NAMASTE NODE.JS SEASON 3

Episode-1
Hand Written Notes

-By Shanmuga Priya

www.linkedin.com/in/shanmuga-priya-e-tech2

Season: 3

Episode: 1

Launching AWS Instance and deploying frontend

1) what is AWS?

Amazon web Service (AWS) is a cloud computing platform that offer a variety of services over the internet.

2) what is cloud computing platform?

→ A cloud computing platform is a network of servers that provide variety of services such as servers, storage, databases, networking & so on to users over the internet.

→ Users can rent access to these services on demand, paying only for what they use.

Steps involved in Setting up AWS for deployment

Step1: sign up in AWS

Step2: Search for EC2 in the search bar provided in Console Home Page.

what is EC2?

EC2 is called Elastic Compute Cloud is a web service that allow users to create and run virtual machines, called instances in the cloud.

Step3: click on Launch instance and add a name to that instance and select the OS which we need to run on that instance from the given list. (ubuntu preferable) and select the instance type.

Step4: Create a new key pair ⓧⓧ

→It is like a secret key which is used to connect to the instance securely.

-> Enter a Key Pair name of your choice and select the RSA algorithm and Pem file format. and click on create Key pair.

-> It will generate a file with the secret key & download automatically.

step 5: click on Launch instance it will create a new instance for us with all the configuration that we setup in the Previous step.

-> the instance will be in pending state later it will turn into running state.

tep6: click on the instance ID it will give all the info about the virtual machine that we have rented like Ip address. Click on connect it will give us a various way of connecting to that instance.

tep7: Select SSH client from it. It is used to connect to the instance using terminal. In terminal follow the steps provided to login to the instance. Now, we have successfully logged in to our instance. To logout from the instance type exit.

tep8: Install a node in the new instance that we created. make sure to use the same version which is used in the Project to avoid misfunctio

How to connect back to instance once logged out?

To connect back to the instance we need to run the same SSH-i Command which we have done in step 7. It basically contains

a SSH-i + secret file + instance name.
(connecting method)

How to run our project in the new instance That we created?

step1: get a https link and place it in the terminal with the command

git clone

" git clone your projects https link from code section."

It will clone our project in the new instance that we created.

steps involved in ~~setting up~~ deploying a project:

Once the instance is setup we will deploy our project. Let's see how to deploy our frontend part.

-> Step 1: Building a project.

before deploying a project we need to bundle it up to a single 'dist' folder which contains the entire project code & packages needed for our project. This can be done by running a npm run build in project's terminal.

=> step 2: Build the project in our new instance.

before building a project we run npm install to download all the packages that our project needs. then we run npm run build to build our project.

=> step 3: Download nginx

what is nginx?

-> Nginx is a opensource software that can be used as web server, load balancer, reverse proxy & so on.

-> It is widely used for serving static web content, handling high traffic and distributing requests to backend servers.

-> we use Nginx to host (or) manage our app on AWS instance.

```
sudo apt update -> to update ubuntu version.
sudo apt install nginx -> to install nginx
sudo systemctl start nginx -> to start nginx
sudo systemctl enable nginx -> to enable nginx
```

-> step 4: copy the code from dist folder to nginx http server.

to copy a dist folder to http server we run a command

```
sudo scp -r dist/* /var/www/html/
```

↓ copy ↓ recursively ↓ everything from dist ↓ http server.

-> This will run our app on port "80" we need to connect this port to our instance's Public Ip address to see our app live.

Step 5: connecting ngnix & our instance.

-> to enable port 80 on our instance click on security tab and go to security group and add a inbound rules to include port 80.

step 6: application Live.

Now we go to Public Ip address our app will be live.