# **GCP Deployment**

**Create a GCP account**

**Step 1: Go to the Google Cloud Website**

* Open your browser and visit **Google Cloud Console**.

**Step 2: Sign in with a Google Account**

* Click **"Sign in"** and use your existing Google account.
* If you don’t have one, click **"Create account"** and follow the instructions.

**Step 3: Start Free Trial (Optional)**

* Google offers a **$300 free credit** for new users.
* Click **“Get started for free”** to access the trial.

**Step 4: Provide Account Information**

* Enter details like:
  + **Country/Region**
  + **Account type** (Individual or Business)
  + **Address and Contact Information**

**Step 5: Add Billing Information**

* Even for the free trial, Google requires a **valid credit/debit card** for verification.
* Google **won't charge you** automatically after the trial period unless you upgrade to a paid account.

**Step 6: Accept Terms and Conditions**

* Read and agree to Google Cloud’s Terms of Service.

**Step 7: Set Up Your First Project**

* Once your account is created, navigate to the **Cloud Console**.
* Click **"Create Project"**, give it a **name**, select a **billing account**, and click **"Create"**.

**Step 8: Enable APIs and Services**

* Enable **Compute Engine, Cloud Storage, Kubernetes, etc.**, depending on your project needs.

**Step 9: Set Up IAM & Security (Optional)**

* Manage access using **Identity and Access Management (IAM)**.
* Set up security policies and permissions.

**Step 10: Start Using GCP**

* Use the **Cloud Shell**, **gcloud CLI**, or the web console to start deploying services.

**Set up a Google Compute Engine instance**

Step 1: Log into Google Cloud Console

* Visit Google Cloud Console.
* Select or create a GCP Project (Projects help organize resources).

Step 2: Enable Compute Engine API

* Navigate to "APIs & Services" > "Library".
* Search for "Compute Engine API" and enable it.

Step 3: Open Compute Engine

* Go to "Compute Engine" > "VM Instances" in the navigation menu.
* Click "Create Instance".

Step 4: Configure Instance Settings

1. Name
   * Enter a unique instance name (e.g., my-vm-instance).
2. Region & Zone
   * Choose a region (e.g., us-central1) and a zone (e.g., us-central1-a).
   * Choose a location close to your users for better performance.
3. Machine Type (CPU & RAM)
   * Choose a predefined machine type (e.g., e2-medium for general use).
   * Click "Customize" if you want specific CPU & RAM values.
4. Boot Disk (OS Selection)
   * Click "Change" under "Boot disk."
   * Select an OS (e.g., Ubuntu, Debian, Windows).
   * Choose disk type (Standard, Balanced, or SSD) and size.
5. Firewall Settings
   * Check "Allow HTTP traffic" and "Allow HTTPS traffic" if needed.

Step 5: Configure Networking (Optional)

* Under "Networking", you can assign a static external IP or configure firewall rules.

Step 6: Create the Instance

* Click "Create" and wait for the VM to launch.

Step 7: Connect to Your VM

1. In the Compute Engine dashboard, find your instance.
2. Click "SSH" to open a terminal in your browser.
3. Alternatively, use gcloud CLI

Step 8: Install Software & Configure

Step 9: Set Up Firewall Rules

* Go to "VPC Network" > "Firewall".
* Click "Create Firewall Rule" to open ports (e.g., TCP:80, 443, 22).

Step 10: Snapshot & Backup

* To create a backup, go to Disks, select your instance’s disk, and click "Create Snapshot".

**Deploy a sample Python script**

**Step 1: Connect to Your VM**

After setting up your VM, connect to it using SSH:

* In **Google Cloud Console**, go to **Compute Engine > VM Instances**.
* Click **“SSH”** next to your instance.

**Step 2: Install Python**

* Most GCP instances come with Python pre-installed. To check

**Step 3: Create the Python Script**

def add\_numbers(a, b):

return a + b

if \_\_name\_\_ == "\_\_main\_\_":

num1 = 5

num2 = 10

result = add\_numbers(num1, num2)

print(f"The sum of {num1} and {num2} is {result}")

**Step 4: Run the Script**

* The sum of 5 and 10 is 15

**Step 5: Automate Execution on Startup**

* @reboot python3 /home/your-user/add\_numbers.py