1. What is difference between Named export, Default export and \* as export?
   * In JS, the export statement is used in modules to expose variables, functions, or classes so that they can be accessed and used in other parts of application or in separate files.
   * *Default export:*
     + A way to share a single value, function, or class as the main thing from a file with other parts of your code.
     + First **export** the component then **import** it.
       - **To export** : export default Header; => [*Header is a component name inside Header file in components repo*]
       - **To import** : import Header from “./components/Header”;

* *Named export:*
  + It allows you to export multiple functions, variables, or classes from a single file as separate entities.
  + To **export** named export :
    - Use export before any const or variable
      * export const CDN\_URL = “”;
* To **import** named export:
  + Import {CDN\_URL} from “../utils/constants”;.
* A file can have both named and default exports together.
  + It means you can export one main thing using *export default*, while also exporting multiple additional values using *export.*
  + This flexibility allows you to organize and share different parts of your code efficiently.

1. What is the importance of config.js file?
   * The config.js file is like the backstage manager for your React application. It’s not something you’ll find automatically in every project, but it serves as a hub where you can gather all the important configurations and settings.
   * In React, there might not be a standard config.js file. Configuration settings are often found in different places, such as within the code, environment variables, or specific configuration files related to tooling like Webpac or Babel or Parcel.
2. What are React Hooks?

* It is a normal pre-built JS utility function which is given to us by React. This function comes with some superpower and it has logic written behind scenes inside React.

1. Why do we need a useState Hook?
   * It is used to create state variable because it maintains the state of component.