Tutorial week 4. React

Tutorial 1:

1. Install NodeJS
2. Open Terminal (or Gitbash if you are in Windows. Install Gitbash) to test npm command
3. Install **create-react-app** by running the following command:

   sudo npm install -g create-react-app
4. Create a new project:

create-react-app myapp
cd myapp
npm start

5. React will open a browser at localhost:3000. You should see your running project

Tutorial 2: Playing with App.js

1. Open myapp project in Visual Studio Code
2. Observe the App.js file. Change it and see the effect.

   For example: <h1>Hello world</h1>

Tutorial 3: Write your first functional component

Write a simple function

```
function Hello(){
 return <span>Hello</span>
}
```

Change the function App()

```
function App() {
 return (
   <div>
```

```
   <Hello/>
   </div>
 );
}
```

Exercises:

- Write a component Header containing one h1
- Write a component Jumbotron containing div, h1, and p
- Write a component Login containing a login form
- Write a component Nav containing 2 hyperlinks

Tutorial 4a: Write a component with props

```
function Welcome(props) {
 return <h1>Hello, {props.name} {props.age}</h1>;
}
```

```
function App() {
 return (
   <div>
   <Welcome name="Thanh" age=”30”/>
   </div>
 );
}
```

Alternatively, we can use this

```
function Header({title, content}){
 return <h1>This is a header {title} {content}</h1>
}
```

```
function App() {
 return (
   <div>
   <Header title="Thanh" content="Test"/>
   </div>
```

```
  );
}
```

Tutorial 4b: Write a component with useState() and useEffect() hook

```
import React, { useState, useEffect } from 'react';

export default function Example() {
  const [count, setCount] = useState(0);


 // Similar to componentDidMount and componentDidUpdate:
 useEffect(() => {
   // Update the document title using the browser API
   document.title = `You clicked ${count} times`;
 });

  return (
    <div>
      <p>You clicked {count} times</p>
      <button onClick={() => setCount(count + 1)}>
        Click me
      </button>
    </div>
  );
}
```

Exercises:

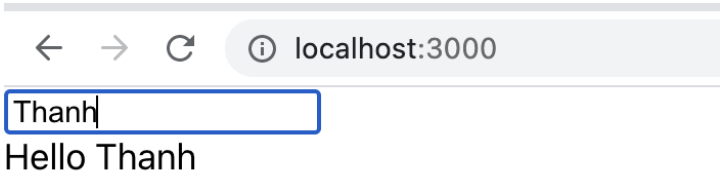
1. Write a component with a button named Decrease. Everytime, users click that button, reduce the count by 1
2. Write a component with a button named Exponential. Everytime, users click that button, increase the count by power of 2

Tutorial 5: Component with forms

Write a component named Hello:

```
function Hello(){
```

```
 const [name, setName] = useState('')
 return (
 <div>
   <input type="text" onChange={e=>setName(e.target.value)}/>
   <div>Hello {name}</div>
 </div>
 )
}
```



Exercises:

1. Write a component named Login with username and password. When users click Login button alert a message containing user and password
2. Write a component named Register with username, password, and retype password. When users click the register button, check if the password is matched. If yes, alert the username and password.

Tutorial 5: Build custom Hooks

This is a custom hook to check if Internet is on or off

```
import {useState, useEffect} from 'react'
export default function useOnline () {
   const [online, setOnline] = useState(navigator.onLine)

   useEffect (() => {
       window.addEventListener('online', function () {
          setOnline(true)
       });

       window.addEventListener('offline', function () {
```

```
          setOnline(false)
        });
    }, [])

    return online
}
```

```
import useOnline from './useOnline'
function App() {
 const online = useOnline()
 return (
    <div>
      {online? "Online": "Offline"}
    </div>
 );
}
```

Exercises:

- **Write a custom hook that displays a timer. It will change the value every 1 second. (Credit to one student in FWC)**
- Try to write a hook for one of the exercises/example above

Tutorial 6: Fetch data from API

We can fetch data from a REST API and display in a React component

```
fetch(endPoint)
.then(res=>res.json())
.then(data=>setData())
```

Use map() function to display data from the fetch()

```
data.map(item=>{
      <li>{item.id} | {item.name}<li>
})
```

Full code for the List.js

```
import { useEffect, useState } from "react"



export default function List(){

   const endPoint = "https://2u7k88arxk.execute-api.us-east-2.amazonaws.com/items"
   const [data, setData] = useState([])

//get data from api
const load = () => {
   fetch(endPoint)
     .then(response => response.json())
     .then(data =>
       {
           console.log(data.Items)
           setData(data.Items)
       }
       );
 }
 //load data automatically
 useEffect(() => {
    load()
  }, [])

   return (
       <div>
           {data.map(item => <li>{item.name}</li>)}
       </div>


   )
}
```

Full code for App.js

```
import logo from './logo.svg';
import './App.css';
import List from './List';
```

```
function App() {

return (
  <div>
     <List/>
  </div>
);
}



export default App;
```

Tutorial 7: Ternary Operator

This operator is very common in ReactJS. Try to master it as it is very handy later.
Define a constant:

```
const [online, setOnline] = useState(true)
```

Use ternary operator ?:

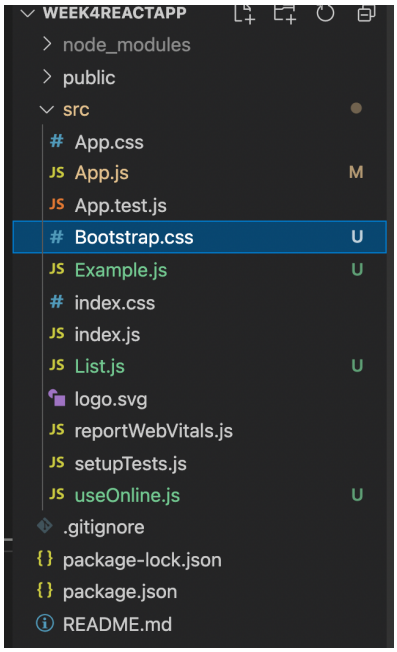
```
{online? “You are online”: “You are offline”}
```

Extra Tutorials:

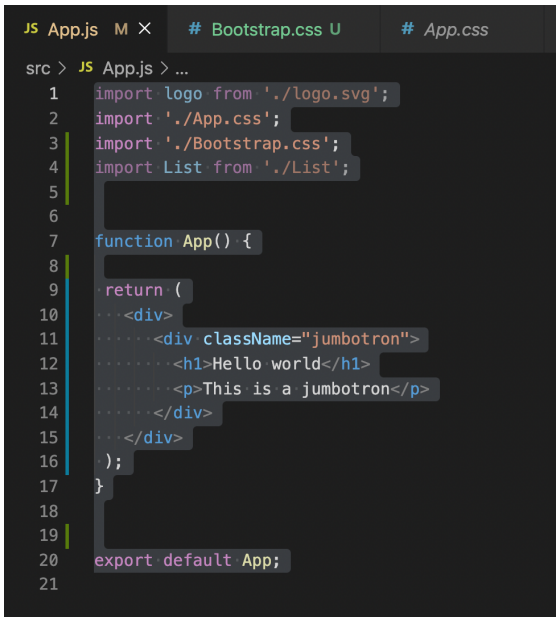
Some students ask me how to use Bootstrap in React. There are plenty of ways to do so but I prefer this following way:

1. Open Bootstrap CSS file and copy it content or download Bootstrap CSS file and place it into your project folder, i.e. next to App.css

https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css



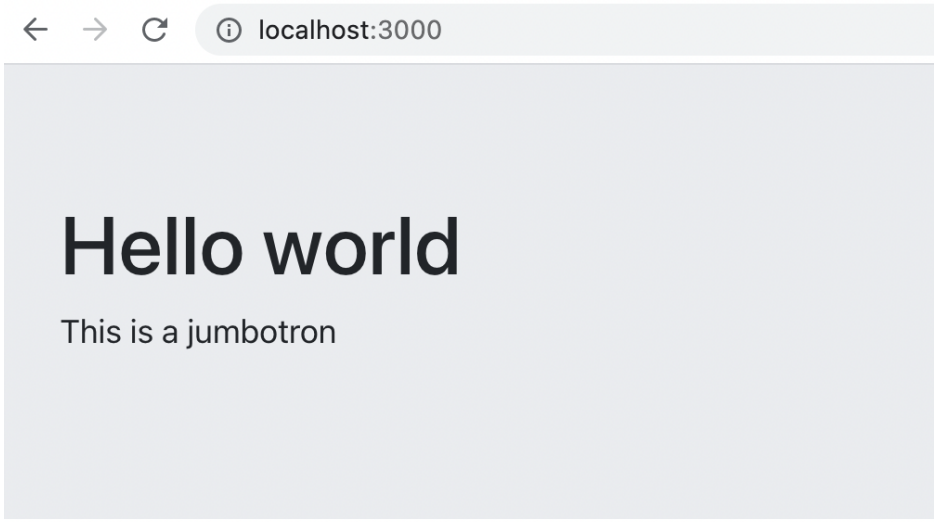
2. Import it into React component as you use with App.css



3. Now use it as usual but make sure you use **className** instead of class

```
import logo from './logo.svg';
import './App.css';
import './Bootstrap.css';
```

```
import List from './List';



function App() {

return (
  <div>
     <div className="jumbotron">
       <h1>Hello world</h1>
       <p>This is a jumbotron</p>
     </div>
  </div>
);
}



export default App;
```



Install bootstrap by npm command (Credit to Minh Vo for his suggestion)

Alternatively, you can install bootstrap by running npm command

npm install bootstrap

Import and use bootstrap as follows:

```
import logo from './logo.svg';
import './App.css';
import List from './List';
import 'bootstrap/dist/css/bootstrap.min.css';

function App() {

return (
  <div>
     <div className="jumbotron">
       <h1>Hello world</h1>
       <p>This is a jumbotron</p>
     </div>

     <button className="btn btn-primary">Click me</button>
  </div>
);
}


export default App;
```