INFOSOFT IT SOLUTIONS

Training | Projects | Placements

HASHI CROP TERRAFORM TRAINING

Core Concepts

1. **Introduction to Terraform**
   - Overview of Infrastructure as Code (IaC)
   - Benefits of Terraform
   - Comparison with other IaC tools (e.g., Ansible, Chef, Puppet)
2. **Terraform Basics**
   - Terraform architecture
   - Key components: Providers, Resources, Variables, Outputs, and Modules

Installation and Configuration

3. **Setting Up Terraform**
   - Installation on different operating systems (Windows, macOS, Linux)
   - Configuring Terraform CLI
   - Initializing a working directory with `terraform init`

Writing and Understanding Terraform Configuration

4. **Hashi Corp Configuration Language (HCL)**
   - Syntax and structure
   - Defining Providers and Resources
   - Using Variables and Outputs
   - Resource Dependencies
5. **Advanced Configuration**
   - Dynamic Blocks
   - Conditional Expressions
   - Built-in Functions

Managing Terraform State

6. **State Management**
   - Purpose and structure of Terraform state files
   - State file locations (local and remote)
   - State locking and consistency
7. **State Manipulation**
   - Commands: `terraform state`, `terraform import`, `terraform taint`, `terraform untaint`
   - Handling state drift
   - Migrating state between backends

Terraform Workflow

8. **Core Terraform Commands**
   - Writing configuration: `terraform plan`, `terraform apply`
   - Destroying infrastructure: `terraform destroy`
   - Formatting and validating: `terraform fmt`, `terraform validate`

Modular Code

9. **Modules**
   - Creating and using modules
   - Organizing and versioning modules
   - Publishing modules in the Terraform Registry

Provisioning and Managing Infrastructure

10. **Resource Provisioning**
    - Provisioning resources in AWS, Azure, GCP, and other providers
    - Using provisioners (e.g., `local-exec`, `remote-exec`)
    - Handling dependencies and lifecycle rules

11. **Variables and Outputs**
    - Defining input variables
    - Variable precedence and overriding

- Output values and data sharing

Collaboration and Automation

12. **Terraform Cloud and Enterprise**
    - Remote state management
    - Collaboration and workspace management
    - Sentinel policies for policy as code
    - VCS integration
13. **CI/CD Integration**
    - Automating Terraform with CI/CD tools (e.g., Jenkins, GitLab CI, GitHub Actions)
    - Best practices for pipeline integration

Security and Compliance

14. **Secrets Management**
    - Integrating with HashiCorp Vault
    - Managing sensitive data in Terraform configurations
15. **Policy as Code**
    - Writing and applying Sentinel policies
    - Enforcing compliance and governance

16. **Debugging and Troubleshooting**
    - Common error messages and solutions
    - Using logging and debugging features
17. **Performance Optimization**
    - Efficient resource management
    - State file performance tuning
18. **Custom Providers and Plugins**
    - Writing custom providers
    - Using third-party plugins

19. **Advanced Topics**

- **Terraform Cloud/Enterprise Overview**

    - Workspaces
    - Version Control Integration
    - Private Module Registry
    - Team Management

- Workspace Automation
- API Driven Workflows

- **<u>Code Management and Optimization</u>**

  - Modularization and Code Organization
  - DRY Programming Concepts
  - Using Null Resources, Dynamic Blocks, and Conditional Operators
  - Secure Variables Management

- **<u>Collaborative Workflows</u>**

  - Terraform Cloud/Enterprise Workflows
  - Automated Testing
  - Sentinel for Policy as Code
  - Policy Development and Management

- **<u>Deployment Strategies</u>**

  - Blue-Green Deployments
  - Canary Tests
  - Feature Flags