CS Schools Experience

Web Development Workshop Handout

Prerequisites

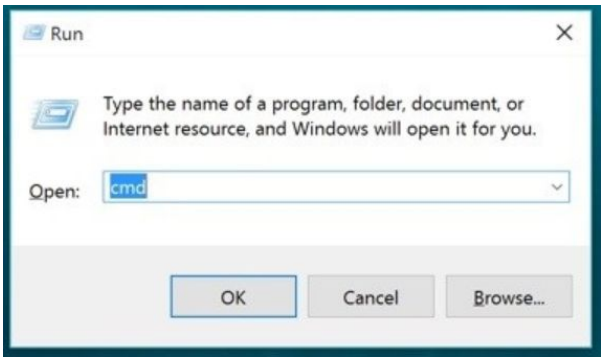
Editor- Visual studio code: https://code.visualstudio.com/
Package Manager - NodeJS: https://nodejs.org/en/
Version Control - Git: https://git-scm.com/downloads
Useful link: https://reactjs.org/

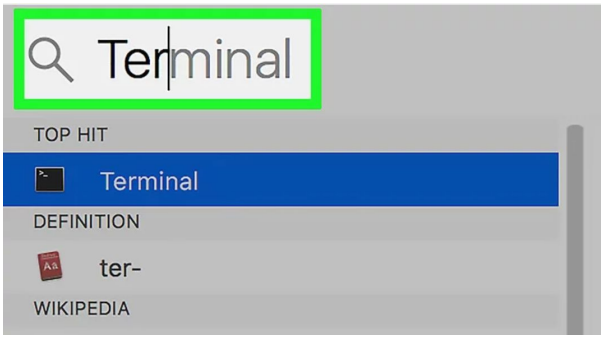
1. The Beginning (Kick-start)

Opening your command prompt:

Windows: Press **Windows+R** to open the “Run” box. Type “**cmd**” and then click “**OK**” to open a regular Command Prompt.

Mac os: Use spotlight by pressing `⌘ Command+Space`





Next, in your command prompt do:

Step 1: Install create-react-app

Type or Copy+Paste command

```
C:\Users\username> npm install -g create-react-app
```

Step 2: Create project

```
C:\Users\username> cd Desktop
C:\Users\username\Desktop> create-react-app myfirstweb
```

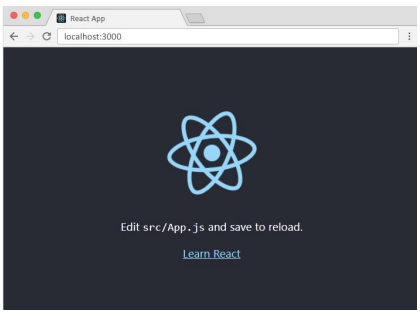
Step 3: Move into your `myfirstweb` project (folder) directory

```
C:\Users\username\desktop> cd myfirstweb
```

Step 4: Start react

```
C:\Users\username\desktop\myfirstweb> npm start
```

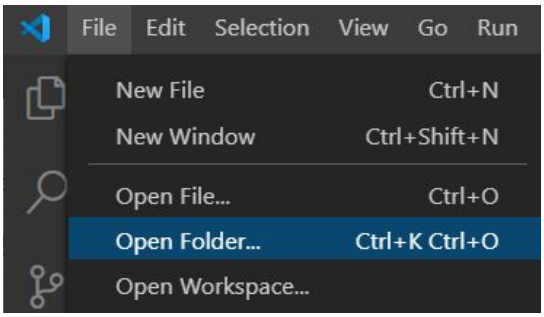
Step 5: You should get this open up in your browser



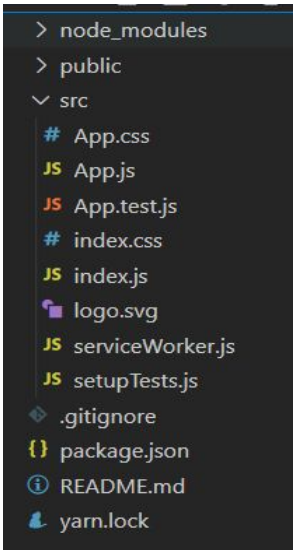
2. Editing your project

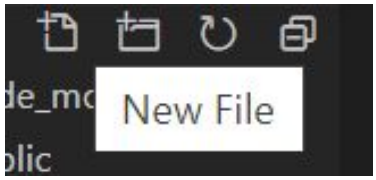
Open your Visual Studio Code and go to File, select Open Folder and go to the project you created. In this case you are going to select the “myfirstweb” folder.

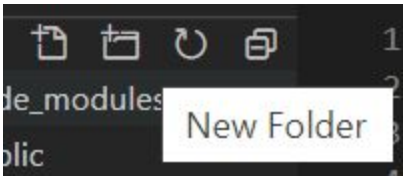
-Go to: **Visual Studio Code -> File -> Open Folder -> Go To Desktop -> Select myfirstweb**



After the project is opened, expand the src (refer as source folder). In general you will have folder structure as shown in the picture below. Your folder structure should be similar. Referring to the second and third picture, starting from the left, the first icon indicates “Create New File” while the second icon indicates “Create New Folder”.







Open App.js file by double clicking, and the code below will be shown in your editor. App.js is the entry point of the project and this file is currently what you are seeing in the browser. Let's have a clean project and we will start building on top of it. We will have to delete everything and leave the code to the state that is shown in the second picture.

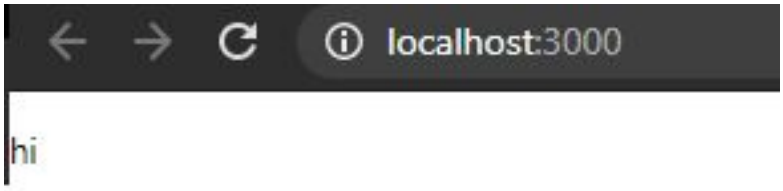
Before:

After:

App.js

```
import React from 'react';



function App() {
  return (
    <div className="App">
     <p>hi</p>
    </div>
  );
}

export default App;
```

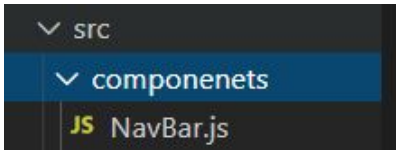
A “hi” can be seen on the webpage. It would be tedious to write everything (long complex code) in a single file as it would be hard to manage and refactor, therefore, we would separate it into small and isolated pieces of codes, namely, **components**.



In our src folder, let’s create a new folder and name it “components”. We will store the files that we are going to build in this folder.

3. Creating Navigation Bar (React-Bootstrap)

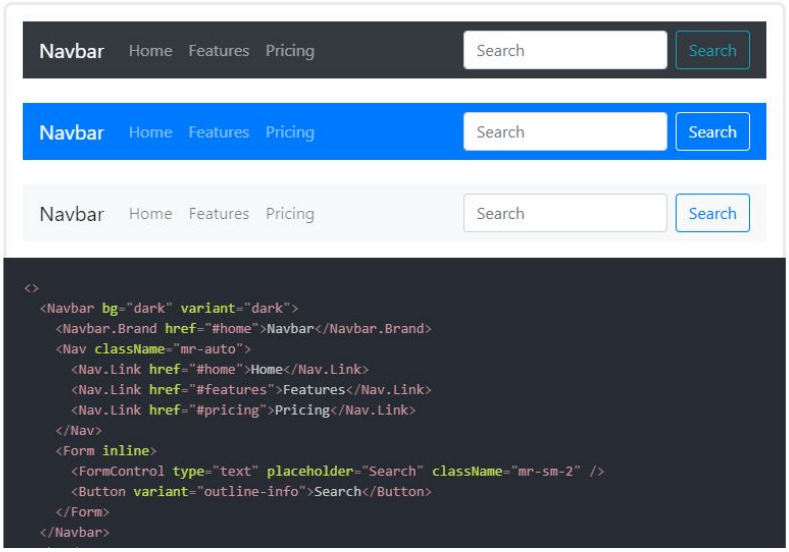
Let's create a navigation bar as our header. To do this, let's create a NavBar.js in our components folder. We will create a class component.



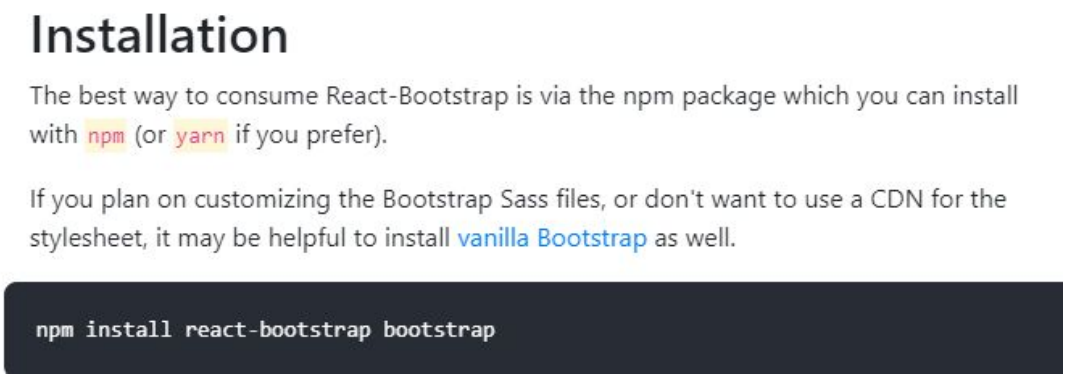
Code for creating a basic class component skeleton:

```
import React, { Component } from 'react'

export default class NavBar extends Component {
    render() {
        return (
            <div>

            </div>
        )
    }
}
```

It would be a hassle to work on CSS now, let's use the BOOTSTRAP. Go to this website https://react-bootstrap.github.io/components/navbar/ , we would select the first navbar as shown in the picture below.



In order to use react-bootstrap, there 3 essential principles that need to be done, let's follow the documentation, we will first need to go to https://react-bootstrap.github.io/getting-started/introduction/ to check the guidelines, then we will copy and paste the command `npm install react-bootstrap bootstrap` into our terminal in Visual Studio Code. To open the terminal in your visual studio code, use Ctrl + ~ and the terminal will pop out. Press enter to install it.



Installation

The best way to consume React-Bootstrap is via the npm package which you can install with `npm` (or `yarn` if you prefer).

If you plan on customizing the Bootstrap Sass files, or don't want to use a CDN for the stylesheet, it may be helpful to install vanilla Bootstrap as well.

```
npm install react-bootstrap bootstrap
```

Copy the slice of code that is needed and paste the code in between the tag of NavBar.js.

```
        <Navbar bg="dark" variant="dark">
                    <Navbar.Brand href="#home">Navbar</Navbar.Brand>
                    <Nav className="mr-auto">
                    <Nav.Link href="#home">Home</Nav.Link>
                    <Nav.Link href="#features">Features</Nav.Link>
                    <Nav.Link href="#pricing">Pricing</Nav.Link>
                    </Nav>
                    <Form inline>
                    <FormControl type="text" placeholder="Search"
className="mr-sm-2" />
                    <Button variant="outline-info">Search</Button>
                    </Form>
                </Navbar>
```

Let's import the Components:

```
import { Navbar, Form, Button, Nav, FormControl} from 'react-bootstrap';
```

Now, the code should look like this:

```
import React, { Component } from 'react'
import { Navbar, Form, Button, Nav, FormControl} from 'react-bootstrap';

export default class NavBar extends Component {
    render() {
        return (
            <div>

                <Navbar bg="dark" variant="dark">
                    <Navbar.Brand href="#home">Navbar</Navbar.Brand>
                    <Nav className="mr-auto">
                    <Nav.Link href="#home">Home</Nav.Link>
                    <Nav.Link href="#features">Features</Nav.Link>
                    <Nav.Link href="#pricing">Pricing</Nav.Link>
                    </Nav>
                    <Form inline>
                    <FormControl type="text" placeholder="Search"
className="mr-sm-2" />
                    <Button variant="outline-info">Search</Button>
                    </Form>
                </Navbar>

            </div>
        )
    }
}
```

Then, import the NavBar component into App.js which is our entry point. The NavBar (Navigation Bar) should be visible on the browser now.

```
import React from 'react';
import NavBar from './components/NavBar';
```

```
function App() {
  return (
    <div className="App">
      <NavBar />
    </div>
  );
}

export default App;
```

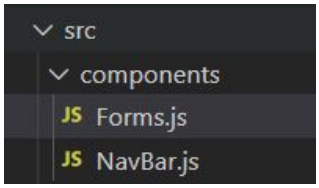
You might realize that the components are different from the Bootstrap website that we extracted the code from, as we are missing a step: paste this `import'bootstrap/dist/css/bootstrap.min.css';` into our App.js. After doing this, you should be able to view the similar NavigationBar.

CSS

```
{/* The following line can be included in your src/index.js or App.js file*/}

import 'bootstrap/dist/css/bootstrap.min.css';
```

4. Creating Form & Card (Ant-Design)

In the same src (source folder), create a new file. Name it Forms.js



We will start by creating a class component.
Forms.js

```
import React, { Component } from 'react'

export default class Forms extends Component {
    render() {
        return (
            <div>

            </div>
        )
    }
}
```

Delete the NavBar in App.js

```
import React from 'react';

function App() {
  return (
    <div className="App">
    </div>
  );
}

export default App;
```

And import into the Forms.js

```
import React, { Component } from 'react';
import NavBar from './components/NavBar';

export default class Forms extends Component {
    render() {
        return (
            <div>

                <NavBar />

            </div>
        )
    }
}
```

Next, we are going to use a different library, called ant-design https://ant.design/docs/react/introduce , remember the 3 principles. First, install it using terminal `npm install antd` . Second, import the style `import 'antd/dist/antd.css';` into App.js. Third, import the components into the file that we are working on.

App.js - Starting from this

```
import React, { Component } from 'react'
import 'bootstrap/dist/css/bootstrap.min.css';
 import 'antd/dist/antd.css';

export default class Forms extends Component {
    render() {
        return (
            <div>

            </div>
        )
    }
}
```

We are going to use the Form in here https://ant.design/components/form/

Forms.js

```
import React, { Component } from 'react';
import NavBar from './NavBar';
import { Form, Input, InputNumber, Button } from 'antd';

export default class Forms extends Component {

    render() {
        return (
            <div>

                <NavBar />

                <div>

                <Form>
                    <Form.Item
                        name={['name']}
                        label="Name"
                        rules={[
                        {
                            required: true,
                        },
                        ]}
                    >
                        <Input />
                    </Form.Item>
                    <Form.Item
                        name={['email']}
                        label="Email"
                        rules={[
                        {
                            type: 'email',
                        },
                        ]}
                    >
                        <Input />
                    </Form.Item>
```

```
                    <Form.Item
                        name={['age']}
                        label="Age"
                        rules={[
                        {
                            type: 'number',
                            min: 0,
                            max: 99,
                        },
                        ]}
                    >
                        <InputNumber />
                    </Form.Item>

                    <Form.Item name={['introduction']}
label="Introduction">
                        <Input.TextArea />
                    </Form.Item>

                    <Form.Item>
                        <Button type="primary" htmlType="submit">
                        Submit
                        </Button>
                    </Form.Item>

                </Form>

                </div>

            </div>
        )
    }
}
```

Let’s add some customized messages to the input validation.

Forms.js

```
import React, { Component } from 'react';
import NavBar from './NavBar';
import { Form, Input, InputNumber, Button } from 'antd';

export default class Forms extends Component {

    render() {

        const layout = {
            labelCol: {
              span: 8,
            },
            wrapperCol: {
              span: 10,
            },
          };

        const validateMessages = {
            required: '${label} is required!',
            types: {
              email: '${label} is not validate email!',
              number: '${label} is not a validate number!',
            },
            number: {
              range: '${label} must be between ${min} and ${max}',
            },
          };

        return (
            <div>

                <NavBar />

                <div>

              <Form {...layout} validateMessages={validateMessages}>
                    <Form.Item
```

```
                        name={['name']}
                        label="Name"
                        rules={[
                        {
                            required: true,
                        },
                        ]}
                    >
                        <Input />
                    </Form.Item>
                    <Form.Item
                        name={['email']}
                        label="Email"
                        rules={[
                        {
                            type: 'email',
                        },
                        ]}
                    >
                        <Input />
                    </Form.Item>
                    <Form.Item
                        name={['age']}
                        label="Age"
                        rules={[
                        {
                            type: 'number',
                            min: 0,
                            max: 99,
                        },
                        ]}
                    >
                        <InputNumber />
                    </Form.Item>

                    <Form.Item name={['introduction']}
label="Introduction">
                        <Input.TextArea />
                    </Form.Item>
```

```
                    <Form.Item wrapperCol={{ ...layout.wrapperCol, offset:
10 }}>
                        <Button type="primary" htmlType="submit">
                        Submit
                        </Button>
                    </Form.Item>

                </Form>

                </div>

            </div>
        )
    }
}
```

App.js

```
import React from 'react';
import'bootstrap/dist/css/bootstrap.min.css';
import 'antd/dist/antd.css'
import Forms from './components/Forms';

function App() {
  return (
    <div className="App">
      <Forms />
    </div>
  );
}

export default App;
```

Adding inline css in Forms.js

Search the after the <NavBar /> and before the <Form>, paste this part of code:

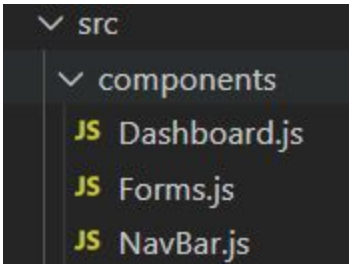
```
                <div style={{marginTop: '10%'}}>
```

Scroll to bottom and look for <Button> tag, paste this part of code:

```
<Button type="primary" htmlType="submit" style={{width: '60%', marginTop:
'5%',backgroundColor:'#8357c5',border: 'none'}}>
```

Card

Now we are going to create our Profile. Create a new file and name it Dashboard.js or Profile.js (personal preference).Create a class component.



Dashboard.js

```
import React, { Component } from 'react'

export default class Dashboard extends Component {
    render() {
        return (
            <div>

            </div>
        )
    }
}
```

Dashboard.js

```
import React, { Component } from 'react'

export default class Dashboard extends Component {
    render() {
        return (
            <div>

                <p> hi from dashboard</p>
```

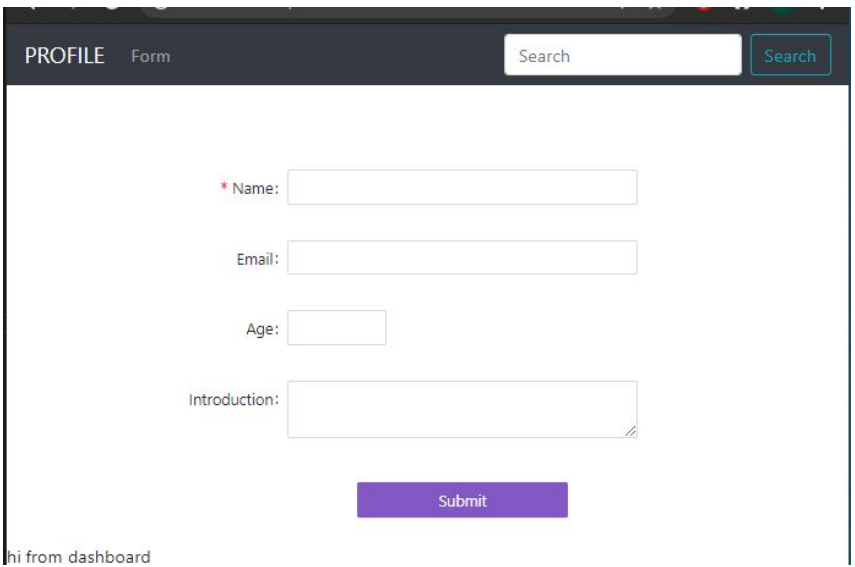
```
            </div>
        )
    }
}
```

Also, remember to import Dashboard.js into App.js for it to be displayed on the browser.

App.js

```
import React from 'react';
import Forms from './components/Forms';
import Dashboard from './components/Dashboard';
import 'antd/dist/antd.css';
import 'bootstrap/dist/css/bootstrap.min.css';

function App() {
  return (
    <div>

        <Forms />

        <Dashboard />

    </div>
  );
}

export default App;
```

Your browser should look like this:



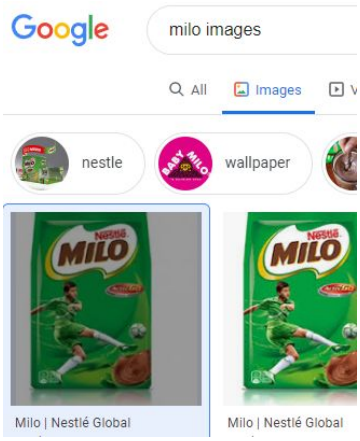
Now we are going to use the Card Components from ant-design. Firstly, let's import that.

Dashboard.js

```
import React, { Component } from 'react';
import { Card } from 'antd';

const { Meta } = Card;

export default class Dashboard extends Component {
    render() {

        return (

            <div>

                <Card>

                    <Card
                    style={{ marginTop: 16 }}
                    type="inner"
                    title="Name"
                    extra={<a href="#">Edit</a>}
                    >
```

```
                        <Meta
                            description="This is the description"
                        />

                        <p>Inner Card content</p>

                    </Card>

                </Card>

            </div>
        )
    }
}
```

Adding images (preferably in jpg/png format) into our project.
Go to the browser and download some images that you like, these images would be used as product photos.



Now, import this image using the image tag , note that the file extension should be entered as well as it is not a javascript file.
In javascript, file extension is not needed.

Dashboard.js

```
import React, { Component } from 'react';
import { Card } from 'antd';
```

```
import ProfilePic from '../images/milo.png';
```

Dashboard.js

```
    <img src={ProfilePic} alt="Profile" />
```

We would now add a NavigationBar to this page. So let's import that to our file.

Dashboard.js

```
import React, { Component } from 'react';
import { Card } from 'antd';
import ProfilePic from '../images/milo.png';
import NavBar from './NavBar';

const { Meta } = Card;

export default class Dashboard extends Component {
    render() {

        return (

            <div>

                <NavBar />

                <div>
                <Card>

                    <img src={ProfilePic} alt="Profile" />

                    <Card
                    style={{ marginTop: 16 }}
                    type="inner"
                    title="Name"
                    extra={<a href="#">Edit</a>}
                    >

                        <Meta
                            description="This is the description"
                        />
```

```
                        <p>Inner Card content</p>

                    </Card>

                </Card>
                </div>

        </div>
        )
    }
}
```

5. Linking Pages (React-Router)

Most of the websites have multiple pages..
Let's create a multi-page website by deploying React-router https://reactrouter.com/

To install we are going to use this command.

```
npm install react-router-dom
```

After finishing installing, go to App.js to setup the router.
Firstly, we are going to import the components

App.js

```
import React from 'react';
import Forms from './components/Forms';
import Dashboard from './components/Dashboard';
import { BrowserRouter, Switch, Route } from 'react-router-dom';
import 'antd/dist/antd.css';
import 'bootstrap/dist/css/bootstrap.min.css';
```

We will place everything in BrowserRouter. Next, add Route and the component we want it to redirect to. So in the first case it is the Dashboard, and second will be the Forms. The other two components we imported previously can be deleted.

App.js

```
import React from 'react';
import Forms from './components/Forms';
import Dashboard from './components/Dashboard';
import { BrowserRouter, Switch, Route } from 'react-router-dom';
import 'antd/dist/antd.css';
import 'bootstrap/dist/css/bootstrap.min.css';

function App() {
  return (
    <div>

          <BrowserRouter>
              <Switch>

              <Route exact path="/" component={Dashboard}/>

              <Route exact path="/froms" component={Forms}/>

              </Switch>
          </BrowserRouter>

    </div>
  );
}

export default App;
```

Now, Router has been set up successfully. We will link it to NavigationBar. Let's go to NavBar.js.

Import the components that we need and this time, we need to import NavLink.

NavBar.js

```
import React, { Component } from 'react';
import { Navbar, Form, Button, Nav, FormControl } from 'react-bootstrap';
import { NavLink } from 'react-router-dom';
```

Change line 11 and line 13 to Navlink Tag and add some styles along it.
NavBar.js

```
import React, { Component } from 'react';
import { Navbar, Form, Button, Nav, FormControl } from 'react-bootstrap';
import { NavLink } from 'react-router-dom';

export default class NavBar extends Component {
    render() {
        return (
            <div>

                <Navbar bg="dark" variant="dark">
                    <NavLink style={{color:"white", marginRight: "3%",
fontSize:"18px"}} to="/">PROFILE</NavLink>
                    <Nav className="mr-auto">
                    <NavLink style={{color:"white"}}
to="/forms">Forms</NavLink>
                    </Nav>
                    <Form inline>
                    <FormControl type="text" placeholder="Search"
className="mr-sm-2" />
                    <Button variant="outline-info">Search</Button>
                    </Form>
                </Navbar>

            </div>
        )
    }
}
```

Now Ctrl+S to save it and we are all set! Load the browser and you should be able to navigate through pages!

Let's make some changes to the card in Dashborad.js. Users should be directed to Forms.js page when "edit" is clicked.

Original Code

```
22                  <Card
23                  style={{ marginTop: 16 }}
24                  type="inner"
25                  title="Name"
26                  extra={<a href="#">Edit</a>}
27                  >
28
```

Apply some changes:
Dashboard.js , line 26

```
extra={<a href="/forms">Edit</a>}
```

Let's make some changes to Forms.js to direct users to the Profile page after clicking the "Submit" button. Having user to click on the Nagivation Bar manually could be frustrating.

Import the components <Link/>

Forms.js

```
import React, { Component } from 'react';
import {Link} from 'react-router-dom'
import NavBar from './NavBar';
import { Form, Input, InputNumber, Button } from 'antd';

export default class Forms extends Component {

    render() {

        const layout = {
            labelCol: {
              span: 8,
            },
            wrapperCol: {
```

```
            span: 10,
          },
        };

      const validateMessages = {
          required: '${label} is required!',
          types: {
            email: '${label} is not validate email!',
            number: '${label} is not a validate number!',
          },
          number: {
            range: '${label} must be between ${min} and ${max}',
          },
      };

      return (
          <div>

              <NavBar />

              <div style={{marginTop: '10%'}}>

            <Form {...layout} validateMessages={validateMessages}>
                  <Form.Item
                      name={['name']}
                      label="Name"
                      rules={
                          [
                              {
                                  required: true,
                              },
                          ]
                      }
                  >
                      <Input
                          onChange={ e => {this.setState({ name:
e.target.value }) }}
                      />
                  </Form.Item>
```

```
                    <Form.Item
                        name={['email']}
                        label="Email"
                        rules={
                            [
                                {
                                    type: 'email',
                                },
                            ]
                        }
                    >
                        <Input
                            onChange={ e => {this.setState({ email:
e.target.value }) }}
                        />
                    </Form.Item>

                    <Form.Item
                        name={['age']}
                        label="Age"
                        rules={[
                        {
                            type: 'number',
                            min: 0,
                            max: 99,
                        },
                        ]}
                    >
                        <InputNumber
                            onChange={ value => {this.setState({ age:
value }) }}
                        />
                    </Form.Item>

                    <Form.Item name={['introduction']}
label="Introduction">
                        <Input.TextArea
                            onChange={ e => {this.setState({ intro:
e.target.value }) }}
                        />
```

```
                    </Form.Item>

                    <Form.Item wrapperCol={{ ...layout.wrapperCol, offset:
10 }}>
                            <Button
                                type="primary"
                                htmlType="submit"
                                style={{width: '60%', marginTop:
'5%',backgroundColor:'#8357c5',border: 'none'}}
                            >
                                <Link to = {{ pathname:'/' }}>
                                    Submit
                                </Link>
                            </Button>

                    </Form.Item>

                </Form>

                </div>

            </div>
        )
    }
}
```

6. State & Props

First, let's go to our form. Let's create some State.

State is an object that is mutable and can be changed. It acts like a variable to store your data. Bear in mind that you need to define your state before the render.
We have 4 data we need to store which is Name, EmailAddress, Age and Introduction. Let's assign an empty string to name, email and intro and assign an integer to age.

Forms.js

```
export default class Forms extends Component {

    state = {
        name: '',
        email: '',
        age: 0,
        intro: '',
    }

    render() {
```

In <Input.> Tag, Let's handle the userInput by using onChange function. onChange function updates the corresponding state which is your data, any changes to the userInput in the form will be stored in your state.

```
<Input  onChange={ e => {this.setState({ name: e.target.value }) }} />

<Input onChange={ e => {this.setState({ email: e.target.value }) }}/>

<InputNumber  onChange={ value => {this.setState({ age: value }) }} />

<Input.TextArea onChange={ e => {this.setState({ intro: e.target.value })
}} />
```

In this stage, our Forms.js will look like this.

Forms.js

```
import React, { Component } from 'react';
import {Link} from 'react-router-dom'
import NavBar from './NavBar';
import { Form, Input, InputNumber, Button } from 'antd';

export default class Forms extends Component {

    state = {
        name: '',
        email: '',
        age: 0,
        intro: '',
    }

    render() {

        const layout = {
            labelCol: {
              span: 8,
            },
            wrapperCol: {
              span: 10,
            },
          };

        const validateMessages = {
            required: '${label} is required!',
            types: {
              email: '${label} is not validate email!',
              number: '${label} is not a validate number!',
            },
            number: {
              range: '${label} must be between ${min} and ${max}',
            },
        };

        return (
            <div>

                <NavBar />
```

```
                <div style={{marginTop: '10%'}}>

              <Form {...layout} validateMessages={validateMessages}>
                    <Form.Item
                        name={['name']}
                        label="Name"
                        rules={
                            [
                                {
                                    required: true,
                                },
                            ]
                        }
                    >
                        <Input
                            onChange={ e => {this.setState({ name:
e.target.value }) }}
                        />
                    </Form.Item>

                    <Form.Item
                        name={['email']}
                        label="Email"
                        rules={
                            [
                                {
                                    type: 'email',
                                },
                            ]
                        }
                    >
                        <Input
                            onChange={ e => {this.setState({ email:
e.target.value }) }}
                        />
                    </Form.Item>

                    <Form.Item
                        name={['age']}
```

```
                        label="Age"
                        rules={[
                        {
                            type: 'number',
                            min: 0,
                            max: 99,
                        },
                        ]}
                    >
                        <InputNumber
                            onChange={ value => {this.setState({ age:
value }) }}
                        />
                    </Form.Item>

                    <Form.Item name={['introduction']}
label="Introduction">
                        <Input.TextArea
                            onChange={ e => {this.setState({ intro:
e.target.value }) }}
                        />
                    </Form.Item>

                    <Form.Item wrapperCol={{ ...layout.wrapperCol, offset:
10 }}>
                            <Button
                                type="primary"
                                htmlType="submit"
                                style={{width: '60%', marginTop:
'5%',backgroundColor:'#8357c5',border: 'none'}}
                            >
                                <Link to = {{ pathname:'/', formInfo:
this.state }}>
                                    Submit
                                </Link>
                            </Button>

                    </Form.Item>

                </Form>
```

```
                </div>

            </div>
        )
    }
}
```

In the button, let's assign our state to an instance object we create and then pass it to the profile. We can do this by doing

```
<Link to = {{ pathname:'/', formInfo: this.state }}>
                                    Submit
                                </Link>
```

Displaying the inputted value from Forms.

Dashboard.js

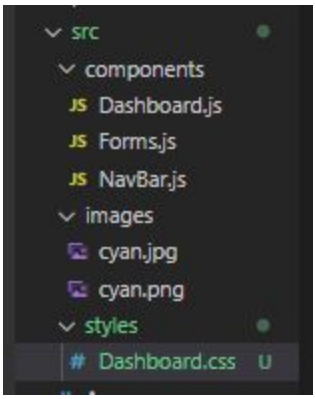
```
import React, { Component } from 'react';
import { Card } from 'antd';
import ProfilePic from '../images/milo.png';
import NavBar from './NavBar';

const { Meta } = Card;

export default class Dashboard extends Component {

    render() {

        let name = "Name"
        let email = "This is user email"
        let introduction = "This user is lazy and did not put introduction"

        if(this.props.location.formInfo != undefined){
            name = this.props.location.formInfo.name
            email = this.props.location.formInfo.email
            introduction = this.props.location.formInfo.intro
```

```
        }
        else{
            name = "Name"
            email = "This is user email"
            introduction = "This user is lazy and did not put introduction"
        }

         return (

             <div>

                 <NavBar />

                 <div>
                 <Card>

                     <img src={ProfilePic} alt="Profile" />

                     <Card
                         style={{ marginTop: 16 }}
                         type="inner"
                         title={name}
                         extra={<a href="/forms">Edit</a>}
                     >

                         <Meta
                             description={email}
                         />

                             <p>{introduction}</p>

                     </Card>

                 </Card>
                 </div>

         </div>
         )
     }
}
```

7. External CSS

For this project, we will separate the style and components files in different folders. The files could be stored in the same folder too, depending on the project. For this project, I suggest to store the style(CSS) and components files differently..In the src folder let's create a folder named styles, and inside the styles folder create a css that corresponds

to the component that you want to apply your css with in our case which is Dashboard.css. You can also name it differently if you like.



Back to Dashboard.js, after creating the css, the first thing you do is to import it. Then we can start adding our style in it. How we add the style is by using className as identifier and then giving your style a name. Then we go to our css and add the styling.

Dashboard.js

```
import React, { Component } from 'react';
import { Card } from 'antd';
import ProfilePic from '../images/milo.png';
import NavBar from './NavBar';
import '../styles/Dashboard.css';

const { Meta } = Card;

export default class Dashboard extends Component {

    render() {

        let name = "Name"
        let email = "This is user email"
        let introduction = "This user is lazy and did not put introduction"

        if(this.props.location.formInfo != undefined){
            name = this.props.location.formInfo.name
            email = this.props.location.formInfo.email
            introduction = this.props.location.formInfo.intro
        }
```

```
        else{
            name = "Name"
            email = "This is user email"
            introduction = "This user is lazy and did not put introduction"
        }

         return (

             <div>

                 <NavBar />

                 <div className={"mainContainer"}>
                 <Card
                     style={{ borderWidth: 6, padding: 50,
borderBottomColor:'#8357c5' }}
                 >

                     <img src={ProfilePic} alt="Profile"
className={"profilePic"}/>

                     <div className={"innerCard"}>
                             <Card
                                 style={{ marginTop: 16, padding: 20 }}
                                 type="inner"
                                 title={name}
                                 extra={<a href="/forms">Edit</a>}
                             >

                                 <Meta
                                     description={email}
                                 />

                                     <p
className={"intro"}>{introduction}</p>

                             </Card>
                     </div>

                 </Card>
```

```
                </div>

        </div>
        )
    }
}
```

Dashboard.css

```
.profilePic {
    width: 10%;
}

.mainContainer{
    margin: 10%;
}

.innerCard{
    margin-left: 20%;
    margin-top: -17%;
}

.intro{
    margin-top: 2%;
}
```

You can also see that I have a couple of inline styles, if you would like to give it a className and move it to the css it is also fine, because it really depends on your preferences.

And save all files, lastly, you will have a better looking one compared to before.