18-09-2021 : Day 1

8 classes

Per day 3 hour classes

Testing and deployment

Front end testing --🡪 Jasmine and Karma

Html/Plain JavaScript-🡪 Jasmine

Angular --🡪 Jasmine and Karma

Backed end testing -🡪 Jasmine with supertest

Mocha with Chai (chai-http)

React JS with JEST

Introduction to grunt

Docker

Introduction to CI and CD with Jenkin

Introduction :

AWS : Cloud computing

S3

EC2

Etc

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

Read a,b 10, 20

//Compute sum = a+b

Write sum 30

Testing mainly divided into 2 types

Black box testing :

Input ----------- >Process------------🡪Output

White box testing :

Input ----------- >Process------------🡪Output

Unit Testing : Unit testing is a kind of software testing method in which each individual and independent part of the source code will be test. Unit means is a small module or function or method or class etc.

Unit testing is a type of white box testing.

Jasmine : Jasmine is open source framework which help to do the testing for JavaScript program it may be client side or server side JavaScript.

Jasmine is DOM less JavaScript testing Framework.

Document Object Model.

Node JS Testing using Jasmine

In Testing

Test Suite : Test suite is s collection of more than test spec as well another Test suite.

Syntax

describe(“Message”,function(){

})

describe(“Message”,()=>{

})

Test Spec : Test spec is mainly use to test the specific function functionality. It can contains more than one expectation.

Syntax

it(“Message”,function(){

})

it(“messasge”,()=>{

})

Expectation : it provide set of pre-defined function which help to do the testing and check actual land expect output.

Create folder Node JS testing with jasmine

Create the package.json file using npm init

npm install jasmine –D ( only development mode)

npm install jamine-node -D

we have to create spec directory or folder.

So jasmine provide command to create the folder ie

jasmine init : This command is use to create spect and jasmine.json configuration file for testing.

To enable the jasmine command we have install the jasmine globally

npm install jasmine –g

create src folder

19-09-2021 : Day 2

Testing hook or life cycle functions.

Using Jasmine testing for Express JS Application

SuperTest is a third party library which is internally use Http Assertion library that allow you to test Node JS Express Module.

Create folder Express JS with Jasmine and SuperTest

Then create package.json file using npm init

npm install express

npm install jasmine –D

npm install jasmine-node –D

npm install supertest –D

Using jasmine init create the spec folder.

Jasmine init

Client JavaScript testing using jasmine with downloaded standalone setup

Angular Framework provide the set for testing

Angular use Jasmine framework to do testing for Angular Application.

Jasmine provide test suit, test spect and expect.

Angular use Karma

Karma is test runner for the JavaScript as well as typescript programs.

Karma : Karma provides a suitable testing environment to any web application.

Karma start a small web server to test JavaScript as well as typescript file to test on browser.

In Angular we are testing component, service, routing file etc.

Angular provided testing utilities files to test angular specific component as well as service.

Angular with Jasmine

React with JEST

Please create new project

ng new angular-testing

please check the spec file which contains one suite and three spec.

to run the application we have to use the command as

ng test

please create component, service and model using command as

ng g c employe : it is use to create the component

ng g s employee : it is use to create service

ng g class employee : it is use to create model

26-09-2021

Open backend project ie Employee Backend REST API

Run npm install to download node\_modules folder.

In mongo db bin folder

Run to command in command prompt ie

mongod : to run the server

mongo : to run the mongo terminal.

npm install nodemon –g

nodemon it a type of external module which help to run express js application.

If we do any changes in express js application automatically it will restart the application.

nodemon app.js

Mock testing :

In general one function depends upon another function or one class depends upon another class.

Component depends upon the service, service class depends upon the backend technologies. Backend technology depends upon the database.

02-10-2021

Day 5 : Docker

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

Virtualization : Virtualization means of employing software (such as hypervisor) to create a virtual version of resource such as server, database, application or tool.

Virtualization let us divide a system into series of application or separate section, each one acting as a distinct individual system.

Base Window 10. Mac, Unix, Linux

Base OS Window 10

RAM 16GB

GOS Unix ---🡪 4gb RAM 50gb

10 OS

1.5

1RM

Virtualization Vs containerization

Virtualization is an abstract version of physical os machine. While containerization is the abstract version of an application.

Docker Container : Running instance of Docker image container turn or run the actual application. A container includes an application and all of its dependencies.

Docker image : The file system and configuration of your application which are use to create the container.

Dockerfile : A docker file is a blueprint or set of instruction that defines how your image is built. It is a series of steps that you have to defined to create the image.

Using Docker file we will create the image.

Docker command

Virtual lap

Open the terminal

docker --version sudo docker --version

docker images : This command is use to check your images

docker pull hello-world : This command is use to pull image from dockerhub

docker run hello-world : This command is use to run the image

Docker registries : Docker stores the images we build in registries. There are two type of registries

public : more than one

private : we can create only one private registries

Docker hub is like a github. In github we can store any type of documents. In Dockerhub we publish our images so other people can pull and run it.

busybox

docker pull busybox

busybox is a tiny unix os.

docker run -it busybox

alpine

docker pull alpine

We will create our own images.

To create the image we have to create the dockerFile which contains configuration details.

According to standard rules files name must be dockerFile (without extension).

**Creating image to display the date information.**

dockerFile

FROM busybox:latest

CMD ["date"]

docker build -t my-busy-box123 . -f dockerFile

creating image for node js application

FROM node:16-alpine

COPY app.js .

CMD ["node","app.js"]

app.js

var a=10;

var b=20;

var sum = a+b;

console.log("Sum of two number is "+sum);

function sayHello(name){

    return "Welcome to Simple Node JS Application "+name;

}

console.log(sayHello("Raj"));

create the image using command as

docker build -t my-node123 . -f dockerFile

then run the image using command as

docker run imageId

creating image for express js application

First create package.json file using command as

npm init

install express js

npm install express

dockerFile

FROM node:latest

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-node1234 . –f dockerFile

if you want to run the image which is use to run the web application we have to use the

command as

docker run –p 9090:9090 imageName/imageId

left side port number to expose the image (this port number may be same or different

must be free in your machine).

right side port number actual port number

docker run –d –p 9191:9090 imageName/imageId

-d : means running the application in detached mode (background).

docker ps (process status) : This command is use to display running container

docker stop containerId :This command is use to stop the container

docker start containerId: This command is use start the container.

docker rmi imageId : This command is use to remove image from local machine.

(note if you get error then –f to remove forcefully)

docker rmi –f imageId : This command is use to remove image force fully

docker rm containerId : This command is use to remove container

**What is nginx?**

It is open source server which help to deploy the deploy the web application.

Nginx default port number is 80.

Run the image which contains nginx server

docker run -d -p 9292:80 my-web123

9292 expose port number

80 : default actual port number of nginx

Docker image for Angular application

ng new angular-docker-info

do some changes in angular project

Then using command build the project

ng build

After build it will create dist folder, inside dist it will create project folder and which contains all build files.

Angular build file we have to provide to backend team -🡪 Java, Asp.net, Node JS, Python,

We can deploy this application alone in AWS or any cloud or any host.

Nginx

Please create Docker file inside build file folder.

dockerFile

FROM nginx

COPY . /usr/share/nginx/html/

docker build -t my-angular-info . -f dockerFile : This command is use to create the image

docker run –d –p 9595:80 my-angular-info : This command is use to run the application on port number

Then open the browser

<http://localhost:9595>

Now we will publish our image in Dockerhub

Before publish we have to create the tagname(it just like a identity or new update for the image like a version).

This command is use to create the tag for the your image

docker tag imageName:latest dockerHubAccountId/imageName

docker tag my-angular-info:latest akashkale/my-angular-info

After created tag successfully

You have to push the images

docker push dockerHubAccountId/imageName

docker push akashkale/my-angualr-info

it may ask username and password : docker login

please provide dockerhub account id and password

docker pull priyadarshinikamaraj/my-angular-info

09-10-2021

Pull the mongo db database

docker pull mongo

docker run -it -d mongo This command is use to run the mongo db container.

-it : run the container in interactive mode.

-d : detached mode or daemon process

Using docker ps : please check the container details.

It provide name for the container.

If we want to provide user-defined names then we have to use the command as

docker run --name my\_db -it -d mongo

Now using command we have to connect to container open the terminal.

docker run –it –link=serviceName:imageName mongo /bin/bash

docker run –it –link=my\_db:mongo mongo /bin/bash

mongo /bin/bash : open the shell in mongo container . this help to open the client terminal.

Now in mongo container shell write the command to open the client mongo terminal

mongo 172.17.0.2:27017 : default ip address is 172.17.0.2 and port number 27017

CI and CD : Continuous Integration and Continuous delivery or development.

Git

Maven

Gradle

Docker : using docker we run one container at time.

When we develop enterprise application we need to run

Multiple container to run different application

One container for angular

One container for express js

One container for mongo db and more

Using docker : using Docker swam

CI and CD – Jenkin

AWS

Kubernets : second option using. It is use to run more than one container which can interact with each others.

Yml : yet another mark up language

CI and CD :

Code changes made by individual team or person in local machine. They have to push this code to Version control (Remote repository) and merged into working software. This phase is known as integration phase.

After merge we have to build the project, run the project, test project and deploy the project.



First pull the image

docker pull jenkins/Jenkins

Then run the image

docker run -p 8080:8080 -p 50000:50000 jenkins/jenkins:lts-jdk11

by default Jenkin default port number is 8080

then open the browser

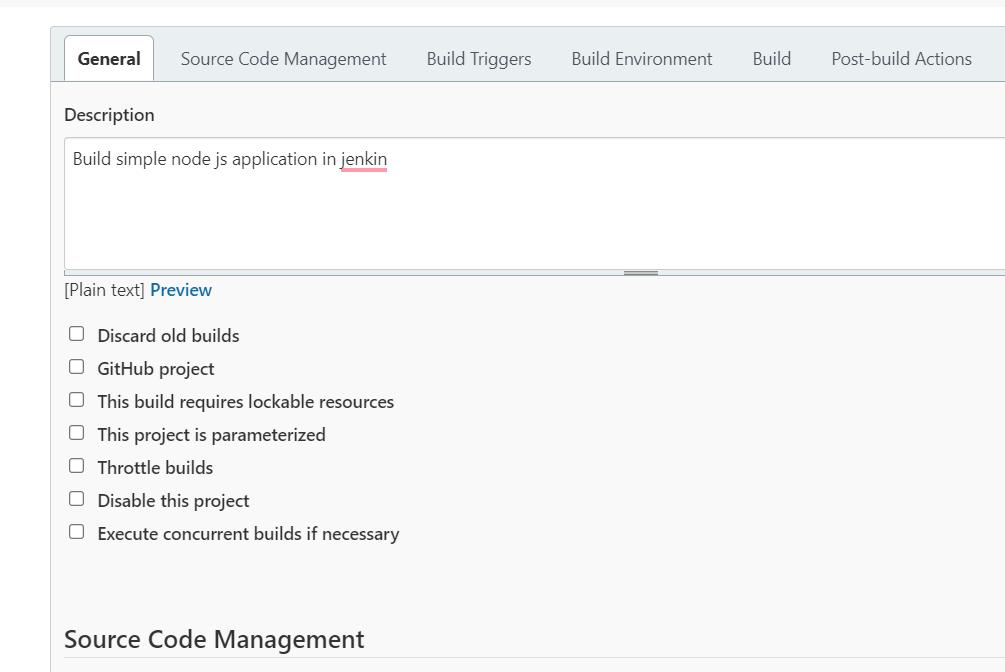
<http://localhost:8080>

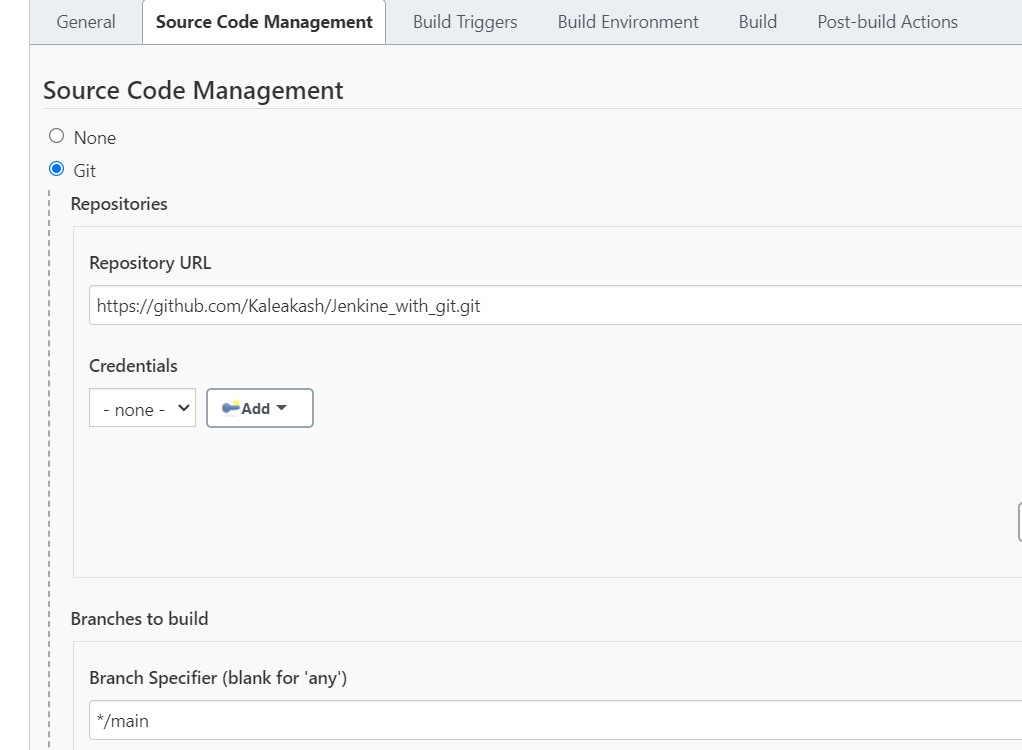
it will ask the password

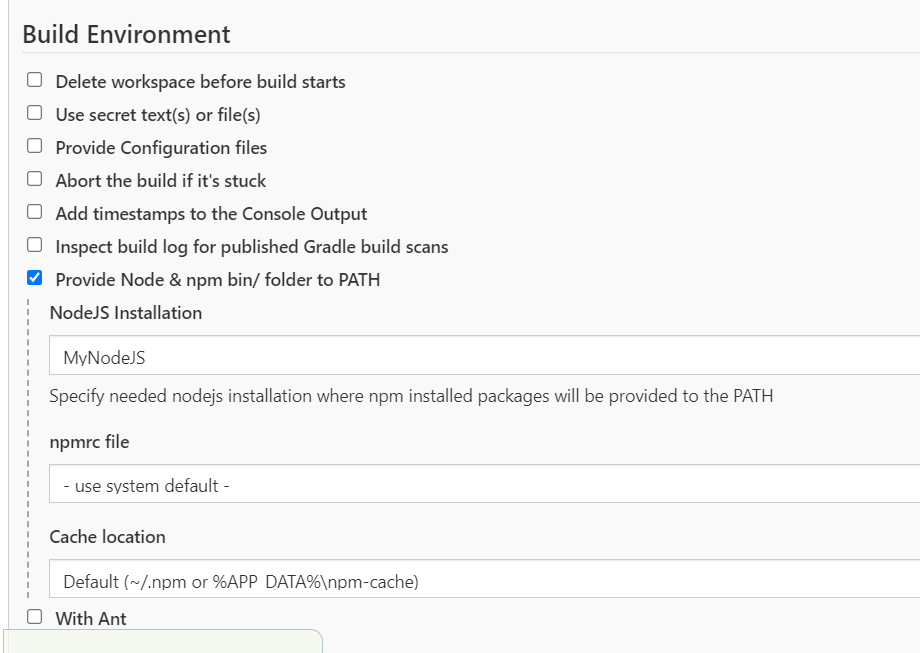
please check password in terminal and copy/paste

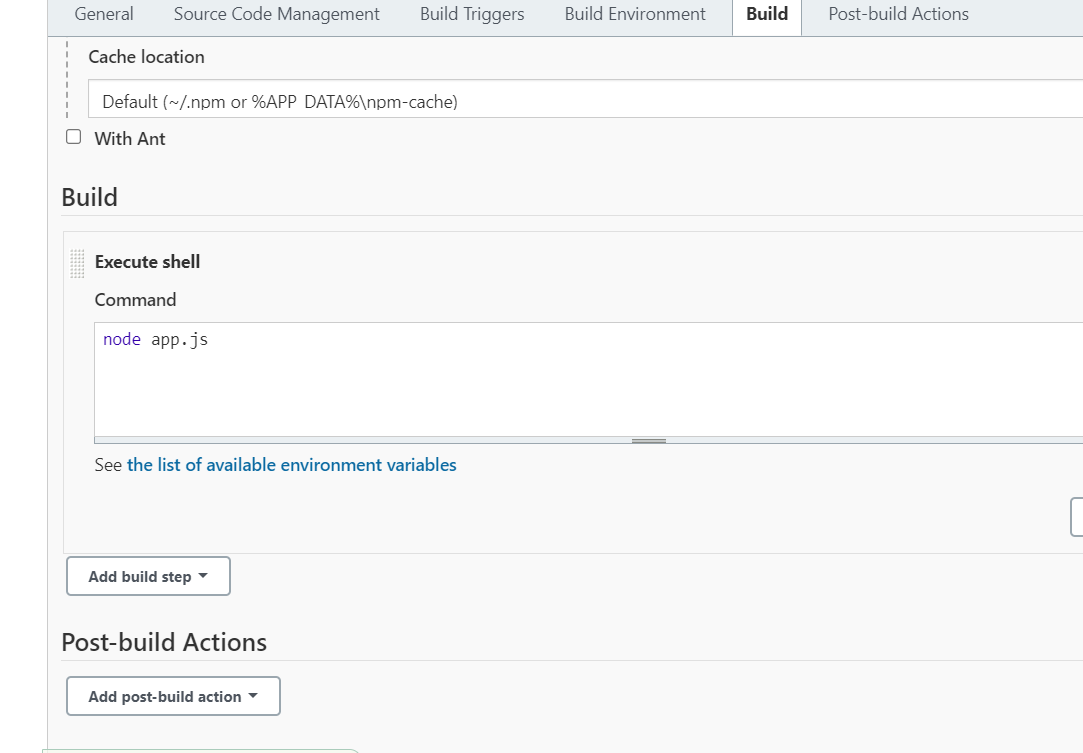
after password it ask to install few plugins please install those plugins.

In Jenkin pull the project and run the node js application









Jenkin Pipe line : Jenkin pipe line is a collection of events or jobs which are integrated with one another in a sequence to do specific task.

pipeline {

agent any

stages {

stage('check all setups ') {

steps {

echo 'installing all software'

}

}

stage('build the applicatin') {

steps {

echo 'build the project'

}

}

stage('test the project ') {

steps {

echo 'testing command'

}

}

}

}

Deploying the application in cloud machine

Iaas : infrastructure as a service

Paas : Platform as a service

Saas : Software as a service

AWS

Azure

Oracle

Google

Etc

AWS : Amazon Web Service : it is type of Cloud computing provide Iaas, Paas and Saas features.

Amazon EC2 : Amazon Elastic Compute Cloud :

This module help to create virtual machine in cloud computing with different type OS based upon our requirement which help to deploy the application.

EC2 instance

To connect the EC2 instance through window

1. We have to use putty software.
2. Using git bash terminal

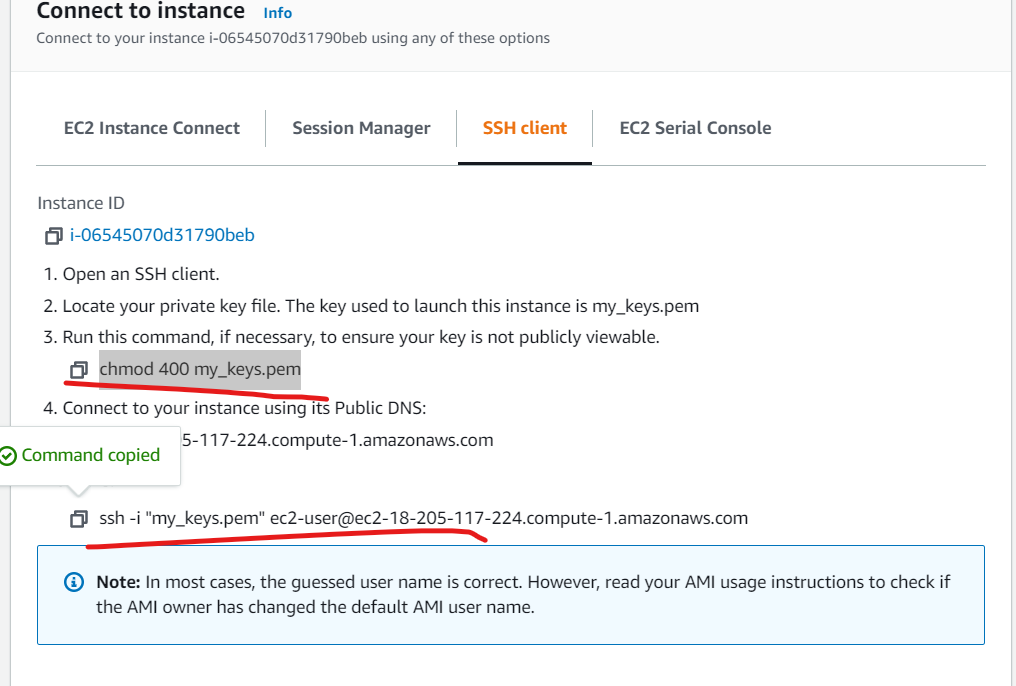
Open the gitbash in place where keyfine present.

chmod 400 my\_keys.pem : This command is use to change the mode of the file

To connect

ssh -i "my\_keys.pem" [ec2-user@ec2-18-205-117-224.compute-1.amazonaws.com](mailto:ec2-user@ec2-18-205-117-224.compute-1.amazonaws.com)

then enter yes



In EC2 instance terminal update using

sudo yum update

Phase 4 project

First Create Angular project

With sample information about yourself.

Using ng build projects.

ng build

Take the help of nginx images and copy all build files in nginx images

Then create the docker file

dockerFile

FROM nginx

COPY . /usr/share/nginx/html/

Push project with dockerFile in Git repository

Or

Publish this image in Docker hub

Then create EC2 instance

Install git and docker

Then create the images in EC2 instance and publish the images and view the images using your IP address.

By default port number of ngnix is 80 . please expose port number within your range while running application ex : custom TCP 🡪 8000-9999

S3 : simplex storage Service in AWS