

**Introduction to React and Gatsby**

**Course Number:** RCT-126
**Duration:** 4 days

**Overview**

React is the most popular framework for building interactive web user interfaces. Gatsby is a static progressive web application (PWA) generator. Gatsby loads only the essential HTML, CSS, JavaScript, and data for your site, ensuring that it loads as quickly as possible. Gatsby also prefetches key resources for other pages so that clicking on links to these pages loads them as quickly as possible.

In this class, experienced JavaScript developers learn the fundamentals of the React framework and how to accelerate the site’s perceived page load times using Gatsby.

**Prerequisites**

All attendees must be experienced JavaScript developers. Knowledge of modern JavaScript concepts such as classes, destructuring, rest/spread operators, Promises, etc., is very helpful. If your team needs to get up to speed with JavaScript, we would be delighted to provide a 1-day primer course on the JavaScript language.

**Materials**

All React and Gatsby training attendees receive comprehensive courseware.

**Software Needed on Each Student PC**

* Google Chrome
* Other modern browsers as desired
* IDE/development environment of your choice
* Other free software and lab files that Accelebrate would specify, including Node

**Objectives**

* Learn the fundamentals of the React framework
* Build functional and class-based React components
* Style components
* Accelerate perceived page loading times with Gatsby
* Use GraphQL to rapidly query data in your applications
* Install and configure Gatsby plugins
* Use Redux for state management
* Deploy a production site

**Outline**

* Introduction
	+ Modern Web Development Tooling
	+ Languages of the Web: HTML, CSS, JavaScript
	+ Role of Data
	+ What is React?
	+ What problem does React solve?
	+ What is Gatsby?
	+ What problem does Gatsby solve?
* Getting Started with React
	+ Three Elements of React Programming
	+ Declarative Programming
	+ Component-based
	+ Learn Once, Write Anywhere
* React Projects
	+ Create React App Generator
	+ Generated Project File Structure
	+ Webpack, Babel, etc...
	+ Hot Reloading
	+ React Create Element and JSX Syntax
* Functional Components
	+ What is a Component
	+ Rendering Components
	+ Model-Driven DOM Changes
* Passing Data to Components
	+ Passing Data with Props
	+ Object Destructuring
	+ Default Props
	+ PropTypes
	+ Composing Components
	+ Composition Patterns
	+ Passing Functions with Props
* React Hooks
	+ Managing State
	+ Understand the Purpose of State
	+ Updating State Properly
	+ Manage Side Effects with Effect Hook
	+ Running Code when a Component Loads
	+ Accessing Elements with the Ref Hook
	+ Setting the Element Focus on Component Load
* Class-based Components
	+ Class-based vs. Functional Components
	+ Managing State
	+ Lifecycle Methods
* Styling React Components
	+ Style and Class Name Props
	+ CSS Modules
	+ CSS in JS
	+ Emotion vs. Styled Components
* Getting Started with Gatsby
	+ Modern Site Generator
	+ What is a static site?
* Gatsby Tooling
	+ Installing the Gatsby CLI
	+ Creating a Gatsby Project
	+ Using Starter Projects
	+ Generated Project File Structure
	+ Configuration Files
	+ Building Projects
	+ Static Site Hosting
* Gatsby Components
	+ What is a Page Component?
	+ Routing and Page Components
	+ Linking Between Pages
	+ What is a Subcomponent?
	+ Reusing Component Logic Across Pages
	+ What is a Layout Component?
	+ Layout Components and CSS
	+ Layout Components and Emotion
	+ Layout Components and Styled Components
	+ CSS Grid and Flexbox
* Working with Data
	+ What is Data?
	+ Using Many Sources of Data
	+ Kinds of Data
	+ Unstructured Data
	+ Create Page API
	+ Query Data with GraphQL
	+ Page Queries
	+ Static Queries
* Wordpress (optional)
	+ Explore how to power you Gatsby website with a Wordpress site
	+ Wordpress as a data source for building the Gatsby site
	+ Wordpress as a data source for the live executing of the web site
* Using GraphQL
	+ GraphQL Queries
	+ Thinking of Data as a Graph
	+ Using GraphIQL
	+ Selecting Properties
	+ Aliasing
	+ Variables
	+ Operation Name
	+ Fragments
* Gatsby Plugins
	+ Installing Plugins
	+ Configuring Plugins
	+ Source Plugins
	+ Transformer Plugins
	+ Explore a Variety of Plugins
* Redux (optional, adds an additional day)
	+ Use Redux to manage state in Gatsby
	+ Using the "gatsby-plugin-react-redux" Plugin
	+ Redux Pattern
	+ Immutable Programming
	+ Actions
	+ Reducers
	+ Selecting State Data
	+ Performing Asynchronous Operations
* Deploying a Production Site
	+ Auditing a Site
	+ Creating a Manifest
	+ Offline Support
	+ Page Metadata
* Conclusion