1. Tell us the differences between uncontrolled and control

In React, a controlled component is a form element whose value is controlled by React through its state. This mea
component's state. When the user interacts with the component, React updates the state and re-renders the com
to the component as a prop, and the component updates its value accordingly. Controlled components are typic
over the user's input. On the other hand, an uncontrolled component is a form element whose value is not contro
words, the value of the component is not stored in the component's state, but rather in the DOM. When the user
in the DOM, and the component is not re-rendered. Uncontrolled components are typically used in simple forms
user's input.

2. How to validate React props using PropTypes?

React provides a library called prop-types which allows you to define the types of props that a component should
the correct types of data are being passed down to the component. Here's an example of how to use prop-types
1. First, install the prop-types package by running the following command: npm install prop-types
2. Import PropTypes at the top of your component file: javascript
import PropTypes from 'prop-types';
3. Add a propTypes object to your component, defining the type for each prop:

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

Node.js is a JavaScript runtime built on the V8 JavaScript engine that allows developers to run JavaScript code ou
for building scalable network applications. Node.js includes features such as a non-blocking I/O model, event-dri
enables developers to write server-side applications in JavaScript.
Express.js is a web application framework for Node.js. It provides a set of features for building web applications s
Express.js simplifies the process of building web applications by providing a higher-level API on top of Node.js's c
applications using an intuitive and flexible API.
In summary, Node.js is a JavaScript runtime that provides a runtime environment for building server-side applicat
simplifies the process of building web applications using Node.js. Express.js is built on top of Node.js and provide

4. What is a custom hook, and why will you create a custom

In React, a custom hook is a reusable function that allows you to share logic between components. Custom hook
other hooks, and are prefixed with the word use.
Custom hooks can be used to abstract away repetitive logic in your components, making them more readable an
between components that do not share a common parent, which can be difficult to achieve with other forms of s
Custom hooks can be used to encapsulate complex behavior such as handling browser events, fetching data from
custom hook, you can extract this logic from your components, making them more focused on their specific resp
For example, you may create a custom hook to handle form validation logic that can be reused across multiple fo
of repetitive code in your components and improve the overall organization of your codebase.