Module 7 - Description

Tested on both the systems below-Macbook Pro (13 inch, 2017, Two Thunderbolt 3 ports) Operating system: mac Catalina (10.15.7) Processor: 2.3 GHz Dual-Core Intel Core i5 Memory 8GB 2133 MHz LPDDR3 Graphics Intel Iris Plus Graphics 540 1536 MB Safari Version 14.0 Terminal 2.1.0

Operating system: Windows 10 home (20H2) – created on this system

OS build:19042.662

Experience: Windows feature experience pack 120.2212.551.0

HP ENVY X360 convertible 15-DS1XXX

Processor AMD Rizen 5 4500U with Radeon graphics 2.38GHz

Memory 16GB

Browser version: Microsoft edge (87.0.664.57)

Tested using windows CMD terminal

TypeScript – 4.1.2

Node.js version (v14.15.1)

Compilation of .jsx files to .js files depends on having done beforehand

- 1. npm install --save-dev @babel/core @babel/cli
- 2. npm install --save-dev @babel/preset-env
- 3. npm install @babel/preset-react --save-dev
- 4. npm install --save-dev @babel/plugin-proposal-class-properties

There must be a .babelrc file with atleast following contents:

```
{
"presets": [ "@babel/preset-react" ],
"plugins": [ "@babel/plugin-proposal-class-properties" ]
}
```

To compile mainEdited.jsx file to mainEdited.js file type command **npx babel mainEdited.jsx** -- **out-file mainEdited.js**

React file: mainEdited.js

All the react code is written in this file. Its main purpose is to keep a **src** of an image as state, and to cycle through a pre-determined list of images, as we press the View Course Images button. This emulates showing a portfolio of course images to clients.

Main parts:

We declare a button which cycles through image numbers 0 through 1 (configurable) and updates the image name to load based on the number for e.g. "/images/image1.jpg". This is a React component.

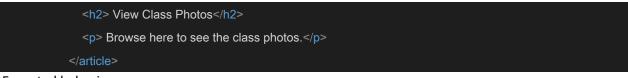
```
img_source = this.props.src;
10
            imagenum = 0;
11
            titleStyle = { color : "black" }
12
            render() {
13
              return (
14
                <div>
15
                  <button onClick={() => {
                    this.imagenum = this.imagenum + 1;
17
                    if (this.imagenum == 2){
18
                      this.imagenum = 0;
19
                    }
20
                    this.img_source = "./images/image" + this.imagenum + ".jpg";
21
                    this.setState({});
22
                  }}>
23
                  Next image
                  </button>
```

Next, we display an image whose src is a state variable and clicking the button changes the image src and causes it to redisplay. We bind this component to id="mainContainer" in html file.

The mainEdited.jsx compiles into a mainEdited.js file which is loaded by the mainpage.html file through the following code.

```
<body onLoad= "start();">
```

```
<article>
<article>
<div class= ".figureelement" id="mainContainer">
</div>
```



Expected behaviour:

1. On opening mainpage.html with a browser, it shows the main courses page and presents a button at the bottom left allowing the user to click and browse through course photos.

React features used:

- **1.** We are maintaining state of the React component using state variables of src and imagenumber. We also update the state on button click.
- 2. We only update part of the html page on button click, not the whole UI, which makes It very swift.
- **3.** We are passing props/properties to React components to set their values with HTML elements. In our case this is title, src.
- 4. We are compiling JSX into JS using babel and simplifying React development.

Screenshot of mainpage.html

