DEPLOY FLASK OR PYTHON WEB APPLICATION USING GIT, GITHUB, JENKINS, TERRAFORM, IN AWS

PREPARED BY.

T.AJAY

CONTENTS:

- Introduction to the tools
- Prerequisites
- Method1:DEPLOY FLASK OR PYTHON WEB APPLICATION MANUALLY BY USING AWS RESOURCES.
- Method2:DEPLOYING PYTHON WEB APPLICATION USING USERDATA.
- Method3:DEPLOYING PYTHON WEB APPLICATION USING GIT, GITHUB AND JENKINS.
- Method4:DEPLOYING PYTHON WEB APPLICATION USING GIT,GITHUB AND TERRAFORM.
- Method5:DEPLOYING PYTHON WEB APPLICATION USING GIT,GITHUB AND BASH SCRIPT.
- Method6:DEPLOYING PYTHON WEB APPLICATION USING GIT, GITHUB AND JENKINS.

INTRODUCTION TO THE TOOLS

- What is python?
  - Python is a high-level, interpreted programming language known for its simplicity and readability.
  - It was created by Guido van Rossum and first released in 1991.
  - It is a dynamically typed language And easy to learn.
- What is flask?
  - Flask is a micro web framework for Python, designed to make it easy to build web applications quickly and with minimal complexity.
- What is Jenkins?
  - Jenkins is an open-source automation server that facilitates the automation of various tasks related to building, testing, and deploying software.
- What is terraform ?
  - Terraform is an open-source infrastructure as code (IaC) tool developed by HashiCorp. It enables users to define and provision infrastructure resources across different cloud providers.

PREREQUISITES

- GIT
- GITHUB
- JENKINS
- TERRAFORM
- AWS RESOURCES
- PYTHON CONFIGURATION FILES

METHOD1:DEPLOY FLASK OR PYTHON WEB APPLICATION MANUALLY BY USING AWS RESOURCES.

- Create a VPC along with subnets, route tables, internet gateway.
- Create security group with respective ports 22,8080,80.
- Create EC2 instance with ubuntu server launch with SSH.
- Run the fallowing commands
  - sudo apt update
  - sudo apt-get full-update –y
  - sudo apt-get install python3-pip
  - sudo git clone https://github.com/ajay77777777/flask-library-app.git
  - cd flask-library-app/
  - pip3 install –r requirements.txt
  - python3 app.py
- Now copy the public ip along with port number.

METHOD2:DEPLOYING PYTHON WEB APPLICATION USING USERDATA.

- Create a VPC along with subnets, route tables, internet gateway.
- Create security group with respective ports 22,8080,80.
- Create EC2 instance with ubuntu server and provide the fallowing userdata.
  - sudo apt update
  - sudo apt-get full-update –y
  - sudo apt-get install python3-pip
  - sudo git clone https://github.com/ajay77777777/flask-library-app.git
  - cd flask-library-app/
  - pip3 install –r requirements.txt
  - python3 app.py
- Now copy the public ip along with port number.

METHOD3:DEPLOYING PYTHON WEB APPLICATION USING GIT, GITHUB AND JENKINS.

- Create a VPC along with subnets, route tables, internet gateway.
- Create security group with respective ports 22,8080,80,9000.
- Create EC2 instance with ubuntu server launch with SSH.
- After connecting instance install Jenkins in that machine and browse the public ip along with port number 8080.
- Create job to clone repository and in build step select execute shell and provide script and build the job.
- Before going to execute the script you need to add Jenkins user for sudo permissions in the sudoers file under /etc directory.
- Now you can browse the public ip along with the port number 9000 .

METHOD4:DEPLOYING PYTHON WEB APPLICATION USING GIT,GITHUB AND TERRAFORM.

- Create EC2 instance with ubuntu server and install terraform in that.
- Next you create the .tf files for aws resources.
- Run the below commands:
  - Terraform init
  - Terraform validate
  - Terraform apply
- Now copy the ip address of newly created instance and browse it along with port number.

METHOD5:DEPLOYING PYTHON WEB APPLICATION USING GIT,GITHUB AND BASH SCRIPT.

- Create a VPC along with subnets, route tables, internet gateway.
- Create security group with respective ports 22,8080,80,9000.
- Create EC2 instance with ubuntu server launch with SSH.
- Next create .sh file and add script to it.
- Add execute permissions to the .sh file.
- Now run the script by using fallowing command
  - ./<file_name>
- Now copy the public ip along with port number 9000.

METHOD6:DEPLOYING PYTHON WEB APPLICATION USING GIT, GITHUB AND JENKINS.

- Create a VPC along with subnets, route tables, internet gateway.
- Create security group with respective ports 22,8080,80,9000.
- Create EC2 instance with ubuntu server launch with SSH.
- After connecting instance install Jenkins in that machine and browse the public ip along with port number 8080.
- Create one repository in github for userdata and copy the url of that repository.
- Create job to clone repository and in build step select execute shell and add the fallowing script and build the job.
    - cd /var/lib/jenkins/workspace/step6
    - sudo chmod +x data.sh
    - ./data.sh
- Before going to execute the script you need to add Jenkins user for sudo permissions in the sudoers file under /etc directory.
- Now you can browse the public ip along with the port number 9000 .