1. **What is the difference between Named export, Default Export and \* as export?**

* With named exports, one can have multiple named exports per file. Then import the specific exports they want surrounded in braces.
* The name of imported module has to be the same as the name of the exported module.
* \* as export means you can import all the named imports into an object.
* One can have only one default export per file. When we import we have to specify a name.
* The naming of import is completely independent in default export and we can use any name we like.

1. **What is the importance of config.js file?**

* Configuration files ("config files" for short) are important to modern computing.
* They allow you to customize how you interact with an application or how an application interacts with the rest of your system.
* To be broadly effective, the most important thing about configuration files is that they are consistent and predictable.
* Configuration formats help developers store the data their applications need and help users store preferences for how they want applications to act.

1. **What are React Hooks?**

* Hooks are functions that let you “hook into” React state and lifecycle features from function components.
* Hooks don’t work inside classes — they let you use React without classes.
* Hooks are a way to reuse *stateful logic*, not state itself.
* In fact, each *call* to a Hook has a completely isolated state — so you can even use the same custom Hook twice in one component.
* React Hooks are simple JavaScript functions that we can use to isolate the reusable part from a functional component.
* Hooks can be stateful and can manage side-effects.

1. **Why do we need a useState Hook?**

* useState is a Hook that lets you add React state to function components.
* It declares a “state variable”
* This is a way to “preserve” some values between the function calls — useState is a new way to use the exact same capabilities that this.state provides in a class.
* Normally, variables “disappear” when the function exits but state variables are preserved by React.
* To manage states. Returns a stateful value and an updater function to update it.
* The **useState()** is a Hook that allows you to have state variables in functional components . so basically useState is the ability to encapsulate local state in a functional component.
* The  useState hook is a special function that takes the initial state as an argument and returns an array of two entries.
* UseState encapsulate only singular value from the state, for multiple state need to have useState calls.