**#100DaysOfDevops** is a challenge to read official documentations, FAQs and white papers of AWS different services and do hands-on practice of AWS services , Docker ,Kubernetes and Jenkins with the motive to get AWS Solution Architect Associate , AWS Developer Associate & Kubernetes Administrator certification.

**Tentative roadmap :**

* AWS
* Bash Scripting
* Docker
* Kubernetes
* Jenkins

**White Papers**

* [AWS Well Architecture Framework](https://bit.ly/3aAfe8J)
* [AWS Blue-Green Deployment](https://bit.ly/2Q2r563)
* [AWS Security Best Practices](https://bit.ly/2Y6A8aH)
* [Implementing Microservices on AWS](https://bit.ly/2CHErSo)
* [Practicing CI/CD on AWS](https://d1.awsstatic.com/whitepapers/DevOps/practicing-continuous-integration-continuous-delivery-on-AWS.pdf)

**AWS hands-on task**

* Create an IAM Role and Configure an EC2 Instance for AWS Systems Manager.
* Build a 3-tier network VPC from scratch.
* Create a Multi-Subnet VPC with secure access to private servers with outbound internet access using VPC , private & public subnet ,NAT & internet gateway, Bastion host.
* Create VPC endpoints and an s3 bucket.
* Managing Data in S3 with Versioning and Lifecycle Rules.
* Secure S3 bucket policies using IAM policies.
* Create an EC2 work station and give limited permission to a IAM user to use it.
* Build Fault Tolerance Website.
* Create a custom AMI.
* Create a status website using EC2, S3 ,IAM roles and bash script.
* Use AWS scheduler to shut down EC2 instances.
* Build a static website with a custom domain, using Route 53 Alias record sets and s3 bucket.
* Build a website from two EC2 instances, and will route traffic using an Application Load Balancer ELB ,create and manage DNS records inside of Route 53.
* Troubleshoot elastic load balancer connectivity.
* Manage the deployment of EC2 instances in an Auto Scaling Group using Lifecycle Hooks.
* Backup and restore from dynamoDB.
* Use the Database Migrations Service (DMS) to migrate a database from an EC2 server to RDS.
* Process DynamoDB stream using lambda.
* Migrating from a Relational Database to DynamoDB.
* Use Data Pipeline to copy DynamoDB data to an S3 bucket.
* Create and configure a CloudTrail trail and a CloudWatch Logs log stream in order to set up monitoring and access alerts for an S3 bucket.
* AWS CloudWatch Logs and Incident Response using Elasticsearch ,Route 53 ,VPC Flow Logs,Elasticsearch Service and Kibana.
* Configure CloudWatch with SNS and send email on CloudWatch Alarm.
* Build ci/cd pipeline using AWS Pipeline to deploy static website on s3.
* Build a simple serverless website with route53 , API gateway and lambda.
* Serverless web page with API gateway and lambda .
* Create a reminder serverless application using a static website hosted on S3 using AWS Step Functions, API Gateway, AWS Lambda.
* Trigger a Lambda function using SQS and dynamoDB.
* See logs using cloudwatch & create billing alarms.
* Deploy a basic infrastructure using CloudFormation Templates.
* Deploy a website canary with cloudformation.
* Roll updates to a highly distributed web application with AWS CodeDeploy.
* Implement AWS CodePipeline to deploy AWS infrastructure through AWS CloudFormation.
* Create a website canary.
* Work with SQS standand queues and FIFO queues.
* Create and subscribe to AWS SNS topics.
* Create a Beanstalk application using CLI .
* Patching live kubernetes deployments and drai a node for maintenance.

**Kubernetes Hands-on task**

* Install and test components of kubernetes cluster.
* Setup kubernetes cluster with docker.
* Deploy Go microservice to kubernetes.
* Create a kubernetes cluster.
* Generate kubeconfig.
* Set up a frontend load balancer.
* Bootstrap kubernetes control panel
* Bootstrap worker nodes.
* AutoScale deployment..
* Schedule pods with taints & tolerance.
* Deploy pod to a node with a label.
* Configure probes for pods.
* Implement state persistence for pods.
* Create PV for pods.
* Perform rolling update of a Go application.
* Configure cron job.
* Roll update of application.
* Set up a self healing application.
* Canary deployment with jenkins and kubernetes.
* Istio in kubernetes
* Pod Security Policy
* Monitor and output logs to a file.
* Monitoring with Prometheus & gafana.

# Create a Service and Discovering DNS Name

* Expose service to the internet.
* Create a cluster role to access PV.
* Repair failed pod
* Create alert rules.
* Practice common kubernetes debugging skill
* Smoke test the cluster
* CKAD Practice exam-1
* CKAD Practice exam-2
* CKAD Practice exam-3.