Vass Computer “Not Just Training – A Path to Your IT Dream Job”

Terraform Administration(30 hours)

Module 1: Introduction to Terraform

- What is Infrastructure as Code (IaC)?
- Why Terraform? Benefits and real-world use cases
- Terraform vs other tools (CloudFormation, Pulumi, Ansible)
- Overview of HashiCorp tool ecosystem

Module 2: Installing and Setting Up Terraform

- Installing Terraform on Linux, Windows, macOS
- Terraform CLI basics
- Setting up VS Code + Terraform plugin
- Writing your first `.tf` file (Hello World on local)

Module 3: Core Terraform Concepts

- Terraform Providers
- Terraform Resources and Data sources
- Terraform State (`terraform.tfstate`)
- Plan, Apply, Destroy workflow
- Understanding `terraform init`, `plan`, `apply`, `destroy`
- Lifecycle of a Terraform deployment

Module 4: Variables and Outputs

- Input variables (`var`)
- Default values and type constraints
- Variable files (`.tfvars`)
- Output values and interpolation
- Sensitive variables

Module 5: Terraform State Management

- Purpose of `terraform.tfstate`
- Local vs remote state
- Backends (local, S3, Consul, etc.)
- State locking
- State manipulation with `terraform state` commands

Module 6: Terraform Modules

- What are modules?
- Creating reusable modules
- Module structure (main.tf, variables.tf, outputs.tf)
- Using public modules from the Terraform Registry
- Nested modules and best practices

Module 7: Provisioners and Dependencies

- `local-exec` and `remote-exec` provisioners
- Resource dependencies with `depends_on`
- Null resource

- Provisioners: when and why to avoid them

Module 8: Data Sources and Dynamic Blocks

- Fetching external data using `data` blocks
- Using `dynamic` blocks for looping over nested structures
- Conditional expressions (`count`, `for_each`, ternary ops)
- Loops with `for`, `for_each`, and `count`

Module 9: Remote Backends and Workspaces

- S3 with DynamoDB locking (AWS)
- Terraform Cloud and Terraform Enterprise
- Workspaces: default vs named
- When to use workspaces vs modules

Module 10: Managing Terraform in Teams

- Code organization strategies
- Separation of dev, staging, and prod environments
- CI/CD with Terraform (GitHub Actions, Jenkins, GitLab CI)
- Collaboration using Terraform Cloud

Module 11: Security and Best Practices

- Managing secrets (AWS credentials, Vault, SSM)
- Sensitive data handling (`sensitive = true`)
- Version pinning of providers and Terraform itself
- Terraform validate, fmt, and lint tools
- Code review and approval process

Module 12: Terraform with AWS (Practical Focus)

- Configuring AWS provider
- Creating EC2 instances, VPC, Subnets, SGs
- Creating IAM roles and policies
- Managing S3, RDS, and Lambda functions
- Using Terraform to manage Route 53 and ELB

Module 13: Advanced Terraform Features

- Resource targeting (`-target`)
- Importing existing infrastructure (`terraform import`)
- Tainting and replacement strategies
- Debugging and logs
- Using `locals` for code clarity
- `terraform console` for testing expressions