AI CREATORS PROGRAM

Setup on Windows:

1. Installation of Node Js on Windows

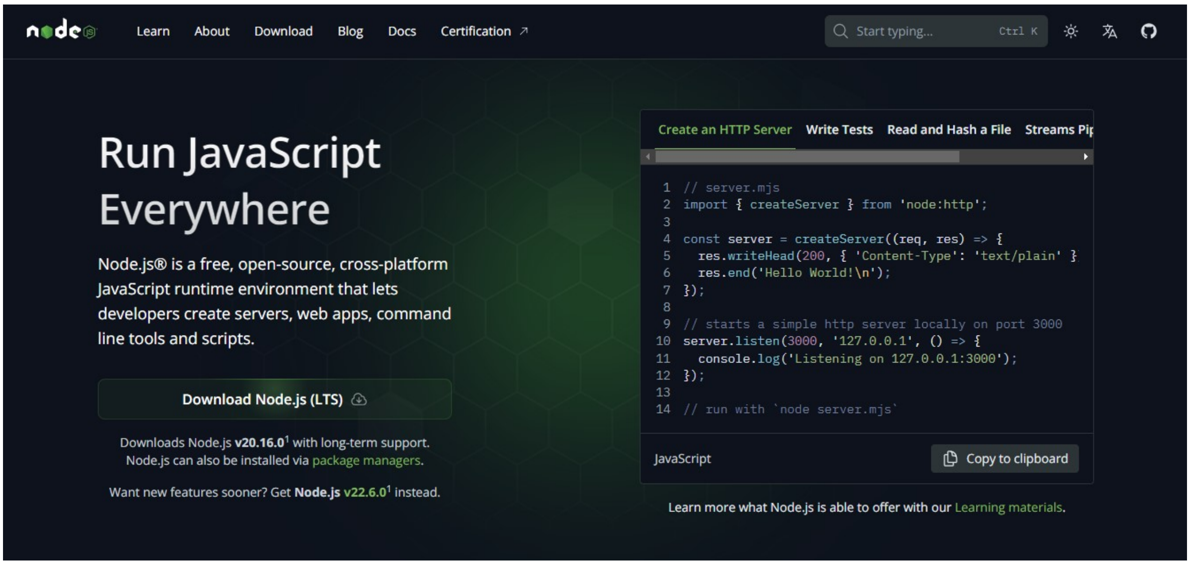
Step 1: Download Node.js Installer

Visit the Node.js website: Go to the official Node.js download page: https://nodejs.org/

Choose the appropriate version:

You can choose between the LTS (Long Term Support) version or the Current version. The LTS version is recommended for most users as it's more stable.

Click on the Windows installer (.msi file) for your system architecture (32-bit or 64-bit).



Run JavaScript Everywhere

Node.js® is a free, open-source, cross-platform JavaScript runtime environment that lets developers create servers, web apps, command line tools and scripts.

Downloads Node.js **v20.16.0**[1] with long-term support. Node.js can also be installed via package managers.

Want new features sooner? Get **Node.js v22.6.0**[1] instead.

```javascript
// server.mjs
import { createServer } from 'node:http';

const server = createServer((req, res) => {
  res.writeHead(200, { 'Content-Type': 'text/plain' }
  res.end('Hello World!\n');
});

// starts a simple http server locally on port 3000
server.listen(3000, '127.0.0.1', () => {
  console.log('Listening on 127.0.0.1:3000');
});

// run with `node server.mjs`
```

Learn more what Node.js is able to offer with our Learning materials.

Step 2: Run the Installer

Locate the downloaded file: Once the download is complete, locate the .msi file in your Downloads folder or the location where you saved it.

Run the installer: Double-click the .msi file to start the installation process.

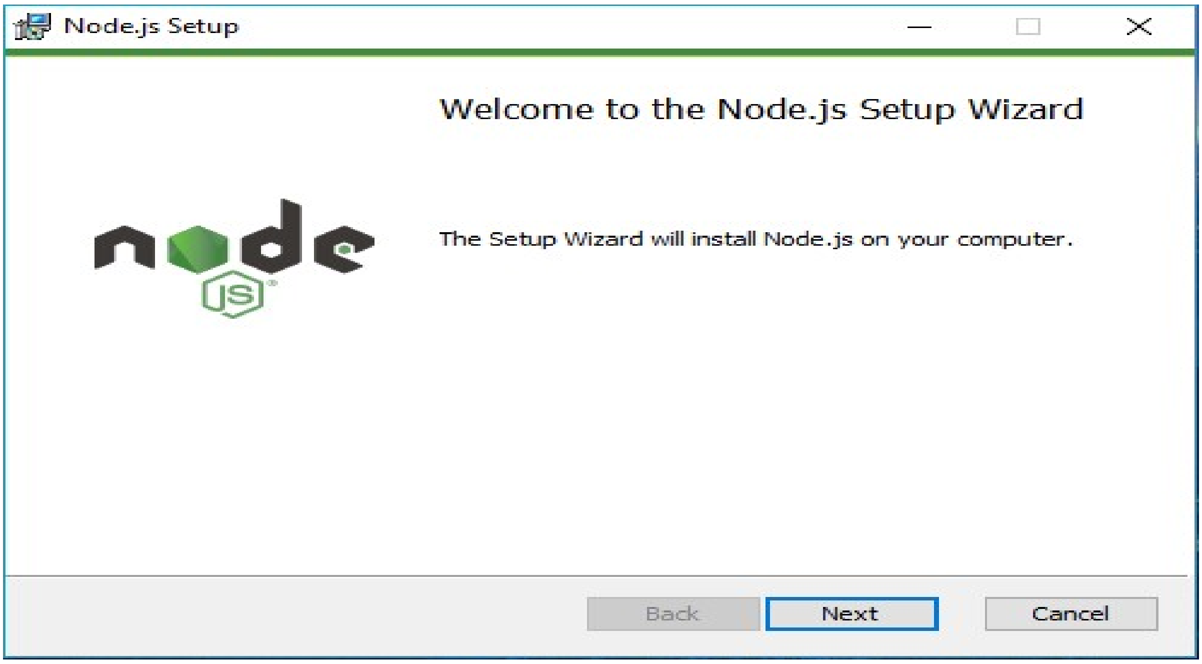
Step 3: Installation Process

Welcome Screen: You'll be greeted with a welcome screen. Click **Next** to proceed.

License Agreement: Read the license agreement, select the checkbox to accept the terms, and then click **Next**.

Destination Folder: Choose the installation path or leave it as default, then click **Next**.

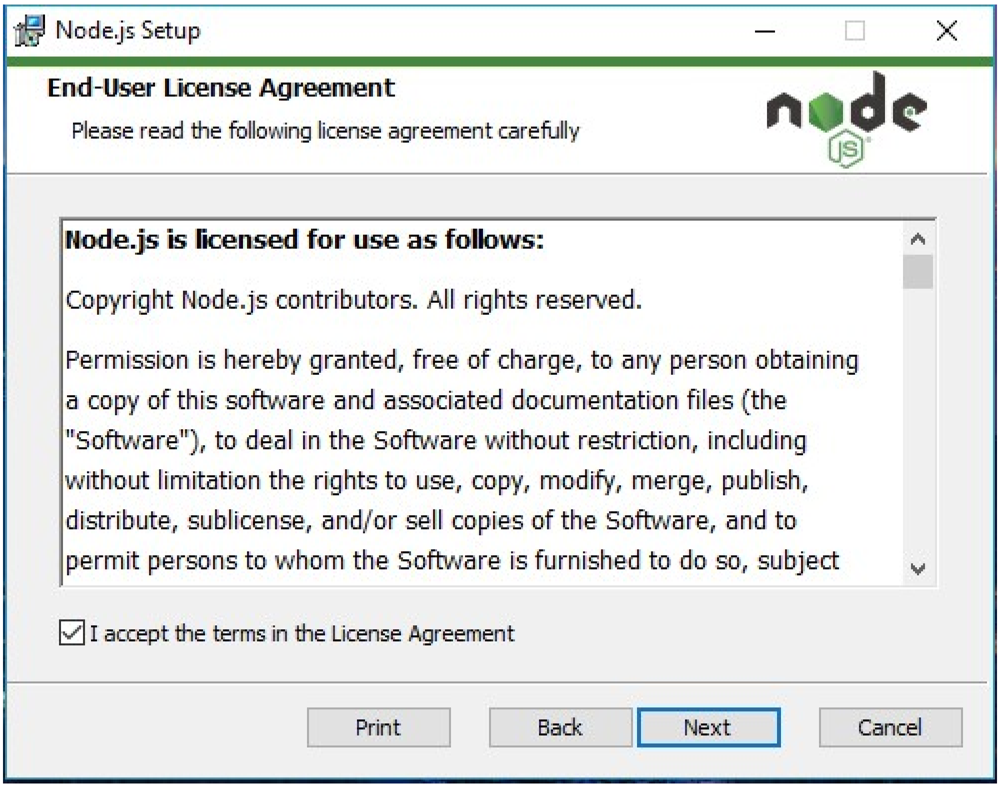
Custom Setup: You'll be given the option to install additional tools. Make sure the boxes for "**Node.js runtime**," "**npm package manager**," and "**Add to PATH**" are checked. Click **Next**.



Node.js Setup

Welcome to the Node.js Setup Wizard

The Setup Wizard will install Node.js on your computer.



End-User License Agreement

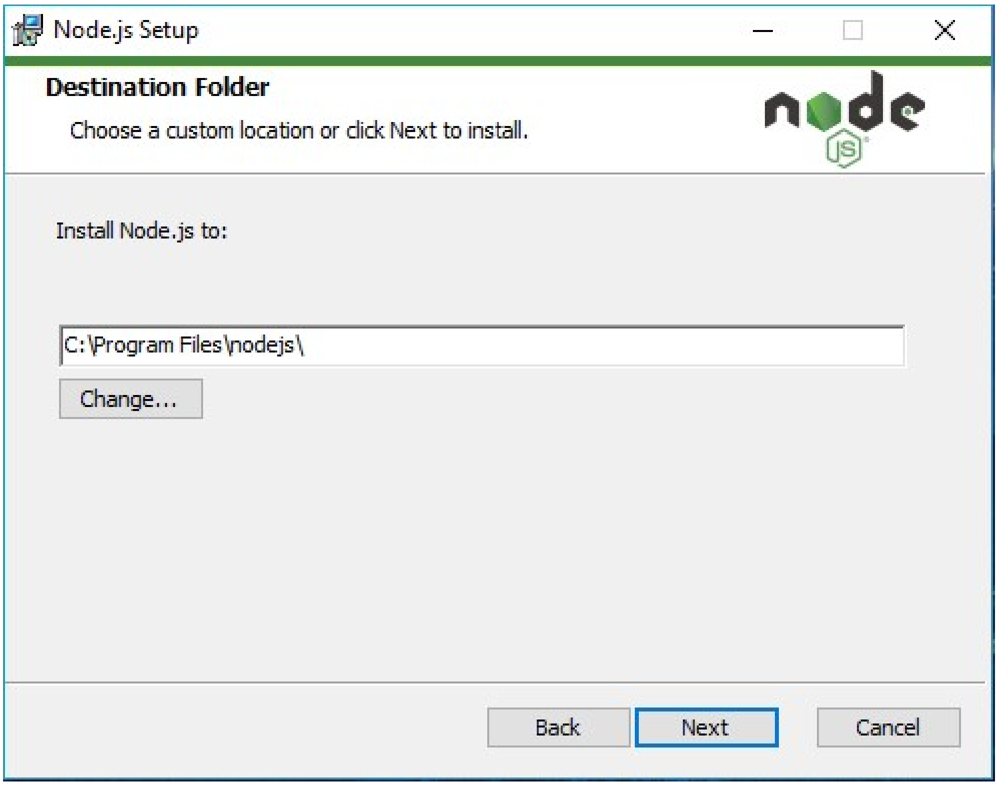
Please read the following license agreement carefully

Node.js is licensed for use as follows:

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject

☑ I accept the terms in the License Agreement

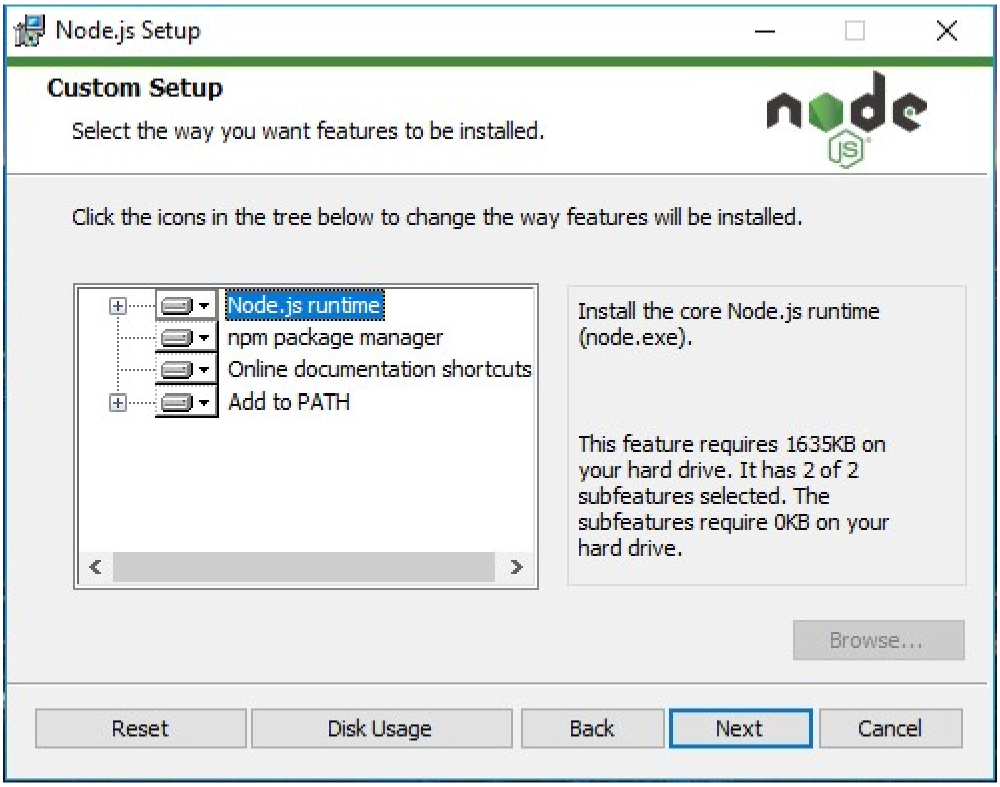


Destination Folder

Choose a custom location or click Next to install.

Install Node.js to:

C:\Program Files\nodejs\



Node.js Setup

Custom Setup

Select the way you want features to be installed.

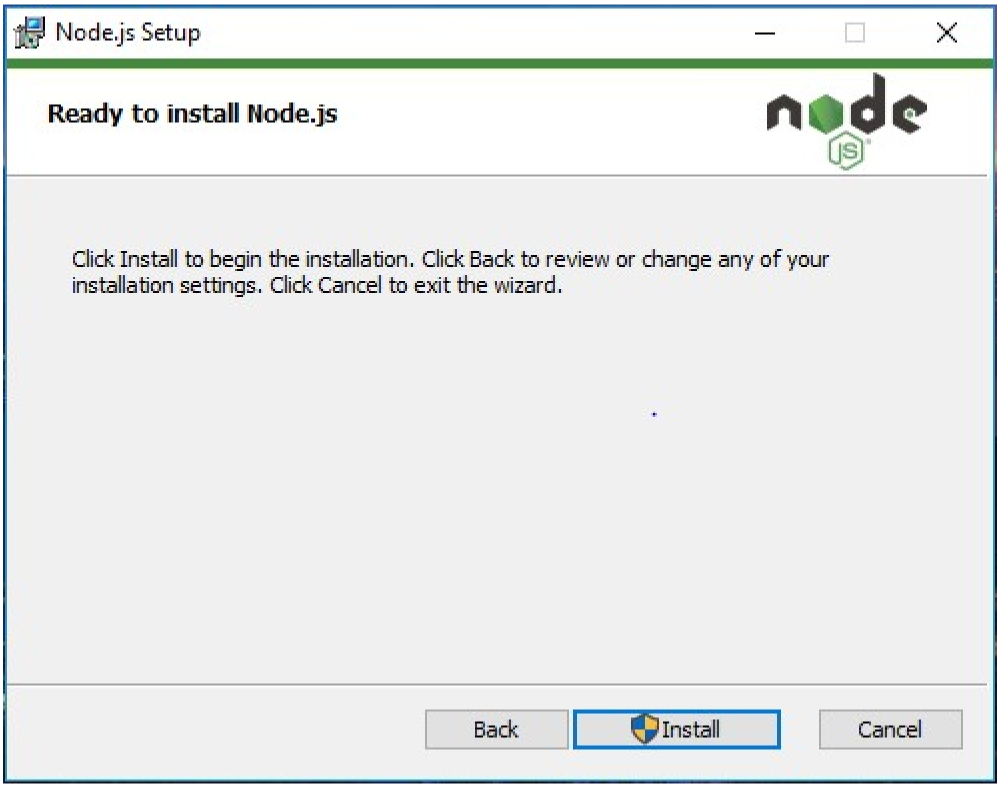
Click the icons in the tree below to change the way features will be installed.

- Node.js runtime
- npm package manager
- Online documentation shortcuts
- Add to PATH

Install the core Node.js runtime (node.exe).

This feature requires 1635KB on your hard drive. It has 2 of 2 subfeatures selected. The subfeatures require 0KB on your hard drive.

Ready to Install: Review your settings and click **Install** to begin the installation process.



Node.js Setup

Ready to install Node.js

Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.

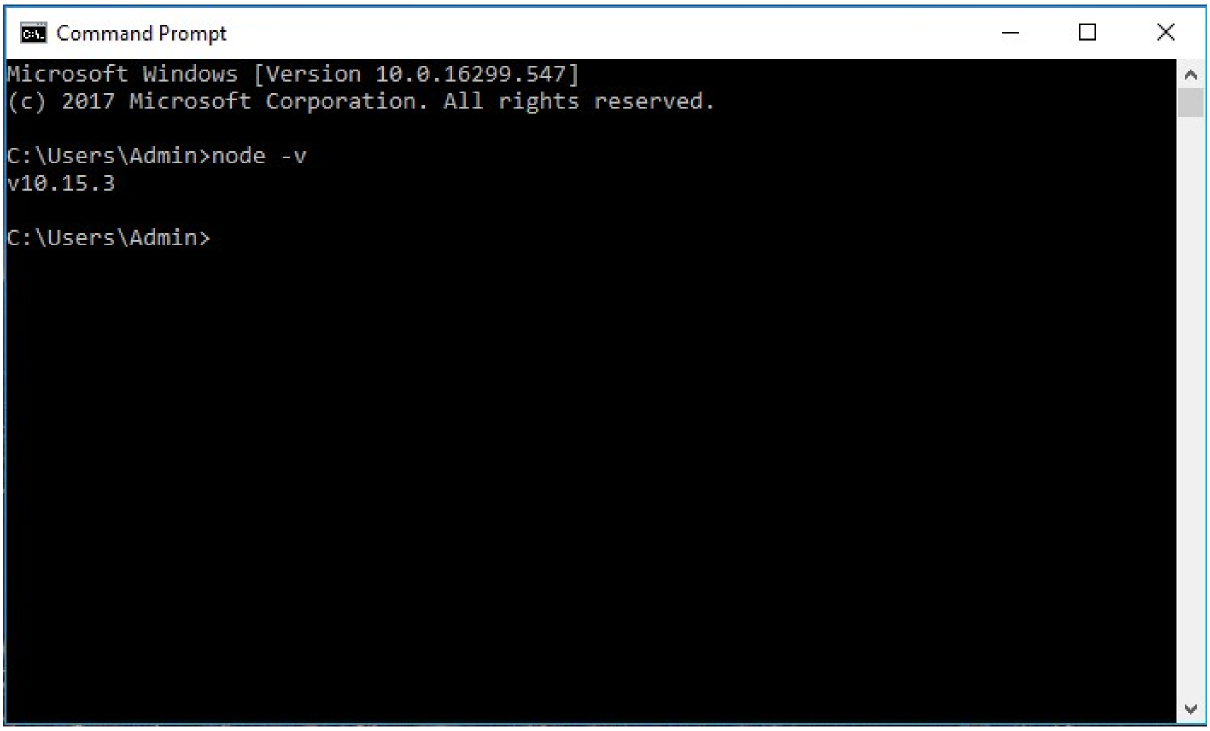
Complete the Installation: Once the installation is complete, click **Finish** to exit the installer.

Step 4: Verify the Installation

Open Command Prompt: Press Win + R, type cmd, and hit Enter.

Check Node.js version: Type **node -v** and press Enter. This command will display the installed Node.js version.

Check npm version: Type **npm -v** and press Enter. This command will display the installed npm version.



```
Microsoft Windows [Version 10.0.16299.547]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Admin>node -v
v10.15.3

C:\Users\Admin>
```

2. Introduction to ReactJS and create-react-app

● **ReactJS**: A JavaScript library for building user

interfaces.

● **create-react-app**: A tool that sets up a new React

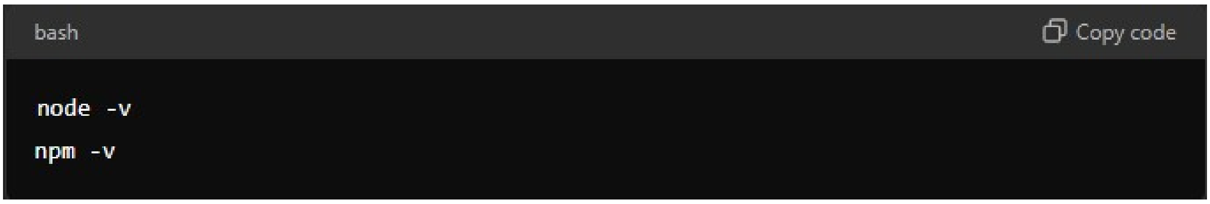
project with a default configuration.

3. Setting up a Basic ReactJS Project using create-react-app on

Windows

Prerequisites

Node.js and npm installed: Ensure that Node.js and npm are installed on your system. You can verify this by running the following commands in Command Prompt:



```bash
node -v
npm -v
```

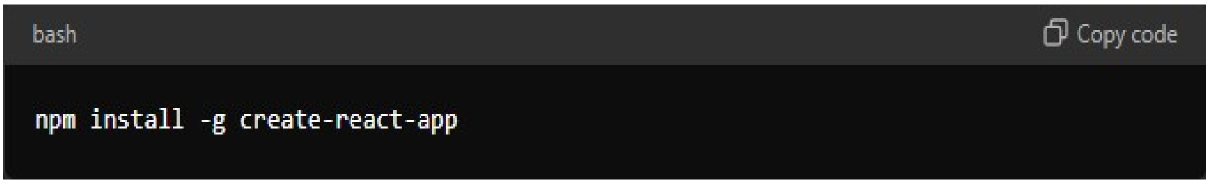
Step 1: Open Command Prompt or PowerShell

Press Win + R, type cmd, and press Enter to open Command Prompt.

Alternatively, you can use PowerShell.

Step 2: Install create-react-app

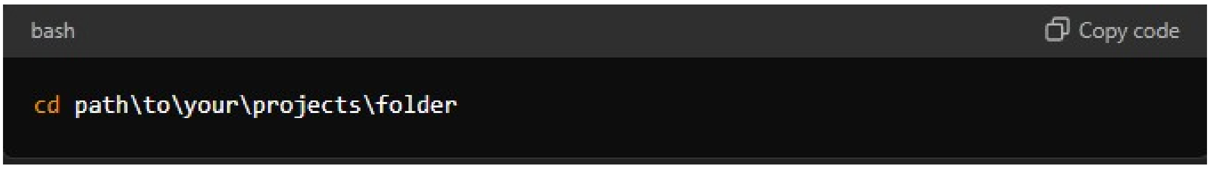
To install create-react-app globally, run the following command:



```bash
npm install -g create-react-app
```

Step 3: Create a New React Application

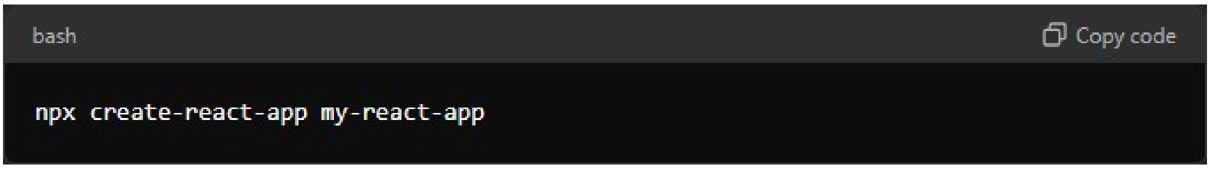
Navigate to your projects folder:



```bash
cd path\to\your\projects\folder
```

Replace **path\to\your\projects\folder** with the path where you want to create your new React project.

Create a new React app:

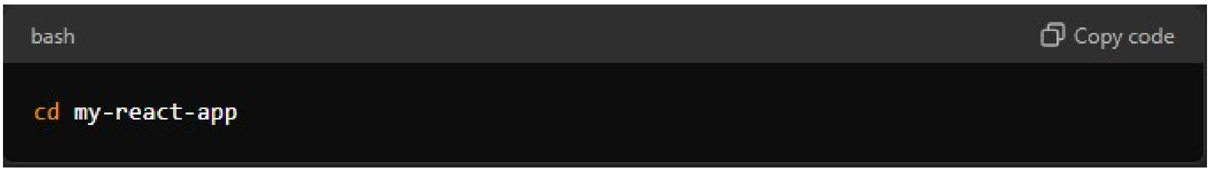


```bash
npx create-react-app my-react-app
```

Replace my-react-app with your desired project name. This will create a new folder named my-react-app and set up the React project inside it.

Step 4: Navigate into Your Project Directory

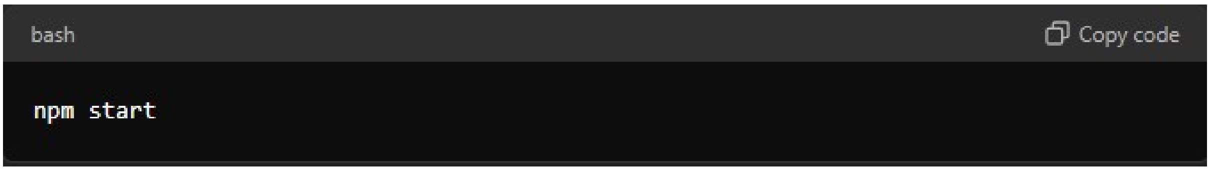
Change into the project directory:



```bash
cd my-react-app
```

Step 5: Start the Development Server

Start the development server with the following command:

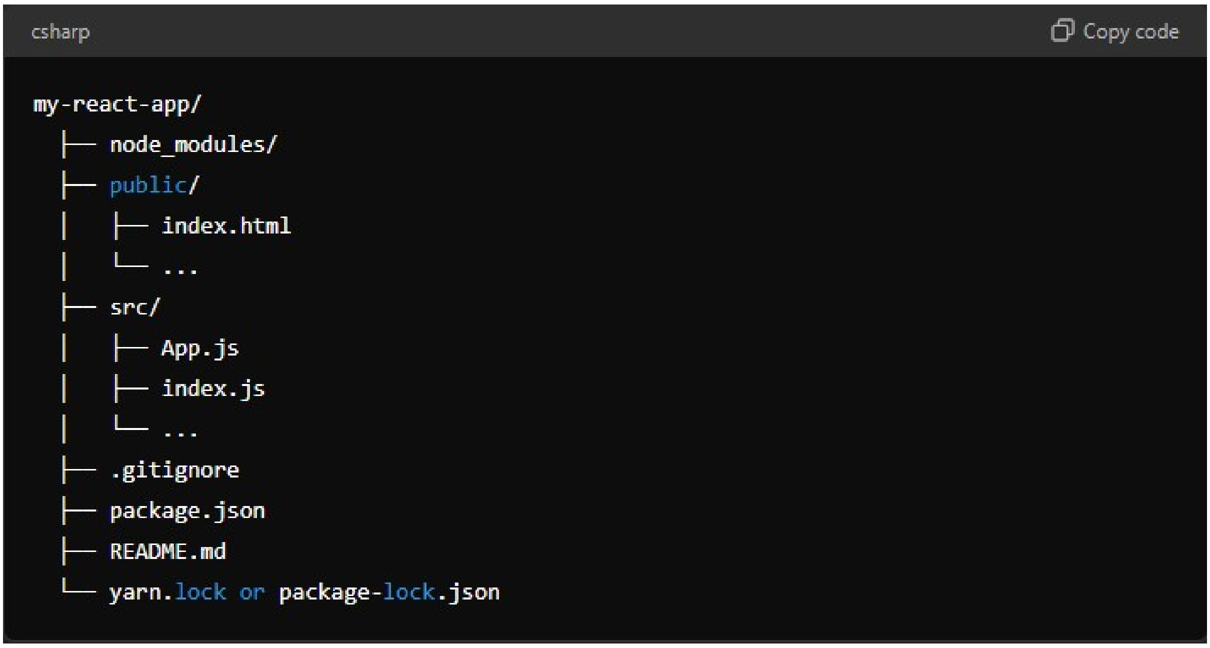


```bash
npm start
```

This will start the development server and open your React application in the default web browser. The application will be running at http://localhost:3000.

Step 6: Explore the Project Structure

After running create-react-app, your project structure will look something like this:



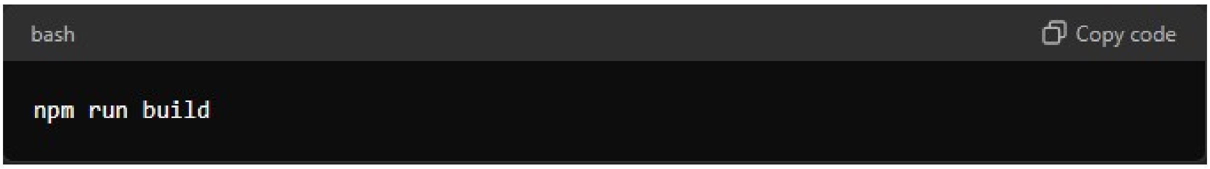
```
my-react-app/
  ├── node_modules/
  ├── public/
  │   ├── index.html
  │   └── ...
  ├── src/
  │   ├── App.js
  │   ├── index.js
  │   └── ...
  ├── .gitignore
  ├── package.json
  ├── README.md
  └── yarn.lock or package-lock.json
```

Step 7: Make Changes and See Live Updates

You can edit files inside the **src** folder, such as **App.js**, and save your changes. The development server will automatically reload and show your changes in the browser.

Step 8: Build the Project for Production

When you are ready to deploy your application, run:



```bash
npm run build
```

Step 9: Additional Tools (Optional)

Visual Studio Code: A lightweight code editor with great support for JavaScript and React.

ESLint and Prettier: For code linting and formatting.

4. Deploying Your React App on Windows

Deployment Options:

○ **GitHub Pages:** Deploy using gh-pages.

○ **Netlify:** Drag and drop the build folder or connect a GitHub repo.

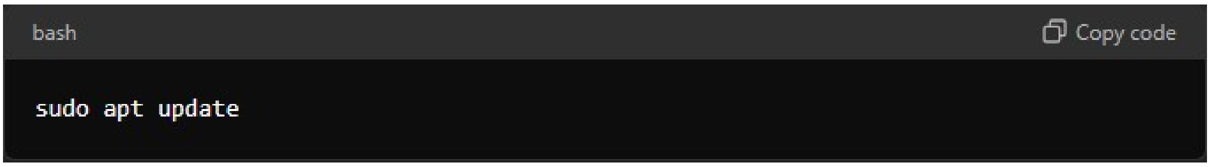
○ **Vercel**: Integrate with your GitHub repo for deployment.

Installation of Node Js on Linux:

1. **Using NodeSource (recommended)**

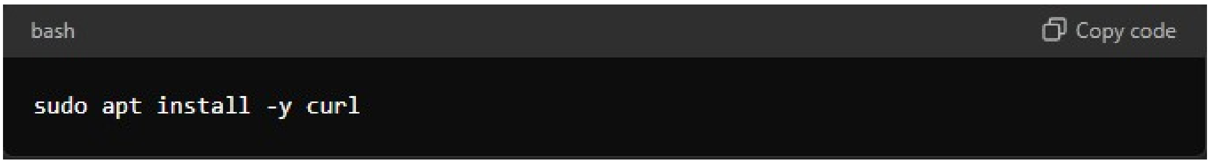
This method ensures you get the latest stable version of Node.js.

Update your package index:



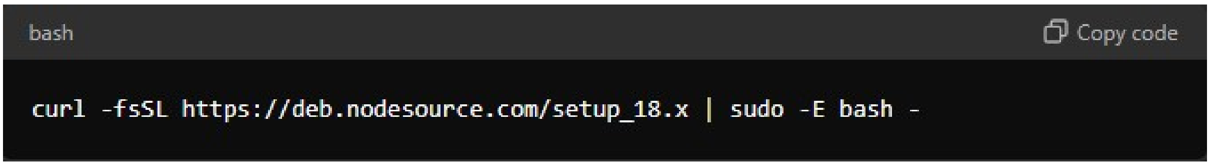
```bash
sudo apt update
```

Install prerequisites:



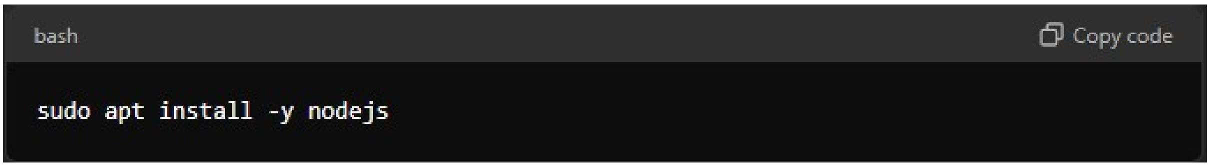
```bash
sudo apt install -y curl
```

Download and install the NodeSource PPA (replace setup_16.x with the version you want, e.g., setup_18.x for Node.js 18):



```bash
curl -fsSL https://deb.nodesource.com/setup_18.x | sudo -E bash -
```

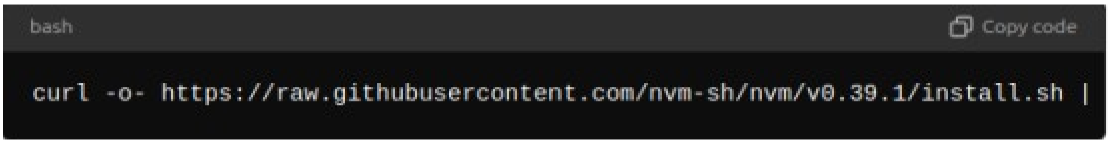
Install Node.js:



```bash
sudo apt install -y nodejs
```

(Optional) Install Specific Version Using Node Version Manager (NVM):

- **Install NVM:**

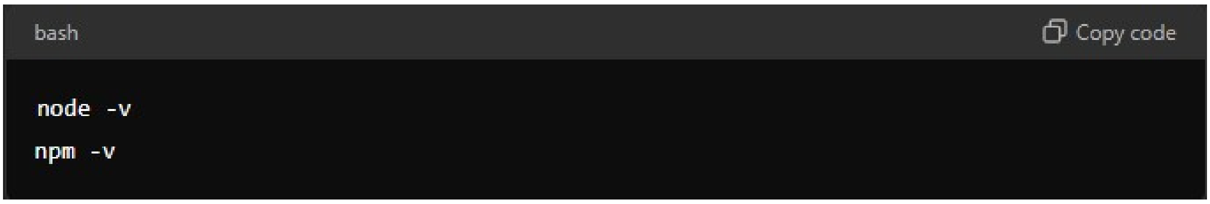


```
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.1/install.sh |
```

Step 2: Verify Installation

Check Node.js Version:

Check npm Version:



```
node -v
npm -v
```

2. Introduction to ReactJS and create-react-app

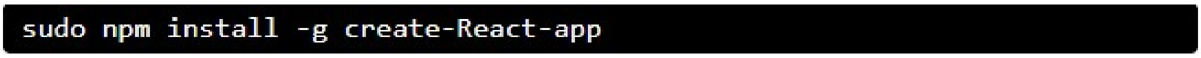
- **ReactJS:** A JavaScript library for building user interfaces.

Install npm:

- **create-react-app**: A tool that sets up a new React project with a default configuration.

3. Setting up a Basic ReactJS Project using create-react-app on Linux

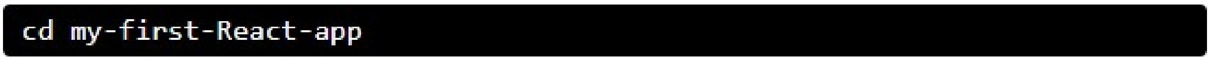
Step 1: Install create-react-app Globally (Optional)



```
sudo npm install -g create-React-app
```

Step 3: Navigate to the Project Directory

- **Change Directory:**

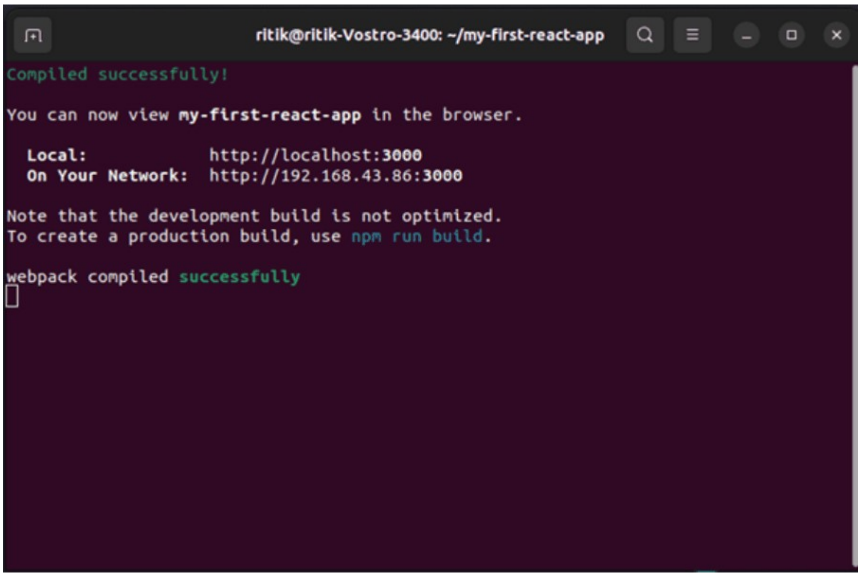


```
cd my-first-React-app
```

Step 4: Run the Development Server

● **Start the Server**:

```
npm start
```



```
Compiled successfully!

You can now view my-first-react-app in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.43.86:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

This should automatically open your default browser with the starter React app loaded!

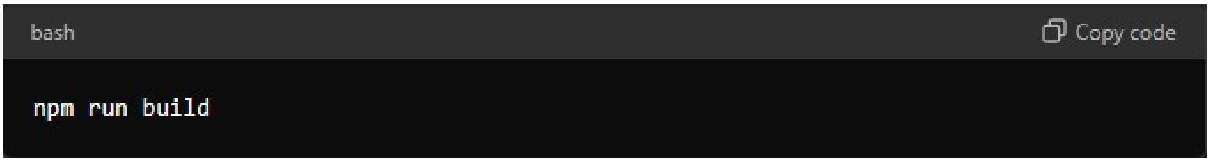
Step 5: Editing the React Project

Open the Project in an Editor: Use an editor like Visual Studio Code, Sublime Text or Vim. Open src/App.js to start modifying the code.

Step 6: Build for Production

Install npm:

● Create a Production Build:



```
npm run build
```

4. Deploying Your React App on Linux

Deployment Options:

○ **GitHub Pages**: Deploy using gh-pages.

○ **Netlify**: Drag and drop the build folder or connect a GitHub repo.

○ **Vercel:** Integrate with your GitHub repo for deployment.

Installation of VS Codium on Windows and Linux:

VS Codium is a free, open-source build of Visual Studio Code, without the Microsoft branding and telemetry. Here's how you can install it on both Windows and Linux:

Installation on Windows:

1. **Download the Installer:**
- Visit the VS Codium website.
- Click on the "Windows 64" or "Windows 32" download link under the "Windows" section.
2. **Run the Installer:**
- Once downloaded, open the installer file (VSCodium-win32-x64-<version>.exe).
- Follow the installation prompts.
3. **Launch VS Codium:**
- After installation, you can launch it from the Start menu or by searching for "VS Codium."

Installation on Linux:

- For Debian/Ubuntu-based distributions:

Add the GPG Key and Repository:

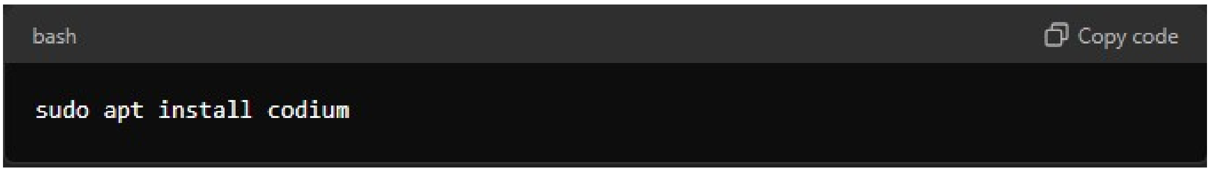
Copy code

```
wget -qO - https://gitlab.com/paulcarroty/vscodium-deb-rpm-repo/raw/master/pub.gpg | sudo apt-key add -

echo 'deb [signed-by=/usr/share/keyrings/vscodium-archive-keyring.gpg] https://deb.vscodium.com/ codium main' | sudo tee /etc/apt/sources.list.d/vscodium.list

sudo apt update
```

Install VS Codium:



```bash
sudo apt install codium
```

For Fedora/RHEL/CentOS:

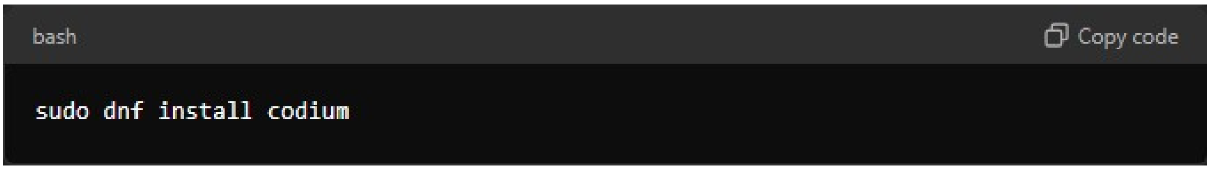
Add the Repository:

Copy code

```
sudo rpmkeys --import https://gitlab.com/paulcarroty/vscodium-deb-rpm-repo/raw/master/pub.gpg

sudo wget https://gitlab.com/paulcarroty/vscodium-deb-rpm-repo/raw/master/vscodium.repo -O /etc/yum.repos.d/vscodium.repo
```

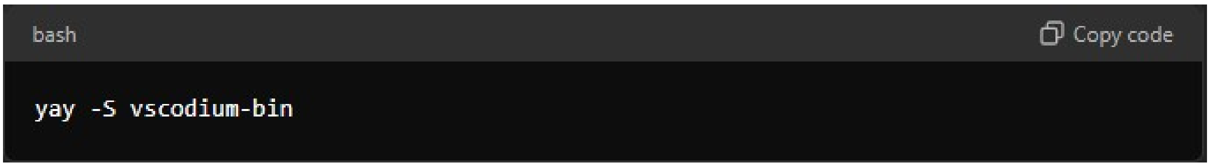
Install VS Codium:



```bash
sudo dnf install codium
```

For Arch Linux:

Using the AUR:

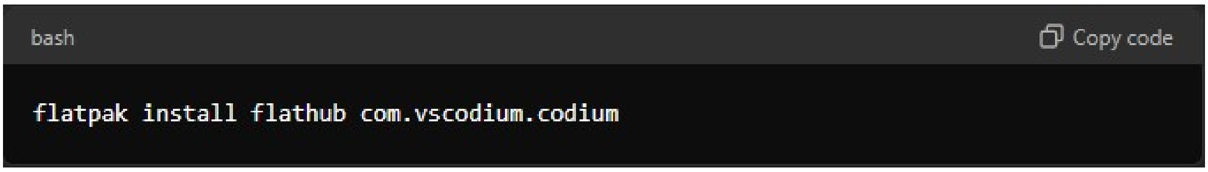


```bash
yay -S vscodium-bin
```

If you don't have yay installed, you can install it or use another AUR helper.

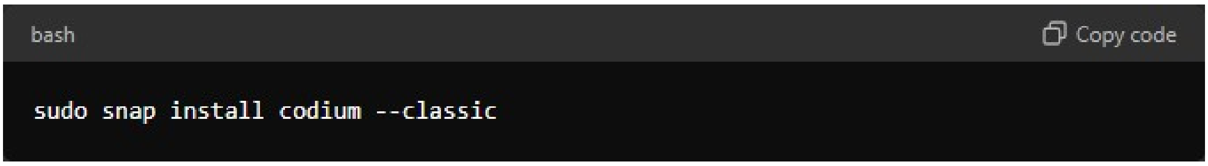
Alternative Installation Methods:

Flatpak (for Linux):



```bash
flatpak install flathub com.vscodium.codium
```

Snap (for Linux):



```bash
sudo snap install codium --classic
```

After installation, you can launch VS Codium from your application menu or via the terminal using the codium command.