.addEventListener('click', () => {
73
74         const input = document.getElementById('question-12-name');
75         const p = document.getElementById('message');
76
77
78         p.innerText = `Hello ${input.value}`;
79
80     });

```

#13

```
53
54     function unhighlight(event) {
55         event.target.style.backgroundColor = 'white';
56         event.target.style.color = 'black';
57     }
58
59
60     const all = document.querySelectorAll('.highlight');
61
62     all.forEach(p => {
63         p.addEventListener('mouseover', highlight);
64         p.addEventListener('mouseleave', unhighlight);
65     });
66
67
68
69
70     const q12 = document.getElementById('say-hello');
71
72     q12.addEventListener('click', () => {
73
74         const input = document.getElementById('question-12-name');
75         const p = document.getElementById('message');
76
77
78         p.innerText = `Hello ${input.value}`;
79
80     });

```

#14

The screenshot shows a code editor window with the tab bar at the top containing files: dom2.html, practice.html, practice2.js (which is the active file), practice.css, mouseover.html, mouseoverjs.js, and click.js. The main area displays the following JavaScript code:

```
practice2.js - gc_ahfe_2021_1
...
88
89     const input = document.getElementById('question-13-name');
90     const ul = document.getElementById('users');
91
92     const li = document.createElement('li');
93
94     li.innerText = input.value;
95
96     ul.append(li);
97 });
98
99
100    const show = document.getElementById('show-button');
101    const hide = document.getElementById('hide-button');
102
103
104    show.addEventListener('click', () => {
105        document.getElementById('secret-text').style.display = 'none';
106    });
107
108
109    hide.addEventListener('click', () => [
110        document.getElementById('secret-text').style.display = 'none';
111    ]);

```

```
practice2.js > ...
88
89     const input = document.getElementById('question-13-name');
90     const ul = document.getElementById('users');
91
92     const li = document.createElement('li');
93
94     li.innerText = input.value;
95
96     ul.append(li);
97 });
98
99
100 const show = document.getElementById('show-button');
101 const hide = document.getElementById('hide-button');
102
103
104 show.addEventListener('click', () => {
105     document.getElementById('secret-text').style.display = 'block';
106 });
107
108
109 hide.addEventListener('click', () => {
110     document.getElementById('secret-text').style.display = 'none';
111});
```

#15

This one did not work because it concatenates not added them

```
110     secret.style.display = 'none';
111 });
112
113
114 const q15 = document.getElementById('add-calc-button');
115
116 q15.addEventListener('click', () => [
117
118     const x = document.getElementById('question-15-x');
119     const y = document.getElementById('question-15-y');
120
121     const p = document.getElementById('question-15-answer');
122
123     p.innerText = x.value + y.value;
124]);
```

The screenshot shows a code editor window with the tab bar at the top containing "practice2.js - gc_ahtf_2021_1". Below the tabs, there are several status indicators for files: "U" for untracked, "M" for modified, and "U" for unstaged changes. The main code area displays the following JavaScript code:

```
90     ul.appendChild(t1);
91
92     });
93
94     });
95
96     const show = document.getElementById('show-button');
97     const hide = document.getElementById('hide-button');
98     const secret = document.getElementById('secret-text');
99
100    show.addEventListener('click', () => {
101        secret.style.display = 'block';
102    });
103
104    hide.addEventListener('click', () => {
105        secret.style.display = 'none';
106    });
107
108
109    const q15 = document.getElementById('add-calc-button');
110
111    q15.addEventListener('click', () => {
112        const x = document.getElementById('question-15-x');
113        const y = document.getElementById('question-15-y');
114
115        const p = document.getElementById('question-15-answer');
116
117        p.innerText = parseInt(x.value) + parseInt(y.value);
118    });
119
120
121
122
123
124
```

#16

```
js practice2.js > ⚙ q16.addEventListener('click') callback
127
128
129 const q16 = document.getElementById('calc-button');
130
131 q16.addEventListener('click', () => {
132
133     const x = +document.getElementById('question-16-x').value;
134     const y = +document.getElementById('question-16-y').value;
135     const operator = document.getElementById('operator');
136     const p = document.getElementById('question-16-answer');
137
138
139
140     if (operator.value === '+') {
141         p.innerText = x + y;
142     }
143
144     if (operator.value === '-') {
145         p.innerText = x - y;
146     }
147
148     if (operator.value === '*') {
149         p.innerText = x * y;
150     }
151
152     if (operator.value === '/') {
153         p.innerText = x / y;
154     }
155
156 });


```

```
JS practice2.js > q16.addEventListener('click') callback
130
131 q16.addEventListener('click', () => {
132
133     const x = +document.getElementById('question-16-x').value;
134     const y = +document.getElementById('question-16-y').value;
135     const operator = document.getElementById('operator');
136     const p = document.getElementById('question-16-answer');
137
138
139
140     if (operator.value === '+') {
141         p.innerText = x + y;
142     }
143
144     if (operator.value === '-') {
145         p.innerText = x - y;
146     }
147
148     if (operator.value === '*') {
149         p.innerText = x * y;
150     }
151
152     if (operator.value === '/') {
153         p.innerText = x / y;
154     }
155
156 });

I
```

```
practice2.js > q16.addEventListener('click') callback
1   const x = +document.getElementById('question-16-x').value;
2   const y = +document.getElementById('question-16-y').value;
3   const operator = document.getElementById('operator');
4   const p = document.getElementById('question-16-answer');
5
6
7   let answer = 0;
8
9   if (operator.value === '+') {
10     answer = x + y;
11   }
12
13   if (operator.value === '-') {
14     answer = x - y;
15   }
16
17   if (operator.value === '*') {
18     answer = x * y;
19   }
20
21   if (operator.value === '/') {
22     answer = x / y;
23   }
24
25
26 });
27
```

New Material

A screenshot of a code editor showing a tooltip for the `querySelectorAll` method. The tooltip is a box with a dark border containing the following text:

```
querySelectorAll(selectors: keyof  
HTMLInputElementTagNameMap):  
NodeListOf<HTMLObjectElement | HTMLElement |  
HTMLAnchorElement | HTMLAppletElement |  
HTMLAreaElement | ... 65 more ... |  
HTMLVideoElement>
```

The code in the editor is:

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
    <meta charset="UTF-8">  
    <meta http-equiv="X-UA-Compatible" content="IE=edge">  
    <meta name="viewport" content="width=q, initial-scale=1.0">  
<title>Document</title>  
</head>  
<body>  
    <button>Click Me 1</button>  
    <script>  
        document.querySelectorAll()  
    </script>  
</body>  
</html>
```

A screenshot of a code editor showing a tooltip for the `forEach` method on a NodeList. The tooltip is a box with a dark border containing the following text:

```
const buttons = document.querySelectorAll('button');  
buttons.forEach(button => button.addEventListener('click', () => {  
    console.log('CLICKED!!!!');  
}))
```

The code in the editor is:

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
    <meta charset="UTF-8">  
    <meta http-equiv="X-UA-Compatible" content="IE=edge">  
    <meta name="viewport" content="width=q, initial-scale=1.0">  
<title>Document</title>  
</head>  
<body>  
    <button>Click Me 1</button>  
    <script>  
        const buttons = document.querySelectorAll('button');  
        buttons.forEach(button => button.addEventListener('click', () => {  
            console.log('CLICKED!!!!');  
        }))  
    </script>  
</body>  
</html>
```

```
bubbling.html - gc_ahef_2021_1
... dom2.html U practice.html M practice2.js U bubbling.html U ● practice.css mouseover.html mouse

D bubbling.html > html > body > script
U 11  <body>
M 12      <button>Click Me 1</button>
U 13
U 14
U 15
U 16      <script>
U 17          const newButton = document.createElement('button');
U 18          newButton.innerText = 'Click Me 2';
U 19
U 20          document.body.append(newButton);
U 21
U 22
U 23      </script>
M 24
U 25  </body>
U 26
U 27  </html>
```

```
bubbling.html > html > body > script > document.body.addEventListener('click') callback
~meta http-equiv="X-UA-Compatible" content="IE-edge"
7     <meta name="viewport" content="width=q, initial-scale=1.0">
8     <title>Document</title>
9  </head>
10
11  <body>
12
13      <button>Click Me 1</button>
14
15      <div id="something">
16
17      </div>
18
19      <script>
20          document.body.addEventListener('click', (event) => [
21              console.log(event.target);
22          ])
23
24
25
26      const newButton = document.createElement('button');
27      newButton.innerText = 'Click Me 2';
28
29      document.getElementById('something').append(newButton); I
30
31  </script>
32
33  </body>
34
35  </html>
```

```
18
19     <script>
20         document.body.addEventListener('click', (event) => {
21             console.log(event.target);
22         })
23
24
25         for (let i = 0; i < 2000; i++) {
26
27             const newButton = document.createElement('button');
28             newButton.innerText = 'Click Me ' + i;
29             document.body.append(newButton);
30
31         }
32     </script>
33
34 </body>
```

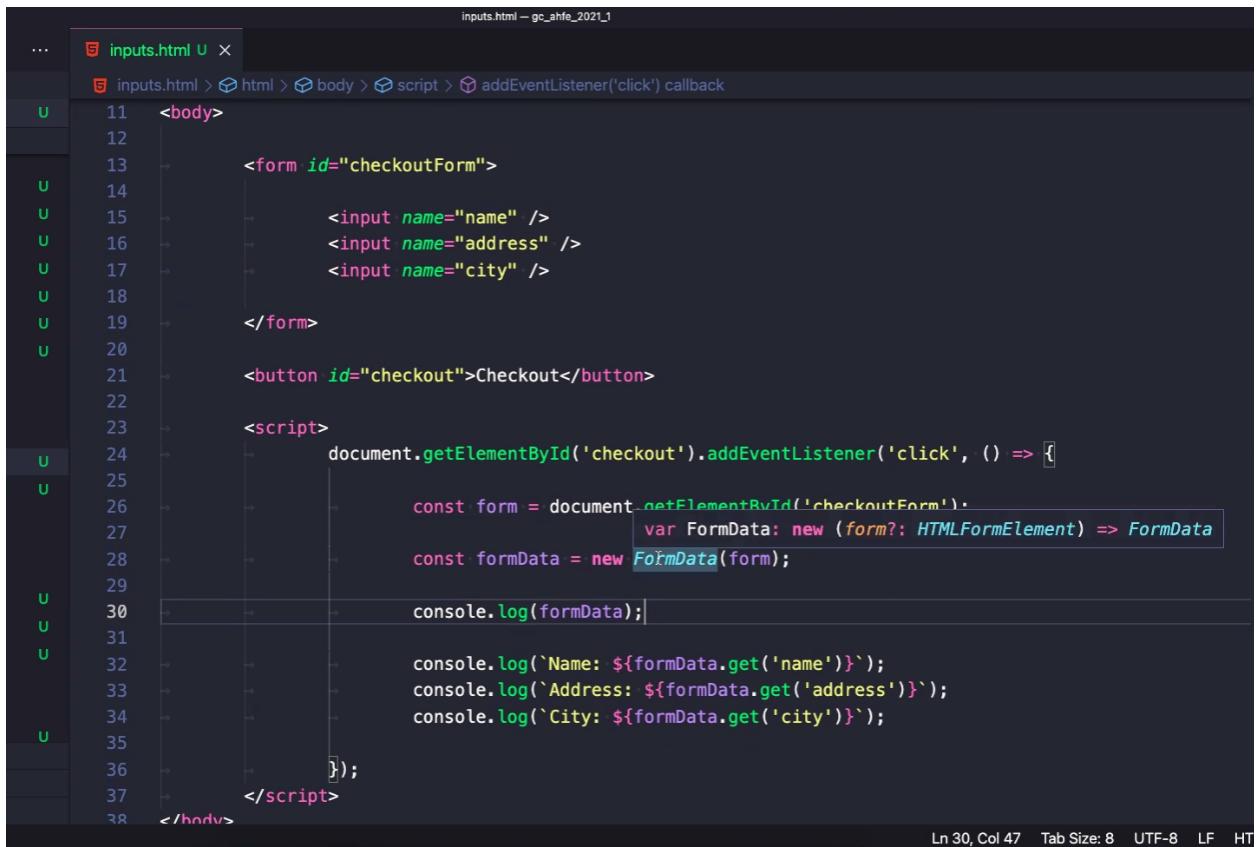
```
8
9     <script>
0         document.body.addEventListener('click', (event) => {
1             event.target.remove();
2         })
3
4
5         for (let i = 0; i < 2000; i++) {
6
7             const newButton = document.createElement('button');
8             newButton.innerText = 'Click Me ' + i;
9             document.body.append(newButton);
0
1         }
2     </script>
3
4 </body>
```

```
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=q, initial-scale=1.0">
8      <title>Document</title>
9  </head>
10
11 <body>
12
13  <ul>
14  <li>Matt</li>
15  <li>Jack</li>
16  <li>Sasha</li>
17 </ul>
18
19  <script>
20  <!--document.querySelector('ul').addEventListener('click', (event) => {
21  -->     event.target.remove();
22  --> });
23  </script>
24 </body>
25
26 </html>
```

Or

```
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=q, initial-scale=1.0">
8      <title>Document</title>
9  </head>
10
11 <body>
12
13  <ul>
14  <li>Matt</li>
15  <li>Jack</li>
16  <li>Sasha</li>
17 </ul>
18
19  <script>
20  <!--document.body.addEventListener('click', (event) => {
21  -->     event.target.remove();
22  --> });
23  </script>
24 </body>
25
26 </html>
```

New Material Inputs & Forms



A screenshot of a code editor showing an HTML file named "inputs.html". The code defines a form with three input fields (name, address, city) and a button labeled "Checkout". A script block adds an event listener to the button that logs the form data to the console. The code uses modern JavaScript features like the FormData object and template literals.

```
inputs.html - gc_ahe_2021_1
...
inputs.html U ×
inputs.html > html > body > script > addEventListener('click') callback
11  <body>
12
13      <form id="checkoutForm">
14
15          <input name="name" />
16          <input name="address" />
17          <input name="city" />
18
19      </form>
20
21      <button id="checkout">Checkout</button>
22
23      <script>
24          document.getElementById('checkout').addEventListener('click', () => [
25
26              const form = document.getElementById('checkoutForm');
27              var FormData: new (form?: HTMLFormElement) => FormData;
28              const formData = new FormData(form);
29
30              console.log(formData);
31
32              console.log(`Name: ${formData.get('name')}`);
33              console.log(`Address: ${formData.get('address')}`);
34              console.log(`City: ${formData.get('city')}`);
35
36          ]);
37
38      </script>
39  </body>
Ln 30, Col 47  Tab Size: 8  UTF-8  LF  HT
```

inputs.html — gc_ahfe_2021_1

inputs.html > html > body > script > addEventListener('click') callback

```
11  <body>
12
13    <form id="checkoutForm">
14
15      <input name="name" />
16      <input name="address" />
17      <input name="city" />
18
19    </form>
20
21    <button id="checkout">Checkout</button>
22
23  <script>
24    document.getElementById('checkout').addEventListener('click', () => {
25
26      const form = document.getElementById('checkoutForm');
27      var FormData: new (form?: HTMLFormElement) => FormData
28      const formData = new FormData(form);
29
30      console.log(formData);
31
32      console.log(`Name: ${formData.get('name')}`);
33      console.log(`Address: ${formData.get('address')}`);
34      console.log(`City: ${formData.get('city')}`);
35
36    });
37
38  </script>
39 </body>
```

Ln 30, Col 47 Tab Size: 8 UTF-8 LF HT

```
<body>

    <form id="checkoutForm">
        <input name="name" />
        <input name="address" />
        <input name="city" />
        <button type="submit" id="checkout">Checkout</button>
    </form>

    <a href="https://stockx.com"></a>

    <script>
        document.getElementById('checkoutForm').addEventListener('submit', (event) => {
            event.preventDefault();
        });
    </script>
</body>

</html>
```

```
10
11 <body>
12
13     <form id="checkoutForm">
14
15         <input name="name" />
16         <input name="address" />
17         <input name="city" />
18
19         <button type="submit" id="checkout">Checkout</button>
20     </form>
21
22     <a href="https://stockx.com">StockX</a>
23
24     <script>
25         document.querySelector('a').addEventListener('click', (event) => {
26             event.preventDefault();
27         });
28     </script>
29 </body>
30
31 </html>
```

