Frontend Development

React Hooks:

React Hooks are functions that allow developers to use state and other React features in functional components. They enable the use of state and lifecycle features in functional components, reducing the need for class components.

In React, a "hook" refers to a special function that allows you to use state and other React features in functional components. React hooks were introduced in React version 16.8 to enable functional components to manage state and side effects, which were traditionally the domain of class components.

A Hook is a special function that lets you "hook into" React Feature. for Example: useState is a Hook that lets you add react State to Function Components. If you write a function component and realize you need to add some state to it, previously you had to convert it to a class.

Now you can use a hook inside the existing function component. Hooks are the new feature introduced in the react 16.8 version. It allows you to use state and other react features without writing a class. Hooks are the functions which “hook into” react state and lifecycle features from function components.

It doesn’t work inside classes. Hooks should always be used at the top level of the react functions. Node version 6 or above. NPM version 5.2 or above is the minimum requirement. In React, a "hook" refers to a special function that allows you to use state and other React features in functional components.

Rules Of Using Hooks In React JS:

Always Write It Inside The Component Or Function. Component name must be Pascal Case (First Letter Should Be Uppercase). We Can Directly Import Or We Can Directly Write It Using React.hookName. Don't call Hooks inside loops, conditions or nested functions.

React hooks were introduced in React version 16.8 to enable functional components to manage state and side effects, which were traditionally the domain of class components. There are so many types of built-in hooks are there, They are as follows: useState, useEffect, useContext, useReducer, useCallback, useMemo, useRef, useImperativeHandle, useLayoutEffect, useHistory and useLocation, useRef, React.forwardRef and many more things are there.

Hooks are the new feature introduced in the react 16.8 version. It allows you to use state and other react features without writing a class. Hooks are the functions which “hook into” react state and lifecycle features from function components. It doesn’t work inside classes. Hooks should always be used at the top level of the react functions. Node version 6 or above. NPM version 5.2 or above is the minimum requirement.