Introduction to React Quiz

Due No due date **Points** 6 **Questions** 6 **Time Limit** None **Allowed Attempts** 3

Instructions

(https://github.com/learn-co-curriculum/phase-2-quiz-introduction-to-react/issues/new)

It's time to check your knowledge! Use this quiz to create a custom study guide. Note any answers that were marked incorrect, so you can study the relevant material and try this quiz again. If you don't know the answer to a question, please do not guess. Instead, select "I don't know". It's OK not to know everything and to admit when we're unsure.

Take the Quiz Again

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	3 minutes	5.75 out of 6

(!) Answers will be shown after your last attempt

Score for this attempt: 5.75 out of 6

Submitted May 3 at 6:51am This attempt took 3 minutes.

Question 1 1/1	pts
Which of the following describe React as a library? Select all that ap	oply.
☑ Component-Based	
	?

Yes!

React encourages us to structure our applications as "components": the building blocks of our web page that handle their own data and UI logic, that can be "composed" together with other components to put together a complete app.

- I don't know.
- A Model-View-Viewmodel Framework
- Declarative

Correct!

React helps us write declarative code, which makes our code easier to understand. Declarative code:

- Describes what a program should accomplish (or what the end result should be)
- Leaves the determination of how to get to the end result up to the program
- Learn Once, Write Anywhere

Nice One! One great thing about React is that it works in a few different environments. Once you learn the key concepts of working with React to create client-side web applications, you can also more easily learn React Native ((https://reactnative.dev/) and write native mobile apps using React, or use tools like Next.js (https://nextjs.org/) and Gatsby.

(https://www.gatsbyjs.com/docs/) to write server-rendered React.

Source/s:

• What is React?

Question 2 1/1 ? Help

Of the code examples below, which uses **declarative** syntax?

I don't know.

const header = <h1 className="main">Hello from React!</h1>;
ReactDOM.render(header, document.querySelector("#container"
));

Correct!

With declarative syntax, we are able to leave the determination of *how* to get to the end result up to the program! Therefore, we *don't* need to lay out every step that needs to happen, like we do with imperative syntax.

```
const header = document.createElement("h1");
header.textContent = "Hi!";
header.className = "main";
const container = document.querySelector("#container");
container.append(header);
```

Source/s:

What is React?

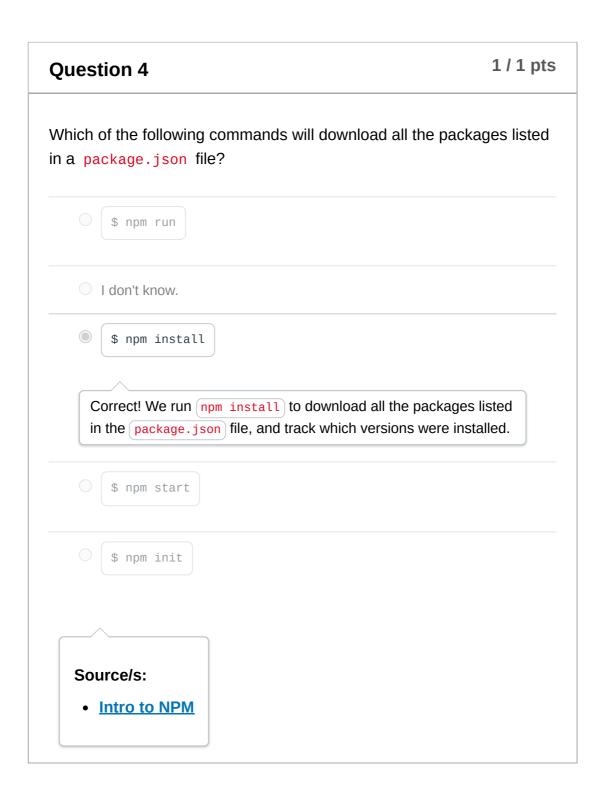
Question 3 1/1 pts Match the term with the correct description. Declarative Programming Describes what a progr; ✓ Imperative Programming Explicitly describes the ✓ Describes what a progr; ✓ Describes what a progr; ✓ Describes what a progr; ✓ Output Describes what a progr; ✓ Describes what a p

) Help

Nailed it!

Source/s:

What is React



Question 5

1/1| (?) Help

is a package manager for JavaScript that works with your JavaScript project directories via the command line, allowing you to install packages of preexisting code. NPM Nailed it!

Source/s:

Intro to NPM

Partial

0.75 / 1 pts **Question 6**

Why is React useful for developers? Select all that apply.

React allows us to create reusable UI components.

Correct!

React encourages us to structure our applications as "components": the building blocks of our web page that handle their own data and UI logic, that can be "composed" together with other components to put together a complete app.

It works in many different environments allowing developers to build not only web applications, but mobile applications and server-rendered React.

Nice One!

One great thing about React is that it works in a few different environments. Once you learn the key concepts of working with React to create client-side web applications, you can also more easily learn React Native (https://reactnative.dev/) and write native mobile apps using React, or use tools like Next.js (https://nextjs.org/) and Gatsby (https://www.gatsbyjs.com/docs/) to write server-rendered React.

I don't know.

React allows us to create applications that can change data, without reloading the page.



React has a great set of developer tools that make it easier to view our components in the browser and debug what's happening under the hood.

100%!

Check out the **React Dev Tools**

(https://chrome.google.com/webstore/detail/react-developertools/fmkadmapgofadopljbjfkapdkoienihi?hl=en) so you can start debugging like a React pro!

Source/s:

- What is React?
- React Dev Tools (https://chrome.google.com/webstore/detail/react-developertools/fmkadmapgofadopljbjfkapdkoienihi?hl=en)
- React Native (https://reactnative.dev/)
- Next.js (https://nextjs.org/)
- Gatsby (https://www.gatsbyjs.com/docs/)