












## Item Navigation

# Additional resources

Here is a list of resources in the form of additional reading materials that will help you gain more insight into project functionality related to the table booking system of your app project.

- The article on [destructuring assignment](#)  syntax allows you to get values out of the array that are returned when the **useState** hook is invoked. This resource describes how the destructuring assignment works in more detail.
- The [Read props inside the child component](#)  link on the Beta version of React docs discusses how to use destructuring assignment to get values out of the props object.
- The [useState reference](#)  on the official React docs website helps you to understand how to work with this hook and some of the caveats involved.
- The [useEffect reference](#)  on the official React docs website helps you to understand the syntax of this hook and goes into some depth to explain how to use and troubleshoot the **useEffect** hook.
- The [rules of hooks](#)  reading on the reactjs.org website gives you an overview of how to work with the hooks as recommended by the React Core team at Meta.
- The [useReducer hook](#)  reference on the reactjs.org website discusses the basics of **useReducer**, along with specifying initial state and lazy initialization.
- The [syntax and usage of the useReducer hook](#)  reading on the reactjs.org website complements the previous additional resource well.
- The [using the useRef hook](#)  reference on the reactjs.org's docs website is a great example of various options that are available when working with this hook.
- The [reusing logic with custom hooks](#)  on reactjs.org discusses the dynamics of custom hooks and goes to some length in giving a few practical examples to complement the theory behind them.

