Module 1: Introduction to Google Cloud Platform

- Introduction to Cloud Computing
- Overview of Google Cloud Platform
- Cloud Service Models [Infrastructure as a service (IAAS), Platform as a service (PAAS),Software as a service(SAAS)]
- Cloud Deployment Model [Public , Private, Hybrid , Multi Cloud]
- Creating a GCP Free Tier Account

Module 2: GCP Global Infrastructure

- Google's Global Backbone and Network
- Understanding Regions, Zones, PoPs and real use cases

Module 3: Organisation Account Setup

- What is a Cloud Identity Account?
- Setting up a Company Organization in GCP
- Understanding the difference between
- Google Workspace vs Cloud Identity
- Linking Cloud Identity to GCP for real-world org structure
- Creating an Organization Node and linking Billing
- Enabling required APIs for Org level

Module 3: Cloud Resource Manager

- What is a Cloud Resource Manager in GCP?
- Understanding the GCP resource hierarchy:
  i. `Organization > Folders > Projects > Resources`
- Creating and managing Projects
  i. Understanding Project id, Project Number, Project Name
- Best practices for project naming, labeling, and grouping

Module 4: Identity and Access Management:

- Introduction to IAM in Google Cloud
- IAM roles: Primitive, Predefined, and Custom
- Understanding Members, Roles, and Policies
- Role inheritance: Organization > Folder > Project > Resource

- Managing IAM policies via `Console`, `gcloud`, and `Terraform`
- Service Accounts creation and authentication

Module 5: SDK and Cloud Shell

- Introduction to Google Cloud SDK (gcloud CLI)
- Installing and configuring the SDK on local machines
- Understanding command-line tools
- Common `gcloud` commands for setting up things in cloud.
- Introduction to Google Cloud Shell
- Differences between SDK vs Cloud Shell
- Cloud Shell Editor and pre-installed tools

Module 6: Virtual Private Cloud (VPC)

- What is a VPC in Google Cloud?
- Default VPC vs Custom VPC
- Subnets: Regional scope and CIDR block planning
- Auto vs Manual subnet creation
- Routes and their types (system-generated and custom)
- Firewall rules: ingress vs egress
- Instance level firewalls using Network Tags
- Implementing Firewall Policy
- Internal vs External IP addressing
- StaticIP, Elastic Ip configuring, use cases
- VPC Peering, Shared VPC implementaion
- Private Google Access , Private Service access
- Serverless VPC
- Intro and Implementing Public NAT

Module 7: Google Compute Engine

- Introduction to GCE: Virtual Machines on Google Cloud
- Machine families: General-purpose, Compute-optimized, Memory-optimized
- Choosing machine types and custom configurations
- Creating and managing VM instances via Console, CLI, and Terraform
- Understanding VM networking and external/internal IPs
- Metadata, startup scripts, and SSH key management
- VM lifecycle: start, stop, suspend, terminate
- Understanding boot disks and attaching additional persistent disks
- Creating Custom Images in and across projects.
- Creating snapshots for backup and restore

Module 8 : Instance Templates & Instance Groups

- What is an Instance Template in GCP?
- Defining machine type, disk, image, metadata, and network in templates
- Creating and managing instance templates via Console, CLI, and Terraform
- Understanding Managed Instance Groups (MIGs)
    i. Auto-healing, auto-scaling, and rolling updates
- Unmanaged Instance Groups – use cases and limitations
- Deploying applications using MIGs across multiple zones
- Using health checks with managed instance groups
- Integration of MIGs with HTTP(S) Load Balancer
- Best practices for using templates in production environments

Module 9 : Load Balancers in GCP

- Overview of Load Balancing in Google Cloud.
- Understanding the importance of load balancing in modern application architecture.
- Exploring Various Types of Load Balancers
- Use cases for each type (web apps, internal services, hybrid connectivity)
- Backend services, health checks, and URL maps
- Forwarding rules, target proxies, and backend instance groups.
- Load balancer integration with Managed Instance Groups (MIGs)
- Session affinity, connection draining, and failover configurations
- Hands-on architecture: Multi-region Global HTTP(S) Load Balancer setup (covered later in labs)
- Best practices for high availability and performance.
- Understanding Google Cloud armor.

Module 10 : Google CDN

- Overview of Google Cloud CDN as part of the Google Cloud Platform.
- Creation of CDN
- Setting up Caching Content with Cloud CDN

Module 11 : Cloud DNS

- Introduction to Cloud DNS in Google Cloud
- Public vs Private DNS zones
- Creating DNS zones and managing records.
- Domain verification and delegation using NS records.
- Integrating DNS with Load Balancers.
- Using DNS for internal service discovery (Private Zones)

Module 12 : SSL/TLS Certificates in GCP

- Introduction to SSL/TLS for secure web applications.
- Google-managed vs Self-managed certificates.
- Creating and managing SSL certificates via Load Balancer

- Attaching certificates to HTTPS Load Balancers
- Auto-renewal of Google-managed certs (DNS verification required)
- Redirection from http to https

Module 13 : Google Cloud Storage

- Introduction to GCS and its use cases.
- Buckets: naming, location types (multi-region, region, dual-region)
- Storage classes: Standard, Nearline, Coldline, Archive
- Object versioning and lifecycle management
- Access control: Uniform vs Fine-grained, IAM vs ACLs
- Uploading/downloading files using Console, gsutil
- Integrating GCS with other GCP services

Module 14 : Cloud Key Management Service

- Introduction to Cloud KMS and its role in securing data
- What is Encryption ? Encryption at Rest , Transit
- Creating and Managing GMEK, CMEK, CSEK

Module 15 : Cloud SQL

- Introduction to Cloud SQL: Fully managed relational databases.
- Choosing the right machine type and storage configuration.
- Creating instances via Console, gcloud, and Terraform
- Public IP vs Private IP connectivity options.
- Connecting to Cloud SQL from Compute Engine, Shell and other clients.
- Securing access: SSL certificates, Cloud SQL Auth Proxy.
- High Availability (HA) configuration and automatic failover
- Creating and Managing Read Replicas in Cloud SQL
- Implementing cross-region replication (DR setup)
- Backups, automated maintenance.
- Database user management and IAM integration.
- Importing and exporting databases (SQL, CSV, and backup formats)

Module 16 : Cloud Run Services

- Introduction to Cloud Run: Fully managed container execution
- Difference between Cloud Run, Cloud Functions, and GKE
- Deploying stateless HTTP containers from source or container registry
- Automatic scaling (scale to zero) and cold start behavior
- Logging and monitoring with Cloud Logging
- How to Rollout and Rollback version in Cloud run
- Traffic splitting in Cloud Run
- Triggering and scheduling Cloud run
- Deploying a java based application in cloud run and connecting to cloud sql private with private service connection.
- Mapping custom domain to cloud run service with https.

Module 17 : Google Artifact Registry:

- What's Artifact Registry
- Creating Docker based repo and pushing images to Artifact repo
- Creating Maven based repo and pushing jar files to Artifact

Module 18 : Cloud Run Jobs

- Introduction to Google Cloud Run Jobs
- Implementing Cloud Run Jobs: one-off and scheduled jobs
- Deploy cloud run job and verify using gcloud

Module 19 : Cloud Build

- Introduction to Cloud Build: CI/CD platform on Google Cloud
- Key components: Triggers, Build Steps, Cloudbuild.yaml
- Creating builds from source repositories (GitHub, Bitbucket)
- Building and pushing Docker images to Artifact Registry
- Automating deployments to GKE, Cloud Run, and Compute Engine.
- Secure builds with IAM permissions and service accounts.
- Implementing approval mechanism for prod deployments.

Module 20 : Hybrid Connectivity

- What is hybrid cloud and why it's needed
- Overview of connectivity options between on-premises and GCP
- Establishing a High available VPN
- Understand Interconnect options
- Implementing CloudRouter for dynamic routing

Module 21 : Cloud Functions

- Introduction to Cloud Functions
- Different Gen functions
- Implementing cloud functions with google services

Module 22 : Cloud Spanner & Memorystore

- Introduction to Cloud Spanner: Globally distributed relational database.
- Combines relational schema, horizontal scalability, and strong consistency
- What is Memorystore: Fully managed in-memory data store
- Supports Redis and Memcached engine.
- Use cases: caching, real-time analytics, session management

Module 23 : Database Migration Service (DMS)

- Introduction to Database Migration Service in GCP
- Use cases: lift-and-shift migration of MySQL and PostgreSQL to Cloud SQL
- Types of migration:

    i. One-time migration (dump and load)
    ii. Continuous replication (for minimal downtime)
  - Monitoring migration progress and logs
  - Validating data integrity post-migration

Module 24 : Cloud Pub/Sub

  - Introduction to Cloud Pub/Sub
  - Implementing pub/sub with gcp resources.

Module 25 : Monitoring and Logging

  - What is Operations?
  - Cloud monitoring concepts
  - Monitoring Workspaces
  - What are Metrics
  - Creating Dashboards
  - Implementing Uptime Checks.
  - Logging in Google Cloud

Module 26 : Google Kubernetes Engine

  - What is Kubernetes?
  - What is Google Kubernetes Engine, how it differs from regular Kubernetes.
  - Creating and connecting to GKE.

Module 27: Interview Preparation

Master Infrastructure as a Code with Terraform

From Fundamentals to Advanced including Certification

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

- Challenges with Traditional IT Infrastructure
- Types of IaC: Declarative vs. Imperative
- Why and What is Terraform?
- Overview of Terraform and its advantages over traditional methods
- Key use cases in GCP

Module 2: Getting Started with Terraform, gcloud, and IDE Setup

- Installing Terraform on Windows, macOS, and Linux
- Installing Google Cloud SDK (gcloud CLI)
- Authenticating gcloud to GCP:
  i. Using Principal Account Authentication
  ii. Using Service Account Authentication
  iii. Application-Based Authentication
- Setting up IDEs for Terraform:
  i. Visual Studio Code with Terraform plugins
  ii. JetBrains IntelliJ with Terraform support

Module 3 : Terraform Architecture

- Understanding Terraform Workflow: Initialization, Planning, Applying, Destroying
- Terraform Core Commands: `terraform init, terraform plan, terraform apply, terraform destroy`
- Terraform Configuration Syntax: Overview of how configurations are structured in .tf files
- Terraform Settings Block: Specifying provider versions, Terraform version, etc.
- Terraform Provider Block: Defining and configuring the Google Cloud provider
- Dependency Lock File: Understanding the .terraform.lock.hcl file and version locking

- Terraform Resource Block: Defining resources such as GCP VM instances, networks
- Terraform Data Block: Using data sources to reference external resources
- Handling Multiple Providers: Managing and configuring multiple cloud providers in the same Terraform configuration
- Terraform State File

Module 4 : Terraform Providers and the Provider Registry

- What are Terraform Providers?
- Understanding the Terraform Provider Registry: Exploring available providers for GCP and other platforms
- Installing and Versioning Terraform Providers: Managing provider versions and ensuring compatibility
- Explain Multi-Cloud and Provider-Agnostic Benefits: Advantages of using Terraform across multiple cloud platforms
- Writing Terraform Configuration Using Multiple Providers: Examples of using multiple providers within a single configuration
- Describe How Terraform Finds and Fetches Providers: How Terraform discovers and downloads providers from the registry

Module 5 : Terraform Settings Block

- Defining the Terraform Settings Block: The purpose and components of the terraform block
- Specifying Required Terraform Version: Ensuring compatibility by specifying the minimum required Terraform version
- Defining Required Providers: Specifying which providers are required for the configuration
- Adding Version Constraints: Setting constraints on provider versions to ensure compatibility and avoid breaking changes
- Configuring Backend Settings: Setting up remote state backends such as GCS or S3

Module 6 : Terraform Resource Block

- What is the Terraform Resource Block?: Understanding the structure and purpose of resource blocks
- Terraform Resource Block Syntax: Proper syntax and structure for defining resources
- Creating Resources: How to define and create resources like VM instances, networks, and more in GCP
- Understanding Resource Arguments and Attributes: Accessing and using resource arguments (e.g., machine types) and attributes (e.g., IP addresses)
- Resource Behavior:
    i. Create resources
    ii. Destroy resources
    iii. Update in-place resources
    iv. Destroy and re-create resources

Module 7 : Terraform Data Sources

- What are Data Sources in Terraform?: Understanding the purpose of data sources
- Using Data Sources to Fetch Existing Infrastructure: Querying existing resources in GCP
- Data Source Syntax: Proper structure and usage of data sources in Terraform
- Common Data Sources in GCP: Examples like google_compute_image, google_project
- Combining Data Sources and Resources: Using data sources to provide input for resource creation

Module 8: Terraform Meta Arguments

- What are Meta Arguments?
- Using count for Creating Multiple Instances: Dynamically creating multiple instances of a resource
- Using for_each for Iterating Over Complex Collections: Handling multiple items with more control over the resource creation process
- Using depends_on to Manage Resource Dependencies: Explicitly defining dependencies between resources
- Using lifecycle to Control Resource Lifecycle: Managing creation, update, and deletion behavior for resources
- Using provider to Override the Default Provider: Assigning a specific provider to a resource block

Module 9: Terraform Input Variables

- Using What are Input Variables?: Understanding how to use input variables for dynamic configuration
- Defining Input Variables: Syntax and structure of variable blocks
- Variable Types: Understanding string, number, list, map, and complex object types
- Providing Input Values: Multiple ways to supply values, including:
- Using default values
    i. Interactive input prompts during execution
    ii. Via CLI using -var or -var-file flags
    iii. Environment variables using TF_VAR_name
    iv. Using var-file for structured input
    v. Using a terraform.tfvars file
    vi. Auto vars in automatically loaded .auto.tfvars files
    vii. Sensitive variables to protect confidential input
- Using Default Values: Setting default values for variables
- Validating Variables: Implementing validation rules for input variables
- Variable Precedence: Understanding the order of precedence for input variables

Module 10: Terraform Output Values

- What are Output Values?: Understanding how to use outputs to display important information
- Defining Output Values: Syntax and structure for defining outputs in a configuration

- Accessing Resource Attributes: Using output values to display attributes from created resources (e.g., IP addresses, VM names)
- Using Output Values Across Modules: Passing values between modules for modular and reusable infrastructure
- Sensitive Outputs: Marking outputs as sensitive to hide confidential information
- Real-Time Implementations for Outputs

Module 11: Terraform Local Values

- What are Local Values?: Understanding the role of local values in Terraform
- Defining Local Values: Syntax and structure for creating local values
- Using Local Values for Reusability: Reducing repetition and improving maintainability in configurations
- Local Values vs. Input Variables vs. Output Values: Understanding the difference between these three types of values
- Real-Time Implementations for Local Values

Module 12: Terraform State

- What is Terraform State?: Understanding how Terraform tracks infrastructure
- Managing Local vs. Remote State: The difference between storing state locally and in a remote backend
- Setting up a GCS Backend for State Management: Using Google Cloud Storage (GCS) as a remote backend for state files
- State Locking: Ensuring consistency with state locking
- State Commands: Common state commands (terraform state list, terraform state show, etc.)
- Implementing terraform import into state file
- Real-Time Implementations for Terraform State

Module 13: Terraform Workspaces

- What are Workspaces?: Understanding how Terraform uses workspaces to manage multiple environments
- Creating and Using Workspaces: Managing environments (dev, staging, prod) with workspaces
- Switching Between Workspaces: Commands for switching and managing workspaces
- Real-Time Implementations for Workspaces

Module 14: Terraform Provisioners

- What What are Provisioners?: Understanding the purpose of provisioners in Terraform
- Types of Provisioners:
  i. Local Provisioners: Running local scripts or commands
  ii. Remote Provisioners: Executing commands on remote resources.
  iii. File Provisioners: Uploading files to remote machines
- Provisioner Connections: Establishing connections to remote instances
- Handling Resource Creation Failures with Provisioners: Understanding the on_failure behavior
- When to Use Provisioners: Best practices and avoiding common pitfalls

- Real-Time Implementations for Provisioners

Module 15: Terraform Functions

- What are Functions in Terraform?: Overview of how functions are used in Terraform
- Below are the few functions we see use cases for.
- String Functions: join(), split(), format()
- Numeric Functions: min(), max()
- Collection Functions: length(), concat()
- Filesystem Functions: file(), filebase64()
- Date and Time Functions: timestamp()
- Real-Time Implementations for Functions

Module 16: Terraform Modules

- What are Terraform Modules?: Understanding how modules help with code reusability
- Creating and Using Modules: Structuring Terraform code using modules
- Module Sources: Using local, Git, and Terraform Registry modules
- Passing Variables Between Modules: Sharing input variables across modules
- Real-Time Implementations for Modules

Module 17: Integrating with CI Server

Module 18: Certification Preparation