

React.js Learning Objectives



At the end of each week, each student should be able to demonstrate that their React project has met the following criteria.

Lesson Name	Topics Covered	Students will be able to...
Lesson 1.1: Project Setup and React Basics	Project setup, React components, JSX, React lists	Generate a new React project, create a new unordered to-do list
Lesson 1.2 React DOM and Components	React DOM, React Components	Incorporate the unordered to-list into a new component, create an "Add to-do" button underneath the list.
Lesson 1.3: Props, State, and Handlers	Handler functions, React state, callback handlers	Create a "To-do list item" component and move new list items into that component, enable form submission, store "new to-do" within React state
Lesson 1.4: Lifting State and Props Handling	Lifting state, controlled components, props handling	Move the hard-coded to-do list variable into state, add new to-do items to the list, destructure props
Lesson 1.5: Hooks and Fragments	Side-effects, custom hooks, fragments	Save the to-do list in their local browser, create custom hooks
Lesson 1.6: Reusable Components, Imperative React	Reusable components, component composition, imperative React, inline handler	Create a reusable input with label component, add a "remove" button to list items
Lesson 1.7: Asynchronous Data, Conditional Rendering, Advanced and Impossible State	Asynchronous data, conditional rendering, advanced state, impossible state	Prepare code for asynchronous data fetching, add loading state, create a conditional loading indicator
Lesson 1.8 Data Fetching, Data Re-fetching, Memorized Handlers	Data fetching/re-fetching, memorized handlers	Connect to Airtable API and fetch data from Airtable
Lesson 1.9: Third-Party Libraries, Async/Await, Forms	Third-party libraries, Async/Await, forms	Install and setup React router, add a new route component

React.js Learning Objectives



Lesson Name	Topics Covered	Students will be able to...
Lesson 2.1: React Router, Class Components	React router; React legacy: class components, state, imperative React	Refactor class components to function components using the <code>useState</code> hook (Note: this lesson does not work with the to-do list project)
Lesson 3.1: CSS, Styled Components, SVG	CSS, CSS modules, styled components, SVG	Add styling to the to-do list project using CSS modules (see rubric)
Lesson 4.1: Performance, TypeScript, Testing, Project Structure	Project structure, typechecking with prop types, unit testing	Install prop types library and assign component prop types
Lesson 5.1: Sorting, Searching, Pagination	Sorting, reverse sorting	Sort their application using Airtable and JavaScript
Final Project Rubric*		

*Rubric for Albatross course. Check the course website or GitHub page to see the most up-to-date grading rubric.