MERN Stack

Phase 2 : Notes

26—7-2021

Day 1

DOM

**JavaScript**

**gitignore**

.gitingore

node\_modules

documents.txt

**JavaScript concept**

**DOM**

**Interact with user**

**DOM :** Document Object Model

**index.html**

<html>

<head>

<title>This is my web page</title>

</head>

<body>

<p>Welcome to HTML Web Page</p>

</body>

</html>

DOM Hierarchy generate in browser memory.

Html

Head body

Title p

textNode textNode

This is my web page Welcome to HTML Web Page

**DOM API :** Document Object Model Application interface.

All programming language like Java, Python, Asp.net as well as JavaScript which provide set of function or methods or classes which help to read, write and update html contents dynamically.

React JS: React JS a open source JavaScript library which help to create web application or pages using components.

Component is use to control the view or part of the view on web page.

For Example

Before React JS and Angular Framework

In multi page application whole page loaded once again in browser ie DOM.

**Index.html welcome.html**

Hyperlink

Submit

Button

Using

JavaScript

**AJAX : Asynchronous JavaScript and XML.**

**SPA (Single Page Application)**

Using SPA rather than loading whole page (DOM) we can load the part of the web page.

Angular and React JS is use to create a SPA.

Angular is a framework, Angular framework base upon MVC(Model View Controller). Angular is part of google.

React JS is library. React JS is View in MVC. React JS is a part of Facebook.

Library provide set of function to read, write and update the DOM(HTML Tags contents) elements. Library is not standard. Library is focus on only one type of task. Library is light weighted.

Framework contains lot pre-defined API which help to do multipurpose. Framework follow standard (design pattern). The implementation of all design pattern is taken care by framework.

MVC :

Model View Controller

View -🡪 Presentation Logic : Look and feel.

Controller -🡪 Intermediate between Model and View

Model ---🡪 Business logic of the application

Compare Angular why is React JS.

Angular is very heavy framework. React JS is light weighted library. Base upon the application requirements we can add extra library to get features. We can make react JS standard using Redux or flux library. Using React we can create mobile application React Native.

React JS provide Virtual DOM. Angular doesn’t provide. React JS is faster than Angular.

React JS doesn’t support 2 way data binding.

We can create react application using different ways.

1. Using online editor
2. Using Node JS

**Node JS : Node JS is a run time environment for JavaScript library or Framework.**

**With node we will get npm (node package manager).**

**Using npm we can download node js external module.**

To create the React JS Project we have to install the react plugin using npm command.

**npm install –g create-react-app**

create-react-app if one of the node js external module which help to create the react js projects.

Syntax to create the react js project

create-react-app project-name

**create-react-app welcome-app**

Move inside a project folder using command

cd project-name

cd welcome-app

To run the project we have to run the command as

npm start

After compiled project successfully then default browser it open with URL as

<http://localhost:3000>

React JS is base upon the components. Component control the view of part of the view in web page

In React JS we can create the component using different ways

1. Using Normal function style ES5
2. Using arrow function style ES6
3. Using ES6 class style

Creating component using function es5 style

function App() {

}

function MyComponent() {

}

In React Every component return JSX.

JavaScript and XML

Inside HTML we are writing JavaScript code using script tags.

<html>

<head>

</head>

<body>

<script type=”text/JavaScript”>

document.write(“Welcome to JS”)

<h1></h1>

</script>

</body>

</html>

Normal JavaScript function return variable values

function add(a,b) {

var sum = a+b;

return sum;

}

function sayHello() {

Return “Welcome”;

}

In JSX if we want to return any single tag the syntax is

<tagName>Contents </tagName>

If we want to return multiple tags then we have to wrap those tags in outer most tags ie div or any others.

<outerMostTag>

<tagName>Contents </tagName>

<tagName>Contents </tagName>

<tagName>Contents </tagName>

</outerMostTag>

Html, table, div, p, h1 to h6 etc.

All html tags can understand by browser. To run html we no need to install any software.

Using Angular and React JS we are creating user-defined tags with help of components and those component return JSX code.

<MyTag></MyTag>

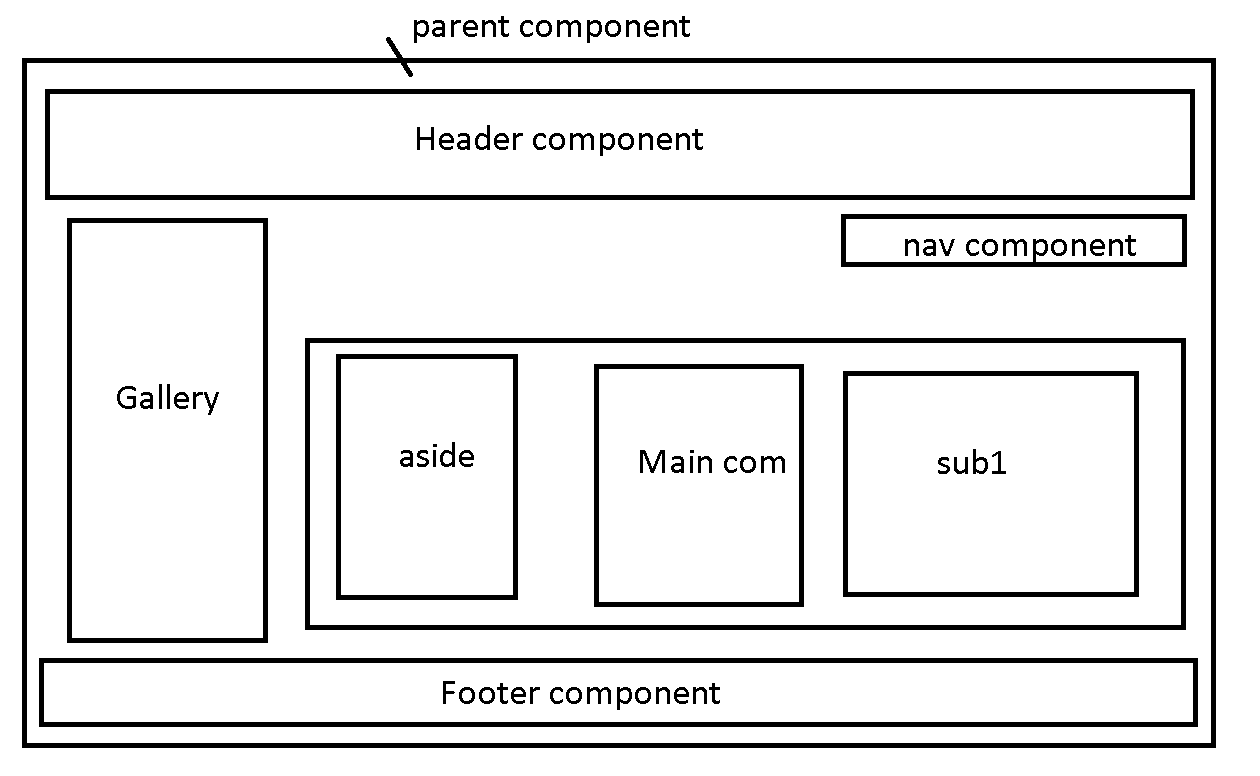
import and export

using import and export we can share function, classes, variable from one file to another files.

React JS provide two pre-defined API

React : This API is use to create the component.

React-DOM : This api is use to pass or send Parent component JSX code to Actual DOM on view side. It provide bridge between React component to Actual DOM.



**ReactDOM.render(<ParentTagName/>,document.getElementById(“root”))**

27—7-2021

Day 2

It is good practise every file must contains only one components.

Creating the component using arrow function style

Syntax

let ComponentName = ()=>JSX Data Data

let Info = ()=><div>Welcome to Arrow component</div>

From ES6 onwards we can use class keyword to create the objects.

So In React JS We can create the component using class syntax ie

Class component

We have to create normal class and that class must be extends React.Component pre-defined class part of react library.

We have to override render() function which help to return the JSX data to view.

object : object is a any real world entity.

have--🡪 state or properties 🡪variable, fields

Person

do/does 🡪 behaviour🡪 functions/methods

Employee

Bank

Car

Animal

To implements object different programming language provide different syntax.

Till ES5 to describe the object we were/are using function keyword.

From ES6 onwards we can use class keyword.

React JS state and props

React Component provide two types of variable ie

state and props

state :

1. state variable is use to describe the component behaviour.
2. State variable is local to that components. We can’t use in another components. State variable consider a private variable to that component where it declare or created.
3. state variable we can’t change or update directly. To do the changes on state variable we have to use the setState() pre-defined functions. Ie state variable is known mutable properties.

props:

1. props is short cut of properties.
2. Props we can’t change ie immutable variables
3. Props is use to pass the value from one component to another components. The relationship between two component must be parent and child. So using props we can pass the value from parent component to child components.
4. Using component attribute and props concept we can pass the value from one component to another components.

If component is created using class syntax (ES6 style). we can use state and props both variable.

If component is created using function syntax (ES5 or ES6(arrow)). We can use only props no state.

React Component are divided into two types.

1. Statefull component : class style component is known as statefull component. Because we can use state and props variable.
2. Stateless component: function or arrow style component is known stateless component. Because we can use only props.

In new Version from React 17.x. React JS provide hooks methods or functions. Using hook method or function we can use state variable in functional style components.

**Function component created using normal function style or arrow style. It is use to receive the props as a parameter and return the JSX.**

28—7-2021

Day 3

React Login

Adding dynamic state property using setState() function

**Independent state property**

this.setState({

…this.state,[newStateName]:stateValue

});

**Nested state property**

this.setState({profile:{

      ...this.state.profile,[ newStateName]: stateValue

}})

**Conditional rendering**

29—7-2021

Day 4

React JS provide external third party library ie axios. Which help to backend technologies REST API Service develop in any language doesn’t matter.

Req(http) java req

Client Amazon Paytm python

Res(http) Java XML Google pay .net

JSON Phone pay :node

python Paypal :php

res Net banking

Web service : giving the service for web application when both application running using different technologies.

1. SOAP Web Service : Simple Object Access Protocol

We can send and receive the data only in the form of XML. XML is heavy

1. REST Full Web Service: We can consume and produce the data xml as well as non xml like json, text, html or any other format. JSON is light weighted.

REST API (Application Programming Interface)

Representational State Transfer

To call REST API develop in any language React js provide axios third party module.

npm install axios

after install using axios we can call get(), post(), put() and delete() methods.

axios all methods return type is promise object.

If function return promise object then we have use then() and catch() function to load successfully data or failure message.

Promise : promise is use to handle asynchronous data or events. Promise can be resolve(success) or reject(failure). If promise resolve then call else catch call.

Node JS provide json-server type of external module which help to create static json file as a server.

In that file we can add, update, delete and retrieve json records.

npm install –g json-server

Mac user

sudo npm install –g json-server

Run the file

**json-server employee.json Then it run in default port number 3000**

**if you want different port number**

**json-server employee.json --port 3001**

After running the server please open the browser

And check URL

<http://localhost:3000/employees>

**Component life cycle**

React component can be created dynamically.

They get created, rendered, added to dom, update in virtual dom then update to actual dom and remove.

Component life cycle divided into 3 phases

Mounting : constructor, componentDidMount()

Updating : render(), componentDidUpate()

Un Mounting : componentWillUmount()

Constructor it call only once.

componentDidMont() : it call at beginning after that if we do any changes state variable it will call once again.

Render(): it will call again and again whenever we do any changes in state variables.

componentDiUpdate(): it call after render to verify state change happen on actual dom or not.

componentWillUnmount() : this function is use to clean any resource before that particular component get destroy.

React JS with CSS or bootstrap

**Component communication**

**React with Redux**

Hook function

**Day 5**

**01-08-2021**

Every component can contains one or more state variables. State variable is private to that components.

Parent Component -🡪App

State variable

Child1 Child2

State variable state variable

Passing the value from parent to child 🡪 using props

Passing or receive the value from child to parent 🡪props with callback function.

We can pass the parent component state variable value to child component through props.

If parent component want to access child component state variables. So child component has to pass the value with help of props and callback function to parent component.

Component1

Component2

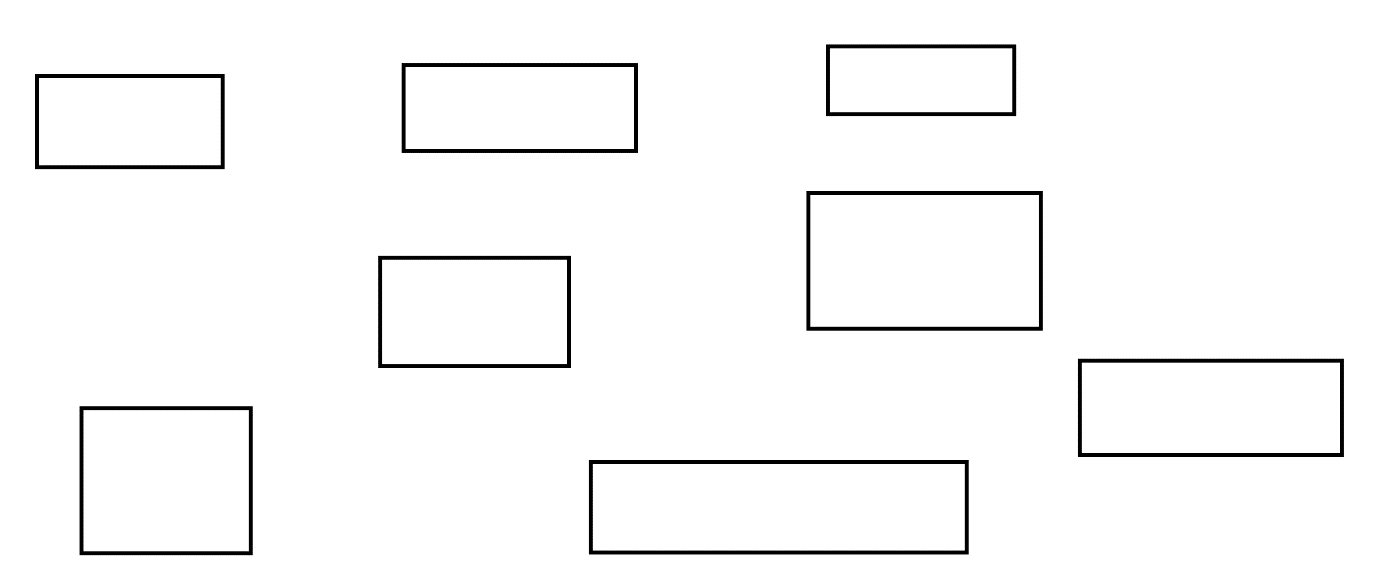
Component3

Component4

Component5

We can share the data from one component to another if component maintain very big hierarchy

1. Using context API. : we can pass the value from parent to child.
2. Using Redux : we can share the data from more than one component doesn’t matter what is the relationship between two component may be parent to child, child to parent or sibling etc.



Redux : Redux state management container tool. Redux provide store concept which help to store state. Every component doesn’t matter what the relationship between two component. The connect to store to retrieve the state, update the state, delete the data or create the state. Component can’t do any change on store directly we have to take the help reducer which takes two parameter action and initial state. So base upon the action they do the change on state.

Reducer is normal JavaScript function which takes initial state and action with payload to do the change on store base upon the action.

**Store** : Like a database which help to store the global state variable.

**Reducer** : is like a table which help to store the data in store.

**action and dispatch** : like a query which help to the changes base upon type of query.

**Records is like state may be id, name, salary**

**Examples**

First create the project using command as

**create-react-app react-with-redux**

npm install redux

npm install react-redux

create class component

export default class App extends Component {

  render() {

    return (

      <div>

        <h2>React with Redux</h2>

      </div>

    )

  }

}

Now create another js file and give the name as **reducer.js**

Reducer is just like a normal JavaScript function which takes two parameter. 1st parameter is initial state and 2nd parameter is action which help to the changes on state base upon the action.

InitialState return from reducer

const initialState = {

    id:100,

    name:"Raj",

    salary:12000

}

export default function reducer(state=initialState,action){

    //change the state base upon the action

    return state;

}

Now in **index.js** file we will create the store.