IST 363

# Lab 10

## This Lab Covers

* React – mapping array items, keys, multiple components, responding to events and JavaScript filtering

## NOTE

1. **Submission:** Please submit screenshots of your code. Download the template.
2. **Web Server:** You will need to have your web server running to view the files you are working on in this lab. If it’s not running, open the folder we worked on in class this week in VS Code, then go to the terminal in VS Code and type npm run dev

## React 1

**A screen shot of a computer program

AI-generated content may be incorrect.Goal:** Today’s lab is a walkthrough of more React.

**Dealing with Lists**In this code, I am looping through a list. Instead of using a for loop, I’m using the JavaScript map function which works well for React. Here’s how my map function works:

* It iterates over a list with item as the loop index
* for each item in the list the function inside of the second set of {} does something -> in this case, it returns the item with an <li> tag around it

A white background with black text

AI-generated content may be incorrect.You will see the map function in other folks’ React code a lot. Here’s a screenshot of the output for this code.

1. A list of information on a white background

   AI-generated content may be incorrect.Using the map function and the following object, write code to loop the object and output the name, year, and major of each student. Your output should look exactly like the screen shot on the right.   
     
   Hint: don’t forget if you need to use any void tags you need to add the / like we saw in class.  
     
   Here's the object. You can copy and paste it into your code.  
     
   const students = [{suid: 123456, name: 'Sue Flay', year: 'senior', major: 'Applied Data Analytics'}, {suid: 234567, name: 'Ella Vader', year: 'junior', major: 'Information Management and Technology'}, {suid: 345678, name: 'Chris P Bacon', year: 'junior', major: 'Innovation, Society and Technology'}];  
     
   Take a screen shot of your code an paste it in the answer template.

**React Keys**  
If you right-click -> inspect -> console you will see an error about keys. Each element in a React list is supposed to have a unique key that helps React know when data is updated. Without it, React updates may become inefficient. In lists, keys are added to the list as an attribute, and need to be unique.

Here’s what a key would look like in a list element:  
  
<li key={some unique value goes here}>

1. In the students object, you have the perfect unique value to use as a key, SUID, Add it to the code! When it refreshes the console error message should disappear.  
     
   Take a screen shot of your code an paste it in the answer template.

**Adding Components**

A screen shot of a computer program

AI-generated content may be incorrect.Eventually our pages are going to have lots of components. But for now, let’s just have two so you can see how this works.

App is the main component. Any other component you define can be included using its name. Check out the syntax I’ve circled in red.

React applications have something called component hierarchies. That is, they will have many components related to each other almost like folders in a file system with parents, children and grandchildren.

Try to chunk your code in different components. You’ll get better at this with experience.

1. For this question, split up the code you wrote in #2 in your App component into two parts. Keep the <h1> line in your App component. Put all the code processing your object into a new component called Students. Make sure to add the code to render the students in the screen even though you moved it. Don’t forget that React components names use PascalCase.  
     
   Take a screen shot of your code an paste it in the answer template.

**Prep for Interactive Data Search**

React is great at reacting to changes so it’s equally great for searching through data in real time on a website page. To prep for interactive searching we need to learn how to handle events in React. Take a moment to look at the events handling info on W3Schools: <https://www.w3schools.com/react/react_events.asp>

1. For this problem write a button element. When the button is clicked a function should run that console logs a message that you pass to the function as a parameter. You can pick the message you want to send.  
     
   Take a screen shot of your code and paste it in the answer template.

**Prep for Interactive Search Filtering Data**

Part of searching is filtering through objects like students. Filtering is a straight JavaScript task. No funky React syntax for this one. Take a moment to look through the W3Schools info on filtering: <https://www.w3schools.com/jsref/jsref_filter.asp>. Then look at this GeeksforGeeks article about filtering objects: https://www.geeksforgeeks.org/how-to-implement-a-filter-for-objects-in-javascript/.

1. For this question we are going to do static filtering. Write code to filter by the name Sue Flay, so only Sue’s record should display.  
     
   Take a screen shot of your code and paste it in the answer template.

**Cliffhanger**

I’m aware I’ve left you hanging as we didn’t implement a dynamic search where you can type in the text and the search happens in real time. We’ll need to learn a few more React concepts before we can do that. Meet me next week in lecture for dynamic searching.

1. Answer the following questions. You may not use artificial intelligence.
2. What are React components, and why are they important?
3. Why do React components need to start with an uppercase letter?
4. What is JSX, and how does it differ from HTML?
5. Why do we use className instead of class in JSX?
6. How do you dynamically insert JavaScript expressions in JSX?
7. What are the benefits of breaking a large React component into smaller components?
8. When should you consider refactoring a React component into multiple components?
9. Can you reuse the same component multiple times in a React app? How?
10. How does React handle form elements like inputs in JSX?

**The End**