ADVANCED DEVOPS
EXPERIMENT NO 5

* Aim :- To understand terraform lifecycle, core concepts/
terminologies and install it on a linux machine
or windows.

* Theory :-

Terraform is a open source infrastructure as
code (IAC) tool developed by HashiCorp. It allows
user to define and provision infrastructure
using a high level configuration language known
as HashiCorp Configuration Language (HCL)
or JSON. Terraform supports a wide range of
cloud providers, such as AWS, Azure, Google Cloud
and on premises solutions, enabling users to
manage infrastructure accross multiple environments
consistently.

Terraform Lifecycle :-

① Write :- Write the configuration file.
② Initialize :- Initialize working directory.
③ Plan :- Execution plan is created based on
          configuration files.
④ Apply :- Apply the desired changes.
⑤ Destroy :- Destroy the infrastructure managed
            by terraform.

Core Concepts and terminologies:-

① <u>Providers</u>:- Plugins that allow it to interact with various APIs.

② <u>Resources</u>:- Represent components of your infrastructure.

③ <u>Modules</u>:- Containers for multiple resources that are used together.

④ <u>State</u>:- A file that keeps track of infrastructure managed by terraform.

⑤ <u>Variables</u>:- Used to make configurations dynamic and reusable.

⑥ <u>Outputs</u>:- Used to extract information from the infrastructure.

✱ <u>Conclusion</u>:-

All the core concepts and terminologies of the terraform were understood along with it's lifecycle. Also installation of terraform was also done successfully.

< Terraform Home

Install Terraform

Operating Systems

macOS

Windows

Linux

FreeBSD

OpenBSD

Solaris

Release information

Next steps

Resources

Tutorial Library

Certifications

Windows

Binary download

386		AMD64	
Version: 1.9.5	Download ⭳	Version: 1.9.5	Download ⭳

Linux

Package manager

[Ubuntu/Debian] CentOS/RHEL Fedora Amazon Linux Homebrew

```
wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/ha
echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashic
sudo apt update && sudo apt install terraform
```

Binary download

386		AMD64	
Version: 1.9.5	Download ⭳	Version: 1.9.5	Download ⭳

About Terraform

Define cloud and on-prem resources in human-readable configuration files that you can version, reuse, and share.

Featured docs

Introduction to Terraform

Configuration Language

Terraform CLI

HCP Terraform

Provider Use

HCP Terraform

Automate your infrastructure provisioning at any scale

Try HCP Terraform for free →

---

terraform   ×   +

← → ↑ ↻   ⧉ > This PC > Local Disk (C:) > terraform   Search terraform

⊕ New ⌄   ✂   ⧉   📋   ⟳   ↗   🗑   ↕ Sort ⌄   ☰ View ⌄   ⋯

Name	Date modified	Type	Size
📄 LICENSE	20-08-2024 23:05	Text Document	5 KB
🗔 terraform	20-08-2024 23:05	Application	88,962 KB

🏠 Home
🖼 Gallery
> ☁ Shreyash - Pe
> ☁ Shreyash - Pe

🖥 Desktop 📌
⬇ Download 📌
📄 Document 📌
🖼 Pictures 📌
🎵 Music 📌
🎬 Videos 📌
💽 Local Disk 📌
🎬 Short Film

```
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.

C:\terraform>terraform
Usage: terraform [global options] <subcommand> [args]

The available commands for execution are listed below.
The primary workflow commands are given first, followed by
less common or more advanced commands.

Main commands:
  init          Prepare your working directory for other commands
  validate      Check whether the configuration is valid
  plan          Show changes required by the current configuration
  apply         Create or update infrastructure
  destroy       Destroy previously-created infrastructure

All other commands:
  console       Try Terraform expressions at an interactive command prompt
  fmt           Reformat your configuration in the standard style
  force-unlock  Release a stuck lock on the current workspace
  get           Install or upgrade remote Terraform modules
  graph         Generate a Graphviz graph of the steps in an operation
  import        Associate existing infrastructure with a Terraform resource
  login         Obtain and save credentials for a remote host
  logout        Remove locally-stored credentials for a remote host
  metadata      Metadata related commands
  output        Show output values from your root module
  providers     Show the providers required for this configuration
  refresh       Update the state to match remote systems
  show          Show the current state or a saved plan
  state         Advanced state management
  taint         Mark a resource instance as not fully functional
  test          Execute integration tests for Terraform modules
  untaint       Remove the 'tainted' state from a resource instance
  version       Show the current Terraform version
  workspace     Workspace management

Global options (use these before the subcommand, if any):
  -chdir=DIR    Switch to a different working directory before executing the
                given subcommand.
  -help         Show this help output, or the help for a specified subcommand.
  -version      An alias for the "version" subcommand.

C:\terraform>
```

```
C:\terraform>terraform --version
Terraform v1.9.5
on windows_amd64

C:\terraform>docker --version
Docker version 27.0.3, build 7d4bcd8

C:\terraform>
```