**Name:** Mayank Arora

1. **Explain devops. Mention the devops stages and tools.**

DevOps is a collaboration between development and operation teams which enables continuous delivery of applications and services to end users.

In simple terms, DevOps combines software development and IT operations. They need to work together for various reasons like continuous delivery, high performance etc.

Stages of DevOps: -

1. Planning (Tools: Git, Jira)
2. Code (Tools: Git, Jira)
3. Building (Tools: Maven, Gradle)
4. Testing (Tools: Selenium)
5. Deploy (Tools: Docker, Chef)
6. Operate (Tools: Docker, Chef)
7. Monitor (Tools: Splunk, Nagios)
8. Integration (Tools: Jenkins, Bamboo)
9. **What is docker? explain the architecture & the components of docker.**

Docker is a DevOps tool used in the deployment stage. Docker is used to automate the deployment of applications in lightweight containers so that applications can work efficiently in different environments.

Architecture of Docker: -

Docker uses a client-server architecture. Docker client and server communicate using REST API. Docker client is a service which runs a command. The command is translated using REST API and is sent to the Docker daemon (server). Then Docker Daemon checks the client request with the operating system to create or manage containers.

Components of Docker: -

1. Docker Client and Server: Docker client is accessed from the terminal and a docker host runs the docker daemon and registry.

A user can build docker images and run docker containers by passing commands from the docker client to docker server.

1. Docker Images: It is a template with instructions for creating docker containers.
2. Docker Containers: Docker container is a software package that consists of all the dependencies required to run an application. We can run 8 containers simultaneously in a host.
3. Docker Registry: It is an open-source server-side service used for hosting and distributing images. Docker also has its own default registry called Docker Hub.
4. **What is kubernetes? Explain the features of kubernetes.**

Kubernetes is a container management tool in DevOps. It is NOT containerization but a tool for container management. Kubernetes can automate container deployment, container scaling and descaling, and container load balancing.

Features: -

1. Automatic Binpacking
2. Service Discovery and Load Balancing
3. Storage Orchestration
4. Self-Healing
5. Secret and Configuration Management
6. Batch Execution
7. Horizonal Scaling
8. Automatic rollbacks and rollouts
9. **Demonstrate docker & kubernetes with screenshots.**

For docker, first we need an Azure Virtual Machine.

**Azure Virtual Machine: -**

1. Select Virtual Machine

Graphical user interface, application

Description automatically generated

1. Go to Create and then Virtual Machine

A screenshot of a computer

Description automatically generated

1. Give resource group name, VM name, select the image and size.

A screenshot of a computer

Description automatically generated

1. Select SSH public key and give a username.

Graphical user interface, text, application, email

Description automatically generated

1. Select Generate new key pair and give a name.

Graphical user interface, text, application

Description automatically generated

1. Select Allow selected ports an SSH (22).

Graphical user interface, application

Description automatically generated

1. Leave everything else as default and click Review + create.

Graphical user interface, application

Description automatically generated

1. Review the settings and change anything if required else if validation has passed click on create.

Graphical user interface, text, application, email

Description automatically generated

1. Select Download private key and create resource

Graphical user interface, text, application, email

Description automatically generated

1. Click on Go to resource.

Graphical user interface, text, application, email

Description automatically generated

1. Select Connect 🡪 SSH

Graphical user interface, text, email

Description automatically generated

1. Copy the command in point 4 and replace <private key path> with the path of your file that was downloaded in step ix.

Graphical user interface, application

Description automatically generated

1. Open command prompt as administrator and go back one directory.

Text

Description automatically generated

1. Paste the command that was created in step xii after providing the path.

Text

Description automatically generated

1. Give yes

Text

Description automatically generated

1. We are now in the Azure VM.

Text

Description automatically generated

**Docker:­ -**

1. First, update your VMs OS to the latest version.

A screenshot of a computer

Description automatically generated with medium confidence

1. Installing Docker.

Text

Description automatically generated

1. Check docker version to confirm it installed properly.

Text

Description automatically generated

1. Download and install a docker image (hello-world is the name of the image)

Text

Description automatically generated

1. List all the installed docker images.

Text

Description automatically generated

1. List docker images with container id.

Text

Description automatically generated

**Kubernetes: -**

1. Select Kubernetes services.

Graphical user interface, text, application

Description automatically generated

1. Select Create 🡪 Create a Kubernetes cluster.

Graphical user interface, text, email

Description automatically generated

1. Give resource group, cluster name.

Graphical user interface, text, application

Description automatically generated

1. In scale method select Manual and node count = 1 to keep the application simple.

Graphical user interface, text, application

Description automatically generated

1. Leave everything else as default and click Review + create.

Graphical user interface, text, application, email

Description automatically generated

1. Review the settings and change anything if required else if validation has passed click on create.

Graphical user interface, text, application

Description automatically generated

1. Click on Go to resource

Graphical user interface, text, application, email

Description automatically generated

1. Open Azure Cloud Shell and select bash (change the region if its not East US by going in advanced settings)

Graphical user interface, text

Description automatically generated

1. Type the command to enter Azure CLI

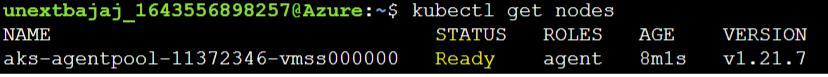
Graphical user interface, text, application

Description automatically generated

1. Now we need to type a command to download the credentials and configure Kubernetes (az aks get-credentials --resource-group <name of resource group> --name <name of Kubernetes service>)



1. Now, we need a command to get the list of nodes and check the connection (Here the STATUS must be ready, then only we can run the application).



1. Next, we open a file named “azure-vote” in VI editor and put the code for the application in this file.



A screenshot of a computer

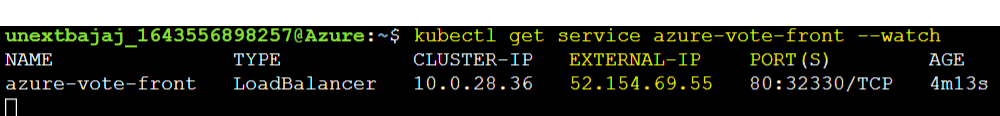
Description automatically generated

1. Next, we need to deploy the application (kubectl apply -f <filename>). Different containers will be created. Out of these we need to expose “azure-vote-front” to the frontend i.e., the Internet.

Text

Description automatically generated

1. Exposing the application to the internet (--watch is used so that we can see what is happening in the frontend). The service would be available on the mentioned External IP address.



1. Application Preview

Graphical user interface

Description automatically generated