***Arpit Singh***

***19BCG10069***

***Krypto – Task 1***

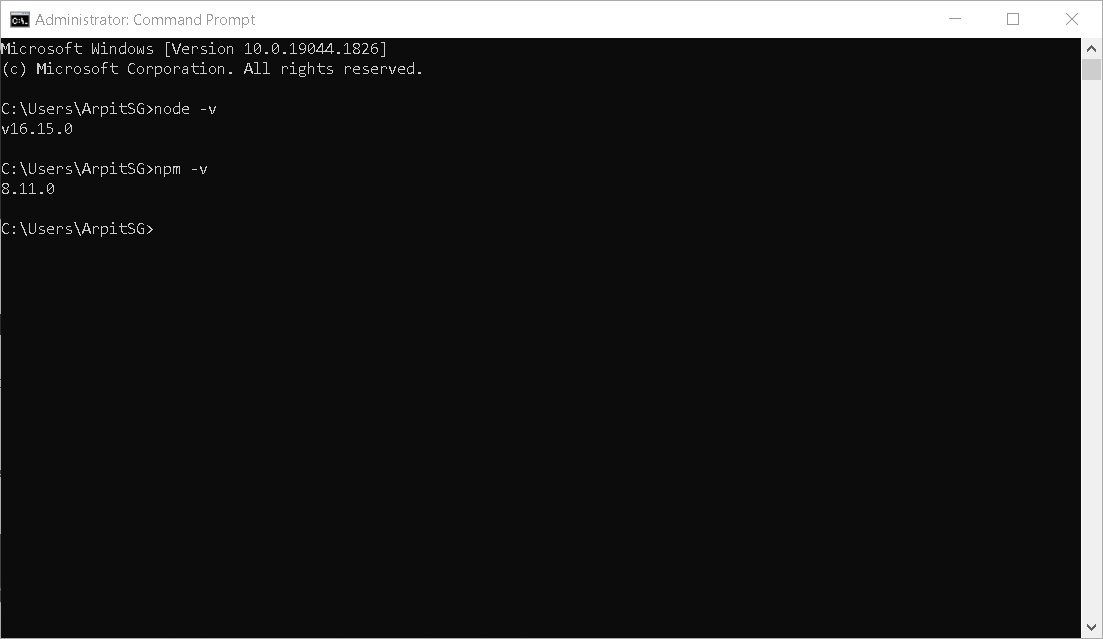
***Technical Task***

***(Front End)***

1. **Installing Node and Setting up the Environment**
   * ***Prerequisites***
     1. Hardware Requirements
     2. RAM 4GB
     3. CPU Intel Core i3TM i3 HQ CPU @2.50 GHz
     4. ROM 256 GB

* Download the **Windows Installer** from [**NodeJs official website**](https://nodejs.org/en/download/). Make sure you have downloaded the latest version of NodeJs. It includes the NPM package manager.
* Install **Node.js and NPM**
* The following features will be installed by default:
  + Node.js runtime
  + Npm package manager
  + Online documentation shortcuts
  + Add to Path
  + **Click** on the **Install** button.

1. **Check Node.js and NPM Version**
   * ***Verify it with “Command Prompt”***
     1. To confirm Node installation, type **node -v** command.
     2. To confirm NPM installation, type **npm -v** command.
     3. You should see the latest versions installed.



* With that done, let’s start off by creating a new React project with the [Create React App](https://www.npmjs.com/package/create-react-app) tool. You can either install this globally, or use ***npx***, like so:

npx create-react-app e-commerce

* For **React Router**
  + ***npm install react-router-dom.***
* **Json-server and Json-server-auth** to create our ***fake back*** end to handle authentication
  + ***npm install json-server json-server-auth.***
* **Jwt - decode** so that we can parse the JWT that our back end will ***respond*** with
  + ***npm install jwt-decode.***

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

***Note:*** I will be using the [**Ecommerce-boilerplate**](https://github.com/gurukishore111/Ecommerce-boilerplate) as instructed in the Readme File so the above commands will not be used as the **basic structure** is already provided.

* I have downloaded the Zip file of the repo (<https://github.com/gurukishore111/Ecommerce-boilerplate>) to ensure that the changes remain private while posting my solution.

# Software’s used:

* **Microsoft Word:** Microsoft Word is word processing software developed by Microsoft. It was first released on October 25, 1983, under the name Multi-Tool Word for Xenix systems.
  + Used for ***Documentation*** Purpose.
* **VisualStudioCode (Vscode):** Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.
  + Used for ***Coding and organizing*** the Application.
* **GitHub/GitLab:** GitHub, Inc. is a provider of Internet hosting for software development and version control using Git. It offers the distributed version control and source code management (SCM) functionality of Git, plus its own features. It provides access control and several collaboration features such as bug tracking, feature requests, task management, continuous integration, and wikis for every project.
  + Used for ***Extracting resources and deploying*** the Application.

# Languages/tech(s)/frameworks used:

* **HTML/CSS/JS/JSX/REACT/JSON/Node/npm/yarn.**

***Note:*** I have used ***YARN*** package manager instead of npm as it is *faster* and provides more *security*.

# Coding and brief Explanation:

***\*NOTE: Proper comments are added in each file for better understanding.\****

1. Git clone <https://github.com/gurukishore111/Ecommerce-boilerplate.git>

* Command to clone base repo to your local machine.

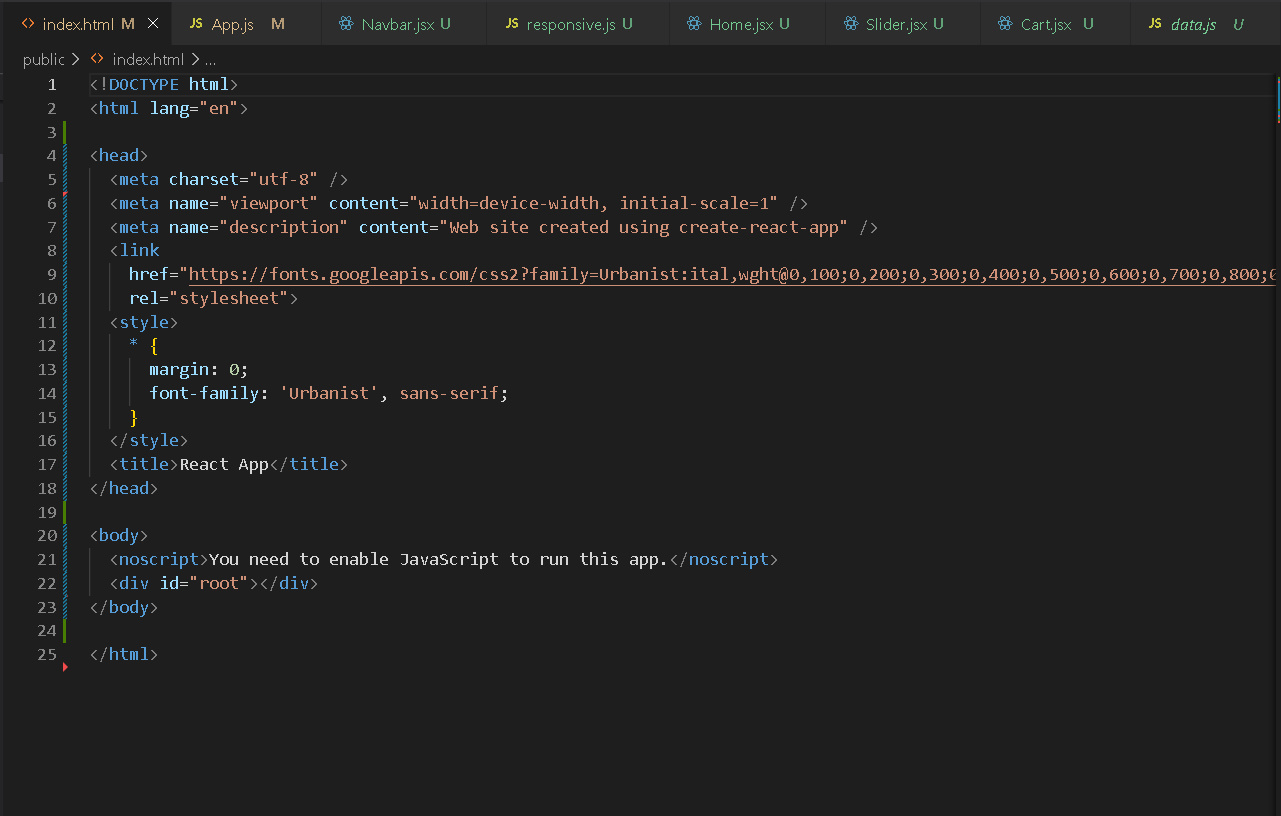
1. **File Structure**:



Here I have made use of components and made separate pages, also I have incorporated styled components in each jsx file.

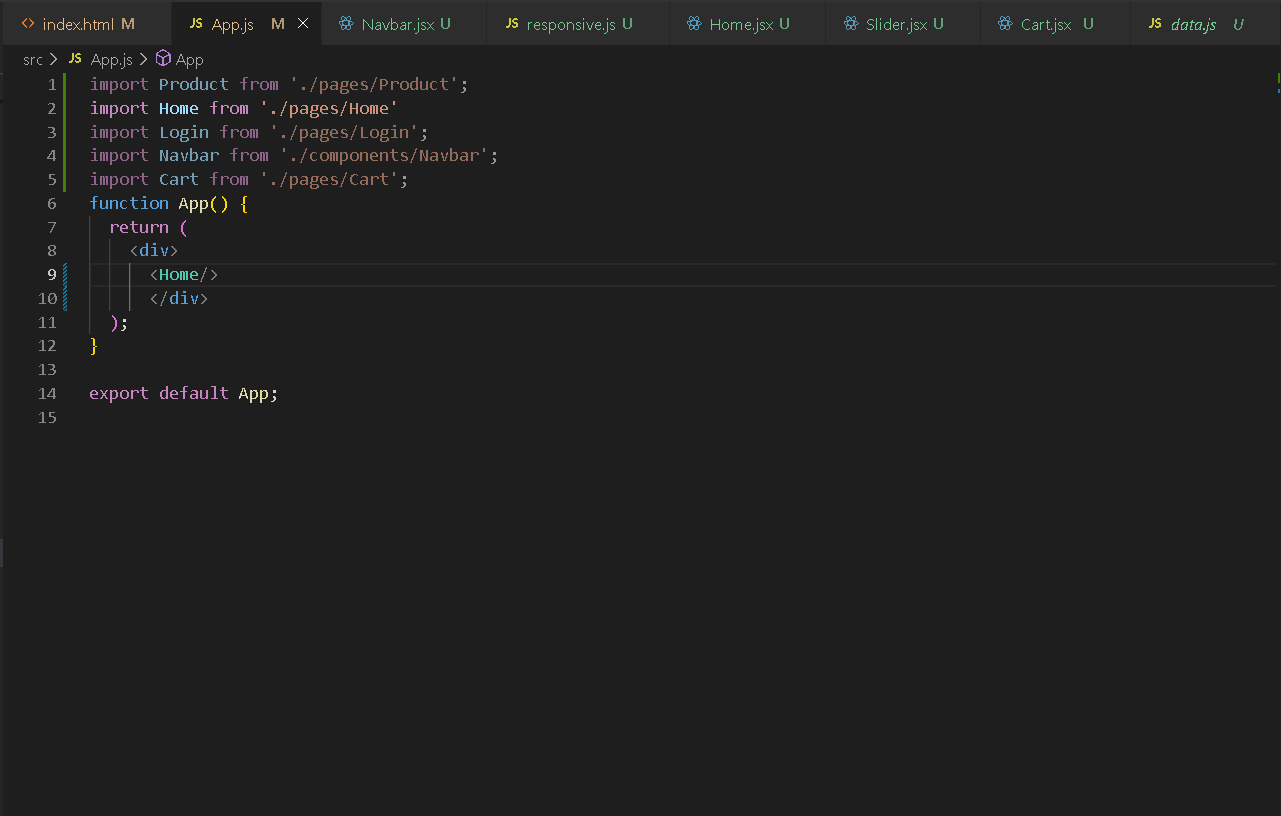
I will explain the flow of the doc and each file in brief below:

* **Index.html:**



Here only 2 extra things have been added. The Google fonts used and its Css styling.

* **App.js:**



Here all the components and pages made are imported and displayed in accordance to the workflow as required.