

## **The differences between uncontrolled and controlled components**

Controlled Components: In a controlled component, the state of the input field is controlled by React. This means that the value of the input field is derived from the component's state, and changes to the value of the input field are handled by updating the state. In other words, the component's state "controls" the value of the input field.

In an uncontrolled component, the state of the input field is managed by the DOM. This means that the value of the input field is derived from the HTML DOM, and changes to the value of the input field are handled by the DOM. In other words, the DOM "controls" the value of the input field.

## **How to validate React props using PropTypes ?**

PropTypes is a built-in validation feature in React that can be used to validate the types of props passed to a component. It is an optional feature and is not required for a React application to function properly, but it can help catch bugs and improve the reliability of our code.

## **The difference between nodejs and express js?**

Node.js is an open-source, cross-platform, server-side JavaScript runtime environment that allows developers to run JavaScript on the server-side, outside of the browser. Express.js, on the other hand, is a popular open-source web application framework built on top of Node.js. Express.js provides a set of features that simplifies the process of building web applications and APIs, such as routing, middleware, template engines, and static file serving.

Here are some key differences between Node.js and Express.js:

Node.js is a runtime environment for executing JavaScript code on the server-side, while Express.js is a web application framework built on top of Node.js.

Node.js provides a set of built-in modules that can be used to build server-side applications, while Express.js provides additional features on top of these modules, such as routing, middleware, and template engines.

Node.js is low-level and requires developers to write more code to build applications, while Express.js provides a higher-level abstraction that makes it easier to build web applications and APIs. Node.js is flexible and can be used to build various types of applications, while Express.js is primarily focused on building web applications and APIs.

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

Custom Hooks are a mechanism to reuse stateful logic (such as setting up a subscription and remembering the current value), but every time you use a custom Hook, all state and effects inside of it are fully isolated. Instead of using High-Order Components (HOCs), one can use custom hooks to improve the readability of the code. Complex codes can become hard to read if layers of providers surround the components, consumers, HOCs, render props, and other abstractions, generally known as wrapper hell.