

## Tell us the differences between uncontrolled and controlled components.

Controlled components are React components that render a form input element and manage the state of that input. They change the value of the input through an `onChange` event handler, which updates the state with the new value. **On the other hand,** uncontrolled components are React components that render a form input element but don't manage the state of that input element. Instead, they rely on the browser to manage the state internally using the DOM.

## How to validate React props using PropTypes?

React provides a package called `prop-types` that allows you to define the expected types and values of the props. This helps to catch errors and improve the stability and maintainability of your codebase.

## Tell us the difference between nodejs and express js.

Node.js is a JavaScript runtime that allows developers to write server-side code using JavaScript. It's built on top of Chrome's V8 JavaScript engine and is used by Google Chrome. Node.js provides an event-driven, non-blocking I/O model that makes it suitable for building data-intensive real-time applications. Node.js comes with built-in modules that allow developers to handle various tasks such as file system operations, network requests, and more. **on the other hand, Express.js,** is a web application framework for Node.js. It provides a set of features that simplify building web applications and APIs using Node.js. Express.js builds on top of Node.js and adds more functionality, such as middleware, routing for mapping URLs to controllers, and template engines for rendering dynamic views.

## What is a custom hook, and why will you create a custom hook?

A custom hook in React is a function that encapsulates reusable logic that can be shared between different components. Custom hooks allow you to extract common functionality from your components and share it across your application. Custom hooks are a powerful tool for writing cleaner, more reusable code. Custom hooks can make your component code more readable and concise by abstracting away implementation details.