Vedang V. Wajge

DISA / 66

Adv. Devops Assignment 2 04/03/24

1) Creating a REST API with serverless framework.

Creating Rest Api with serverless framework is an efficient way to deploy serverless applications that can scale automatically without managing servey.

i) Serverless framework : A powerful tool that deployment of services and serverless applications across various app cloud providers like Aws.

ii) Serverless Architecture: This design model allows developers to build applications without worrying about underlying infrastructure, enabling focus on code.

iii) Rest Api : Representational state transfer is architecture style for designing network applications.

Steps for creating Rest Apis for serverless framework:

1) Install serverless framework
2) Creating a Node.js serveless project.
3) Project structure define.
4) Create a Rest Api Resource
5) Deploy the Service.
6) Testing the API's.
7) Storing data in Dynamic DB's
8) Adding more functionalities : 'list all candidates', etc.
9) Aws IAM Permissions.
10) Monitoring and Maintanence.

Q-2] Case study for sonarQube.
Creating own profile in sonarqube for testing project quality. Use sonarqube to analyze your Github code. Install sonarlint in your intellij id and analyze java code. Analyze python project with sonarqube.

→ Sonarqube is an open source platform used for continuous inspection of code quality.

1) Profile updation/creation in sonarqube:
Quality profiles in sonarqube are essential configurations that define rules applied during code analysis. Each project has a quality profile for every supported language with default being 'sonarqube'.

2) Using sonarcloud to analyze Github code.
Sonarqube / sonar cloud is cloud based counterpart of sonar Qube that integrates directly with Github, BitBucket, Azure and Gitlab Repos. To get started with sonarcloud via Github sign up via sonarcloud setup, complete result can be viewed in both sonar-cloud & Github including security import issue.

3) Sonarlint in Java IDE:
Sonarlint is an IDE that performs on the fly code analysis as you write code. It helps developers detect bugs, security vulnerabilities and code smells directly in the development environment such as Intellij Idea or Ellipse

4] Analyzing Python Projects with SonarQube:
Sonarqube supports Python test coverage, reporting, but it requires third party too like coverage.py to generate the coverage port. To enable coverage adjust your blind process so that coverage tool runs before sonar scanner and ensures report.

5) Analyzing Node.js projects with Sonarqube.
For node.js project sonarqube can analyze Javascript and Typescript of code similar to the python setup you can configure sonarqube to analyze node.js projects by installing the appropriate.

At a large organization, your centralized operations team may get repetitive infrastructure requests. You can use Terraform to build a 'self-serve' infrastructure.
Terraform's self-serve infrastructure provides a powerful use case in large organizations.
i) Self serve infra: By using Terraform modules, you can create reusable and standardized infrastructure config. Module creation in Terraform, main.tf, variables.tf, and outputs.tf.
Also after module creation its standardization is equally important.
ii) Enabling self-service for Product Teams:
Create a self service or version control access and

and provide pre-configured Terraform workflows onboard and train product teams, and the most important RBAC (Role based Access Control) for preventing unauthorized Access

3] Automate Infrastruct Request via Ticketing systems:

Integrate Terraform Cloud or Terraform Enterpr connect terraform with the Ticketing system, automate approval workflows and monitor & log requests.

4] Workspaces setup for Environment Segregation.

To manage different environments, Terraform workspaces were set up. This ensured that teams could deploy the same infrastructure across different environments without overlap.