Devops Q&A:

**How have you utilized Infrastructure as Code (IaC) in your previous roles?**

I have designed and implemented Terraform scripts to provision and manage resources across **AWS, Azure, and GCP** depending on project needs. My responsibilities included:

* **Modular Design & Reusability**: Built Terraform modules for commonly used components such as VPCs, security groups, EC2 instances, EKS clusters, IAM policies, and S3 buckets. This modular approach improved reusability and reduced duplication across multiple projects.
* **State Management & Collaboration**: Configured **remote state management** using S3 + DynamoDB (for AWS) and integrated with Git for version control, ensuring that multiple team members could safely collaborate without state conflicts.
* **Environment Strategy**: Managed multiple environments (dev, QA, staging, prod) using Terraform workspaces and variable files, ensuring consistency and compliance across the SDLC.
* **CI/CD Integration**: Integrated Terraform with **Jenkins/GitHub Actions pipelines**, enabling automated plan and apply stages, with peer review and approvals before deployment. This reduced manual interventions and improved auditability.
* **Governance & Security**: Enforced tagging standards, IAM role-based access, and implemented guardrails for cost control and compliance.
* **Impact**:
  + Reduced infrastructure provisioning time from days to under 30 minutes.
  + Improved disaster recovery readiness by codifying infrastructure blueprints.
  + Enabled development teams to spin up their own environments safely and on-demand.
  + Brought consistency, scalability, and compliance to cloud operations.

Overall, IaC with Terraform has allowed me to deliver **scalable, secure, and automated infrastructure**, while also mentoring junior engineers on best practices and reviewing their IaC implementations."