## Package.json in react JS

name, version, dependency, start command

## ​ component in react

piece of reusable code/ function ex header, footer, banner, menu

Type of components

* Function components
* Class components
* high order components
* pure components
* controller components
* uncontrol components

class Welcome extends React.Component {

render() {

return <h1>Hello, {this.props.name}</h1>;

}

}

Component Lifecycle

* Mounting(constructor(),render(),componentDidMount())
* Updating
* Unmounting

## ​ class component in react JS

See demo UserCc

## ​ what is JSX in react

* JavaScript xml it use as normal html/xml tag
* Angular & react can work together

## ​ Click event in react JS

onClick={Submit} where submit is a function

## ​ State in react functional component

State basically an object

import {useState} from 'react';

let [message,setmessageState]=useState;

function UpdateSubmit{

setmessageStatemessage+

}

## ​ state in class component in react

## ​ props in the functional component in react

props are the property

## ​ props in class component

Pass data from one comp to other

## ​ get input box value

## ​ show and hide element toggle

## ​ basic form tutorial

## ​ if-else conditions

## ​ pass function as props

## ​ introduction to react JS life cycle

Each component in React has a lifecycle which you can monitor and manipulate during its three main phases. The three phases are: Mounting, Updating, and Unmounting.

## mount updates unmount

## ​ constructor life cycle method

## ​ render life cycle in React JS

## ​ component did mount react JS life cycle

## ​ component did update react JS life cycle

## ​ Should Component update in react JS life cycle

## ​ componentWillUnmount in react JS

## ​ Hooks in react JS

State,lifecycle,pure component.hook start with use key word (useEffect,useRef, useState, useContext)

## ​ useEffect hook in react JS

## ​ useEffect with state and props in react JS

## ​ Style type in react JS

## ​ install bootstrap in react JS

npm install react-bootstrap bootstrap

## npm install react-bootstrap bootstrap

## ​ Handle array with list in react JS

## ​ List with bootstrap in react JS

## ​ Nested list with a nested array in react tutorial

## ​ How to reuse component in reactJS

## ​ Nested list with a nested array in react tutorial

## ​ React JS send data child to parent

## ​ Pure Component in React JS

## ​ Use memo in react functional component

## ​ Ref in React JS

## ​ useRef in react functional component

## ​ React JS forward ref

## ​ React JS Controlled Component

has input filed

## ​ React JS Uncontrolled Component

has input filed. handle directly through DOM

## ​ React JS High order component HOC

## ​ React JS Routing basic

## ​ React JS Routing example

npm i react-router-dom

## ​ 404 page in react routing

## ​ React JS Dynamic routing

## ​ React JS Call Get Method API- HTTP call

## ​ What is postman

## ​ How to call post method API in react JS

## ​ React JS delete method API

## ​ React JS pre-filled form

## ​ React JS update API method

## ​ React JS get the previous state in the functional component

## ​ get previous props in react JS with hooks

## ​ Context API in react JS

Data transform

## provider-set the data

## ​ React JS interview Questions

## react js component to to component communication

* Parent-to-child with props
* Parent-to-child with props
* Functional props
* Passing arguments to functional props
* Passing functional props through Router

## react js project structure

* Assets Folder
* Components Folder
* Middleware Folder
* Routes Folder
* Config Folder
* Utils Folder
* Api
* packages.json
* package version,depence
* packages-lock.json
* package version history
* git ignour
* app.js
* app.test.js
* index.js
* entry point
* setupTests.js
* all text case
* public
* manifost.json

## Security in React

* XSS
* Broken Authentication
* Disable HTML markups

## Improve performance of React

* webpack
* Using a CDN
* Lazy loading
* Code Splitting in React

Yarn is alternative npm

npm start

index.js---load first

app.js