React Js Imp Topics

1) Definition of React Js?

Ans) React js is **JavaScript library** and it is used to **build user interface**.

2) Differences between Angler, React and Vue js?

Ans) **Angler:**

1) Component based on **UI** framework uses like **TypeScript**

2) All the **packages are Default Installed.**

React Js:

1) Component based on **UI** framework uses like **JavaScript**

2) We need to install all The Packages.

Vue Js:

1) Component based on **UI** framework uses like **TypeScript**

2) All the **packages are Default Installed.**

3) Explain About Class Based Component?

Ans) **Definition:**

Class components are more complex than functional components. It requires you to extend from React. Component and create a render function which returns a React element.

The best future for class based component is **state** and **life cycle**

- Before 2019 most of the people use class based component.
- After 2019 used functional component because react got updated version (**React 16.8**) They provided best future it's called **React Hooks.**
- React Hooks futures are Intergration both **state** and **life cycle.**
- We Can use both future (**state and lifecycle**) in one component.

Example Program:

```
import React, { Component } from 'react'
      export default class App extends Component {
         state={
               name:"Koushik",
               }
     render() {
       return (
             <div>
                 <center>
                       <h1>Hai i am {this.state.name}</h1>
                 </center>
             </div>
   )
  }
}
```

4) Explain About Function Based Component?

Ans) ## Definition:

A **React functional component** is a simple JavaScript function that accepts props and returns a React element.

The Best future of Function based component is react hooks which was introduces in react 16.8 version

- Before 2019 most of the people use class based component.
- After 2019 used functional component because react got updated version (**React 16.8**) They provided best future it's called **React Hooks.**
- React Hooks futures are Intergration both **state** and **life cycle.**
- We Can use both future (**state and lifecycle**) in one component.

5) Explain About State?

Ans) **Definition:**

To Declare multiple variables in one component we use state.

Example Code:

```
import React, { Component } from 'react'

export default class State extends Component {
   state ={
      name : "koushik",
   }
 render() {
   return (
     <div>
       <h1>{this.state.name}</h1>
     </div>
   )
 }
}
```

5) Explain About Props?

Ans) Definition:

To send value from one component to another component we use props.

Example Code:

Props

```
import React, { Component } from 'react'
import Props1 from './Props1'
export default class Props extends Component {
    state ={
        name :"Hello this is props"
    }
  render() {
    return (
      <div>
        <Props1 data={this.state.name}/>
      </div>
    )
  }
}
```

Props1

```
import React, { Component } from 'react
export default class Props1 extends Component {
  render() {
    return (
      <div>
        {this.props.data}
      </div>
    )
```

}}

6) Explain About Onclick?

Ans) Definition:

The onClick event generally occurs when we use click on event.

It Allows the program to execute a function when an event gets clicked.

Example Code:

```
import React from 'react'
const OnClick = () => {
  return (
    <div>
      <button onClick={()=> console.log("why did you touch me")}>Touch Me</button>
    </div>
  )
}
export default OnClick
```

7) Explain About useState?

Ans) Definition:

- We need to use Hooks in Functional Component only not in class component.
- useState is React Hooks that allows you to add state to a functional component.
- It returns an array into two values.
- The current state and a function to update it.

Example Code:

```
import React, { useState } from 'react'
const Usestate = () => {
    const[name,setName]=useState("")
    const[count,setCount]=useState(0)
  return (
    <div>
```

```
      <h1>{name}</h1>
     <button onClick={()=>setName("How dare you to touch me")}>Touch Me</button>
     <h1>{count}</h1>
     <button onClick={()=> setCount(count+1)}>Increment</button><button onClick=  {()=> setCount(count-1)}>Decrement</button>

   </div>
  )
}

export default Usestate
```

8) Explain About useEffect?

Ans) **Definition:**

- To Execute the program after the return statement we use useEffect.
- It is a Lifecycle method.
- Lifecycle is nothing but to execute function one after another function.
- The useEffect Hooks allows you to perform side Effects in your components.

Example Code:

```
import React, { useEffect, useState } from 'react'
const UseEffect = () => {
   const[count,setCount]=useState(0)
   useEffect(()=> console.log(count),[count])
  return (
   <div>
      <h1>You Have Clicked on Button {count} Times</h1>
      <button onClick={()=>setCount(count+1)}>Clicked</button>
</div>  ) }  export default UseEffect
```

8) Explain About onChange?

Ans) Definition:

- The onChnage event in react detects when the value of an input element changes.

Example Code:

```
import React, { useState } from 'react'
const OnChange = () => {
   const [user,setUser]=useState("")
   const handler = e =>{
      setUser(e.target.value)
   }
  return (
   <div>
     <input type='text' placeholder='username' value={user} name='user' onChange={handler}/>
     <br/>
     {user}
   </div>
  )
}
export default OnChange
```

9) Explain About onSubmit?

Ans) Definition:

- The user provided data to submit data to the backend.

Example Code:

```
import React, { useState } from 'react'
const OnSubmit = () => {
   const [data,setData]=useState({
```

```
        user:'',
        password:'',
    })
    const {user,password}= data;
    const handler = e =>{
        setData({...data,[e.target.name]:[e.target.value]})
    }
    const submitHandler = e =>{
        e.preventDefault();
        console.log(data);
    }
return (
    <div>
      <form onSubmit={submitHandler}>
      <input type='text' placeholder='username' value={user} name='user'
onChange={handler}/>
      <br/>
      <input type='password' placeholder='password' value={password}
name='password' onChange={handler}/>
      <br/>
      <input type='submit' name='submit'/>
      <br/>
      </form>
    </div>
  )
}
 export default OnSubmit
```

10) Explain About Map Function?

Ans) **Definition:**

- To do array of value hydrate and print into individual component or to use in individual component.
- Not only array of value array of object hydrate and print into individual component or to use in individual component.

Example Code:

```
import React from 'react'
const Map = () => {
   const arr1=["heelo Roomates"]
   const arr=[
      {
         id:1,
         title:"Aijay"
      },
      {
         id:2,
         title:"Santhosh"
      },
      {
         id:3,
         title:"Koushik"
      },
      {
         id:4,
         title:"Room"
      }
   ]
  return (
   <div>
```

```
        {
          arr1.map(
            (value1, index) =><h1 key={index}>{value1}</h1>
          )
        }
        {
          arr.map(
            (value) =><li key={value.id}>{value.id}  {value.title}</li>
          )
        }
    </div>
  )
}
 export default Map
```

11) Explain About Filter Function?

Ans) Definition:

- To filter array of values and array of objects we can use it.

Example Code:

```
import React from 'react'
const Filter = () => {
    const arr=[10,20,30,40,50,60]
    const filterd =arr.filter(item=>item>20)
    const name=["naveen","anil","anwar"]
    const frilterd1=name.filter(name=>name.includes('a'))
  return (
    <div>
      {
```

```
        filterd.map(
            (item,index)=><li key={index}>{item}</li>
        )
      }
      {
        frilterd1.map(
            (name,index)=><li key={index}>{name}</li>
        )
      }
    </div>
  )
}
export default Filter
```

12) Explain About Fetch?

Ans) **Definition:**

- To access Api data we use Fetch.

Example Code:

```
import React, { useEffect, useState } from "react";
const Fetch = () => {
  const [data, setData] = useState([]);
  useEffect(() => {
    fetch("https://jsonplaceholder.typicode.com/posts")
      .then((res) => res.json())
      .then((json) => setData(json));
  }, []);
  return (
```

```
    <div>
      <table>
        <tr>
          <th className="text-center">S.no</th>
          <th className="text-center">Title</th>
          <th className="text-center">Body</th>
        </tr>
        {data.map((item) => (
          <tr key={item.id}>
            <td className="text-center">{item.id}</td>
            <td className="pl-3 pr-3">{item.title}</td>
            <td className="pl-3 pr-3">{item.body}</td>
          </tr>
        ))}
      </table>
    </div>
  );
};
export default Fetch;
```

13) Explain About Axios?

Ans) **Definition:**

- To access Api data we use Axios.
- It is a library while using this library we can access any http request like get , post, delete.
- To use axios we to instal it first.
- To install **npm install axios**.

Example Code:

```
import React, { useEffect, useState } from "react";
Import axios from “axios”
```

```
const axios = () => {
  const [data, setData] = useState([]);
  useEffect(() => {
    axios.get("https://jsonplaceholder.typicode.com/posts")
      .then((res) => res.json())
      .then((json) => setData(json));
  }, []);
  return (
    <div>
      <table>
        <tr>
          <th className="text-center">S.no</th>
          <th className="text-center">Title</th>
          <th className="text-center">Body</th>
        </tr>
        {data.map((item) => (
          <tr key={item.id}>
            <td className="text-center">{item.id}</td>
            <td className="pl-3 pr-3">{item.title}</td>
            <td className="pl-3 pr-3">{item.body}</td>
          </tr>
        ))}
      </table>
    </div>
  );
};export default Axios;
```

14) Explain About React Router?

Ans) **Definition:**

- To Develop multiple pages we use it.
- To use this method we need to install first.
- To install **npm install react-router-dom.**
- Import {BrowserRouter,Switch,Route}.
- Import {Link}.

15) Explain About UseContext?

Ans) Definition:

- To Send pro values from one component to another component.
- To get code click on link (https://www.youtube.com/watch?v=TcWB3L4UFgE&list=PLWnZ0qt0PImVaDkDbF96dnRGO0_lXVLKf&index=36)

16) Explain About UseRef?

Ans) Definition:

- It is used to get data from user and assign the value proper to variable
- **UseState** will re-render when the content change and update in ui
- **UseRef** doesn't notify you when its content changes.

Example:

```
import React, { useEffect, useRef } from 'react'
const Useref = () => {
   const data =useRef()
   const submitHandler = e =>{
      e.preventDefault();
      console.log(data.current.value)
   }
   useEffect (()=>{
      data.current.focus();
   },[])
  return (
   <div>
      <center>
```

```
        <form onSubmit={submitHandler}>
            <input ref={data} type='text' placeholder='Exnter Your Name' />
            <br />
            <input type='submit' />
        </form>
        </center>
    </div>
  )
}
export default Useref
```

17) Explain About Higher-Order-Componenet?

Ans) Definition:

- A Higher-Order-Component (HOC) is an advanced technique in React for Reusing component logic.
- Hoc is a pure function with zero side-effects and doesn't modify the input component.
- Higher-Order-Component is a function that takes a component and returns a new component.

Git-Hub

TO Push Code

→ git init
→ Git add .
→ Git commit –m "name"
→ Git remote add oringin "url"
→ Git git remote –v
→ Git push –u Oring master
→ Or git push –f orign master

TO Pull (Get) Code

→Git clone “url”

TO Remove Reposiiter

→Git remote remove oringin