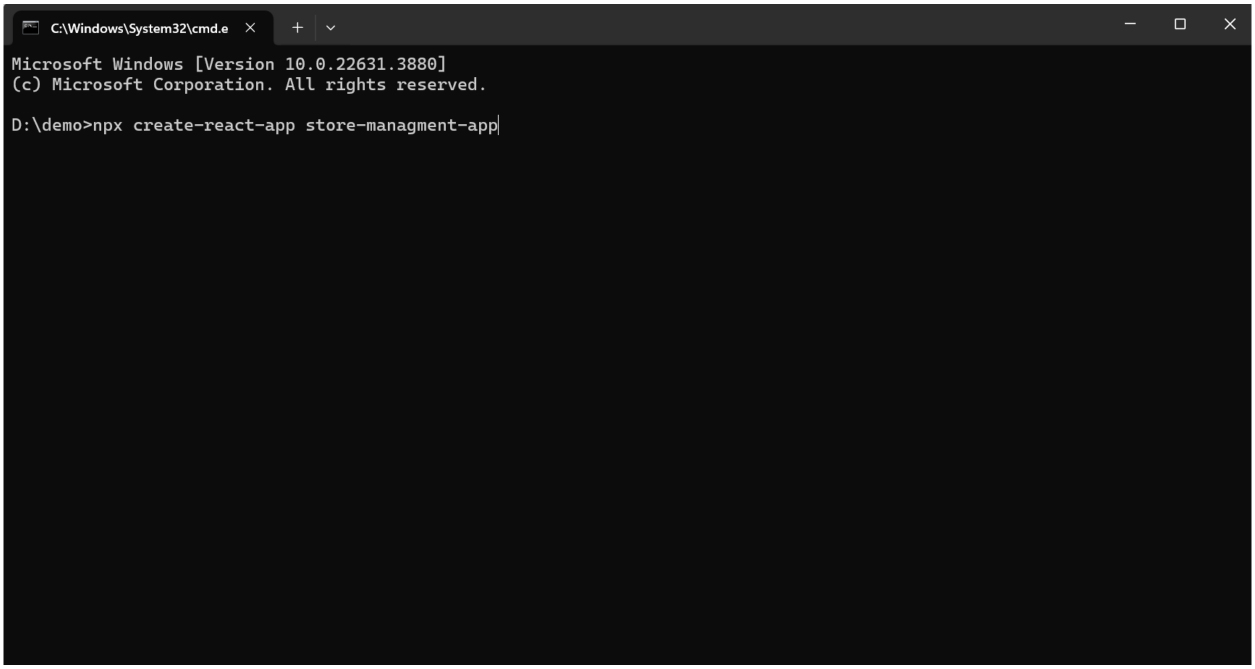
Store Management Application

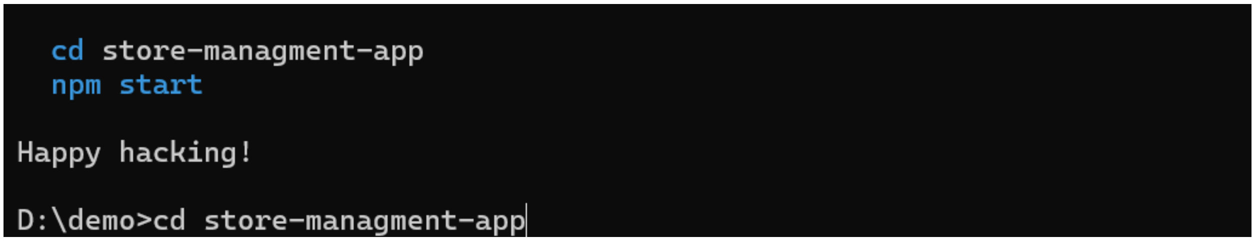
In This Store Management Application, we will perform all CRUD operations in React project to understand the implementation of real-time projects.

Step 1. Create React Project With the title ‘store-management-App’.



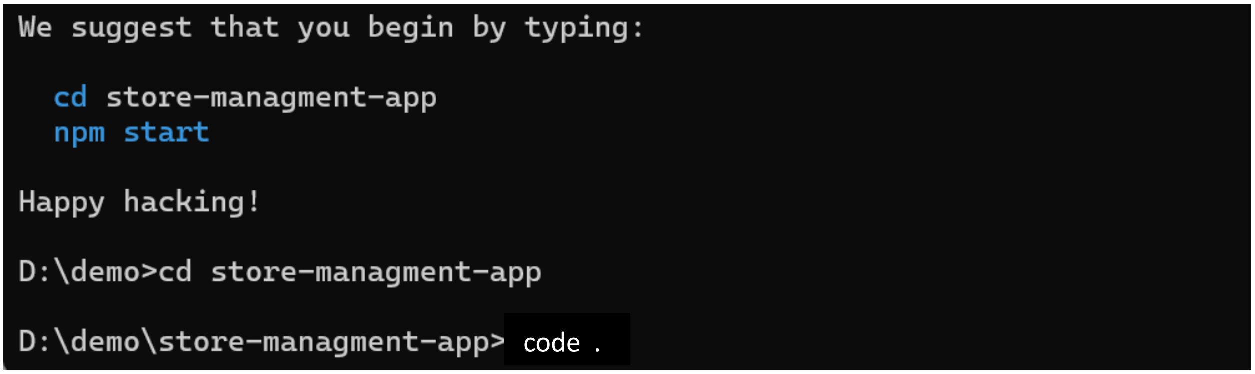
```
Microsoft Windows [Version 10.0.22631.3880]
(c) Microsoft Corporation. All rights reserved.

D:\demo>npx create-react-app store-managment-app
```

Step 2. Once the project is created by CRA then move cmd to the project directory.



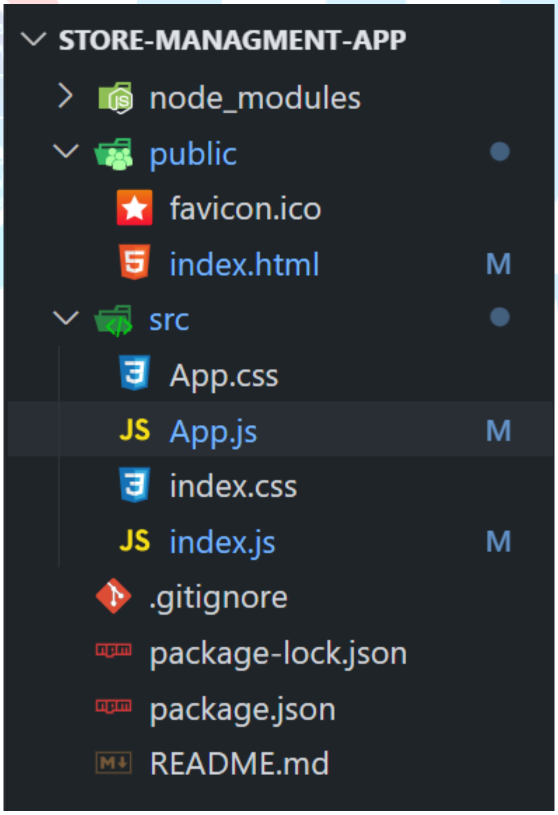
```
   cd store-managment-app
   npm start

Happy hacking!

D:\demo>cd store-managment-app
```

Step 3. Use ‘code .’ command to launch VScode ide into the project directory.



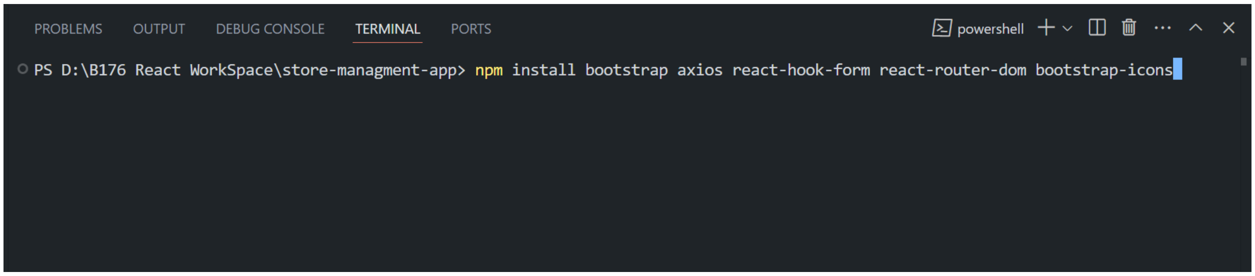
```
We suggest that you begin by typing:

  cd store-managment-app
  npm start

Happy hacking!

D:\demo>cd store-managment-app

D:\demo\store-managment-app> code .
```

Step 4. Remove unnecessary files and code form project.

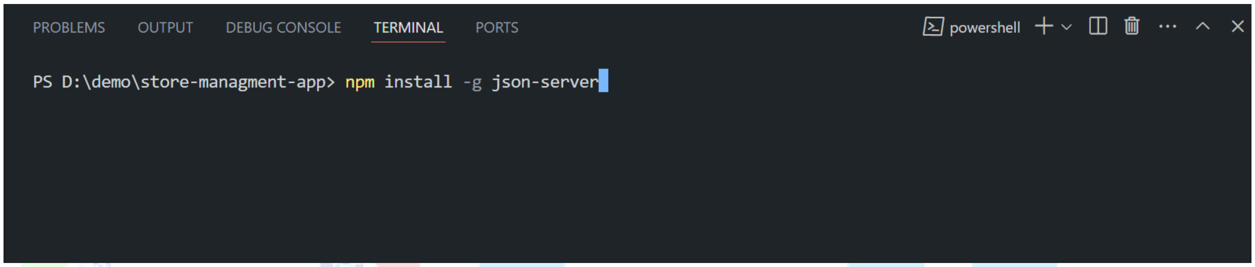


```
STORE-MANAGMENT-APP
  > node_modules
  v public                 ●
      favicon.ico
      index.html           M
  v src                    ●
      App.css
      App.js               M
      index.css
      index.js             M
    .gitignore
    package-lock.json
    package.json
    README.md
```

Step 5. Install all required libraries by using npm command as bellow.

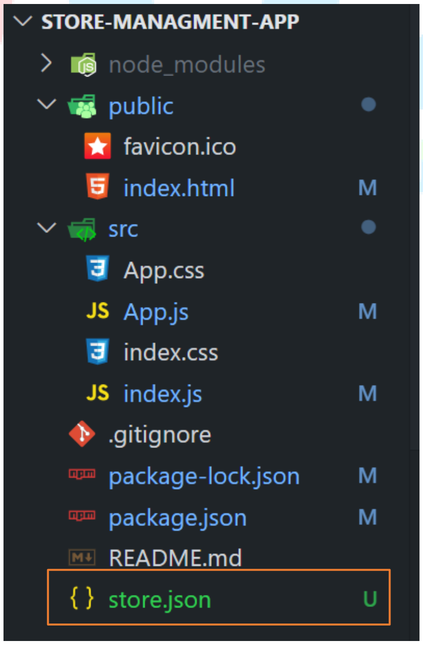


```
PS D:\B176 React WorkSpace\store-managment-app> npm install bootstrap axios react-hook-form react-router-dom bootstrap-icons
```



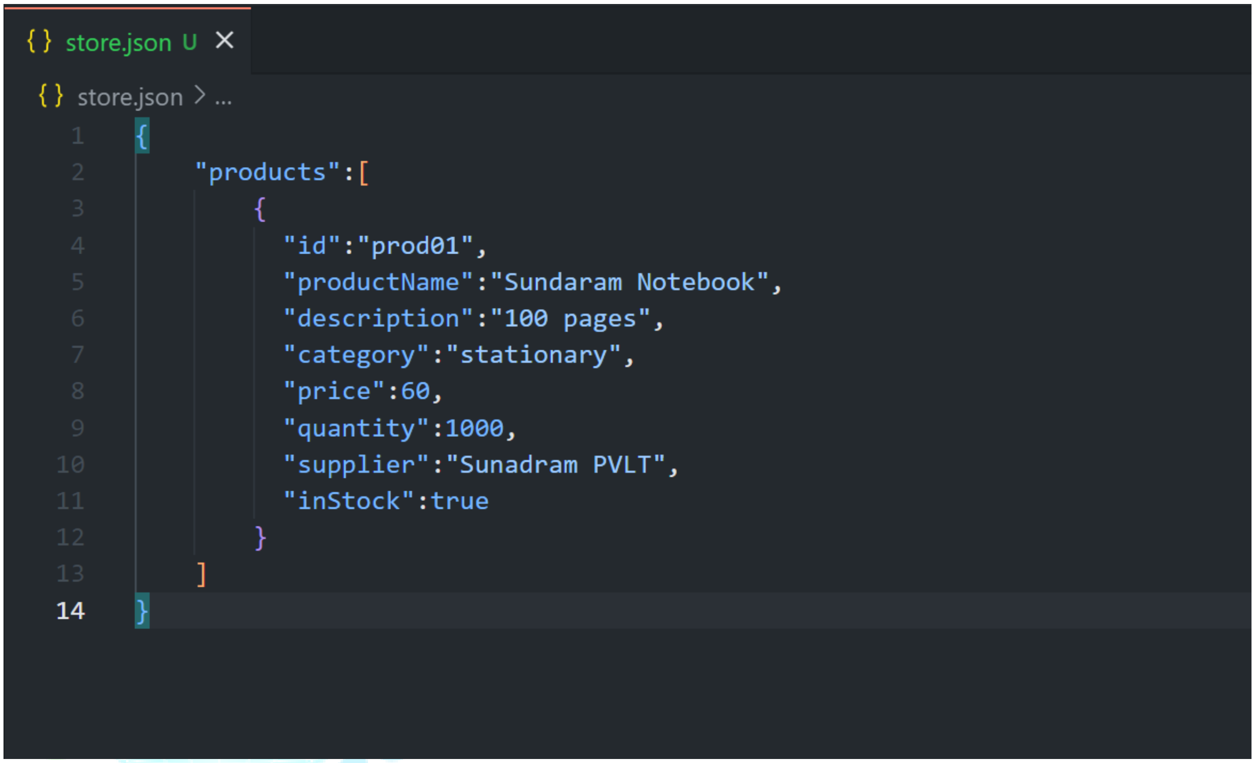
```
PS D:\demo\store-managment-app> npm install -g json-server
```

Step 6. Create dummy data base in store.json file.



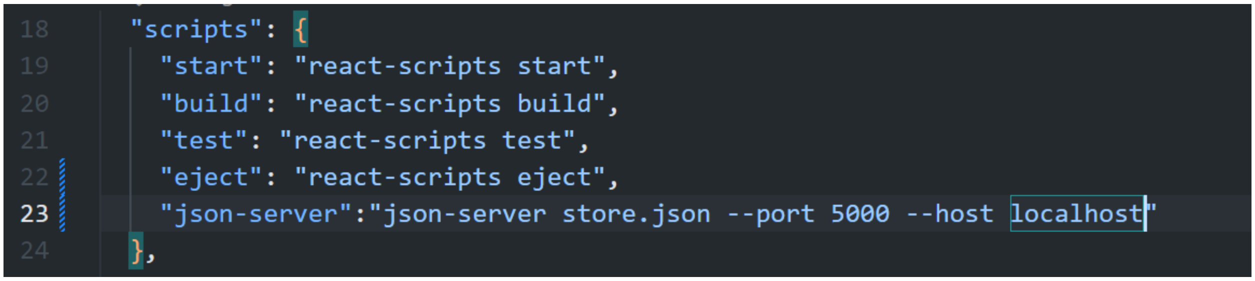
```
STORE-MANAGMENT-APP
  > node_modules
  v public
      favicon.ico
      index.html          M
  v src
      App.css
      App.js              M
      index.css
      index.js            M
    .gitignore
    package-lock.json     M
    package.json          M
    README.md
    store.json            U
```

Step 7. In strore.json add some dummy data as shown below.



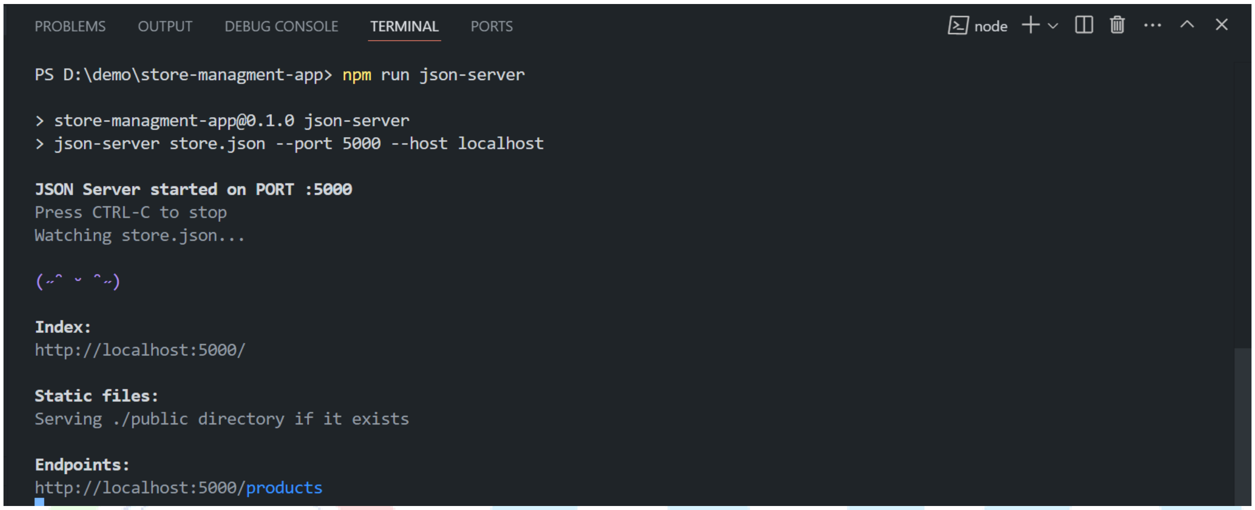
```json
{
    "products":[
        {
            "id":"prod01",
            "productName":"Sundaram Notebook",
            "description":"100 pages",
            "category":"stationary",
            "price":60,
            "quantity":1000,
            "supplier":"Sunadram PVLT",
            "inStock":true
        }
    ]
}
```

Step 8. In package.json add running script for the json-server.



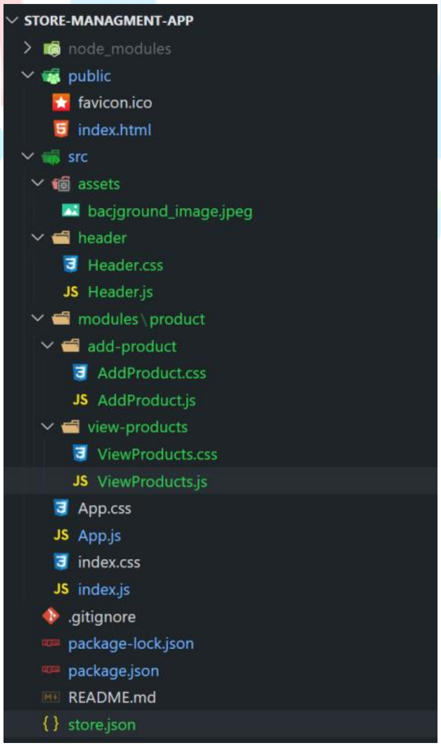
```json
  "scripts": {
    "start": "react-scripts start",
    "build": "react-scripts build",
    "test": "react-scripts test",
    "eject": "react-scripts eject",
    "json-server":"json-server store.json --port 5000 --host localhost"
  },
```

Step 8. Run the store.json database on json-server.



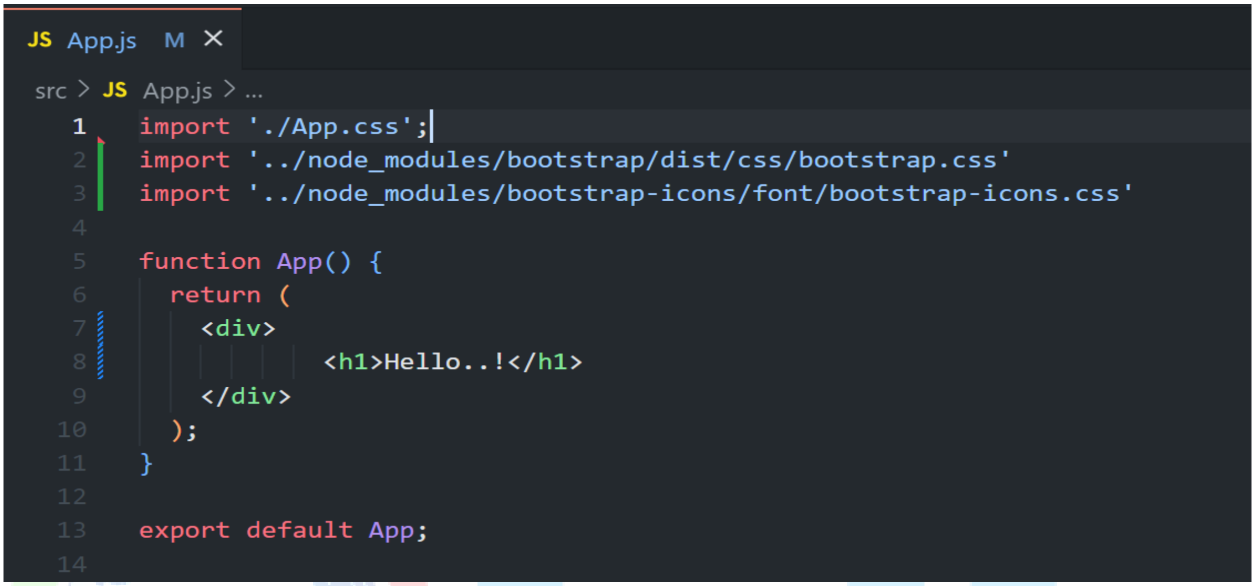
```
PS D:\demo\store-managment-app> npm run json-server

> store-managment-app@0.1.0 json-server
> json-server store.json --port 5000 --host localhost

JSON Server started on PORT :5000
Press CTRL-C to stop
Watching store.json...

(˶ᵔ ᵕ ᵔ˶)

Index:
http://localhost:5000/

Static files:
Serving ./public directory if it exists

Endpoints:
http://localhost:5000/products
```

Step 8. Create required folder structure and files in src directory.(assets for images)



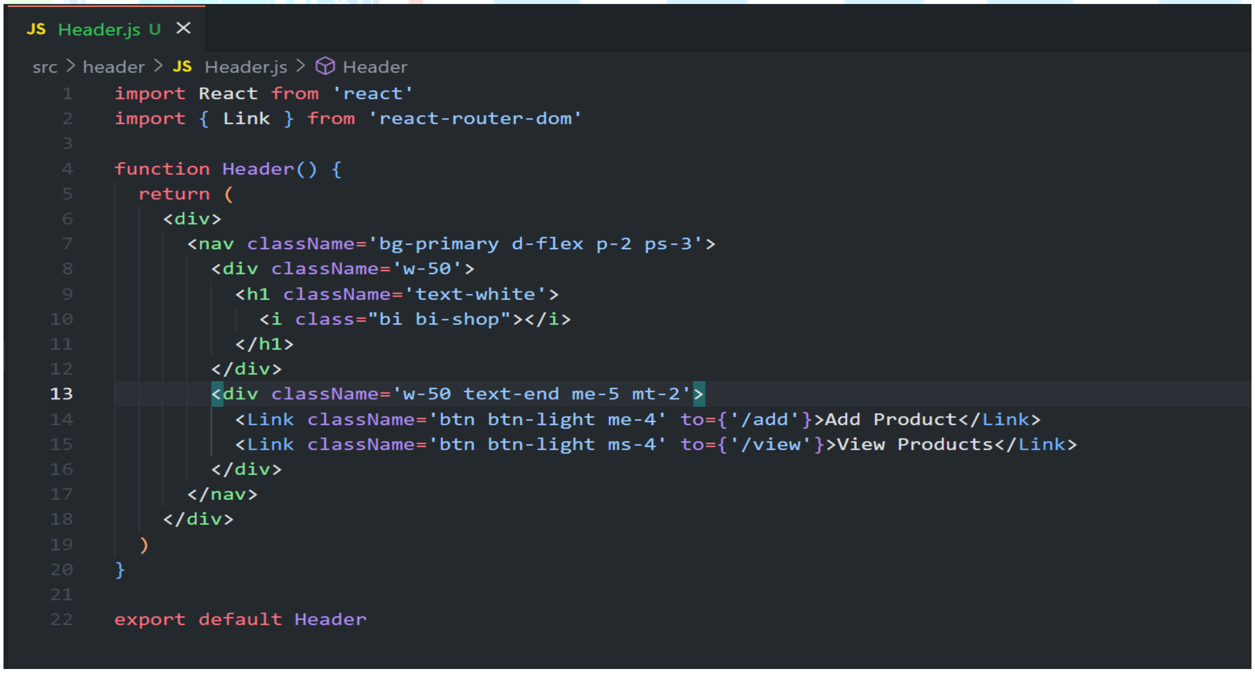
```
STORE-MANAGMENT-APP
 > node_modules
 v public
     favicon.ico
     index.html
 v src
   v assets
       bacjground_image.jpeg
   v header
       Header.css
       Header.js
   v modules\product
     v add-product
         AddProduct.css
         AddProduct.js
     v view-products
         ViewProducts.css
         ViewProducts.js
     App.css
     App.js
     index.css
     index.js
   .gitignore
   package-lock.json
   package.json
   README.md
   store.json
```

Step 9. Configure bootstrap and bootstrap-icons library.



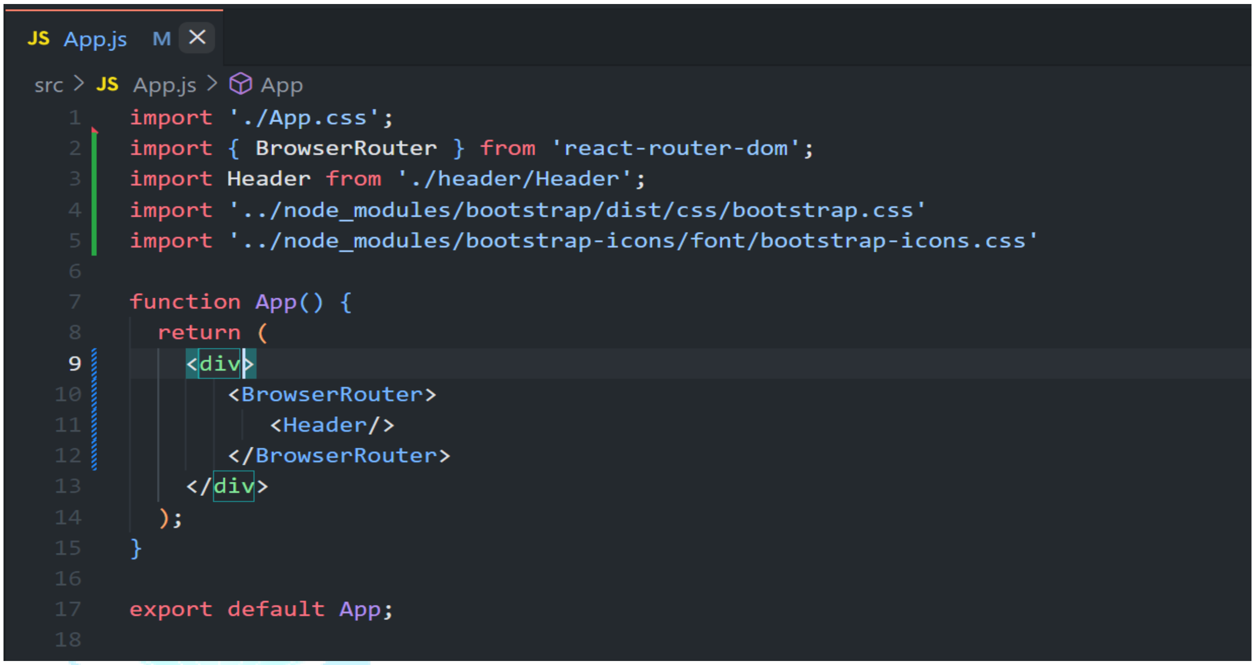
```
import './App.css';
import '../node_modules/bootstrap/dist/css/bootstrap.css'
import '../node_modules/bootstrap-icons/font/bootstrap-icons.css'

function App() {
  return (
    <div>
        <h1>Hello..!</h1>
    </div>
  );
}

export default App;
```

Step 9. In Header.js create Header component as shown below.



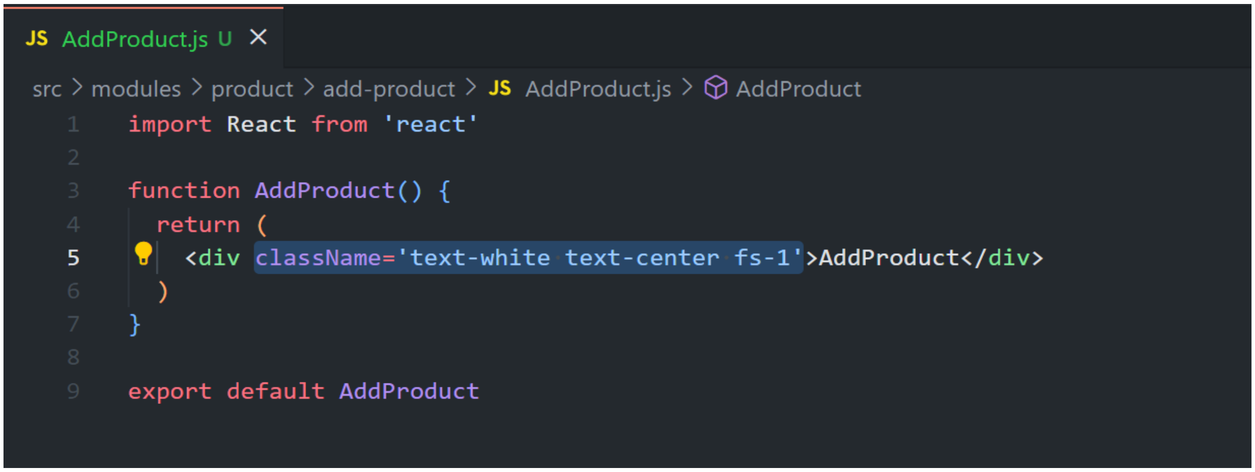
```
import React from 'react'
import { Link } from 'react-router-dom'

function Header() {
  return (
    <div>
      <nav className='bg-primary d-flex p-2 ps-3'>
        <div className='w-50'>
          <h1 className='text-white'>
            <i class="bi bi-shop"></i>
          </h1>
        </div>
        <div className='w-50 text-end me-5 mt-2'>
          <Link className='btn btn-light me-4' to={'/add'}>Add Product</Link>
          <Link className='btn btn-light ms-4' to={'/view'}>View Products</Link>
        </div>
      </nav>
    </div>
  )
}

export default Header
```

Step 10. Add the selector of Header component in App component as children of BrowserRouter components.



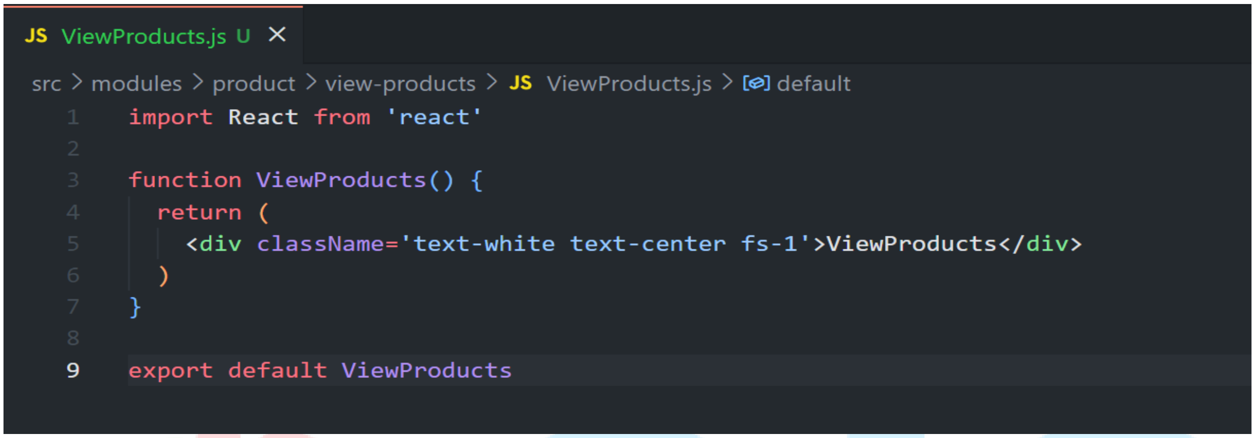
```
import './App.css';
import { BrowserRouter } from 'react-router-dom';
import Header from './header/Header';
import '../node_modules/bootstrap/dist/css/bootstrap.css'
import '../node_modules/bootstrap-icons/font/bootstrap-icons.css'

function App() {
  return (
    <div>
       <BrowserRouter>
          <Header/>
       </BrowserRouter>
    </div>
  );
}

export default App;
```

Step 11. Create AddProduct component as below.


JS AddProduct.js U ✕
src > modules > product > add-product > JS AddProduct.js > AddProduct

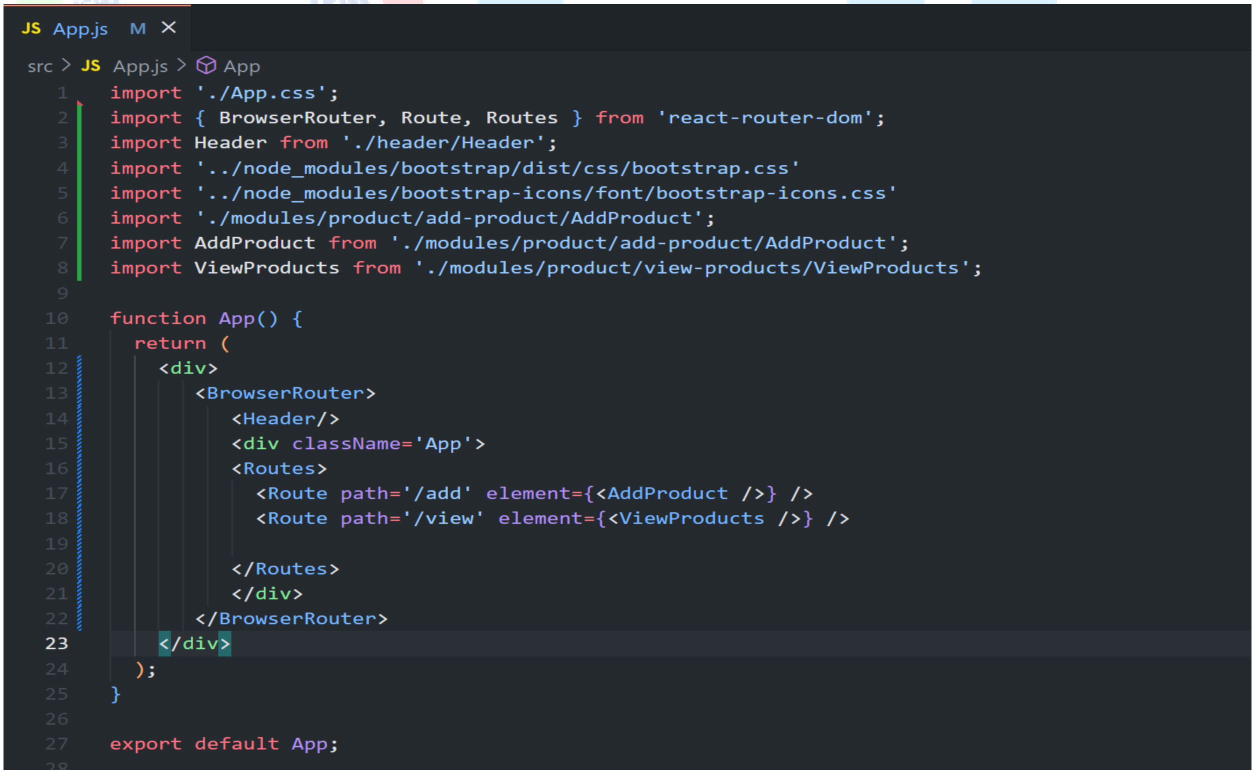
```
import React from 'react'

function AddProduct() {
  return (
    <div className='text-white text-center fs-1'>AddProduct</div>
  )
}

export default AddProduct
```

Step 12. Create ViewProducts component as below.


JS ViewProducts.js U ×
src > modules > product > view-products > JS ViewProducts.js > default

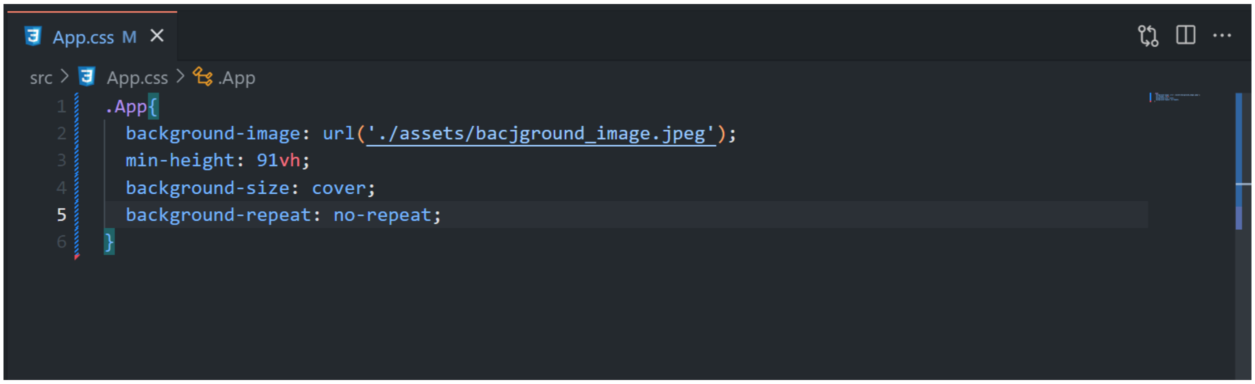
```
import React from 'react'

function ViewProducts() {
  return (
    <div className='text-white text-center fs-1'>ViewProducts</div>
  )
}

export default ViewProducts
```

Step 13 In App component add routs for AddProduct component and ViewProducts component as define into the Link component.



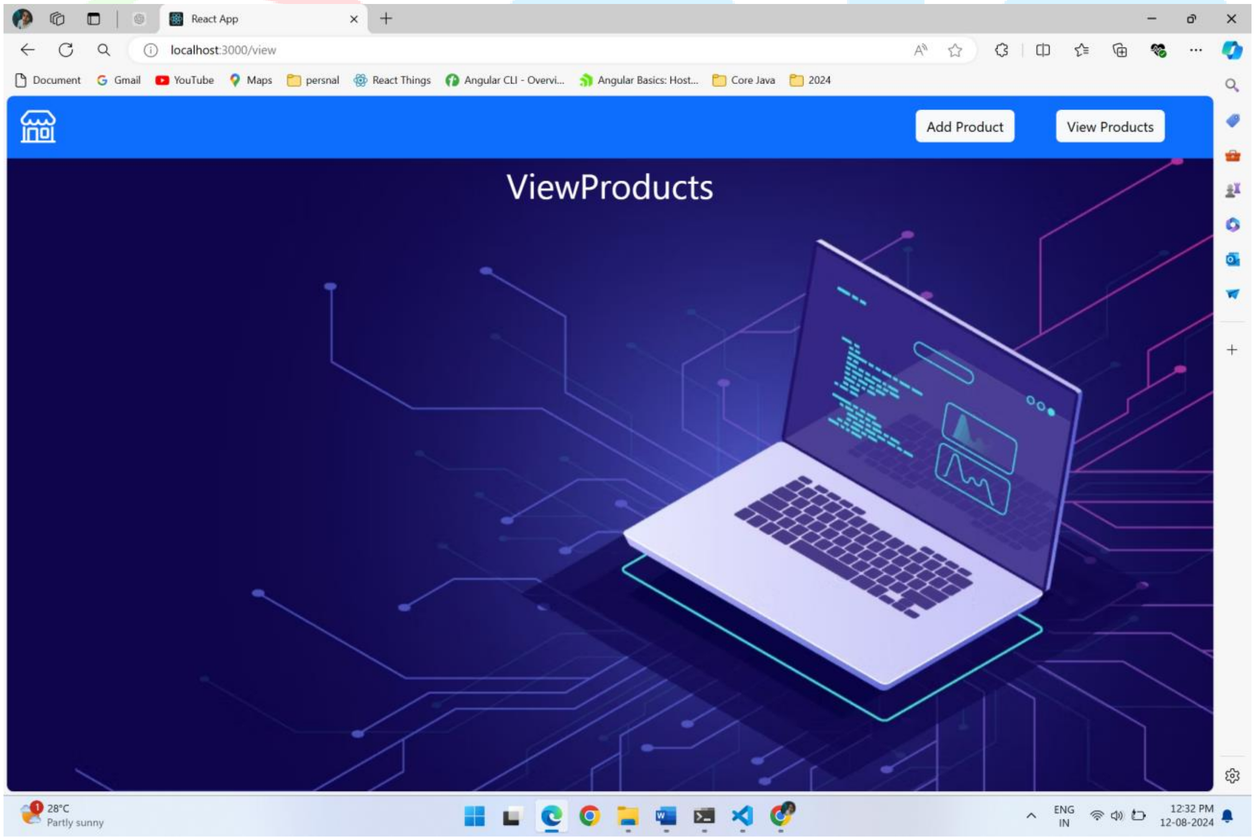
```
import './App.css';
import { BrowserRouter, Route, Routes } from 'react-router-dom';
import Header from './header/Header';
import '../node_modules/bootstrap/dist/css/bootstrap.css'
import '../node_modules/bootstrap-icons/font/bootstrap-icons.css'
import './modules/product/add-product/AddProduct';
import AddProduct from './modules/product/add-product/AddProduct';
import ViewProducts from './modules/product/view-products/ViewProducts';

function App() {
  return (
    <div>
      <BrowserRouter>
        <Header/>
        <div className='App'>
        <Routes>
          <Route path='/add' element={<AddProduct />} />
          <Route path='/view' element={<ViewProducts />} />

        </Routes>
        </div>
      </BrowserRouter>
    </div>
  );
}

export default App;
```

Step 14. In App.css file change '.App' class as bellow.

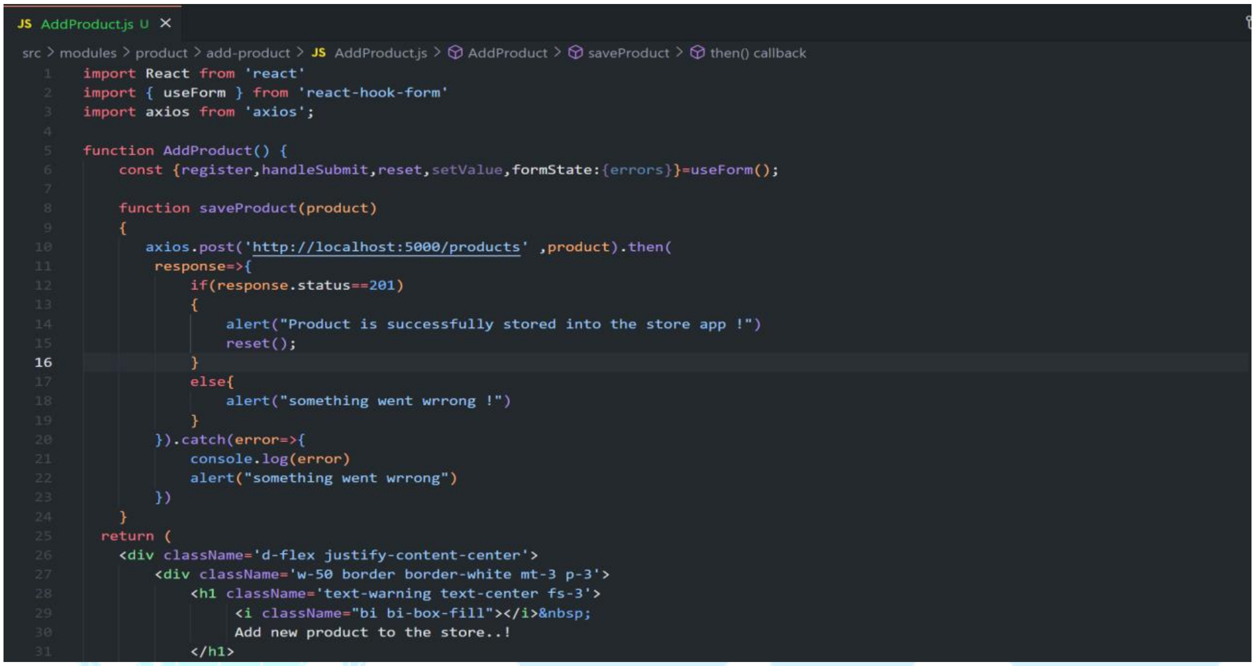


```css
.App{
  background-image: url('./assets/bacjground_image.jpeg');
  min-height: 91vh;
  background-size: cover;
  background-repeat: no-repeat;
}
```

Step 15. Check Output. (navigate to AddProduct & View Products Component)

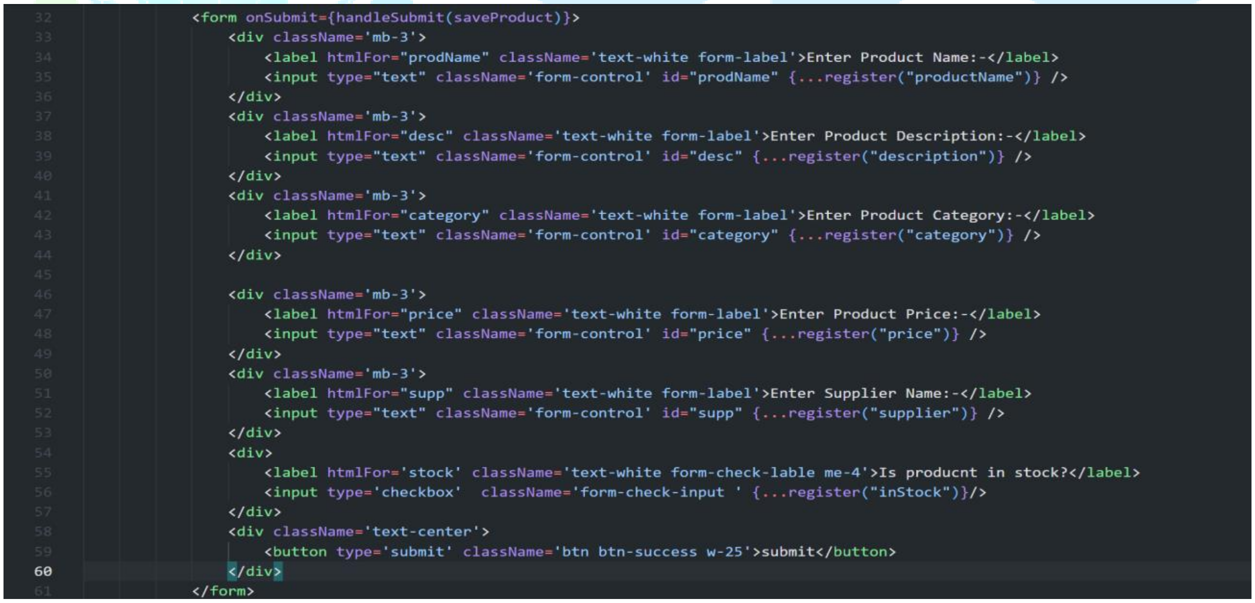


Step 16. Create Product registration form in AddProduct component and save the product details into the database by consuming post API using axios.

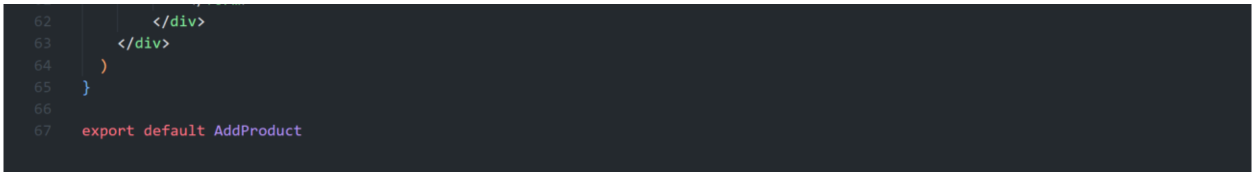


JS AddProduct.js U ×
src > modules > product > add-product > JS AddProduct.js > AddProduct > saveProduct > then() callback

```
import React from 'react'
import { useForm } from 'react-hook-form'
import axios from 'axios';

function AddProduct() {
    const {register,handleSubmit,reset,setValue,formState:{errors}}=useForm();

    function saveProduct(product)
    {
       axios.post('http://localhost:5000/products' ,product).then(
        response=>{
            if(response.status==201)
            {
                alert("Product is successfully stored into the store app !")
                reset();
            }
            else{
                alert("something went wrrong !")
            }
        }).catch(error=>{
            console.log(error)
            alert("something went wrrong")
        })
    }
  return (
    <div className='d-flex justify-content-center'>
        <div className='w-50 border border-white mt-3 p-3'>
            <h1 className='text-warning text-center fs-3'>
                 <i className="bi bi-box-fill"></i>&nbsp;
                 Add new product to the store..!
            </h1>
```

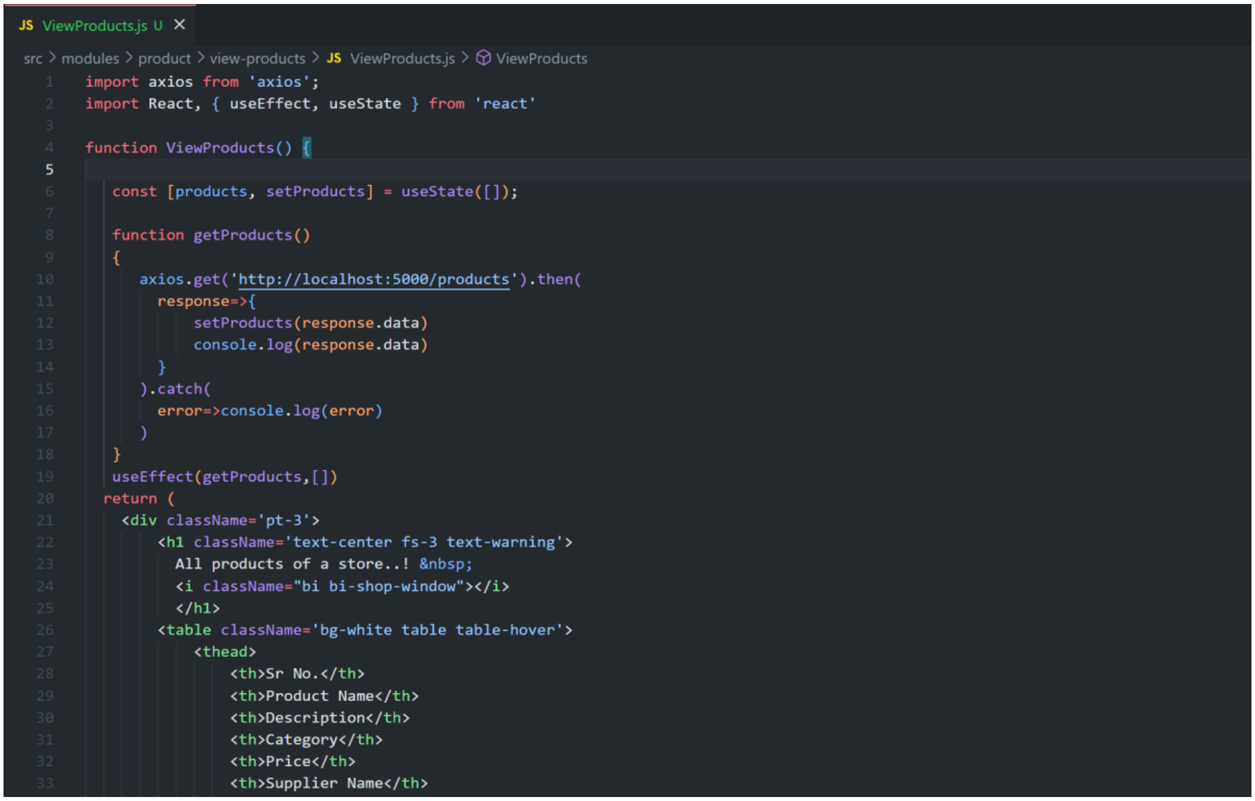


```
            <form onSubmit={handleSubmit(saveProduct)}>
                <div className='mb-3'>
                    <label htmlFor="prodName" className='text-white form-label'>Enter Product Name:-</label>
                    <input type="text" className='form-control' id="prodName" {...register("productName")} />
                </div>
                <div className='mb-3'>
                    <label htmlFor="desc" className='text-white form-label'>Enter Product Description:-</label>
                    <input type="text" className='form-control' id="desc" {...register("description")} />
                </div>
                <div className='mb-3'>
                    <label htmlFor="category" className='text-white form-label'>Enter Product Category:-</label>
                    <input type="text" className='form-control' id="category" {...register("category")} />
                </div>

                <div className='mb-3'>
                    <label htmlFor="price" className='text-white form-label'>Enter Product Price:-</label>
                    <input type="text" className='form-control' id="price" {...register("price")} />
                </div>
                <div className='mb-3'>
                    <label htmlFor="supp" className='text-white form-label'>Enter Supplier Name:-</label>
                    <input type="text" className='form-control' id="supp" {...register("supplier")} />
                </div>
                <div>
                    <label htmlFor='stock' className='text-white form-check-lable me-4'>Is producnt in stock?</label>
                    <input type='checkbox'  className='form-check-input ' {...register("inStock")}/>
                </div>
                <div className='text-center'>
                    <button type='submit' className='btn btn-success w-25'>submit</button>
                </div>
            </form>
```



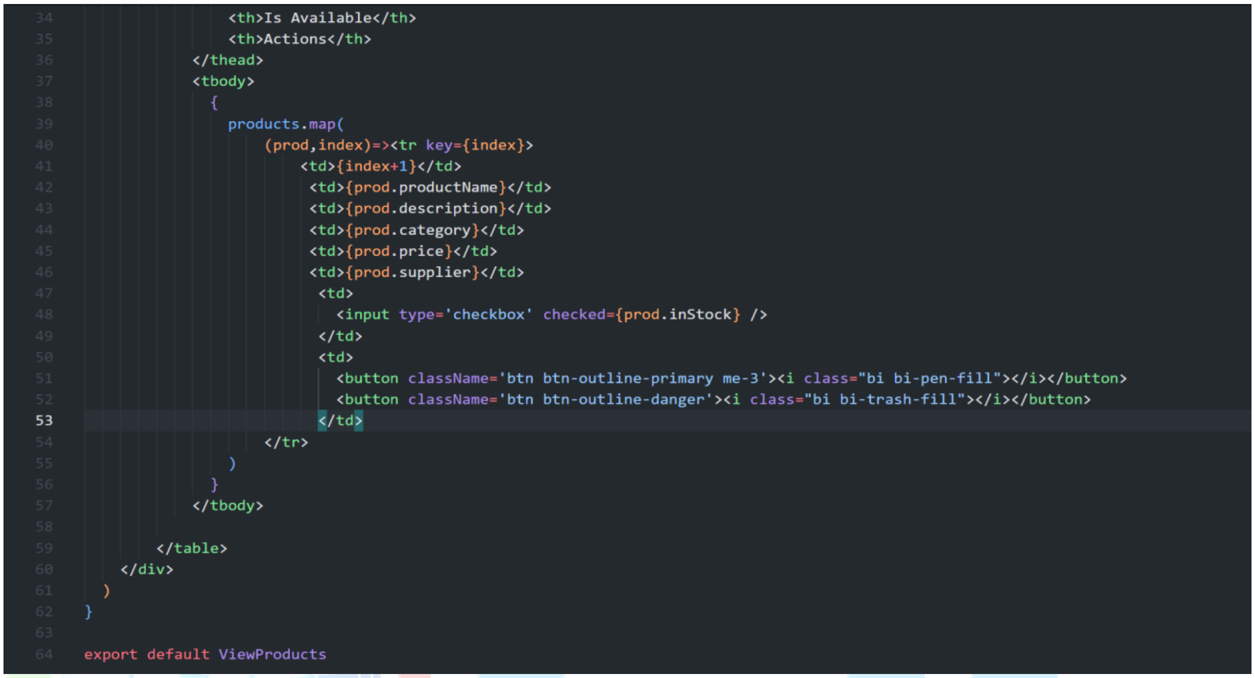
```
        </div>
    </div>
  )
}

export default AddProduct
```

Step 16. Create table in ViewProducts component to render the product details from the database by consuming get API using axios.



JS ViewProducts.js U ×
src > modules > product > view-products > JS ViewProducts.js > ViewProducts

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

function ViewProducts() {

   const [products, setProducts] = useState([]);

   function getProducts()
   {
      axios.get('http://localhost:5000/products').then(
        response=>{
            setProducts(response.data)
            console.log(response.data)
        }
      ).catch(
        error=>console.log(error)
      )
   }
   useEffect(getProducts,[])
  return (
    <div className='pt-3'>
        <h1 className='text-center fs-3 text-warning'>
          All products of a store..! &nbsp;
          <i className="bi bi-shop-window"></i>
          </h1>
        <table className='bg-white table table-hover'>
            <thead>
                <th>Sr No.</th>
                <th>Product Name</th>
                <th>Description</th>
                <th>Category</th>
                <th>Price</th>
                <th>Supplier Name</th>
```



```
                <th>Is Available</th>
                <th>Actions</th>
            </thead>
            <tbody>
              {
                products.map(
                    (prod,index)=><tr key={index}>
                        <td>{index+1}</td>
                         <td>{prod.productName}</td>
                         <td>{prod.description}</td>
                         <td>{prod.category}</td>
                         <td>{prod.price}</td>
                         <td>{prod.supplier}</td>
                          <td>
                            <input type='checkbox' checked={prod.inStock} />
                          </td>
                          <td>
                            <button className='btn btn-outline-primary me-3'><i class="bi bi-pen-fill"></i></button>
                            <button className='btn btn-outline-danger'><i class="bi bi-trash-fill"></i></button>
                          </td>
                    </tr>
                )
              }
            </tbody>

        </table>
    </div>
  )
}

export default ViewProducts
```