**Assignment 5**

* What is the difference between named export, default export, and \* as export?

**Named export**: With named export, we can share multiple objects, functions or component, variables from a single file

These values can be imported in another module using the 'import' keyword and the exact name of the export.

Header.js

export const Title = () =>{

…..

}

export const Header = () =>{

…..

}

Import {Title, Header} from ‘./Header.js’;

**Default Export**: Default exports are created by including a default tag in the export. Usually, you see default exports happen at the bottom of a file, but it’s possible to define them when your component is declared.

const MyComponent = () => {}

export default MyComponent

Or

function MyComponent() {}

export default MyComponent

Or

export default function() {}

Or

export default () => {}

**\* as import**: A wildcard \* as export can be used to export all exports of a module in a single import. It can only be used with a named import. For example:

// myModule.js

export const namedExport1 = ...;

export const namedExport2 = ...;

// import.js import \* as myModule from './myModule';

console.log(myModule.namedExport1);

console.log(myModule.namedExport2);

* What is the importance of config.js file ?
* To manage static global variable across components, making it easier to manage and maintain codebase, making it easier to change as and when needed in single file without touching entire codebase.
* Easy to maintain for different environments production, testserver and local
* Can maintain sensitive information such as api key and secret separate from rest of the code.
* What are react hooks ?
* React hook is new addition to react introduced in react 16.8
* React hooks are normal js functions
* React hooks are used for functional component
* Some of the hooks are useState, useEffect, useReducer etc.,
* Can also create custom hook in react
* They allow for code reuse
* Making easier to share logic between components
* Why do we need use state hook ?
* useState is to manage state
* Normal variable doesnot rerender but useState rerender state value
* useState is used in functional component to manage state variable, object, functions
* Components often need to change some content on screen as result of interaction.
* The component needs to remember things like current input value, clicked value, current image and shopping etc., to maintain that we need state so react provides us hook in functional component namely useState
* Usestate return array of two element namely variable and function
  + Variable retains state to rerender
  + Function to update the state variable
  + To add state import useState

Import {useState} from ‘react’;

Const [input, setInput] = useState(“”);

useState(“”) -> can set initial value for the state like boolean or string or numbers

SetInput -> set is the naming convention used by react

Returned value is array i.e., state and setState() -> here we are destructuring the value