

**Question :** Differences between uncontrolled and controlled components.

**Uncontrolled components:** The component's state is managed internally by the DOM, not by React. You can get the value of the input fields using DOM API such as `document.getElementById` or `ref`. Changes are not immediately recorded or reflected in the component's state, so you will need to add a callback function to handle the value change.

**Controlled components:** The component's state is managed by React, and its current value is stored in the state. You will need to use the `onChange` event to handle the value change and update the state.

**Question :** How to validate React props using PropTypes

PropTypes is a package that allows you to validate the props that are passed to a component. It provides a way to define the type of props that a component expects, ensuring that they are of the correct type and preventing errors. To use PropTypes, you need to import it from the `prop-types` package: `import PropTypes from 'prop-types';`

**Question :** Tell us the difference between nodejs and express js.

**Node.js** is a JavaScript runtime built on the V8 engine. It allows you to run JavaScript code outside of a web browser and can be used for both client-side and server-side development.

**Express.js** is a lightweight web framework for Node.js. It provides a set of features to build web applications, such as routing, middleware, and templating. Express.js is built on top of Node.js and uses its features to provide a more developer-friendly way to build web applications.

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

A custom hook is a function that allows you to reuse stateful logic across multiple components. Custom hooks are just plain JavaScript functions that use React hooks. You might want to create a custom hook if you find yourself repeating the same logic in multiple components. For example, if you have several components that fetch data from an API and display the result, you can create a custom hook that encapsulates the API call and returns the data. This way, you can reuse the logic in multiple components without duplicating code.