Welcome to Front-End Web Development with React: Additional Resources

PDFs of Presentations

0-Course-Overview.pdfPDF File
How to Learn.pdfPDF File

React Resources

- [React Site](#)

Coursera Resources

- [Coursera Learner Help](#)
- [Switching to a Different Session](#)

Full Stack Web Development: The Big Picture: Objectives and Outcomes

This lesson gives you a big picture view of the Full Stack Web Development. The lecture gives you an overview of full stack web development. At the end of this lesson, you will be able to:

- Understand what is meant by full stack in the context of web development
- Distinguish between front-end, back-end and full stack web development
- Understand the position of this course in the context of this specialization

Full Stack Web Development: Additional Resources

PDFs of Presentations

FSWD-BigPicture.pdfPDF File

Useful Links

- [What is a Full Stack developer?](#)
- [Wait, Wait… What is a Full-stack Web Developer After All?](#)
- [The Myth of the Full-stack Developer](#)
- [Multi-tier Architecture](#)
- [What is the 3-Tier Architecture?](#)

Setting up your Development Environment

Software Requirements

1. **Text editor of your choice**: Any text editor that you are already familiar with can be used for editing the project files. I will be using Visual Studio Code (https://code.visualstudio.com/) as the editor of choice in this specialization. You may also consider other editors such as Brackets (http://brackets.io/), Sublime Text (http://www.sublimetext.com/), or Atom (https://atom.io/).
2. **Browser of your choice**: You may use your preferred browser. I will be using Chrome as the browser in all the exercises. All the exercises and assignments in this course have been tested using Chrome v. 46. Please note that not all browsers may support all the HTML5 features to the same extent. You might encounter problems when using other browsers. I strongly urge you to use the latest Chrome browser for the exercises and assignments in this course so that any problems are minimized.
3. **Command line shell**: Familiarity with the command-line shell will be essential for the exercises. In Windows a cmd window or power shell with admin privileges would be needed. On a Mac or in Linux, a terminal window can be used. Please get familiar with the "sudo" command in OS X and Linux.
4. **Files required for the exercises**: We will provide additional starter files for the exercises wherever needed. Links to download the files will be provided inline in the **exercise instructions** that follow each exercise video. Please download the files provided there, if any, before beginning the exercise. The links are also available through the **Additional Resources** of the specific lesson.

Note: Please remember to retain the folders and all the files that you create in the exercises. Further exercises will build upon the files that you create in the preceding exercises. DO NOT DELETE the files at the end of the exercises, unless otherwise instructed. You may wish to set up your exercise folder as a Git repository and commit the files to the repository at the end of each exercise.

Setting up your Development Environment: Git and Node: Additional Resources

PDFs of Presentations

Git.pdfPDF File
Git-Exercises.pdfPDF File
NodeJS.pdfPDF File
Exercises-Node-NPM.pdfPDF File

Setting up your Development Environment: Git and Node: Additional Resources

PDFs of Presentations

Git.pdfPDF File
Git-Exercises.pdfPDF File
NodeJS.pdfPDF File
Exercises-Node-NPM.pdfPDF File

Additional Resources (Git)

- Git site http://git-scm.com.
- Installing Git chapter from Pro Git
- Git reference manual
- Quick reference guides: GitHub Cheat Sheet (PDF) | Visual Git Cheat Sheet (SVG | PNG)
- Atlassian comprehensive Git tutorial

Additional Resources (Node.js and NPM)

- Nodejs.org
- Npmjs.com
- Node API Documentation
- NPM Documentation
- lite-server

Introduction to React: Objectives and Outcomes

In this lesson you will be given a quick overview of JavaScript frameworks and libraries and then introduced to React in particular. We will learn some basics of React and how to configure a React application using the create-react-app the command line tool. At the end of this lesson, you will be able to:

- Get a basic overview of JavaScript frameworks and libraries
- Understand the architecture of an React application
- Scaffold out a starter React application using *create-react-app,* the command line tool

Introduction to React: Additional Resources

PDFs of Presentations

1-JavaScript-Frameworks.pdfPDF File
2-Intro-React.pdfPDF File
3-React-App-Overview.pdfPDF File
4-Intro-JSX.pdfPDF File

React Resources

- Reactjs.org
- create-react-app
- reactstrap
- reactstrap Navbar
- Introducing JSX
- Convert JSX using Online Babel Compiler

Definitions

- Framework
- Hollywood Principle
- Inversion of Control
- Imperative vs Declarative Programming
- Imperative vs Declarative

Blog Articles

- 5 Best JavaScript Frameworks in 2017
- Top JavaScript Frameworks & Topics to Learn in 2017
- Declarative vs. Imperative Programming for the Web
- Is React library or a framework?
- Is React a library or a framework and why?
- An Introduction to the React Framework

- [React is a framework](#)
- [Why isn't React called framework? What does it lack to be a framework?](#)

React Components: Objectives and Outcomes

In this lesson you will learn about React components and how we construct an React component and design its views. At the end of this lesson you will be able to:

- Create a React component
- Construct the React component code and the view for your component using JSX and JavaScript

React Components: Additional Resources

PDFs of Presentations

5-Component-Part1.pdfPDF File
6-Component-Part2.pdfPDF File
7-Lifecycle-Methods.pdfPDF File

React Resources

- [React Components](#)
- [React Component State](#)
- [React Component Props](#)
- [reactstrap Media Object](#)
- [React.Component and Lifecycle Methods](#)
- [reactstrap Card](#)
- [Bootstrap unstyled list](#)
- [React Dev Tools](#)
- [React Dev Tools Chrome Extension](#)

In this assignment you will add a new component to the React application to show the details of a selected dish. You will use the Card component and the Bootstrap unstyled list component to prepare the view for this new component.

Step-By-Step Assignment Instructions

Objectives and Outcomes

In this assignment, you will continue to work with the React application that you have been developing in the exercises. You will add a new component named *DishdetailComponent* that will display the details of a selected dish. You will then design the view for the component using the card component. At the end of this assignment, you should have completed the following tasks:

- Created a new DishdetailComponent and added it to your React application.

- Updated the view of the DishdetailComponent to display the details of the selected dish using an reactstrap card component.
- Updated the view of the DishdetailComponent to display the list of comments about the dish using the Bootstrap unstyled list component.

Assignment Requirements

This assignment requires you to complete the following tasks. Detailed instructions for each task are given below. The picture of the completed web page included below indicates the location within the web page that will be updated by the three tasks.

Task 1

In this task you will be adding a new ***DishdetailComponent*** to your React application and include the component into the menu component's view so that the details of a specific dish are displayed there:

- Replace the card showing the selected dish in MenuComponent's view with the DishdetailComponent, and make sure to pass the selected dish information as props to the DishdetailComponent.
- Create a new *DishDetail* class in a file named *DishdetailComponent.js* in the *components* folder
- Export the *DishDetail* class from this file so that it can be imported in *MenuComponent.js* and used to construct the view of the selected dish.
- Return a <div> from the render() function. This <div> should use the Bootstrap *row* class to position the content within the <div>. This div will display both the details of the dish in a Card and the list of comments side-by-side for medium to extra large screens, but will stack them for xs and sm screens.
- The card should be enclosed inside a <div> appropriate Bootstrap column classes so that it occupies the entire 12 columns for the xs and sm screen sizes, and 5 columns for md screens and above. Also apply a class of m-1 to this div.
- The comments should be enclosed in a <div> to which you apply appropriate column classes so that it occupies the entire 12 columns for the xs and sm screen sizes, and 5 columns for md screens and above. Also apply a class of m-1 to this div.
- If the dish is null then you should return an empty <div>

Task 2

In this task you will be adding a card component to the ***DishdetailComponent*** view to display the details of the dish given above:

- Implement a function named renderDish() that takes the dish as a parameter and returns the JSX code for laying out the details of the dish in a reactstrap Card. You have already seen this as part of the MenuComponent class in the exercise earlier.

- Display the name of the dish as the Card title, and the description as the Card text.

Task 3

In this task you will use the comments that are included in the dish object above to display a list of the comments for the dish. Please use your JavaScript knowledge to recall how you would access an inner

property in a JavaScript object that itself points to an array of JavaScript objects (comments). This task involves the following steps:

- Implement a function named renderComments() that takes the comments array as a parameter and lays out each comment as shown in the image below. You should use the Bootstrap list-unstyled class on the list.
- Each comment should be displayed on two lines, the first one showing the comment, and the second line showing the comment author's name and the date.
- The comments should contain a <h4> header with the word "Comments".
- Remember to enclose the header and comments inside a <div> before returning the JSX code. Otherwise React will not do the layout correctly.
- If the comments are null, then you should return an empty <div>.

Uthappizza

Zucchipakoda

Vadonut

ElaiCheese Cake

Ristorante Con Fusion

Uthappizza

Zucchipakoda

Review criteria

Upon completion of the assignment, your submission will be reviewed based on the following criteria:

Task 1:

- A new ***DishdetailComponent*** has been added to your React application.
- Included the DishDetail into your MenuComponent's view to show the selected dish.
- Passing the selected dish as props to the DishDetail Component.
- Used the appropriate Bootstrap classes to the card so that it occupies the entire row for xs and sm screen sizes, and 5 columns for md screens and above.
- Used the appropriate Bootstrap classes to the div containing the list of comments so that it occupies the entire row for xs and sm screen sizes, and 5 columns for md screens and above.

Task 2:

- Used the Card component to display the details of the dish.

Task 3:

- Included a list of comments about the dish into the dishdetail view.

Assignment 1: React Components: Additional Resources

Assignment 1 Screenshots

React Resources

- [React Components](#)
- [React Component State](#)
- [React Component Props](#)
- [reactstrap Media Object](#)
- [reactstrap Card](#)
- [Bootstrap unstyled list](#)
- [Bootstrap Grid](#)