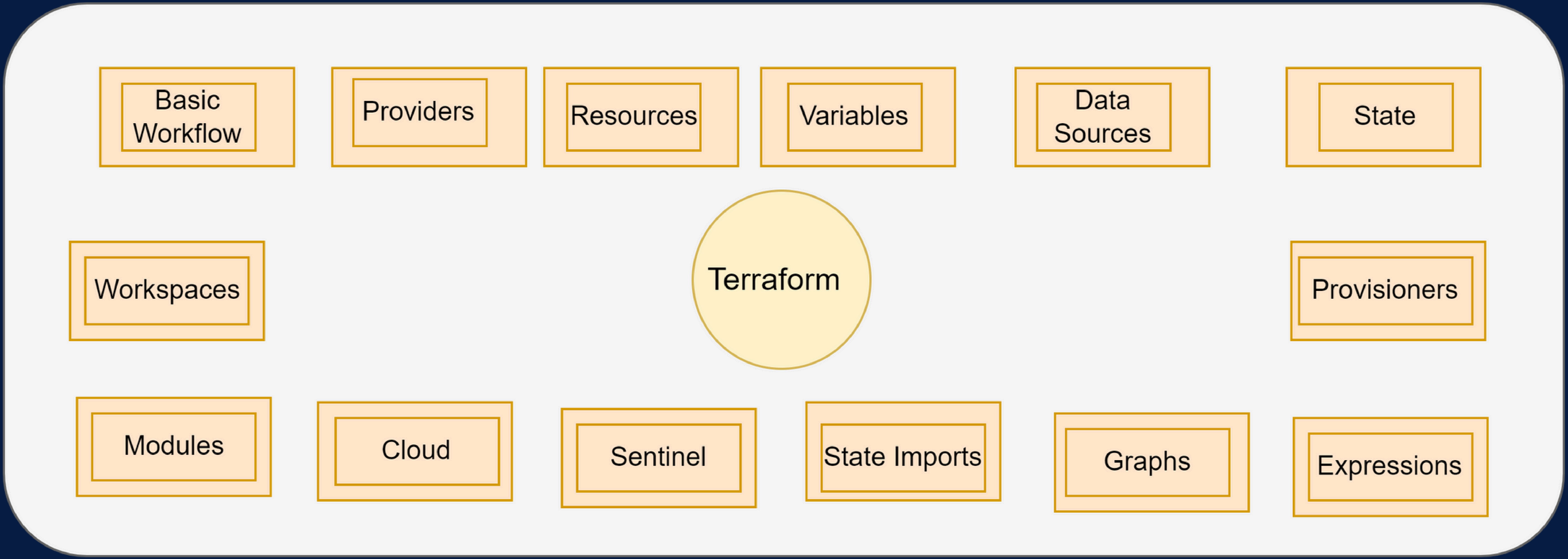
HashiCorp
Terraform

Day 1: Introduction to Terraform and Infrastructure as Code (IaC)

What is Terraform?

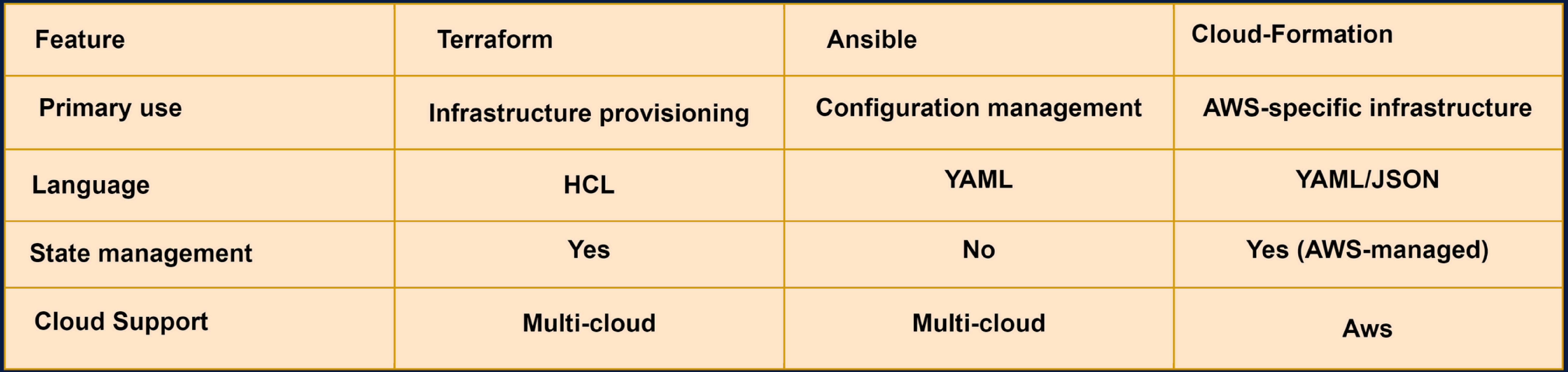
Terraform is an open-source Infrastructure as Code (IaC) tool developed by HashiCorp.
It allows users to define, provision, and manage infrastructure resources across various cloud providers and services using a declarative configuration language.



Key features of Terraform:

1. Declarative language (HashiCorp Configuration Language - HCL).
2. Multi-provider support.
3. State management.
4. Plan and apply workflow.
5. Modular architecture.

Comparison with other IaC tools:



Feature	Terraform	Ansible	Cloud-Formation
Primary use	Infrastructure provisioning	Configuration management	AWS-specific infrastructure
Language	HCL	YAML	YAML/JSON
State management	Yes	No	Yes (AWS-managed)
Cloud Support	Multi-cloud	Multi-cloud	Aws

Case study:

A global financial services company leverages Terraform to manage its infrastructure across AWS, Azure, and on-premises data centers.
By using Terraform, they achieve:
1. Consistent resource provisioning across environments.
2. Rapid replication of environments for development, testing, and production.
3. Version-controlled infrastructure changes.
4. Reduced risk of configuration drift.
5. Improved compliance and auditing capabilities.

Hands-On Lab:

1. MacOS (using Homebrew):

```
brew tap hashicorp/tap
brew install hashicorp/tap/terraform
```

2. Linux (Ubuntu/Debian):

Update Package List and Install Dependencies

```
sudo apt-get update && sudo apt-get install -y gnupg
software-properties-common curl
```

Add HashiCorp GPG Key:

```
curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key add
```

Add HashiCorp Linux Repository

```
sudo apt-add-repository "deb [arch=amd64] https://apt.releases.hashicorp.com $(lsb_release -cs) main"
```

Install Terraform

```
sudo apt-get update && sudo apt-get install terraform
```

Windows:

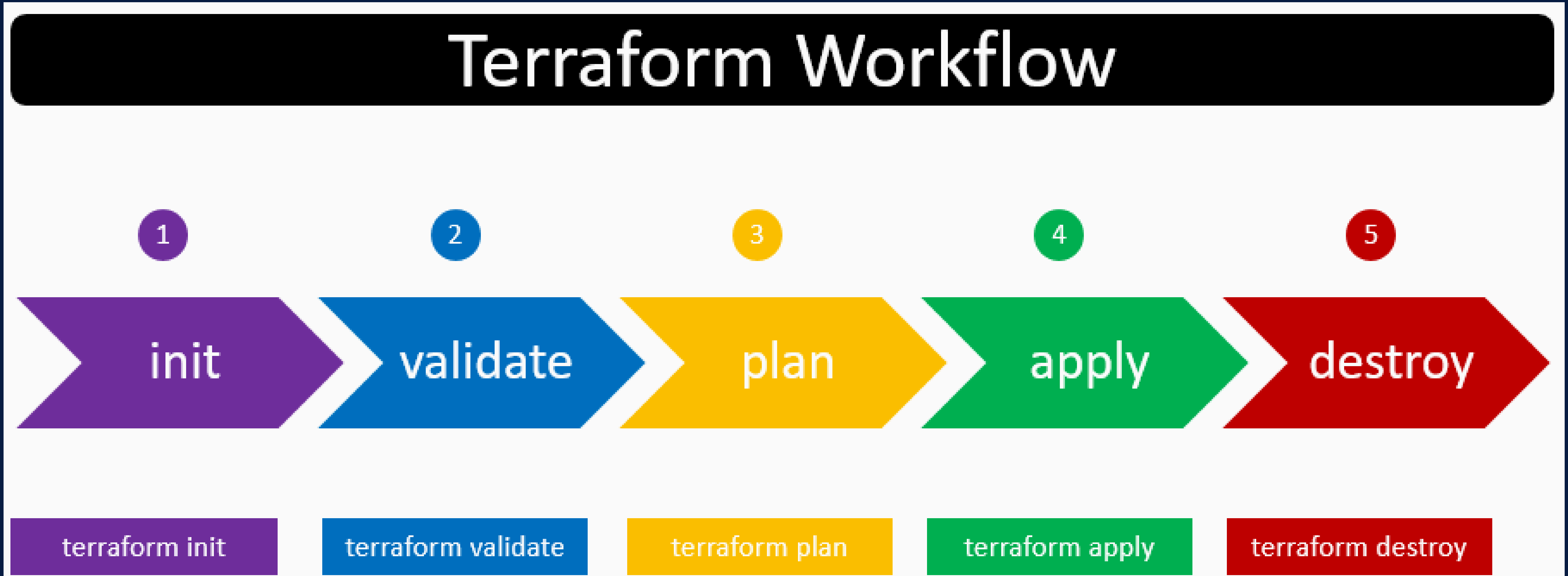
Download the Terraform ZIP file from the official HashiCorp website
Extract the ZIP file to a directory of your choice (e.g., `C:\terraform`)

Add the directory to your system's PATH environment variable
After installation, verify Terraform is installed correctly:

By Using This Command: terraform version

Terraform Workflow

terraform init
terraform validate
terraform plan
terraform apply
terraform destroy



Terraform Workflow

1 init — terraform init

2 validate — terraform validate

3 plan — terraform plan

4 apply — terraform apply

5 destroy — terraform destroy

Questions on the advantages of IaC and Terraform's purpose

1. What are the primary benefits of using Infrastructure as Code? (Select all that apply)

a) Version control
b) Consistency across environments
c) Automated provisioning
d) Reduced human error
e) All of the above

2. Which of the following best describes Terraform's primary purpose?
a) Configuration management
b) Infrastructure provisioning
c) Application deployment
d) Network monitoring

3. Terraform uses a ________ approach to define infrastructure.
a) Imperative
b) Declarative
c) Procedural
d) Reactive

4. Which of the following is NOT a key feature of Terraform?

a) Multi-cloud support

b) State management

c) Container orchestration

d) Resource graph

5. In the context of Terraform, what does HCL stand for?

a) HashiCorp Configuration Language

b) High-level Computing Language

c) Hybrid Cloud Logic

d) Hardware Control Layer