

Functional Components	Class Components
A functional component is just a plain JavaScript pure function that accepts props as an argument and returns a React element(JSX).	A class component requires you to extend from React. Component and create a render function which returns a React element.
There is no render method used in functional components.	It must have the render() method returning JSX (which is syntactically similar to HTML)
Functional components run from top to bottom and once the function is returned it can't be kept alive.	Class component is instantiated and a different life cycle method is kept alive and being run and invoked depending on the phase of the class component.
Also known as Stateless components as they simply accept data and display them in some form, they are mainly responsible for rendering UI.	Also known as Stateful components because they implement logic and state.
React lifecycle methods (for example, componentDidMount) cannot be used in functional components.	React lifecycle methods can be used inside class components (for example, componentDidMount).
Hooks can be easily used in functional components to make them Stateful.	It requires different syntax inside a class component to implement hooks.
example: <code>const [name,SetName]= React.useState('')</code>	example: <pre>constructor(props) { super(props); this.state = {name: ''} }</pre>