

# Ex 9 - Interactivity in react

## Responding to Events

### Challenge 1 :

#### Challenge 1 of 2: Fix an event handler

Clicking this button is supposed to switch the page background between white and black. However, nothing happens when you click it. Fix the problem. (Don't worry about the logic inside `handleClick` —that part is fine.)

App.js

Download Reset Fork

```
1 export default function LightSwitch() {
2   function handleClick() {
3     let bodyStyle = document.body.style;
4     if (bodyStyle.backgroundColor === 'black') {
5       bodyStyle.backgroundColor = 'white';
6     } else {
7       bodyStyle.backgroundColor = 'black';
8     }
9   }
10
11  return (
12    <button onClick={handleClick}>
13      Toggle the lights
14    </button>
15  );
16 }
17
```

Toggle the lights

^ Show less

### Challenge 2 :

#### Challenge 2 of 2: Wire up the events

This `ColorSwitch` component renders a button. It's supposed to change the page color. Wire it up to the `onChangeColor` event handler prop it receives from the parent so that clicking the button changes the color.

After you do this, notice that clicking the button also increments the page click counter. Your colleague who wrote the parent component insists that `onChangeColor` does not increment any counters. What else might be happening? Fix it so that clicking the button *only* changes the color, and does *not* increment the counter.

ColorSwitch.js

Reset Fork

```
1 export default function ColorSwitch({
2   onChangeColor
3 }) {
4   return (
5     <button onClick={e => {
6       onChangeColor();
7       e.stopPropagation();
8     }}>
9       Change color
10    </button>
11  );
12 }
13
```

Change color

Clicks on the page: 0

## State: A Component's Memory

# Challenge 1 :

## Challenge 1 of 4: Complete the gallery

When you press "Next" on the last sculpture, the code crashes. Fix the logic to prevent the crash. You may do this by adding extra logic to event handler or by disabling the button when the action is not possible.

After fixing the crash, add a "Previous" button that shows the previous sculpture. It shouldn't crash on the first sculpture.

App.js

data.js

src > App.js > Gallery > handleMoreClick

```
4 export default function Gallery() {
5   const [index, setIndex] = useState(0);
6   const [showMore, setShowMore] = useState(false);
7
8   function handleNextClick() {
9     setIndex(index + 1);
10  }
11
12  function handlePreviousClick() {
13    setIndex(index - 1);
14  }
15
16  function handleMoreClick() {
17    setShowMore(!showMore);
18  }
19
20  let sculpture = sculptureList[index];
21  return (
22    <>
23      <button
24        onClick={handleNextClick}
25        disabled={index === sculptureList.length - 1}
26      >
27        Next
28      </button>
29      <button onClick={handlePreviousClick} disabled={index === 0}>
30        Previous
31      </button>
32      <h2>
33        <i>{sculpture.name}</i>
34        by {sculpture.artist}
35      </h2>
36      <h3>
37        ({index + 1} of {sculptureList.length})
38      </h3>
39      <button onClick={handleMoreClick}>
40        {showMore ? "Hide" : "Show"} details
41      </button>
42      {showMore && <p>{sculpture.description}</p>}
43      <img src={sculpture.url} alt={sculpture.alt} />
44    </>
45  );
46 }
```

Preview

https://djtt5x.csb.app

Next

Previous

Homenaje a la Neurocirugía by  
Marta Colvin Andrade

(1 of 12)

Show details

