Phase 4 : Testing and deployment

8 classes and each day 3 hour

Day 1 : 21-05-2022

Testing

Testing is use to find the defects or error or bugs in application.

Read a, b

Compute sum =a+b

Write b

Jasmine : Jasmine is type of open source web framework which provided lot of re-defined api which help to do the testing the JavaScript programs (it may be client side as well as server side(node js)).

Jasmine is use to do the unit testing.

In Testing

Test suite : it is like a container which contains more than one test cases. To make the function as a test suite all testing framework provided pre-defined function to make suite it

describe()

syntax

describe(“message”,callback);

describe(“message”,()=> {

})

Test case : test case help use to write the function which help to do the testing function functionality. To make the test case all testing framework provide pre-defined function ie

it()

syntax

it(“messag”,()=> {

})

describe(“message”,()=> {

it(“1st test case ”,()=> {

})

It(“2nd test case “ ()=> {

})

})

expect functions : Then jasmine provided lot of expect functions which help to check actual and expected output.

describe(“message”,()=> {

it(“1st test case ”,()=> {

coding…..

expect(expectedoutput).toXXX(actualOutput)

expect(expectedoutput).toXXX(actualOutput)

})

It(“2nd test case “ ()=> {

Coding….

expect(expectedoutput).toXXX(actualOutput)

expect(expectedoutput).toXXX(actualOutput)

})

})

Mocha

JEST

UI Testing

Plain Java Script Testing

Backend Testing

We can set the testing environment using sample testing template or using Node JS.

Please create the folder as testing , inside this folder create the as frontend and inside that folder create the folder as plain javascript testing

Life cycle function or hook of Jasmine

beforeEach(callback) : it is a life cycle hook it will call before each it function automatically.

beforeAll(callback) : it is a life cycle hook it will call before All it function automatically only once.

afterEach(callback) : it is a life cycle hook it will call after each it function automatically.

afterAll(callback) : it iwi life cycle hook it will call after all it function automatically only once.

Day 2 : 22-05-2022

UI Testing

Angular Testing : Angular framework use jasmine framework to do the angular api testing. Angular internally provided for us jasmine configuration.

Jasmine is testing framework which provide suite, spec and expect functions.

describe, it and more than one expect.

Karma: it is known as test runner for the Jasmine or any other testing framework.

But still angular internally use Karma as runner to run angular testing application.

Angular provide their api ie TestBed which help to do the testing for Angular component as well as service.

Please create new project

ng new angular-testing

routing -🡪 no

style 🡪 css

ng test to test the program

Resting backend technologies REST API

<https://fakestoreapi.com/products>

ng g s product service class

ng g class product model class

Day 3 :

28-05-2022

Mock object : we have to create the mock for the HttpClient. Fake or proxy or dummy data we have to provide.

To create the mock service we create the new service class

ng g s product-mock

Node JS (Server Side JavaScript testing)

We can use Server Side JavaScript testing using Jasmine or Mock with Chai also possible.

Inside testing folder

Create backend folder

Node js testing folder

Create the package.json file using npm init

Then install two modules

npm install jasmine -D

npm install jasmine-node -D

npm install jasmine --location=global

jasmine init (this command is use to create the spec folder in current project)

Testing using Node JS with Rest API

Create the folder as rest api testing

Create the package.json file using npm init

npm install jasmine -D

npm install jasmine-node -D

npm install supertest -D this will help to do the testing for http methods.

npm install express

jasmine init (this command is use to create the spec folder)

Day 4

29-05-2022

Docker is an advanced OS Virtualization software platform that makes it easy to create, deploy and run the application in a Docker container.

Virtualization: Virtualization is the means of employing software (such as Hypervisor) to create virtual version of resources such as server, tool, database or application.

Virtualization lets you divide a system into series of separate section, each one acting as a distinct individual system. The virtual environment is known as virtual machine.

VM ware software

Oracle VM

Base Machine contains 16 GB RAM

1 TB

Virtual Machine

1 to 10 VM

4 GM RAM

100 GM

Container : container is known as run time environment or engine which help to run the application.

The Docker container is a very light weighted package that allows the developer or programmer to package up an application and deploy it as one with help of in built libraries and other dependencies.

Virtualization Vs Containerization

Virtualization is an abstract version of physical machine.

Containerization is the abstract version of an application or app or program.

Docker Container : This is a running process or instance of images. Running the instance of Docker image it turn the actual application.

Docker image Docker images are the source code for our container or Docker image contains everything that we need to run our application.

Docker file : Docker file is a blue print or template which help to create the image or

Docker file contains set of instruction that defines how our images is built.

Docker registry : it is use to store the image or it is use to publish the image.

Docker hub : docker hub is like a git hub which is use to publish the image. In Docker hub we will the docker the registry. So anyone can publish the image the image as well as pull the image from docker hub and run in local machine.

Day 5

04-06-2022

1. Helloword
2. Busybox
3. Alpine
4. We created user-defined images on busybox to display message as well date

Now we will create the image to run node js

Dockerfile

FROM node:14-slim

COPY app.js .

CMD ["node","app.js"]

docker build –t my-node . –f Dockerfile

Creating the image to run the Express JS Application

First create the folder as express js images

Then create the package.json file using npm init

Then using npm command install express module like

npm install express

Then create the app.js file and create few rest api

App.js

let express = require("express");

let app = express();

let products = [

    {pid:100,pname:"TV",price:550000},

    {pid:101,pname:"Computer",price:45000}

]

app.get("/",(req,res)=> {

    res.send("Welcome to Docker using Express JS")

})

app.get("/user/:name",(req,res)=> {

    let name = req.params.name;

    res.send("Welcome user to docker "+name)

})

app.get("/products",(req,res)=> {

    res.json(products);

})

app.listen(9090,()=>console.log("Server running on port number 9090"));

**Dockerfile**

FROM node:14-slim

RUN mkdir /usr/src/app

WORKDIR /usr/src/app

COPY package.json /usr/src/app/

RUN npm install

COPY app.js /usr/src/app/

CMD ["node","app.js"]

docker build –t my-express . –f Dockerfile

To run the image if image contains to run the web application we have to use the command as

Docker run –p 9090:9090 my-express

Left side is expose port number it may be same or different

Right is actual port number of express j

docker run -d -p 9091:9090 my-express docker run -d -p 9091:9090 my-express

-d means detached mode (background)

-p port number

docker ps (ps mean process status) This command is use to display all running container.

docker ps –a (this command is use to display all container present in our machine it may be running or stop)

docker start containerId : This is use to start the container

docker stop containerid : This is use to stop the container

docker rm containerid : This is use to remove the container

if you get any error then remove forcefully

docker rm containerId –f

docker rmi imageId/imageName

if you get any error may be image link with some container then plz remove container and remove image

or

docker rmi imageId/imageName –f

lot of server are there

apache tomcat

web logic

IIS

Nginx server

Nginx is a open source server which help to deploy the any type of application on server.

Nginx default port number is 80.

Creating the image for Angular application

ng new angular-app-with-docker

routing 🡪 no

styling 🡪 css

after created the project do the coding according to your requirements.

Then we have to build the project using command as

ng build

After build you can see dist folder inside a angular project folder

Now you have to create the image with build file

Dockerfile

FROM nginx

COPY /dist/angular-app-with-docker /usr/share/nginx/html

Now you have to create the image

docker build –t my-angular . –f Dockerfile

05-06-2022

Before push the image in docker hub we have to provide the tag it is like a identity or version

docker tag imageName dockerHubAccount/imageName:version

docker tag my-angular akashkale/my-angular:1.2

after created tag for image we have to push the image

docker push dockerhubaccount/imageName

docker push akashkale/my-angular:1.2

if you get the error as access denied then

docker login

provide username and password

please pull my image using command as

CI and CD : Continuous Integration and Continuous Delivery or deployment

Code changes made by individual team members in their machine and after done changes they have push this code to remote repository and they have to merge the code in working software. This phase is known as integration phase.

Integration phase is very hard because after merge other team code the working software may be work or not.

docker pull akashkale/my-angular:1.2

docker run –d –p 80:80 akashkale/my-angular:1.2