

Tutorial

[\(/tutorial/\)](#) [Python 3 Basic \(/tutorial/python-3-basic-tutorial/\)](#) [Tkinter \(/tutorial/tkinter-tutorial/\)](#) [Python Modules \(/tutorial/python-modules-tutorial/\)](#)
[JavaScript \(/tutorial/javascript/\)](#) [Python Numpy \(/tutorial/python-numpy/\)](#) [Git \(/tutorial/git/\)](#) [Matplotlib \(/tutorial/matplotlib/\)](#) [PyQt5 \(/tutorial/pyqt5/\)](#)
[Data Structure \(/tutorial/data-structure/\)](#) [Algorithm \(/tutorial/algorithm/\)](#)

HowTo

[\(/howto/\)](#) [Python Scipy \(/howto/python-scipy/\)](#) [Python \(/howto/python/\)](#) [Python Tkinter \(/howto/python-tkinter/\)](#) [Batch \(/howto/batch/\)](#)
[PowerShell \(/howto/powershell/\)](#) [Python Pandas \(/howto/python-pandas/\)](#) [Numpy \(/howto/numpy/\)](#) [Python Flask \(/howto/python-flask/\)](#) [Django \(/howto/django/\)](#)
[Matplotlib \(/howto/matplotlib/\)](#) [Docker \(/howto/docker/\)](#) [Plotly \(/howto/plotly/\)](#) [Seaborn \(/howto/seaborn/\)](#) [Matlab \(/howto/matlab/\)](#) [Linux \(/howto/linux/\)](#)
[Git \(/howto/git/\)](#) [C \(/howto/c/\)](#) [Cpp \(/howto/cpp/\)](#) [HTML \(/howto/html/\)](#) [JavaScript \(/howto/javascript/\)](#) [jQuery \(/howto/jquery/\)](#)
[Python Pygame \(/howto/python-pygame/\)](#) [TensorFlow \(/howto/tensorflow/\)](#) [TypeScript \(/howto/typescript/\)](#) [Angular \(/howto/angular/\)](#) [React \(/howto/react/\)](#)
[CSS \(/howto/css/\)](#) [PHP \(/howto/php/\)](#) [Java \(/howto/java/\)](#) [Go \(/howto/go/\)](#) [Kotlin \(/howto/kotlin/\)](#) [Node.js \(/howto/node.js/\)](#) [Csharp \(/howto/csharp/\)](#)
[Rust \(/howto/rust/\)](#) [Ruby \(/howto/ruby/\)](#) [Arduino \(/howto/arduino/\)](#) [MySQL \(/howto/mysql/\)](#) [MongoDB \(/howto/mongodb/\)](#) [Postgres \(/howto/postgres/\)](#)
[SQLite \(/howto/sqlite/\)](#) [R \(/howto/r/\)](#) [VBA \(/howto/vba/\)](#) [Scala \(/howto/scala/\)](#) [Raspberry Pi \(/howto/raspberry-pi/\)](#)

Reference

[\(/api/\)](#) [Python \(/api/python/\)](#) [Python Pandas \(/api/python-pandas/\)](#) [Numpy \(/api/numpy/\)](#) [Scipy \(/api/scipy/\)](#) [JavaScript \(/api/javascript/\)](#)

(<https://www.ezoic.com/what-is-ezoic/>)
report this ad

setState and prevState in React

🏠 (/) > [HowTo \(/howto/\)](#) > [React Howtos \(/howto/react/\)](#)


> [setState and prevState in React \(/howto/react/react-setstate-prevstate/\)](#)

✍️ [Rana Hasnain Khan](#) 🕒 Dec 21, 2022 📅 Dec 30, 2021

🔗 [React \(/tags/react/\)](#) [React setState \(/tags/react-setstate/\)](#) [React prevState \(/tags/react-prevstate/\)](#) 

([http://www.facebook.com/sharer.php?](http://www.facebook.com/sharer.php?src=bm&u=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&t=setState%20and%20prevState%20in%20React)

[src=bm&u=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-](http://www.facebook.com/sharer.php?src=bm&u=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&t=setState%20and%20prevState%20in%20React)

[prevState%2f&t=setState%20and%20prevState%20in%20React](http://www.facebook.com/sharer.php?src=bm&u=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&t=setState%20and%20prevState%20in%20React))  ([https://twitter.com/intent/tweet/?](https://twitter.com/intent/tweet/?text=React%20Howtos-setState%20and%20prevState%20in%20React&url=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&hashtags=web,development)

[text=React%20Howtos -](https://twitter.com/intent/tweet/?text=React%20Howtos-setState%20and%20prevState%20in%20React&url=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&hashtags=web,development)

[setState%20and%20prevState%20in%20React&url=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-](https://twitter.com/intent/tweet/?text=React%20Howtos-setState%20and%20prevState%20in%20React&url=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&hashtags=web,development)

[setstate-prevstate%2f&hashtags=web,development](https://twitter.com/intent/tweet/?text=React%20Howtos-setState%20and%20prevState%20in%20React&url=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&hashtags=web,development))  ([https://www.linkedin.com/shareArticle?](https://www.linkedin.com/shareArticle?mini=true&url=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&title=React%20Howtos-setState%20and%20prevState%20in%20React&source=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&summary=Short%20summary)

[mini=true&url=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-](https://www.linkedin.com/shareArticle?mini=true&url=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&title=React%20Howtos-setState%20and%20prevState%20in%20React&source=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&summary=Short%20summary)

[prevState%2f&title=React%20Howtos -](https://www.linkedin.com/shareArticle?mini=true&url=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&title=React%20Howtos-setState%20and%20prevState%20in%20React&source=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&summary=Short%20summary)

[setState%20and%20prevState%20in%20React&source=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2f](https://www.linkedin.com/shareArticle?mini=true&url=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&title=React%20Howtos-setState%20and%20prevState%20in%20React&source=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&summary=Short%20summary)

[setstate-prevstate%2f&summary=Short%20summary](https://www.linkedin.com/shareArticle?mini=true&url=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&title=React%20Howtos-setState%20and%20prevState%20in%20React&source=https%3a%2f%2fwww.delftstack.com%2fhowto%2freact%2freact-setstate-prevstate%2f&summary=Short%20summary))



We will introduce `setState` and `prevState` and use them in React.



How To Print Integer In C With Source Code Free Download 2020 | C Projects with Sou...

🔗 `setState` and `prevState` in React

`setState()` and `prevState()` are `useState` hooks that are used to change state in a React class component.

`setState()` indicates that this component and its children components are changed and need to be re-rendered with the updated state. `setState` is the primary method used to update the UI in response to event handlers and server responses.

`prevState()` is the same as the `setState` but the only difference between them is that if we want to change the state of a component based on the previous state of that component, we use `this.setState()`, which provides us the `prevState`.

Let's check an example of a counter app. That can increment, decrement, reset numbers.

First of all, we will create a new file, `Counter.js`. In `Counter.js` we will import React from `"react"`; and create a function called `Counter()`.

In `Counter()` we will initialize our constants `initialNum`, `num`, and `setNum`

```
# react
import React from "react";

function Counter() {
  const initialNum = 0;
  const [num, setNum] = React.useState(initialNum);
}
```

We will return the HTML with buttons and a counter display. So, `Counter.js` will look like below.

```
# react
import React from "react";

function Counter() {
  const initialNum = 0;
  const [num, setNum] = React.useState(initialNum);

};

return (
  <div>
    <p>Counter: {num}</p>
    <button onClick={() => setNum(initialNum)}>Reset</button>
    <button onClick={() => setNum(num + 1)}>Increment</button>
    <button onClick={() => setNum(num - 1)}>Decrement</button>
  </div>
);
}
export default Counter;
```

Let's include `Counter.js` in `App.js` and add its component. `App.js` will look like below.

```
# react
import "../styles.css";
import Counter from "../Counter.js";

export default function App() {
  return (
    <div className="App">
      <Counter />
    </div>
  );
}
```

Now, let's test our counter.

Output:

Counter: 0

Let's add another button that will increment the counter by five.

```
# react
<button onClick={IncrementByFive}>Increment By 5</button>
```

Let's create a function `IncrementByFive`.

```
# react
const IncrementByFive = () => {
  for (let i = 0; i < 5; i++) {
    setNum(num + 1);
  }
};
```

This function will loop for five times to increment it by 5.

Now, let's test our counter app.

Output:

Counter: 0

When we click on the `Increment by 5` button, it only increments the counter by 1.

It is because in `setNum(num + 1);` there are multiple `setNum` calls, so React grouped all these calls and updates the state only in the last call of `setNum`, and because in the last call `setNum` is not yet updated and still have the same value, so it only increments it by 1.

That's when `prevState` comes in handy and solves our problems like a champ. Now, let's rewrite our counter app using `prevState`.

Basically, instead of passing in a value of the new state variable, we pass in a function that has access to the old state value. So, now we will change `IncrementByFive` function by changing `setNum(num + 1);` to `setNum(prevNum => prevNum + 1);`.

So, our function will look like below.


```
# react
const IncrementByFive = () => {
  for (let i = 0; i < 5; i++) {
    setNum(prevNum => prevNum + 1);
  }
};
```

Now, let's test our counter app.

Output:

Counter: 0

Reset Increment Decrement Increment By 5

In the above result, when we click on the `Increment by 5`, it increments it by 5.

Let's correct our other buttons as well using `prevState`.

`Counter.js` will look like below.

```
# react
import React from "react";

function Counter() {
  const initialNum = 0;
  const [num, setNum] = React.useState(initialNum);

  const IncrementByFive = () => {
    for (let i = 0; i < 5; i++) {
      setNum((prevNum) => prevNum + 1);
    }
  };

  return (
    <div>
      <p>Counter: {num}</p>
      <button onClick={() => setNum(initialNum)}>Reset</button>
      <button onClick={() => setNum((prevNum) => prevNum + 1)}>
        Increment
      </button>
      <button onClick={() => setNum((prevNum) => prevNum - 1)}>
        Decrement
      </button>
      <button onClick={IncrementByFive}>Increment By 5</button>
    </div>
  );
}

export default Counter;
```

So, in this guide, we learned about the issues which can be resolved using `prevState`, and we also built a basic counter app in React using `setState` and `prevState`.

Author: **Rana Hasnain Khan** (/author/rana-hasnain-khan/)

(/author/rana-hasnain-khan/)

(/author/rana-hasnain-khan/)

Rana is a computer science graduate passionate about helping people to build and diagnose scalable web application problems and problems developers face across the full-stack.

 LinkedIn (<https://www.linkedin.com/in/rana-hasnain-3059a2165/>)

(<https://www.ezoic.com/what-is-ezoic/>)

[report this ad](#)

