**JavaScript Concepts**

* JS is single threaded. It takes only single core of the computer to work. This is why it is called that JS is a less powerful language.
* Loosely Typed language – Type of variable is not required.
* Asynchronous in Nature – Means control shifts once that task is done
* Does setInterval or setTimeout always works/gets executed in the specified time?  
  Answer is no. Since JS is single threaded, if the system in between gets involved in expensive task, then these setInterval or setTimeout functions may get delayed.  
  You can play and watch how setTimeout/setInterval/async functions utilise Call Stack, Web Apis and Callback Queue for execution and control flow. Also called Event Loop. 🡪 [Link](http://latentflip.com/loupe)

**REACT JS Concepts**

* React as reconciler.
* Virtual DOM? Why virtual DOM? How used in optimisation?
* VDOM maintains total structure of DOM in variables and calculates diff and update. If there would be no VDOM, then we need to read DOM again and again to calculate diff which would be very costly.
* VDOM helps to batch the updates. DOMS are actually updated after certain interval of time, not instantly. (If too many changes are taking place frequently in complex applications, VDOM helps to batch these updates rather than manipulating DOM constantly)
* Bug : setInterval, lets assume set for 1 sec, but it getting called more than one time in 1 sec. Why?  
  Because each time the component is repainted new setInterval (or setTimeout) instance is called is called and hence runs infinitely many times. But it should be called at once only.   
  One way to remove this error is to use if else and use global flag variable.   
  But in react this is generally fixed using the same inside useEffect hooks.
* State Management :   
  Dirty way of doing is that keep everything on the top component i.e., app.js. But this is not good because if there is any change then the all component will get rendered. So its advised to keep the state management in the lowest common ancestor wherever it is necessary. This is why state management library is preferred over inbuilt context-api in REACT.
* Note: Try not to hard reload the page when login/logout. Use state management in a more efficient way.
* use Custom Hook to fetch data from backend. This increases reusability and code readability. - Custom fetch hook