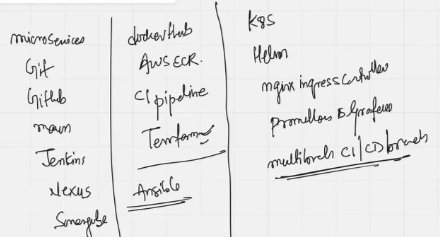
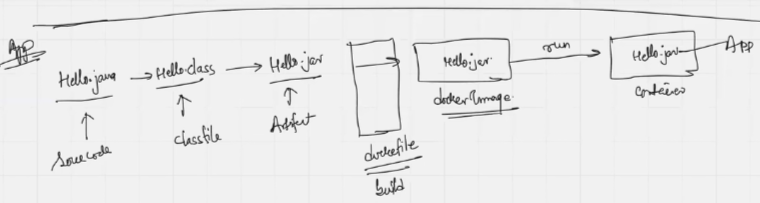
Kubernetes

Things completed so far and summary of upcoming activities:





Container lifecycle: Start, stop, reboot, logs, network, storage, security

What is kubernetes capable of?

Kubernetes simplifies the management of 100 containers across 100 applications, offering automated deployment, scaling, and updates while ensuring high availability and resource efficiency. It achieves this through features like self-healing, service discovery, and load balancing, allowing for smooth operation and minimal downtime even with complex deployments.

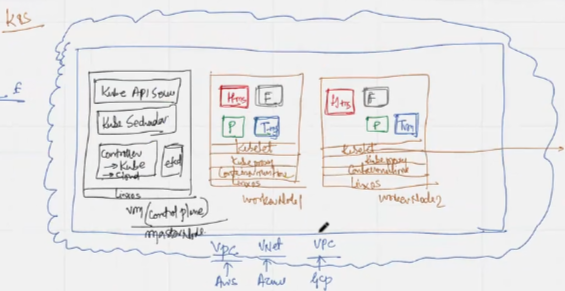
Kubernetes helps you:

* Run containers at scale across clusters of servers
* Automate deployment, rollouts, rollbacks
* Ensure high availability and fault tolerance.
* Perform load balancing across services
* Manage configurations and secrets
* Monitor and self-heal failed containers automatically

Architecture

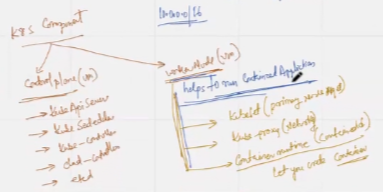
A kubernetes cluster is made up of multiple VMs. One of the VM is called a control-plane, the remaining 2-3 VMs are called worker nodes. Worker nodes define the size of the kubernetes cluster. The small kubernetes cluster has 1cp & 1wn, the biggest kubernetes cluster has 1cp & 5000wn.

Takes analogy from society like:







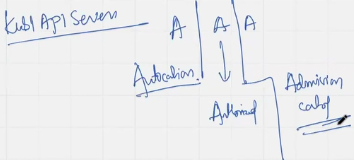


Each worker node has a capacity of running 6 applications, by this we ensure that more than 1 copy of an application is available beyond 1 working node. Payment, Hotel, Flight can have an app running in more than 1 working node → Fault tolerance.

Each container  shown with a perforated border has an application Hotel-microservice running in a container is called a POD.

Every time a set of 6 containers gets installed, the cloud-controller initiates a new worker-node so-called as the power of Kubernetes. In the event of a worker-node going down, the applications still remain safe on rest nodes.

Understanding control-plane



>> Kube-scheduler is responsible to pass the information to particular worker-node >> Point A

Understanding worker-node

1. Kubelet takes the information from kube-scheduler, and it is responsible for creating, updating, deleting POD on the worker-node

