

ReactJS Basics: A Beginner's Guide

If you're new to ReactJS, this is the perfect place to start. In this guide, we'll cover the basics of ReactJS, including what it is, how to use JSX, and the importance of the Virtual DOM. We'll also dive into props, passing props to child components, components rendering, and filter map in React. Let's get started!



by shubham ss

The Fundamentals of ReactJS

JSX

In ReactJS, you use a syntax called JSX to describe what you want your UI to look like. JSX looks like HTML, but it allows you to use JavaScript expressions to create dynamic content.

Virtual DOM

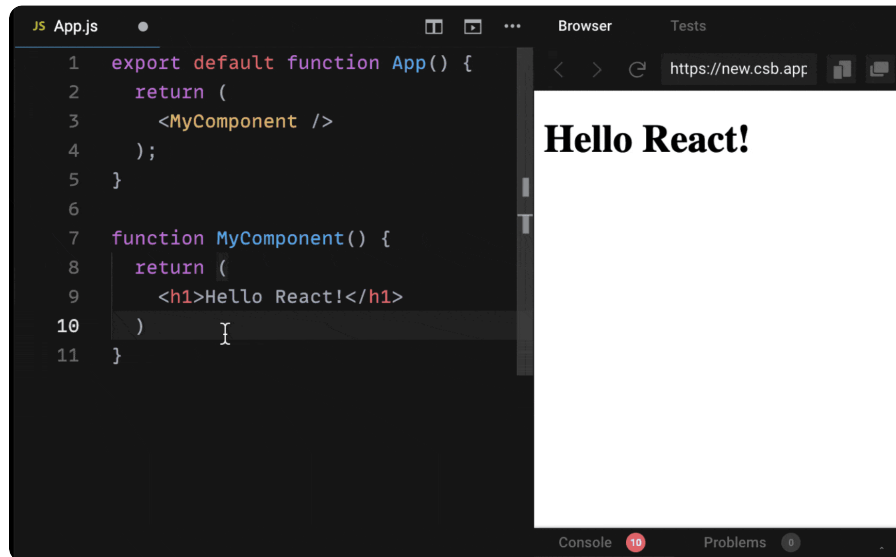
The Virtual DOM is a lightweight copy of the real DOM that ReactJS uses to keep track of changes. This makes it easier and faster for ReactJS to update your UI, resulting in a better user experience.

Components

ReactJS is built around components, which are reusable pieces of code that represent parts of your UI. You can think of a component as a function that takes some input (props) and returns some output (the UI).

Passing Props to Child Components

One of the most powerful features of ReactJS is the ability to pass props from a parent component to a child component. This allows you to create complex UIs that are easy to manage and maintain.



```
1 export default function App() {  
2   return (  
3     <MyComponent />  
4   );  
5 }  
6  
7 function MyComponent() {  
8   return (  
9     <h1>Hello React!</h1>  
10  )  
11 }
```

The screenshot shows a code editor with a dark theme. The left pane displays the code for App.js, which exports a default function App() that returns a JSX element containing a MyComponent. The right pane shows the rendered output of the application, which is a white box with the text "Hello React!" in a large, bold, black font. The browser's address bar shows the URL https://new.csb.app. The bottom of the editor shows tabs for Console, Problems, and Tests.

Passing props from parent to child component is as simple as passing an object as a parameter.

Components Rendering

| Component | Description |
|----------------------|---|
| Functional Component | A simple JavaScript function that returns some JSX. They are easier to read and test than class components. |
| Class Component | A more powerful component class that you can use to implement advanced features. |
| Hooks | A way to create stateful functional components that can handle lifecycle events. |

Using Filter Map in React

Filter

The `filter()` method allows you to create a new array that contains only the elements that meet certain criteria. For example, you can use `filter()` to create a list of items that match a certain category.

Map

The `map()` method allows you to create a new array by applying a function to each element of an existing array. This is useful when you want to transform your data into a new format.

The Power of Props

Props are a powerful way to pass data between components in ReactJS. They allow you to create complex UIs that are easy to maintain and customize. With props, you can create reusable components that can be used in many different places throughout your app.

1 Props vs. State

Props are immutable, while state is mutable. This means that props cannot be modified by the component that receives them, while state can be modified by the component itself.

2 Default Props

You can set default values for your props using the `defaultProps` property. This is useful when you want to provide a default value for a prop in case it's not passed to the component.