Parul® University

Specialization - Cloud Computing - I

Prof. Dhruv Shah

Unit – 6

Google Cloud Platform (GCP)

The GCP Console

Introduction

- The GCP Console is a web-based interface used to manage GCP resources and services.
- It provides a graphical interface for interacting with Google Cloud services, monitoring, and configuring projects.

Benefits

- User-friendly interface
- Centralized management of cloud resources
- Role-based access controls for security

Key Features

- Resource Management: Create and manage GCP projects, users, and billing accounts.
- Service Navigation: Access various services like Compute Engine, Storage, Kubernetes, etc.
- Monitoring & Logging: View real-time logs and monitor cloud resources via Stackdriver.
- Security Controls: Manage Identity and Access Management (IAM) settings.
- Billing & Budgets: Track usage and costs.

Consol

Understanding GCP Projects

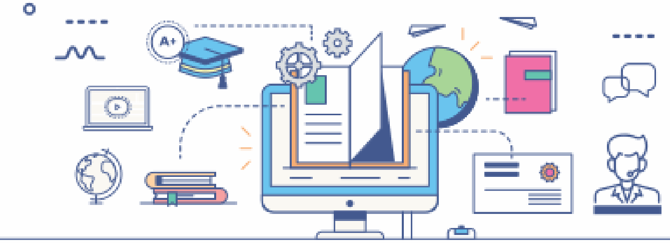
What is a GCP Project?

- A GCP Project is a fundamental organizing entity for cloud resources.
- Each project contains services, APIs, billing information, and IAM roles.

Key Components of a GCP Project

- Project ID: A unique identifier for your project.
- Project Number: A system-generated number.
- Billing Account: Links your project to billing information.
- APIs & Services: Enable and manage GCP APIs.
- IAM Policies: Control access and permissions.

Project Console



Project Lifecycle

- Create a Project
- Configure Services & APIs
- Set Up IAM Roles & Policies
- Deploy Applications & Services
- Monitor & Manage Resources
- Delete (If No Longer Needed)

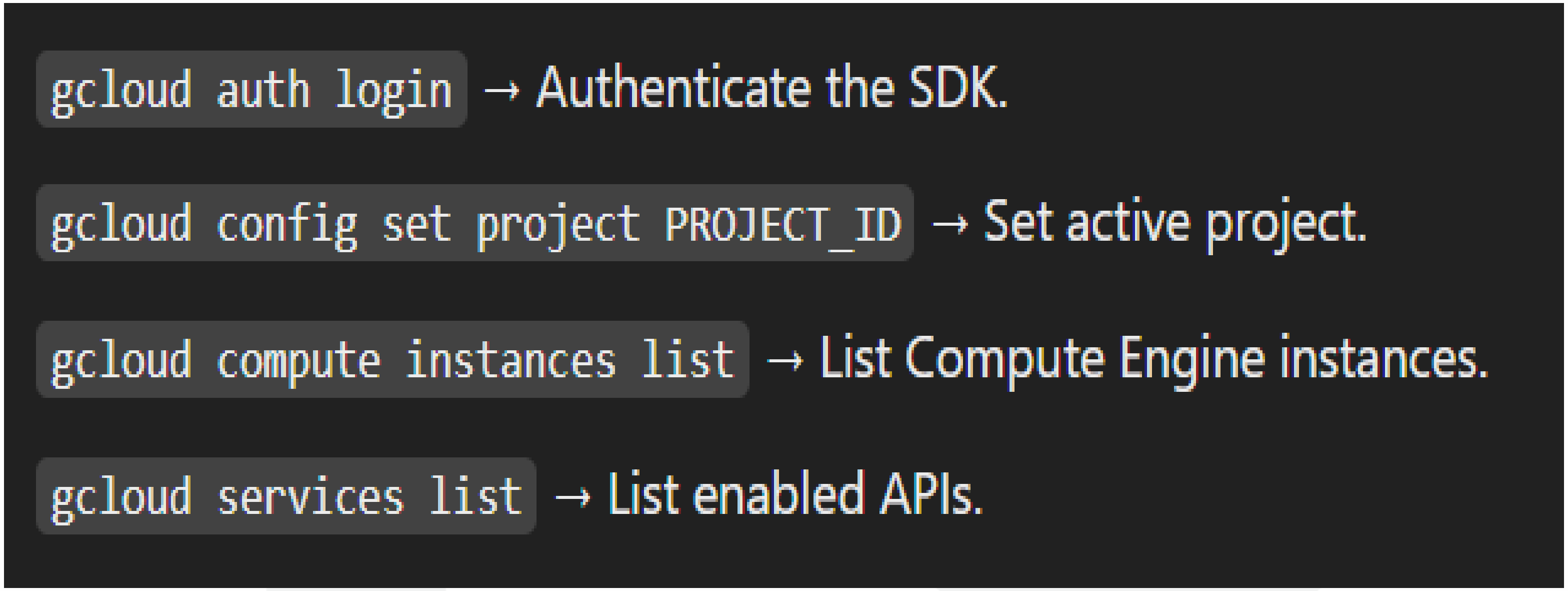
Install and Configure Cloud SDK

- Google Cloud SDK (Software Development Kit) is a set of command-line tools for managing GCP resources.
- It includes the gcloud, gsutil, and bq commands.

Installation Steps

- Download SDK: Get the installer from Google Cloud SDK website.
- Install the SDK: Follow installation steps based on OS (Windows, Mac, Linux).
- Initialize the SDK:
- Run gcloud init to configure the SDK.
- Authenticate using gcloud auth login.
- Set your preferred project: gcloud config set project [PROJECT_ID].

Key CLI Commands



`gcloud auth login` → Authenticate the SDK.

`gcloud config set project PROJECT_ID` → Set active project.

`gcloud compute instances list` → List Compute Engine instances.

`gcloud services list` → List enabled APIs.

Cloud Shell

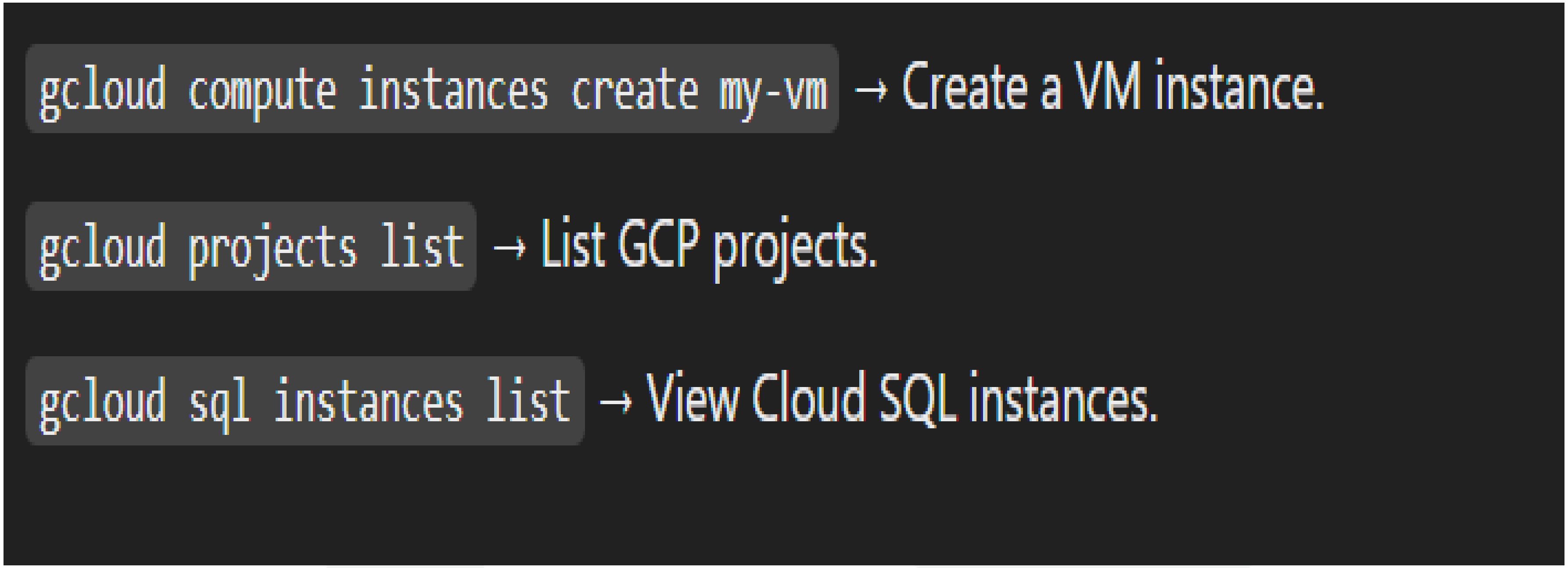
What is Cloud Shell?

- Cloud Shell is an interactive shell environment available directly in the GCP Console.
- It provides pre-configured tools (Cloud SDK, Git, Python) for managing GCP resources.

Advantages

- No local setup required.
- Comes with pre-installed SDK and essential tools.
- Persistent storage (5GB per user).

Common Cloud Shell Commands



`gcloud compute instances create my-vm` → Create a VM instance.

`gcloud projects list` → List GCP projects.

`gcloud sql instances list` → View Cloud SQL instances.

GCP APIs

Overview

- GCP APIs allow developers to programmatically interact with GCP services.
- They provide REST and gRPC interfaces for automation.

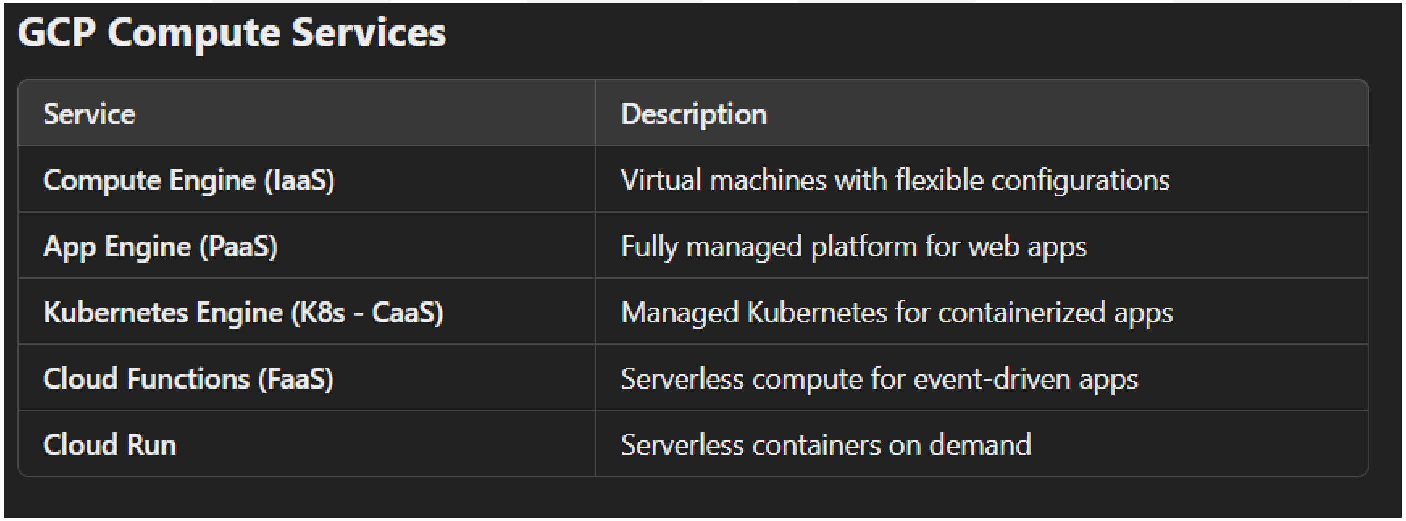
Common GCP APIs

- Compute Engine API → Manage virtual machines.
- Cloud Storage API → Store and retrieve files.
- BigQuery API → Run SQL-like queries on large datasets.
- Cloud Functions API → Deploy serverless functions.

Compute Options in the Cloud

Introduction

- Compute resources power cloud applications.
- GCP offers multiple computing services, each suited for different workloads.



GCP Compute Services

Service	Description
Compute Engine (IaaS)	Virtual machines with flexible configurations
App Engine (PaaS)	Fully managed platform for web apps
Kubernetes Engine (K8s - CaaS)	Managed Kubernetes for containerized apps
Cloud Functions (FaaS)	Serverless compute for event-driven apps
Cloud Run	Serverless containers on demand

What is Compute Engine?

- Go to GCP Console > Compute Engine > VM Instances.
- Click “Create Instance”.
- Configure instance:
- Machine Type: Select CPU and RAM.
- Boot Disk: Choose OS (e.g., Debian, Ubuntu).
- Firewall Rules: Allow HTTP/S traffic if needed.
- Click Create and start the VM.

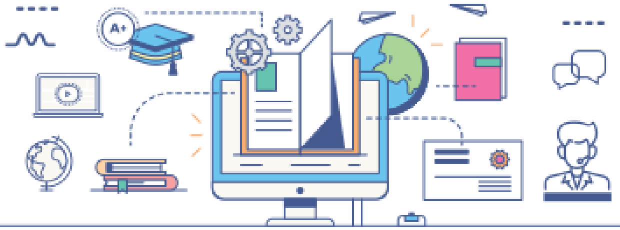
Configuring Elastic Apps with Autoscaling

What is Autoscaling?

- Autoscaling automatically adjusts the number of VM instances based on demand.
- Helps optimize cost and performance.

Types of Autoscaling in GCP

- Compute Engine Autoscaler: Adjusts VM count based on CPU usage, requests per second, etc.
- Kubernetes Autoscaler: Scales containers dynamically.
- App Engine Autoscaler: Scales web applications automatically



Steps to Configure Autoscaling

- Navigate to Compute Engine > Instance Groups.
- Click Create Instance Group.
- Set Scaling Policy:
- Scale based on CPU utilization (e.g., scale up at 60% CPU usage).
- Scale based on request rate.
- Set Min & Max Instances.
- Save the configuration and test by generating traffic.

Conclusion

- GCP provides a powerful suite of cloud computing tools.
- Projects help organize and manage cloud resources efficiently.
- Cloud SDK and Cloud Shell simplify management via CLI.
- Compute options like Compute Engine, Kubernetes, and Cloud Functions enable flexible application deployment.
- Autoscaling ensures cost-effective and scalable cloud applications.

DIGITAL LEARNING CONTENT

Parul® University

www.paruluniversity.ac.in