HOOK IN REACT

Hooks allow us to "hook" into React features such as state and lifecycle methods.

* Hooks can only be called inside React function components.
* Hooks can only be called at the top level of a component.
* Hooks cannot be conditional

**Use of *‘this’* keyword:** The first reason has to do more with javascript than with React itself. To work with classes one needs to understand how ‘this’ keyword works in javascript which is very different from how it works in other languages. It is easier to understand the concept of props, state, and uni-directional data flow but using ‘this’ keyword might lead to struggle while implementing class components. One also needs to bind event handlers to the class components.

**Simplifying complex scenarios:** While creating components for complex scenarios such as data fetching and subscribing to events it is likely that all related code is not organized in one place are scattered among different life cycle methods.

* Hooks are available for React version 16.8 or higher.
* Hooks are completely opt-in. Use it partially for a few components or base the whole project on it as per your needs without rewriting any existing code.
* Hooks don’t contain any breaking changes and are 100% backward-compatible.
* The react team has no plan to remove classes from React.

**WITH HOOK WE CAN USE CLASS COMPONENT FEATURES IN FUNCTIONAL COMPONENTS COMPONENTS SUCH AS STATE ,LIFE CYCLE, PURE COMPONENT.**

**JSX**

Stands for JavaScript XML. It is simply a syntax extension of JavaScript. It allows us to directly write HTML in React (within JavaScript code). It is easy to create a template using JSX in React, but it is not a simple template language instead it comes with the full power of JavaScript.

It has tag name,makes react code simpler

const element = <h1>Welcome to tgc.</h1>;

**Props and State**

**State is owned locally and the component itself updates it.** **Props are owned and read-only by a parent**

Props are passed to components via HTML attributes. props stands for properties.