Setting up a React project in VS Code (Optional)

To complete the exercises in this course you have been provided with a dedicated lab environment set up specifically for you to apply the skills that you have learned. You can find out more about Working with Labs in this course in the following reading.

You can also use VS Code to practice these exercises on your local machine as an alternative option.

To follow along in this reading, you need to have Node.js and VS Code already installed on your computer. If you don’t have this setup, please refer to the following resources in this course: Setting up VS Code and Installing Node and NPM.

In VS Code, you're ready to start a brand new React project.

You can do it using npm.

**What is npm?**

When Node.js is installed on a computer, **npm** comes bundled with it.

With **npm**, you can:

1. Author your own Node.js modules ("packages"), and publish them on the npm website so that other people can download and use them
2. Use other people's authored modules ("packages")

So, ultimately, npm is all about *code sharing* and *reuse*. You can **use other people's code** in your own projects, and you can also **publish your own Node.js modules** so that other people can use them.

An example npm module that can be useful for a new React developer is [create-react-app](https://create-react-app.dev/). While this npm module comes with its own website, you can also find some info on the [create-react-app project on GitHub](https://github.com/facebook/create-react-app).

Whenever you run the npm command to add other people's code, **that code, and all other Node modules that depend on it, get downloaded** to your machine.

However, although it's possible to do so, this is not really necessary, at least in the case of the **create-react-app** Node module.

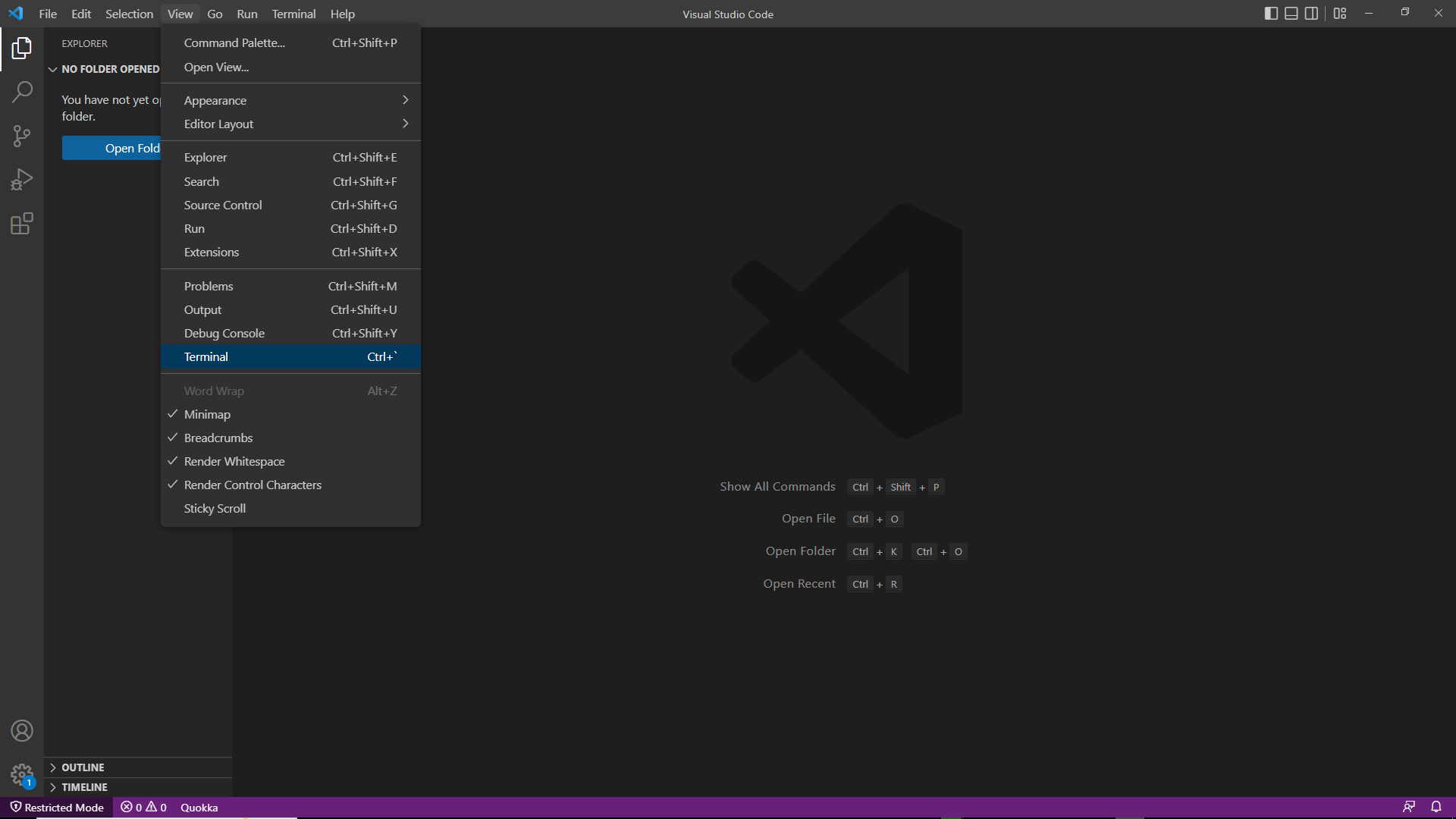
In other words, you can avoid installing the **create-react-app** package but still use it.

You can do that by running the following command: **npm init react-app example**, where “example” is the actual name of your app. You can use any name you’d like, but it’s always good to have a name that is descriptive and short.

In the next section, you'll learn how to build a brand new app that you can name: **firstapp**.

**Opening the built-in VS Code terminal and running *npm init react-app* command**

In VS Code, click on *View*, *Terminal* to open the built-in terminal.



Now run the command to add a brand new React app to the machine:

1

npm init react-app firstapp

The installation and setup might take a few minutes.

Here's the output of executing the above command:

removed 1 package, and audited 1422 packages in 3s

190 packages are looking for funding

                       run `npm fund` for details

6 high severity vulnerabilities

To address all issues (including breaking changes), run:

npm audit fix --force

Run `npm audit` for details.

Created git commit.

Success! Created firstapp at /home/pc/Desktop/firstapp

Inside that directory, you can run several commands:

    npm start

    Starts the development server.

    npm run build

    Bundles the app into static files for production.

    npm test

    Starts the test runner.

    npm run eject

    Removes this tool and copies build dependencies, configuration files

    and scripts into the app directory. If you do this, you can’t go back!

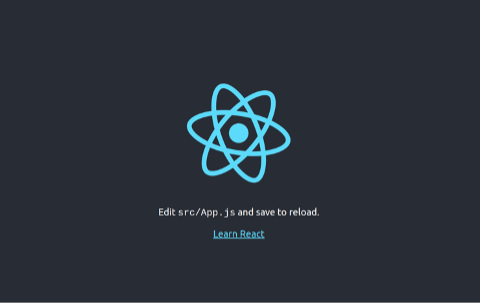
We suggest that you begin by typing:

cd firstapp

npm start

If you follow the suggestions from the above output, you'll run: **cd firstapp**, and then **npm start**.

Following the instructions, opening a browser with the address bar pointing to [http://localhost:3000](http://localhost:3000/), will show the following page in your browser:



This means that you've successfully:

* Set up your local development environment
* Run the create-react-app npm package (without installing it!)
* Built a starter React app on your local machine
* Served that starter React app in your browser

After you've built your starting setup, in Module 2 you'll start working with the basic building blocks of React: components.