

1.

Get started with React 1100 XP

31 min • Module • 10 Units

★★★★★ 4.8 (78)

React.js is the most popular front-end JavaScript framework. Developers use JSX, a combination of HTML and JavaScript, to create views in a natural way. Developers can also create components for blocks that can be reused across their applications. This module introduces React and the core skills developers need to use this powerful framework.

[Overview](#) ▾

[Save](#)

Working with data and properties in React components 1000 XP

26 min • Module • 9 Units

★★★★★ 4.6 (55)

The goal of React components is to create reusable blocks of code and UI. To support reusability, you can pass data into components through the use of properties. We will see how to add properties to a component, and work with complex data types such as arrays.

[Overview](#) ▾

[Save](#)

2. Realizar una comparación de los siguientes códigos.

[learning-react-star-rating-2 - CodeSandbox](#)

[learning-react-star-rating-3 - CodeSandbox](#)

a. ¿Cuál es el código que utiliza props? Colocar el link.

El código que usa props es [learning-react-star-rating-2 - CodeSandbox](#), aunque en el archivo StartRating.js se tiene una variable que asigna 5 estrellas en el archivo App.js se asignan 10 y ese valor sobrescribe al que este asignado en StartRating.js

b. Anota las diferencias que puedas notar.

- En rating-2 en el archivo App.js se sobrescribe la variable totalStars.
- En rating-3 en el archivo StartRating.js al importar React se importa un objeto llamado useState.
- En rating-3 se crea un arreglo llamado selectedStars.
- En rating-3 se usa una etiqueta vacía para guardar todo el componente.
- En rating-3 se crea un contador de las estrellas seleccionadas.

3.

React

React is a popular JavaScript library for building reusable, component-driven user interfaces for web pages or applications.

React combines HTML with JavaScript functionality into its own markup language called JSX. React also makes it easy to manage the flow of data throughout the application.

In this course, you'll learn how to create different React components, manage data in the form of state props, use different lifecycle methods like `componentDidMount`, and much more.

▼ Collapse courses

○ 20/47

- ✓ Create a Simple JSX Element
- ✓ Create a Complex JSX Element
- ✓ Add Comments in JSX
- ✓ Render HTML Elements to the DOM
- ✓ Define an HTML Class in JSX
- ✓ Learn About Self-Closing JSX Tags
- ✓ Create a Stateless Functional Component
- ✓ Create a React Component
- ✓ Create a Component with Composition
- ✓ Use React to Render Nested Components
- ✓ Compose React Components