THEORY

21A95A0414

Q1) Build a manual ci-cd pipeline for a java application and also write a simple test.sh for testing, after that convert the same in automated ci-cd pipeline using jenkins but you have to also use maven for java application.

\* Firstly go to jenkins and then open manage jenkins on the dashboard

\*Then open available plugins in that install maven integration and maven pipeline integration

\*Then click on install without restart

\*Then back to new page and open back to restart page and select new item and select the new name for the item and select pipeline integration in that.

\*Then save it and enter git repository URL then click build whenever a snapshot dependency is built and git hub hook trigger for GITSCM polling

\*Then select nodejs and enter repository name as in docker

\*Apply and save and open the dashboard->manage jenkins->global tool configuration

\*Then in that select git->path to git executable

\*Enter the jdk file and click on it

\*Select the name as jdk and enter its version

\*Finally select the build now and now we can observe that the build is success.

Q2. Create an EC2 instance for a REACTJS application and deploy the application and provide the public IP for it, also enable the cloud monitoring on these instances.

A) EC2 INSTANCE:

\*EC2 is used to create virtual machines in aws platform like LINUX OS, WINDOWS OS

\*The creation of EC2 consists of seven steps:

a) CHOOSE AN AMAZON MACHINE IMAGE

b) CHOOSE AN INSTANCE TYPE

c) CONFIGURE INSTANCE DETAILS

d) ADD STORAGE

e) ADD TAGS

f) CONFIGURE SECURITY GROUPS

g) REVIEW

\*Here we have to use UBUNTU as amazon machine image then connect the instance

\*Run the commands as sudo apt update, sudo apt install npm, npm i , npm start

\*Then a new REACTJS application is created in our instance

\*Then select security group and then edit inbound rules with the port number and select anywhere ipv4 in custom and then save rules

\*Then copy the public URL for the running instance

\*Finally paste it in the new tab and port number

\*It will displayed the empty white blank page

Q3. Create an RDS connection with ec2 instance and use it to create an SQL database and a sample table.

\*first we have to create an EC2 instance using ubuntu as AMI image and run the created instance

\*Then create RDS in that select free tier project, MYSQL, password and confirm password and then select respective VPCs and availability zones

\*Then don’t allow public access and select security groups and create the database.

\*Then edit the inbound rules in that select MYSQLAURORA with the port 3306 and then save the rules

\*Then select the created database then actions then click on

modify -> connectivity-> Additional settings-> then make the database as public and confirm.

\*Then select switch to EC2 configuration in RDS and connect it with the instance

\*Then go to ec2 and connect it and run the following commands

a) sudo apt update

b)sudo apt install mysql-client

c)mysql –version

\*Then enter our confirm password then create a table using SQL

\*SQL is used to access and manipulate databases

\*SQL is a relational database🡪it is nothing but the Elloboration of rows and columns it means a table format

\* Atlast create a table using SQL language

🡪CREATE DATABSE VIJAYA;

🡪USE VIJAYA;

🡪CREATE TABLE STUDENT (id INT NPT NULL AUTO\_INCREMENT,NAME VARCHAR(50),BRANCH VARCHAR(20),PRIMARY KEY(id));

🡪INSERT INTO STUDENT (NAME,BRANCH) VALUES(“”NITHYA”,”ECE”),(“NEELIMA”,”ECE”);

🡪SELECT \* FROM STUDENT;

\*Then it will display you a table format.

Q4) Create an S3 bucket and deploy the REACTJS static application on it and provide with the public URL for use.

\* first we have create an empty bucket in S3

\*Then download nodejs in chrome with windows of msi of 64bit

\*Then create a build folder

\*Then clone the git repository to a new folder and open the command prompt from the amazon create folder

\*for that we have to go to vs code and check the version and run the command as npm run build

\*Then upload the files and folders to the s3 bucket which are in the build folder

\*Then enable static website hosting in the properties of the bucket

\*Then edit the object ownership and ACL’S should be enabled

\*Then make public the objects in the Actions

\*Then copy the object URL from the properties and past it in the NEWTAB so it will display the output

\*It is a object based storage, data can be replicated across multiple facilities

\*There are different types of storage classes are there

1)S3 INFREQUENT ACCESS

2)S3 ONEZONE-IA

3)S3 GLACIER

4)S3 STANDARD

\*Then we can copy the object URL and we can observe the output.