Name - Aman Yadav
Class - DIT C, Roll no - 60
Sub - Adv. DevOps

Assignment no. 2

03

1. Create a REST API with the Serverless framework.

2. Steps: Install Serverless framework globally using the following commands on the terminal.
- npm install -g serverless
This command allows you to create, manage & deploy serverless applications across various cloud providers including Aws.
- Create a new service with Aws Nodejs template.
Initialize a new Serverless service called REST API.
It creates a folder containing basics and a template specifically configured for building a serverless application.
- Navigate to the project directory.
- cd rest-api.
- Initialize nodejs project & install dependencies.
- npm init -y
- npm install express serverless-http
Express & serverless-http integrate building the REST API with Aws lambda.
- Edit the serverless.yml file to include - service: rest-api

provider:
  name: aws
  runtime: nodejs14.x
  stage: dev
  region: us-east-1

functions
  name: aws
  runtime handler.app

- http:
path: /
method: any

This configuration specifies the service name, aws provider settings and define this lambda functions wit http event trigger.

- Edit handler.js to add the Express app.

```
Const express = require('express');
Const app = express();
app.get('/hello', (req, res) => res.json({ message
```

This creates a simple express app with a single route /hello & exports it in lambda compatible format.

- Deploy the app & the API gateway. A virtual lab is generated for testing.
- Test the deployed API.
- Curl https://<api_id>.execute-api.<region>.amazonaws.com/dev/hello

Using the above returns a JSON response {Message: 'hello'}.

(2) Case study for Sonarqube:

: Setting up your profile for Sonarqube. Creating a profile on Sonarqube allows devops to analyze the quality of the projects and track improvements over time.
Steps: Install Sonarqube & set it up locally.
. once logged in create a new project in Sonarqube by providing a project name & key.
. Add the Sonar-project properties file to the root directory of the project which contains necessary configurations.
. Use Sonarqube scanner to analyze your projects & uploads the result to the dashboard

: Using Sonarqube to analyze github code:
Steps: Signhub & connect your github repository.
. Setup Github actions to run sonarcloud whenever Code is pushed into the repository. This ensures Continuous code quality analysis.

: Sonarlint for real time code analysis in IDES: Sonarlint is a plugin for intellij IDEA & Eclipse that performs code analysis.
Steps: Install the plugin for intellij IDEa goto files > Settings > plugins, fin Sonarlint & install.

. Sonarlint can be linked to sonarqube instance to sync rules & quality profits.

- Sonarlint runs automatically as you write code flag issues directly in the editor.

iv Analyzing Python project with Sonarqube.
- Steps: Ensure Sonarqube is running Configure Sonarqube for python (Specify resource files)
- Execute Sonar scanner from the root of your project.
- Analysis results will be uploaded to Sonarqu

v Analyzing Nodejs project with Sonarqube. →
- Steps: Verify Javascript plugin is available your sonarqube instance.
- Configure project by adding properties in the Sonar Project file.
- You can also contain Sonarlint with Sonarqube for a more comprehensive Javascript analysis.
- Use Sonarqube scanners to analyze the proje

8. At a large organization your centralized app operation team may get infrastructure requests. You can use terraform to build a service infrastructure model that lets product teams manage infrastructure independently. Create & use Terraform modules that modify the standards for deploying & managing services in your organization. Terraform cloud can also integrate with ticketing systems like service Now & automatically generate new infrastructure requests.

→ Step 1: Define infrastructure standards: Establish & car standards & best practices for infrastructure deployment including resource types, tagging policies, & security compliances.

Step 2: Create terraform module based on the organizations standards. Develop a common resource using EC2 instances & S3 buckets.

eg- variable "instance-type" {
default = "t2.micro" }

resource = aws instance "example" {
ami = ami-12345678"
instance type = var.instance type"
tags = {
Name = "example-instance"
}
}

- Sonarlint runs automatically as you write code flag issues directly in the editor.

(iv) Analyzing python project with sonarqube
steps: Ensure Sonarqube is running Configure Sonarqube for python (Specify the source files).
- Execute sonar scanner from the root of your pro
- Analysis results will be updated to Sonarqube.

v) Analyzing Nodejs project with Sonarqube.
- steps: Verify Javascript plugin is available in your sonarqube instance.
- Configure project by adding properties in the Sonar project file.
- You can also combine sonarlint with Sonarqube for a more comprehensive Javascript analysis.
- Use Sonar-Scanner to analyze the project.