

ReactJS - Installation

React provides CLI tools for the developer to fast forward the creation, development and deployment of the React based web application. React CLI tools depends on the Node.js and must be installed in your system. Hopefully, you have installed Node.js on your machine. We can check it using the below command –

```
node --version
```

You could see the version of Node.js you might have installed. It is shown as below for me,

```
v14.2.0
```

If *Node.js* is not installed, you can download and install by visiting <https://nodejs.org/en/download/>.

Toolchain

To develop lightweight features such as form validation, model dialog, etc., React library can be directly included into the web application through content delivery network (CDN). It is similar to using jQuery library in a web application. For moderate to big application, it is advised to write the application as multiple files and then use bundler such as webpack, parcel, rollup, etc., to compile and bundle the application before deploying the code.

React toolchain helps to create, build, run and deploy the React application. React toolchain basically provides a starter project template with all necessary code to bootstrap the application.

Some of the popular toolchain to develop React applications are –

- Create React App – SPA oriented toolchain
- Next.js – server-side rendering oriented toolchain
- Gatsby – Static content oriented toolchain

Tools required to develop a React application are –

- The *serve*, a static server to serve our application during development
- Babel compiler
- Create React App CLI

Let us learn the basics of the above mentioned tools and how to install those in this chapter.

The *serve* static server

The *serve* is a lightweight web server. It serves static site and single page application. It loads fast and consume minimum memory. It can be used to serve a React application. Let us install the tool using *npm* package manager in our system.

```
npm install serve -g
```

Let us create a simple static site and serve the application using *serve* app.

Open a command prompt and go to your workspace.

```
cd /go/to/your/workspace
```

Create a new folder, *static_site* and change directory to newly created folder.

```
mkdir static_site  
cd static_site
```

Next, create a simple webpage inside the folder using your favorite html editor.

```
<!DOCTYPE html>  
<html>  
  <head>  
    <meta charset="UTF-8" />  
    <title>Static website</title>  
  </head>  
  <body>  
    <div><h1>Hello! </h1></div>  
  </body>  
</html>
```

Next, run the *serve* command.

```
serve .
```

We can also serve single file, *index.html* instead of the whole folder.

```
serve ./index.html
```

Next, open the browser and enter `http://localhost:5000` in the address bar and press enter. serve application will serve our webpage as shown below.

Hello not found or type unknown

The serve will *serve* the application using default port, 5000. If it is not available, it will pick up a random port and specify it.

```
| Serving! |
|         |
| - Local: http://localhost:57311 |
| - On Your Network: http://192.168.56.1:57311 |
|         |
| This port was picked because 5000 is in use. |
|         |
| Copied local address to clipboard! |
```

Babel compiler

Babel is a JavaScript compiler which compiles many variant (es2015, es6, etc.,) of JavaScript into standard JavaScript code supported by all browsers. React uses JSX, an extension of JavaScript to design the user interface code. Babel is used to compile the JSX code into JavaScript code.

To install Babel and it's React companion, run the below command –

```
npm install babel-cli@6 babel-preset-react-app@3 -g
...
...
+ babel-cli@6.26.0
+ babel-preset-react-app@3.1.2
updated 2 packages in 8.685s
```

Babel helps us to write our application in next generation of advanced JavaScript syntax.

Create React App toolchain

Create React App is a modern CLI tool to create single page React application. It is the standard tool supported by React community. It handles babel compiler as well. Let us install *Create React App* in our local system.

```
> npm install -g create-react-app
+ create-react-app@4.0.1
added 6 packages from 4 contributors, removed 37 packages and updated 12 packages in 4.693s
```

Updating the toolchain

React Create App toolchain uses the *react-scripts* package to build and run the application. Once we started working on the application, we can update the *react-script* to the latest version at any time using *npm* package manager.

```
npm install react-scripts@latest
```

Advantages of using React toolchain

React toolchain provides lot of features out of the box. Some of the advantages of using React toolchain are –

- Predefined and standard structure of the application.
- Ready-made project template for different type of application.
- Development web server is included.
- Easy way to include third party React components.
- Default setup to test the application.

Revision #1

Created 14 December 2022 10:36:10 by Admin

Updated 14 December 2022 10:37:08 by Admin