# **Course Introduction**

## Introduction to the course: React Basics

n/a

## How is React used in the real world?

n/a

# **React Components and Where They Live**

## Why React?

n/a

## React.js overview

When meta released the React library, it included the concept of something called component-based architecture. This is essentially a design philosophy for building software based on reusable components of code. Each component consists of well-defined functionality that can be inserted into an application without requiring modification of other components. Because components are reusable, they can be used multiple times and easily inserted anywhere we're needed. This results in components that can exist within the same space yet interact independently from each other.

*True or false? React enables developers to build SPAs (single page applications).*

* *True*

## Introduction to functional components

React provides two types of components, functional components and class components. They behave very similar in React to traditional functions, and classes in JavaScript.

In the default React application, only one component is rendered and it's the app components located inside the index.js file.

It's important to know that every React app must contain at least one component, and it's called the root components. This component is loaded using the import statement.

The syntax to render a component is very similar to a self-closing tag in HTML, you just place the component name inside the left and right-angle brackets and don't forget the forward slash.

A screenshot of a computer

Description automatically generated with low confidence

React is scripted using a special syntax called JavaScript XML or JSX. For many Reacts developers, this is known as a syntax extension to JavaScript.

Text, letter

Description automatically generated

JSX syntax looks very similar to HTML. What are its advantages is that it allows you to write JavaScript code inside what looks like HTML elements. In fact, you can think of JSX as a combination of custom HTML and JavaScript. This allows you to make your website dynamic.

Notice that the first letter of the component name is capitalized. This is because there's a difference in how React treats capitalized, and non-capitalized component names. It's important to remember that all component names in React must be capitalized. Why is this? Well, because React treats lowercase components as regular HTML elements. Capitalizing a component name helps React to distinguish JSX elements from HTML elements.

Graphical user interface, text, application

Description automatically generated

This rendering happens behind the scenes because of something called transpiling. You can think of transpiling as a process of converting JSX to HTML and you'll learn more about this later.

*A React component is much like a regular JavaScript function.*

* *True*

## Creating React components

Graphical user interface, application

Description automatically generated

The dot character instructs VS Code to run this command in the current folder. In other words, I'm using Create React app to build a new app for me inside this folder.

I can start the app by typing npm then a space followed by start.

Graphical user interface, text, application

Description automatically generated

*True or false? For a component to render something on the page, it needs to return one or more JSX elements.*

* *True*

## The React project structure

n/a

*As a new developer, what's the part of the create-react-app-generated starter app structure that you should focus on?*

* *the 'src' folder*

## Importing components

In order to create a fully functioning React app, you need to create a collection of components.

what is the difference between modules and components, since they are both essentially just JavaScript files? You are right. While they have similarities, it can help to think of a component as a single part or small piece of functionality like a button. Then you can think of a module as something that's larger than just one component like a series of components. This technique of splitting your code into several modules is known as modular programming, and it complements the component-based architecture of React.

*Imagine that you have an App component and a Footer component in the same location, at the root of the src folder. Choose the proper way to import the Footer component into the App component.*

* *import Footer from "./Footer";*

# **Components Use and Styling**

## Principles of components: Props

Recall that in JavaScript, you can make your functions more flexible by declaring them with parameters that allow you to pass in values as arguments when you call the function. Well, in React, you can perform a similar action using something called properties, which are represented as props.

Props allow you to pass data from one component to another. It's helpful to think of props as arguments a component can accept and are passed using JSX syntax, much like HTML attributes. Inside the function, you use the keyword props.

A picture containing diagram

Description automatically generated

When two components communicate with each other, the component sending the props data is known as the parent, and the component receiving the props data is known as the child. As you've just learned with the example earlier, this parent-child relationship allows parent components of past data down to child components using props. It's also possible for parent components to send the same data to multiple child components.

However, it's important to know that this communication is a one-directional data flow. It's not possible to communicate from the child component back to the parent components using props. Instead, developers use other approaches.

*In React, functional components are reusable blocks of code that act like a JavaScript function and you can pass data from one component to another using props. Which of the following statements are true? Select all that apply.*

* *When two components communicate with each other, the component sending the props data is known as the parent and the one receiving the data is known as the child.*
* *Props are like a JavaScript object.*
* *Props are passed using JSX syntax.*

Another important limitation has to do with something called pure functions. In programming, a pure function is a function that will always return the same outputs for the same argument values that are passed in.

## Using props in components

*Which of the following can be defined as a parameter within a functional component?*

* *Props*

## Introducing JSX

*True or false. You need to define how React should render a component. Can you use a regular JavaScript function to complete this action?*

* *True*

*Graphical user interface, text, application, email

Description automatically generated*

It's also important to remember that the HTML code must be wrapped in a top-level element, such as a div tag. If you do not want to add extra div elements to the DOM, you can use something called a fragment instead. This is like an HTML open and close tag without the tag name.

## Practical styling

Recall that there are three techniques to include CSS and HTML documents. The first is in line by utilizing the style attribute inside the HTML element. Next is internal by using the style elements in the head section. Finally, external by linking to an external CSS file using the link element.

*True or false? Using React, you can easily convert a CSS rule to a JavaScript object, where each key-value pair describes a CSS declaration.*

* *True*

## Embedded JSX expressions

Embedded expressions allow developers to insert the values of JavaScript variables into the HTML of the resulting React elements. Embedded expressions can also embed the outputs of functions.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

*You would like to use JSX to display a user’s location on a webpage, and you have defined the following JavaScript variable:*

*const location = Boston;*

*Which of the following lines of code uses the correct syntax for embedding this variable in an HTML paragraph element?*

* *const result = <p>{location}</p>;*

## Embedding in attributes

Text

Description automatically generated

*True or false? When using an img element in your React component, you can set the value of the src attribute using a JSX expression.*

* *True*

## sss