appscrip
ACCELERATING BUSINESS

DevOps Interview Assignment

Objective

Create a complete CI/CD infrastructure pipeline on AWS using Terraform, Kubernetes, ArgoCD, and optionally, Ingress + DNS.

Time line 1 to 2 days Maximum

Task Requirements

1. Provision AWS EKS Cluster using Terraform

- Use Terraform to:
  - Create a VPC (optional: use EKS module with VPC support)
  - Deploy an EKS Cluster
  - Create IAM roles and node groups
  - Output kubeconfig credentials to access the cluster.

2. Deploy an NGINX Application using a Kubernetes Manifest

- Create Kubernetes manifest files (YAML) for:
  - Deployment
  - Service (NodePort or ClusterIP)
- Apply the manifests using kubectl apply or sync using ArgoCD.

3. Set Up ArgoCD on EKS

- Install ArgoCD in the argocd namespace.
- Expose the ArgoCD server using a LoadBalancer or port-forward.
- Create an ArgoCD Application resource that points to a Git repository with the NGINX manifests.

4. Access the NGINX Application

- Show that NGINX is deployed and accessible via:
  - kubectl port-forward OR
  - LoadBalancer Service (optional)

5. (Optional Bonus) Expose NGINX via Ingress + Custom Domain

- Install an Ingress Controller (e.g., AWS ALB Ingress Controller or NGINX Ingress Controller).
- Create an Ingress resource for the NGINX application.
- Point a domain (can be a free domain or subdomain) to the Ingress IP using Route 53 or any DNS provider.
- Provide screenshot or terminal output showing DNS resolution.

Deliverables

GitHub Repository with:
- terraform/ folder: EKS cluster infrastructure code.
- manifests/ folder: Kubernetes manifests for NGINX.
- argocd/ folder: ArgoCD Application resource YAML.
- README.md with:
  - Steps to provision the cluster
  - ArgoCD login instructions
  - Port-forward or public access URL for NGINX
  - (Optional) Ingress domain details

Evaluation Criteria

Criteria	Description
Terraform Setup	Modular, reusable code for EKS
Kubernetes Manifests	Correct syntax, functional deployment
ArgoCD Usage	GitOps principles properly applied
Accessibility	NGINX accessible via port-forward or Ingress
Documentation	Clear, reproducible steps in README
Bonus	Ingress + DNS domain mapping (extra points)