**Context in React**

1. To avoid props drilling in React we use Context API

2. Context is used for transfer your props or data from parent comp to child comp or final comp without passing down the

props at every entry level of React comp

3. In Context, react provide one method called createContext()

1. Context have two properties -

i. Provider - Parent Comp i.e App.js

ii. Consumer - Child Comp

b. In Context, we use Provider component and it wrap around the child comp

c. In context, we use value keyword for transferring the data

**useContext () hook -**

1. Hooks are like a predefined function

2. This hook has its own syntax

3. Hooks are only used in functional comp

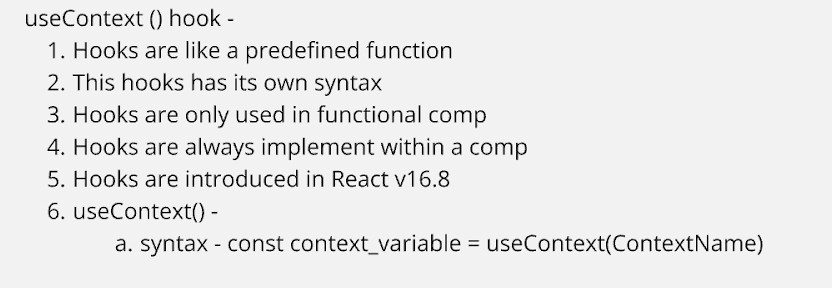
4. Hooks are always implemented within a comp

5. Hooks are introduced in React vl 6.8

6. useContext() -

syntax - const context\_variable = useContext(ContextName)

7.useContext() hook return context value.

**App.js**

import { createContext } from 'react'; import './App.css';

import ChildA from './childComp/ChildA'

const appName = createContext() const name = createContext() const abc = createContext() const xyz = createContext()

//Provider - Parent Comp

//Consumer - Child Comp function App() {

return (

<div className="App">

<h1>Props Drilling In React</h1>

<h2>Parent Comp - App.js</h2>

<appName.Provider value='Welcome to React'>

<name.Provider value='Jordan Walke'>

<abc.Provider value='Good Morning Everyone'>

<xyz.Provider value='React v18.3'>

<ChildA />

</xyz.Provider>

</abc.Provider>

</name.Provider>

</appName.Provider>

</div>

);

}

export default App;

//Named Type

export {appName, name, abc, xyz}

**ChildA.js**

import ChildB from './ChildB' const ChildA = () =>{

return(

<>

<h2>I am ChildA Comp</h2>

<p>Msg</p>

<ChildB/>

</>

)

}

export default ChildA

**ChildB.js**

import ChildC from './ChildC' const ChildB = () =>{

return(

<>

<h2>I am ChildB Comp</h2>

<p>Msg - </p>

<ChildC/>

</>

)

}

export default ChildB

**ChildC.js**

import ChildD from './ChildD' import { name } from '../App' import { useContext } from 'react' const ChildC = () =>{

const devname = useContext(name) return(

<>

<h2>I am ChildC Comp</h2>

<h3>Name - {devname}</h3>

<ChildD/>

</>

)

}

export default ChildC

**ChildD.js**

import { useContext } from "react" import { appName } from "../App" import { abc } from "../App" import ChildE from "./ChildE"

const ChildD = () =>{

const msg = useContext(appName) const greet = useContext(abc) return(

<>

<h2>I am ChildD Comp</h2>

<h3>msg - {msg}</h3>

<h3>Greeting of the day - {greet}</h3>

<ChildE/>

</>

)

}

export default ChildD

