Day 1 : 03-04-23

Testing :

Jasmine and Karma

Cucumber

Angular routing

Angular binding

Frontend technologies

https://[www.google.com](http://www.google.com) -🡪 URL (Uniform resource locator)

req(http/https)---🡪

Client Server

🡨-----Res(http/https) html

Html 1,2,3,4,5

Css (cascading style sheet)

JS

Static web page / dynamic web page

<tagName>

</tagName>

DOM : document object model

Index.html

Dom hierarchy

Html

Head body

Title div content (textNode)

Meta

Script

Style

Types of css

Inline css

<tagName style=”property:value;property:value;”></tagName>

P,h1,h6,div,span,body etc

Internal css

<style type=”text/css”>

Selector{property:value}

</style>

1. Universal selector : \*
2. Specific selector : tagName{property:value}
3. Class selector
   1. Local class selector
   2. Global class selector
4. Id selector

Class selector vs id selector

<p class=”abc” id=”a1”>first</p>

<h1 class=”abc” id=”a2”>second</h1>

<p class=”xyz” id=”a3”>third</p>

<p class=”abc”>fourth</p>

<p class=”xyz”>fifth</p>

Class : means group of the tags which have same name or different names.

Id : ever tag we have to provide unique id which help to read, write and update dom using javascript.

External css

Every tag

We have attribute as

Name, class, id, style

<input type=”radio” name=”gender” value=”male”/>

<input type=”radio” name=”gender” value=”female”/>

Bootstrap is open source css responsive web framework.

Bootstrap provided lot of pre defined classes base upon dom element to apply formatting style for web page.

Like

Button classes

Form classes

Table classes

Container div tag

Alert classes

Etc

Grid layout :

It is use to arrange the html component in table format using row and column

By default in grid layout every row divided into 12 columns.

JavaScript

JavaScript was object based interpreter scripting language. JavaScript was/is use to do validation on client side.

ECMA or ES . ECMA is a concept. JavaScript is a one of the implementation of ES

ES5 JS.

Now a day we can do validation using javaScript as well as html 5 features.

<input type=”text” name=”emailid” required/>

MEAN Stack or MERN Stack or node js

Using JavaScript we can do programming on web page without server.

In JavaScript we can declare the variable using var keyword

let and const

till ES5 javascript to declare the variable we are using var keyword but from ES6 onward to declare variable we can use let as well as const.

operator

if statement

looping

function

Day 1 : 03-05-23

JavaScript events

Event is a interaction between user and component (html dom elements like p, div, button, radio button checkbox etc).

Event provide the bridge between html and javascript code.

Type of events

In JavaScript all event start with pre fix on followed by event name

Like

onClick

ondblclick

onMouseOver

onMouseOut

onKeyUp

onKeyDown

onSubmit

onChange

onFocus

onBlur

onLoad

onUnload

JavaScript object

Object : object is any real world entity.

Property or state -🡪 have -🡪variable, number, string, Boolean etc

Person

Behaviour -🡪do/does -🡪 functions or methods.

Bank

Car

Customer

Manager

In JavaScript object are divided into two types

Pre defined object

BOM : Browser Object Model

DOM : document object model

User defined object

<input type=”button” value=”Click here” onClick=”fun()”/>



JavaScript follow object hierarchy

Object property

Behaviour

Object property

Behaviour

Object property

Behaviour

Object

DOM API Document object model application programming interface.

Lot of programming language like Java, python, C#, JavaScript provided lot pre defined function or methods which help to read, write and update dom dynamically.

JavaScript : using JavaScript read, write and update dom become more complex when we use enterprise application.

jQuery : jQuery is external Javascript library which provided lot of pre defined function which help to read, write and update dom easily.

jQuery doesn’t follow any standard rules.

Angular is a framework.

If we develop any application using framework. Framework internally follow standard rules. 70 to 80% task is taken care by framework. Design pattern. Best practise or solution of repeating problem.

The implementation of design pattern is taken care by framework.

Multi page application

Own dom own dom

Index.html welcome.html

Hyper link

Form submit button

Using JavaScript

SPA (Single page application )

In single page application rathern than loading whole page from scratch we load only part of the web page.

Angular Framework

React JS is library .react JS doesn’t follow standard. We can make rest js standard with react with redux.

Gmail page

Inbox

Outbox

Sent

MVC :Model View controller / component

Node JS

Html, css, JavaScript, bootstrap

Server side technologies

Java

Servlet/jsp , spring framework and spring boot

Asp.net

Php

Python

Node js

Before node js JavaScript is known as client side scripting language which help to create web application on frontend side.

But after node js using JavaScript we can create server side technologies. Means after node js javascript also known as server side scripting language.

MEAN or MERN

Angular framework is base upon typescript. Typescript is super set of JavaScript. Browser doesn’t understand typescript we need to convert ts to js.

Npm (node package manager) which help to download external node js module.

Ng (Angular)

MVC

React is only view

node --version

npm --version

npm install -g @angular/cli : this command is use to enable ng command.

ng version

to create the angular project

ng new project-name

routing 🡪no

styling 🡪 css

cd project-name

open the project in vs code

ng serve after compiled 100% we need to open the browser and type <http://localhost:4200>

or

ng serve -open

in project

expand

src

app folder

app.component.html -🡪 template page

app.component.css -🡪 external css page

app.component.ts 🡪 component

JavaScript as well as TypeScript

We can create user defined object using 3 ways

1. Literal style
2. Function style
3. Class style

Component is use to control the view or part of the view in web page. Using angular we are creating user defined tags with help of selector.

templateUrl attribute help to connect html page.

styleUlr : it is use to connect css file like external css link tag

app.module.ts file

module is a collection of more than one component. Module is like a package in java.

Declaration : in this section we have to provide details about more than one component

Imports : in this section we can import pre defined as well as user defined modules.

BrowserModule.

Providers: in this section we have provide angular service class details.

Bootstrap : in this section we have to provide parent component details.

HttpClient one of the pre defined api which help to call rest api develop in any language connected with any database

MEAN Stack

Mongo db

Express JS (rest api using javascript or typescript)

Angular

N node js

Day 3 : 03-11-23

App.component.html

App.component.css

App.component.ts

app.module.ts

main.ts

by default we have only one module and inside that module one component.

LoginModule CustomerModule ProductModule AppModule

signInComponent

signComponent

passwordChangeComponent

ng generate component componentname

or

ng g c component name

Data binding

Data binding provide the bridge between template and component. It is use to sync the data between template to component.

It divided into two types

1. One way data binding
2. String interpolation : it is a type of one way data binding . component to view

Component -------------------🡪View

Syntax

{{}} we need to use in html page

Example

{{variableName}}

{{10+50}}

{{functionName()}}

1. Property binding : it is a type of one way data binding. Component to view

Component ----------------🡪View

Syntax

[]

Component

lname = “Patil”;

in html page

<input type=”text”/>

<input type=”text” value=”Vijay”/> default value Vijay consider

<input type=”text” value=”lname”/> value itself is lname consider

<input type=”text” [value]=”lname”/> property binding

<p [innerText]=”lname”></p>

<div [innerText]=”lname”></div>

1. Even binding : event binding also type of one way data binding. Template to component.

Template --🡪 component (ts file)

Syntax

()

JavaScript event angular event

onClick (click)

onDblClick (dblclick)

onSubmit (ngSubmit)

event binding with property binding or string interpolation we can achieve two way data binding.

Template reference : it use to pass the value of textfield, radiobutton, checkbox etc from template or html page to components.

<input type=”text” #nameRef/>

# is prefix and reference name is nameRef.

1. Two way data binding

Two way data binding ie component to view and vice-versa. So we can achieve two way data binding using ngModel attribute. This attribute is a part of FormsModule. So while achieving two way data binding using ngModel we need to import FormsModule in app.module.ts file in import section.

Syntax

[(ngModel)]

Directives : directives is use to add the extra behaviour or functionality for existing dom or html page.

Types of directives

1. Component directive : using @Component decorator we are creating user defined tags with help of selector. Component is a type of directive which help to create user defined tags with help of selector.
2. Structure directive

Using structure directive we can add or remove html code from a dom.

1. ngIf
2. ngFor

with help of structure directive we can do if statement as well as looping in html page.

1. Attribute directive

ngStyle

ngClass

model class : it can be class or interface.

Object : object is any real world entity.

Product

Employee

Customer

Order

Login

To provide the entities information we need to create the model class.

We will work on product entity

ng g c product (component)

ng g class product (model class)

Or

Ng g interface product (model interface)

Day 4 : 03-12-2023

Interface is use to provide the specification

Class is use to provide implementation base upon the specification.

Typescript or angular

interface Bank {

accno:number;

withdraw():void; // in complete function

}

class Hsbc implements Bank {

accno:number=100;

withdraw() : void {

}

}

Create interface with only variable for one purpose

Create interface with incomplete function to provide specification.

interface Employee{

empid:number;

name:string;

}

let emp1 = {id:100,name:”Ravi”,age:21};

let emp2={srno:”abc123”,fname:”Mahesh”,city:”Bangalore”};

let emp3:Employee={empid:123,name:”Ravi”,}

let emp = new Employee();

interface Bank {

withdraw():void; // in complete function

deposit():void

transfer():void

}

class Hsbc implements Bank {

withdraw():void{

}

deposit():void

transfer():void

}

class Sbi implements Bank {

}

ng new angular-forms-service

routing 🡪yes

styling 🡪 css

if we want to pass the value from template to component we use template reference.

<input type=”text” #nameRef/>

If we want to pass the more than one or group of values then we can use angular forms.

Angular forms mainly divided into two types

1. Template driven forms
2. Model or reactive driven forms.

Template driven forms --🡪 flow of the application template to component

Easy to develop. Good for simple form.

In this form we use ngForm and ngModel attribute in template side. These attribute are part of FormsModule. So we need to import FormsModule in app.module.ts file in import section.

Model driven form -🡪 flow of the application component to template.

Complex to develop. Good for complex form.

In this form we use FormGroup, FormControl, FormArray api in component side and fromGroup, formControlName are attribute in template side. These api and attribute part of ReactiveFormsModule. So we need to import ReactiveFormsModule in app.module.ts file in import section.

ng g c tdf-login-page

ng g c mdf-login-page

In template driven we can create the form reference

<form #loginRef=”ngForm”>

</form>

Pre defined classes provided by angular to do the validation using template driven form as well as model driven forms.

Valid, invalid, touched, dirty, pristine

Template driven form

Model driven form

Promise : Promise is a pre defined object provide by JavaScript which help to handle asynchronous event of data.

Promise can be resolved or reject or pending etc.

Creating user defined promise

To handle the promise we have to use then() and catch()

If promise resolved then get called if promise rejected catch get call.

Backend technologies

Web Service :

Restfull web service.

XML or JSON.

Frontend technologies

JavaScript provided pre defined function ie fetch() which help to call rest api develop in any language

Plain javacript Fetch function return type is promise object.

Axios which help to call rest api with third party library

axios.get(“URL”).then().catch();

axios we use in react js

in angular we have to use HttpClient pre defined api to call rest api

httpclient return type is Observable. Observable is use to handle asynchronous event of data.

Rxjs : third party library we use in angular.

Angular service

If we write any business logic in component that logic become local to that component only.

MVC

View Component Model

Template component service

Tdf-login signIn() dtf-component

signIn()

Mdf-login signIn() mdf-component

We can create service class object using

1. New keyword
2. Using DI

IOC : Inversion of control :IOC is a concept. Rather than creating any resource explicitly allow to created by container. If container create it will maintain properly. As developer pull the resource from container, use and leave it.

DI Dependency Injection : DI is a implementation of IOC.

Angular support only one type of DI ie constructor base.

18-03-2023

ng new product-rest-api

routing 🡪 yes

styling 🡪 css

ng g c product component

ng g s product service

ng g class/interface product model class or interface.

If we are planning to call rest api develop in any language we need to use HttpClient api provided by angular. This HttpClient API we have to do DI in service or component.

AJAX :

Asynchronous JavaScript and XML

fetch(“url”).then().catch();

HttpClient all http methods get, post, put, patch, delete return type is Observable. It is part of rxjs.

Observable is use to handle asynchronous event of data.

But in observable we will use subscribe to handle the asynchronous event of data.

This method take 3 parameter as callback.

1st next : it is use to load the data one by one like loop.

2nd error : if any error generate at beginning or middle or last 2nd parameter get called to handle the error.

3rd complete : if no error third parameter get called after loaded all data successfully.

HttpClient API is a part of HttpClientModule so we need to import HttpClientModule in app.module.ts file.

REST API (Representational State Transfer).

Two actors

Admin : singin no signup

Username : [admin@gmail.com](mailto:admin@gmail.com)

Password : admin@123

Add Category/ View Category / Update Category / Delete Category

Add Product / View product / Delete product / Update product

View all customer / user details

View all customer order details.

Shipment

User / Customer : signin and signup

Sign in or Register / Signup

View All product details.

Add product to cart / Wishlist

Place the order one or more than one product

View its own order details.

Add account details

Payment : dummy payment gateway

Node js provided third party module ie json-server This module help us to run static json file as server.

npm install -g json-server

open the command prompt in the location and run the command as

json-server sampledb.json by default run on port number 3000.

<http://localhost:3000>

ng new e-shoping-admin-app

routing 🡪 yes

styling 🡪 css

ng new e-shoping-user-app

routing 🡪 yes

styling 🡪 css

open the e-shoping-admin-app in vs code

app.comonent.html : remove all the code

<div>

    <h2>E - Shopping Admin App</h2>

</div>

ng g c aboutus

ng g c contactus

ng g c signin

ng g c landing

routing is use navigate from one component template to another component template with or without condition.

Angular provided pre defined tag <router-outlet></router-outlet>. This tag is like a place holder which help to load the component page content base upon the path provided in routing file.

ng g s auth/auth service class

ng g class model/admin model class

19-03-2023

ng g c dashboard

auth guard : angular provide set of auth guard (they are interface which contains let of life cycle methods) which help to make restrict the user not to move component directly.

ng g guard auth/auth

ng g c logout

component communication :

if we want to share the data between one component to another components.

1. Parent child relationship 🡪 parent to child : @Input decorator
2. Child to parent relationship 🡪 child to parent @Output decorator with EventEmiter or @ViewChild decorator.
3. Sibling relationship :
   1. Session storage : html 5 features
   2. Common shared service
   3. Observable RxJS

ng g c product

ng g class product

ng g s product