

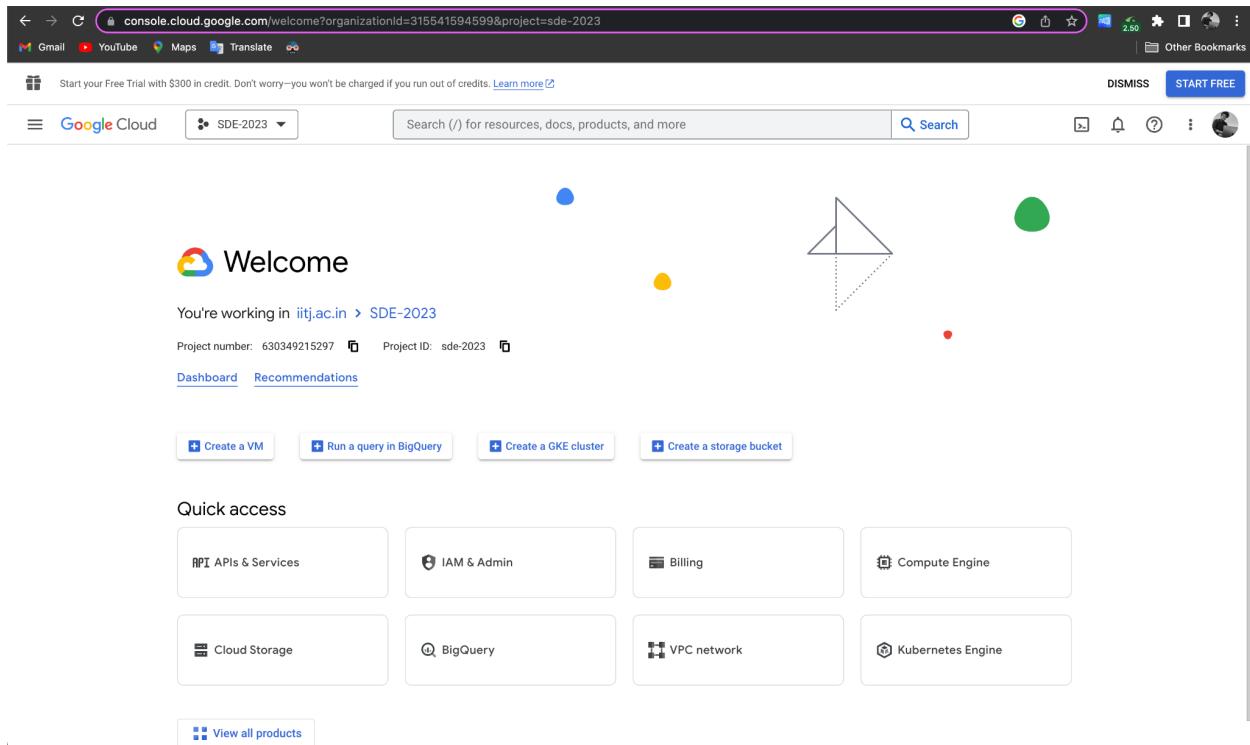
SDE Assignment-2 Report

Bikash Dutta (D22CS051)

Task 1: Setting Up a Virtual Machine (VM) on GCP (30 points)

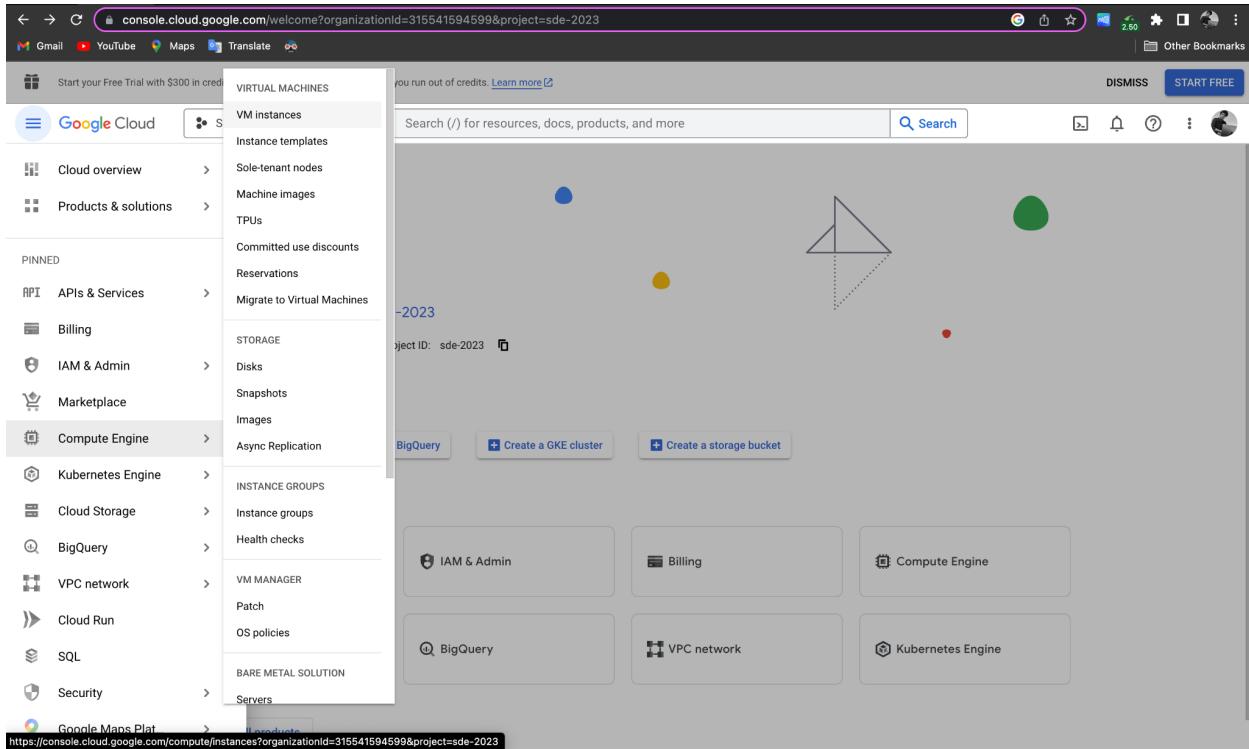
1. Create a GCP Account:

- Please access the official website of Google Cloud Platform at the following URL:
<https://cloud.google.com/>
- Please use the google educational credits that have been provided.
- Please adhere to the provided instructions in order to establish your GCP account, ensuring that you furnish the necessary billing details as requested.
- Since no project has been present at the beginning of the setup, we need to create a project.
- At the end of this, the dashboard will look like below.



2. Set Up a Virtual Machine Instance:

- Please access the Google Cloud Platform (GCP) Console by logging in at the following URL: <https://console.cloud.google.com/>.
- To use the "Compute Engine" feature in the GCP Console, users should locate and choose it from the menu located on the left-hand side of the interface.
- Please navigate to the "VM instances" tab.



- To initiate the creation of a new virtual machine (VM) instance, please select the "Create" button.

3. Choose an Appropriate Operating System:

- On the "Create an instance" page, it is necessary to assign a name to the virtual machine (VM).
- In the section titled "Boot disk," users are advised to click on the "Change" option in order to designate their desired Linux distribution (such as Ubuntu or CentOS) and its corresponding version.
- Adjust additional parameters such as the region and zone according to your personal preferences.
- Go through the configuration once again before creating the VM instance
- To initiate the creation of the virtual machine, please click on the "Create" button.

4. Configure the VM with Custom Machine Specifications:

- On the "Create an instance" page, To modify the CPU, RAM, and other machine specs based on your requirements, please navigate to the "Machine type" section and select the "Change" option.
- If necessary, users may choose to personalize supplementary configurations, including GPUs, boot drive capacity, and network settings.
- To complete the virtual machine creation process, please click on the "Create" button.

Start your Free Trial with \$300 in credit. Don't worry—you won't be charged if you run out of credits. [Learn more](#)

DISMISS **START FREE**

Google Cloud SDE-2023 Search (/) for resources, docs, products, and more **Search**

Create an instance **EQUIVALENT CODE**

To create a VM instance, select one of the options:

- New VM instance** Create a single VM instance from scratch
- New VM instance from template** Create a single VM instance from an existing template
- New VM instance from machine image** Create a single VM instance from an existing machine image
- Marketplace** Deploy a ready-to-go solution onto a VM instance

Name * instance-1

MANAGE TAGS AND LABELS

Region * asia-south2 (Delhi) **Zone *** asia-south2-a

Machine configuration

Try the new H3 machine series, optimized for HPC. **TRY NOW**

General purpose **Compute optimized** **Memory optimized** **GPUs**

Machine types for common workloads, optimized for cost and flexibility

Series	Description	vCPUs	Memory	Platform
C3	Consistently high performance	4 - 176	8 - 1,408 GB	Intel Sapphire Rapids
C3D	PREVIEW	4 - 360	8 - 2,880 GB	AMD Genoa
E2	Low cost, day-to-day computing	0.25 - 32	1 - 128 GB	Based on availability
N2	Balanced price & performance	2 - 128	2 - 864 GB	Intel Cascade and Ice Lake
N2D	Balanced price & performance	2 - 224	2 - 896 GB	AMD EPYC
T2A	Scale-out workloads	1 - 48	4 - 192 GB	Ampere Altra Arm
T2D	Scale-out workloads	1 - 60	4 - 240 GB	AMD EPYC Milan
N1	Balanced price & performance	0.25 - 96	0.6 - 624 GB	Intel Skylake

Machine type
Choose a machine type with preset amounts of vCPUs and memory that suit most workloads. Or, you can create a custom machine for your workload's particular needs. [Learn more](#)

PRESET **CUSTOM**

e2-small (2 vCPU, 1 core, 2 GB memory)

vCPU	Memory
0.5-2 vCPU (1 shared core)	2 GB

ADVANCED CONFIGURATIONS

Availability policies
VM provisioning model Standard

Choose "Spot" to get a discounted, preemptible VM. Otherwise, stick to "Standard". [Learn more](#)

VM PROVISIONING MODEL ADVANCED SETTINGS

Display device
Enable to use screen capturing and recording tools.

Enable display device

Confidential VM service [?](#)
Confidential Computing is disabled on this VM instance

ENABLE

Container [?](#)

VM provisioning model
Standard

Choose "Spot" to get a discounted, preemptible VM. Otherwise, stick to "Standard". [Learn more](#)

VM PROVISIONING MODEL ADVANCED SETTINGS

Display device
Enable to use screen capturing and recording tools.
 Enable display device

Confidential VM service
Confidential Computing is disabled on this VM instance
[ENABLE](#)

Container
Deploy a container image to this VM instance
[DEPLOY CONTAINER](#)

Boot disk

Name	instance-1
Type	New balanced persistent disk
Size	10 GB
License type	Free
Image	Debian GNU/Linux 11 (bullseye)

[CHANGE](#)

Identity and API access

Service accounts
Service account
Compute Engine default service account

Requires the Service Account User role (roles/iam.serviceAccountUser) to be set for users who want to access VMs with this service account. [Learn more](#)

Access scopes

- Allow default access
- Allow full access to all Cloud APIs
- Set access for each API

Firewall

Add tags and firewall rules to allow specific network traffic from the Internet

- Allow HTTP traffic
- Allow HTTPS traffic
- Allow Load Balancer Health Checks

Can be added later also.

Observability - Ops Agent

Monitor your system through collection of logs and key metrics.

- Install Ops Agent for Monitoring and Logging

Advanced options

Networking, disks, security, management, sole-tenancy

[CREATE](#) [CANCEL](#) [EQUIVALENT CODE](#)

- Once created VM can be seen as below.

Start your Free Trial with \$300 in credit. Don't worry—you won't be charged if you run out of credits. [Learn more](#)

[DISMISS](#) [START FREE](#)

Google Cloud SDE-2023 Search (/) for resources, docs, products, and more [Search](#)

Compute Engine

Virtual machines

- VM instances
- Instance templates
- Sole-tenant nodes
- Machine images
- TPUs
- Committed use discounts
- Reservations
- Migrate to Virtual Machines

Storage

- Disks
- Snapshots
- Images
- Async Replication

VM instances

INSTANCES **OBSERVABILITY** **INSTANCE SCHEDULES**

Your project's VMs use global DNS names by default. To reduce the risk of cross-regional outages, we recommend you use zonal DNS instead. [Learn more](#)

[USE ZONAL DNS](#) [DISMISS](#)

VM instances

Filter Enter property name or value

Status	Name	Zone	Recommendations	In use by	Internal IP	External IP	Connect
<input type="checkbox"/>	gke-sde-a2-75090f07-9x0t	asia-south2-a		gke-sde-a2-default-pool-75090f07-grp_a...	10.190.0.9 (nic0)	34.131.142.100 (nic0)	SSH
<input type="checkbox"/>	gke-sde-a2-default-pool-8bb6b253-4v1c	asia-south2-b		gke-sde-a2-default-pool-8bb6b253-grp_af...	10.190.0.10 (nic0)	34.131.31.79 (nic0)	SSH
<input type="checkbox"/>	gke-sde-a2-default-pool-e048aa3f-1zw5	asia-south2-c		gke-sde-a2-default-pool-e048aa3f-grp_af...	10.190.0.8 (nic0)	34.131.244.73 (nic0)	SSH
<input type="checkbox"/>	sde-a2	asia-south2-a			10.190.0.2 (nic0)	34.131.119.19 (nic0)	SSH

Related actions

[^ HIDE](#)

Start your Free Trial with \$300 in credit. Don't worry—you won't be charged if you run out of credits. [Learn more](#)

[DISMISS](#) [START FREE](#)

Google Cloud SDE-2023 Search (/) for resources, docs, products, and more [Search](#)

Compute Engine sde-a2 [EDIT](#) [RESET](#) [CREATE MACHINE IMAGE](#) [CREATE SIMILAR](#) [START / RESUME](#) [STOP](#) [OPERATIONS](#) [Help](#)

Servers Networks Volumes NFS Shares Procurements [NEW](#) Maintenance Events [NEW](#)

Logs [Logs](#) Serial port 1 (console) [SHOW MORE](#)

Basic information

Name	sde-a2
Instance Id	76193658554108297
Description	None
Type	Instance
Status	Running
Creation time	Sep 25, 2023, 1:49:52 AM UTC+05:30
Zone	asia-south2-a
Instance template	None
In use by	None
Reservations	Automatically choose
Labels	None
Tags	—
Deletion protection	Disabled
Confidential VM service	Disabled
Preserved state size	0 GB

Machine configuration

Machine type	c2-small
CPU platform	Intel Broadwell
Minimum CPU platform	None
Architecture	x86_64
vCPUs to core ratio	—
Custom visible cores	—
Display device	Enabled
Enable to use screen capturing and recording tools.	
Gpus	None
Resource policies	

Networking

Public DNS PTR Record	None
Total egress bandwidth tier	—
NIC type	—

[VIEW IN NETWORK TOPOLOGY](#)

Firewalls

HTTP traffic	On
HTTPS traffic	On
Allow Load Balancer Health checks	On

Network tags

[http-server](#) [https-server](#) [lb-health-check](#)

Network interfaces

Name	Network	Subnetwork	Primary Internal IP address	Alias IP ranges	IP stack type	External IP address	Network
sde-a2	default	default	10.190.3.2		IPv4	34.131.119.19 (Ephemeral)	Prem.

Storage

Boot disk

Name	Image	Interface type	Size (GB)	Device name	Type	Architecture	Encryption	Mode	Wb
sde-a2	ubuntu-22.04-jammy-20230919	SCSI	10	sde-a2	SSD persistent disk	x86_64	Google-managed	Boot, read/write	Del

Local disks

None

Additional disks

None

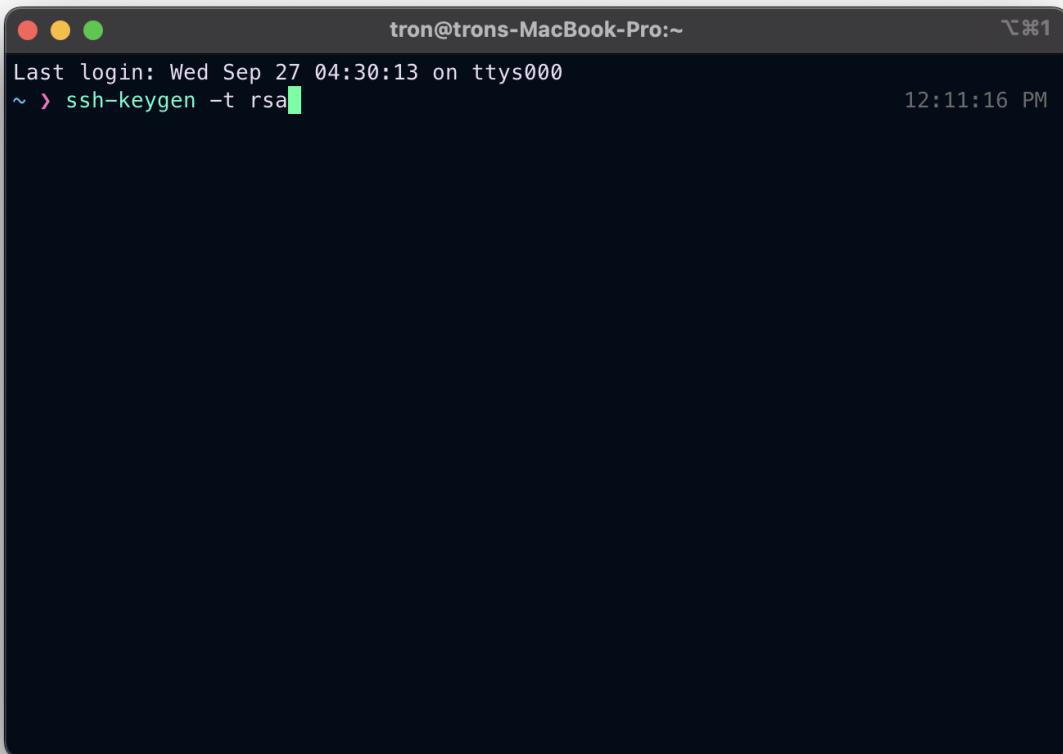
Security and access

5. Access the VM Using SSH from your local machine:

- First we need to generate a public-private key.

```
Unset
```

```
ssh-keygen -t rsa
```



A screenshot of a macOS terminal window titled "tron@trons-MacBook-Pro:~". The window shows the command "ssh-keygen -t rsa" being typed at the prompt. The terminal has a dark background with white text. The status bar at the bottom right shows the date and time as "12:11:16 PM".

- Then, add the content of your public key to your instance.

```
Unset
```

```
cat .ssh/id_rsa.pub #your private key may be at different locataion
```

```
tron@trons-MacBook-Pro:~ Last login: Wed Sep 27 04:30:13 on ttys000
~ > ssh-keygen -t rsa
~ > cat .ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAABgQDRfj9yTH0V7c6m2vN06r0bLlT6l08RXSj6B6xGRI+4
khgHw8Y2GyXD0fIZ7+te8wNC+o3RbLfZVHxJSqGPk3KnM9WBjIWw4oJYqsoxEI3l+D33c+VQTl2JEvJ3
LqQ+gSvDnAJfim89fQ/i+IElDY3+r+qEVuBHo9fKhq0awuJf3u5k0dBjdLTc+Ds9e4cjjkuVL3TCzu7f
qInCkj6n7JTn6cJ3XMbg34t/S6o+NIBD8839U8yjV53BPRVIuSUkDh2g4ogweebK2sGI2tRcSlYsloLU
NjvZWjoPB3Z0pQvyardFV0CjynvvH+jAlMTZZjohuy5taGT1aE5xENi7unMy4N+qskBvYdmv6C9YQ5qe
0NB5e9nlDExR6FSwUiG4P5gmVGL1dA3sxTtzuCTf3FNSCPuuY6WiVPSN3T0G34/TVQXFm3EUWes5Ab9t
0wMNkh/UevF6kdC6kXVh5FNap3yrYJ+zgWx36d21f2tKGT+2VQYMFelrLvFCTvvQxpg0gA0= tron@tr
ons-MacBook-Pro.local
~ > █
```

- Click on the Edit option, then navigate to the security and access configurations.
- Click on the add key in the SSH section, then paste your generated public key.

Security and access

Shielded VM ?

To edit Shielded VM features you need to stop the instance first.

Turn on all settings for the most secure configuration.

Turn on Secure Boot ?

Turn on vTPM ?

Turn on Integrity Monitoring ?

SSH Keys

These keys allow access only to this instance, unlike project-wide SSH keys. [Learn more](#)



Block project-wide SSH keys

When checked, project-wide SSH keys cannot access this instance. [Learn more](#)

SSH key 1 *

ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQgQDRfj9yTHOV7c6m2vN06rO

Enter public SSH key

SSH key 2 *

ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmldHAyNTYAAAAbmldHA

Enter public SSH key

SSH key 3 *

ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCHJF4k4Q2YAcskNQ2+Kgr

Enter public SSH key

+ ADD ITEM

Identity and API access ?

Service accounts ?

You must stop the VM instance to edit its service account

Service account

Compute Engine default service account



Requires the Service Account User role (roles/iam.serviceAccountUser) to be set for users who want to access VMs with this service account. [Learn more](#)

Access scopes ?

You must stop the VM instance to edit its API access scopes

Allow default access

Allow full access to all Cloud APIs

Set access for each API

Management

- One Done Click on the save at the end of the page & verify.

[sde-a2](#) [EDIT](#) [RESET](#) [CREATE MACHINE IMAGE](#) [CREATE SIMILAR](#) [START / RESUME](#) [STOP](#) [⋮](#) [0](#)

[DETAILS](#) [OBSERVABILITY](#) [OS INFO](#) [SCREENSHOT](#)

SSH Keys

SSH keys

Username	Key
tron	ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQgQDRfj9yTHOV7c6m2vN06rObLLT6l08RXSj6B6xGRI+4khgHw8Y2GyXD0fIZ7+te8... ▾
d22cs051	ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTbmlzdHAyNTYAAAAlbmlzdHAyNTYAAABBB0y2y0WHXiblPezEgQX/VR473G2yw... ▾
d22cs051	ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCHJF4k4Q2YAcskNQ2+Kgr6MYBYEyF16+fFQLbkcZWA7bH0ScMfZePe8iXLZz... ▾

Block project-wide SSH keys [Off](#)

API and identity management

Service account [630349215297-compute@developer.gserviceaccount.com](#)

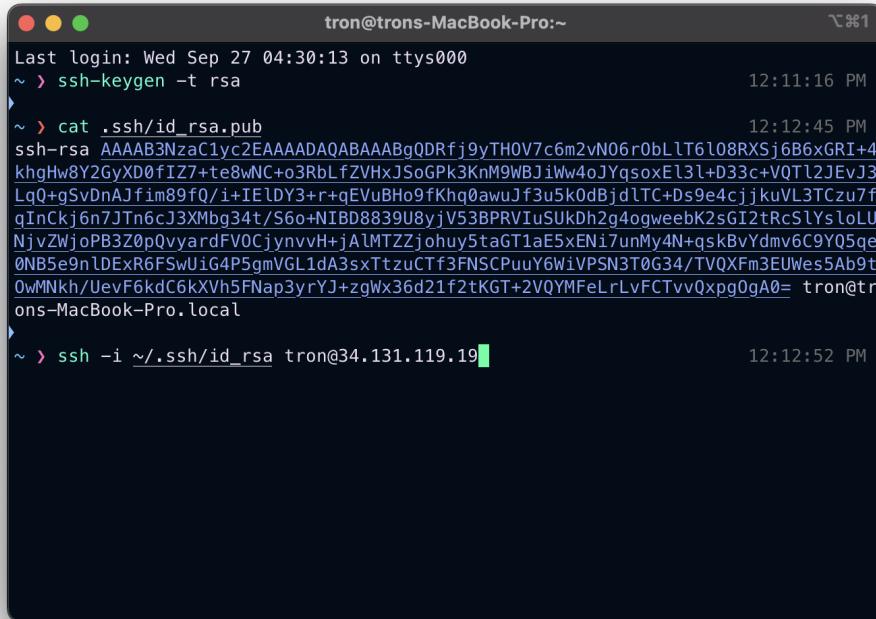
Cloud API access scopes [Allow default access](#)

[▼ SHOW DETAILS](#)

- Now, for connecting to the VM, use the below command.
- NOTE: your private and public keys will differ from mine (~/.ssh/id_rsa) & (tron@34.131.119.19).**

Unset

```
ssh -i ~/.ssh/id_rsa tron@34.131.119.19 #your username and ip will be differnt
```



```
tron@trons-MacBook-Pro:~ Last login: Wed Sep 27 04:30:13 on ttys000
~ > ssh-keygen -t rsa
~ > cat .ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQgQDRfj9yTHOV7c6m2vN06rObLLT6l08RXSj6B6xGRI+4khgHw8Y2GyXD0fIZ7+te8wNC+o3RbLfZVHxJSogPk3KnM9WBjIw4oJYqsoxE13l+D33c+VQTl2JEvJ3LqQ+gSvDnAJfim89f0/i+IElDY3+r+qEVuBHo9fKhq0awuJf3u5k0dBjdLTC+Ds9e4cjjkuVL3TCzu7fqInCkj6n7JJTh6cJ3XMbq34t/S6o+NIBD8839U8yjV53BPRVIuSUkDh2g4ogweebK2sGI2tRc5lYsLoLU NJjvZWjoPB3Z0pQvyardFVOcjynvvH+jAlMTZZjohuy5taGT1aE5xENi7unMy4N+qskBvYdmv6C9YQ5qe0NB5e9nlDExR6FSwUjG4P5gmVGL1dA3sxTzucTf3FNSCPuuY6wiVPSN3T0G34/TVQXFm3EUWes5Ab9t0wMNkh/UevF6kdC6kXVh5FNap3yryJ+zsWx36d21f2tKGT+2VQYMFelrLvfCTvvQxpg0gA0= tron@trons-MacBook-Pro.local
~ > ssh -i ~/.ssh/id_rsa tron@34.131.119.19
```

```
tron@sde-a2: ~
```

```
~ > ssh -i ~/.ssh/id_rsa tron@34.131.119.19          12:12:52 PM
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-1014-gcp x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

System information as of Wed Sep 27 06:54:20 UTC 2023

System load:  0.0          Users logged in:          0
Usage of /:   68.1% of 9.51GB  IPv4 address for br-09f0debb0480: 172.19.0.1
Memory usage: 39%          IPv4 address for docker0:    172.17.0.1
Swap usage:   0%          IPv4 address for ens4:      10.190.0.2
Processes:    113

* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Tue Sep 26 23:00:17 2023 from 220.158.144.58
tron@sde-a2:~$
```

6. Install Nginx Web Server:

- To update the package lists and install Nginx in the SSH terminal, execute the provided commands, taking into consideration that the specific commands may differ depending on the Linux distribution being used.
- To update the system and install the Nginx web server on Ubuntu/Debian, the following commands can be executed:

Unset

```
sudo apt update; sudo apt install nginx -y
```

```
tron@sde-a2: ~
Get:11 http://asia-south2-a.gce.clouds.archive.ubuntu.com/ubuntu jammy-updates/unive
rse amd64 Packages [986 kB]
Get:12 http://asia-south2-a.gce.clouds.archive.ubuntu.com/ubuntu jammy-updates/unive
rse Translation-en [215 kB]
Get:13 http://asia-south2-a.gce.clouds.archive.ubuntu.com/ubuntu jammy-updates/unive
rse amd64 c-n-f Metadata [21.9 kB]
Get:14 http://asia-south2-a.gce.clouds.archive.ubuntu.com/ubuntu jammy-updates/multi
verse amd64 Packages [41.6 kB]
Fetched 3758 kB in 4s (966 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
W: https://packages.cloud.google.com/apt/dists/google-cloud-ops-agent-jammy-all/InRe
lease: Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the D
EPRECATION section in apt-key(8) for details.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nginx is already the newest version (1.18.0-6ubuntu14.4).
The following package was automatically installed and is no longer required:
  libnuma1
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
tron@sde-a2:~$
```

- Verify your installation by the following command & typing your external IP in browser.

```
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nginx is already the newest version (1.18.0-6ubuntu14.4).
The following package was automatically installed and is no longer required:
  libnuma1
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
tron@sde-a2:~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
  Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: ena>
  Active: active (running) since Sun 2023-09-24 20:48:47 UTC; 2 days ago
    Docs: man:nginx(8)
 Main PID: 3561 (nginx)
    Tasks: 3 (limit: 2337)
   Memory: 3.9M
      CPU: 336ms
     CGroup: /system.slice/nginx.service
             └─3561 "nginx: master process /usr/sbin/nginx -g daemon on; master_pro>
               ├─3562 "nginx: worker process" " " " " " " " " " " " " " " " " " " " " " " >
               ├─3563 "nginx: worker process" " " " " " " " " " " " " " " " " " " " " " " >
               └─3564 "nginx: worker process" " " " " " " " " " " " " " " " " " " " " " " >

Sep 24 20:48:47 sde-a2 systemd[1]: Starting A high performance web server and a rev>
Sep 24 20:48:47 sde-a2 systemd[1]: Started A high performance web server and a rev>
lines 1-15/15 (END)
```

7. Display a Custom Webpage:

- Once Nginx has been installed, it is possible to generate a personalized webpage in order to showcase the successful configuration.
- Utilize a text editor to generate an HTML file for the purpose of constructing a personalized webpage, such as naming it index.html. The choice of editor, whether it be nano or vim, is contingent upon personal preference.

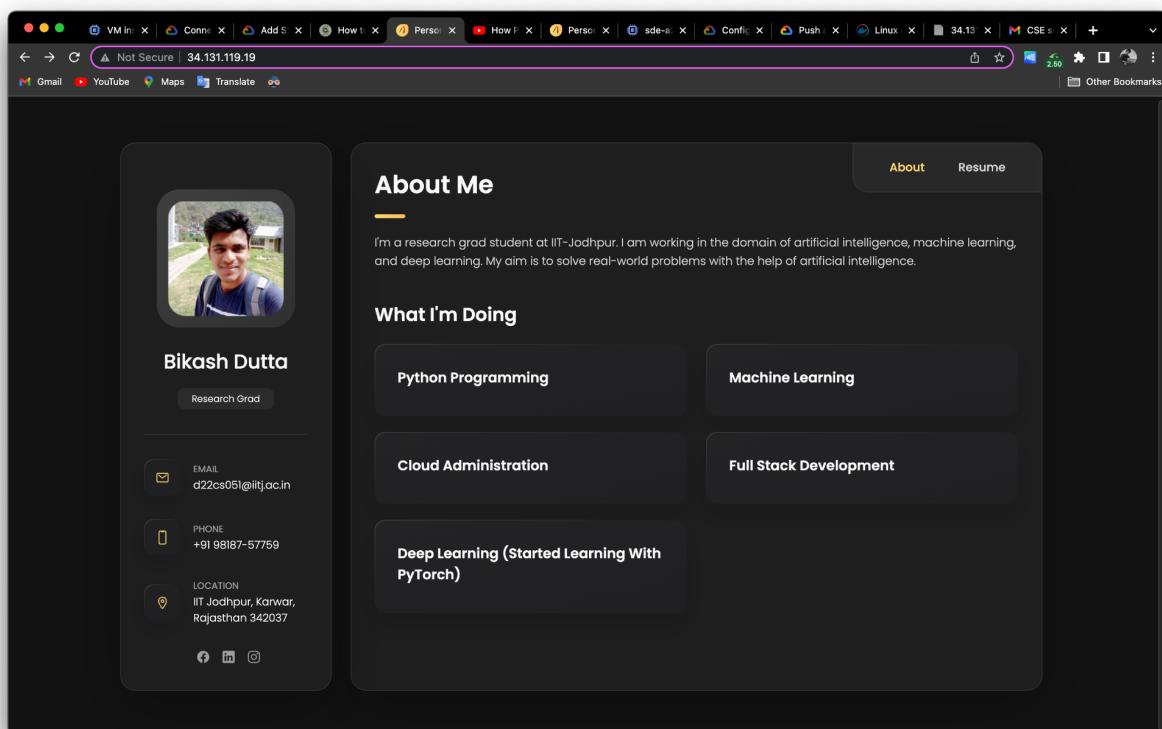
```
Unset
```

```
sudo nano /var/www/html/index.html
```

- Copy code
- Add your HTML content to this file, save it, and exit the editor.
- Restart the Nginx to apply changes.

```
Unset
```

```
sudo systemctl restart nginx
```



Task 2: Docker Containerization (40 points)

1. Install Docker on Your Local Machine (or VM):

- follow the installation instructions for your operating system:
<https://docs.docker.com/get-docker/>
- Our VM is running Ubuntu, hence following appropriately.

```
Unset
```

```
# removing old installations if exists
for pkg in docker.io docker-doc docker-compose podman-docker containerd runc;
do sudo apt-get remove $pkg; done
```

```
Unset
```

```
# Add Docker's official GPG key:
sudo apt-get update
sudo apt-get install ca-certificates curl gnupg
sudo install -m 0755 -d /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o
/etc/apt/keyrings/docker.gpg
sudo chmod a+r /etc/apt/keyrings/docker.gpg

# Add the repository to Apt sources:
echo \
"deb [arch=$(dpkg --print-architecture)"
signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu
\
"$(./etc/os-release && echo "$VERSION_CODENAME")" stable" | \
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update
```

```
Unset
```

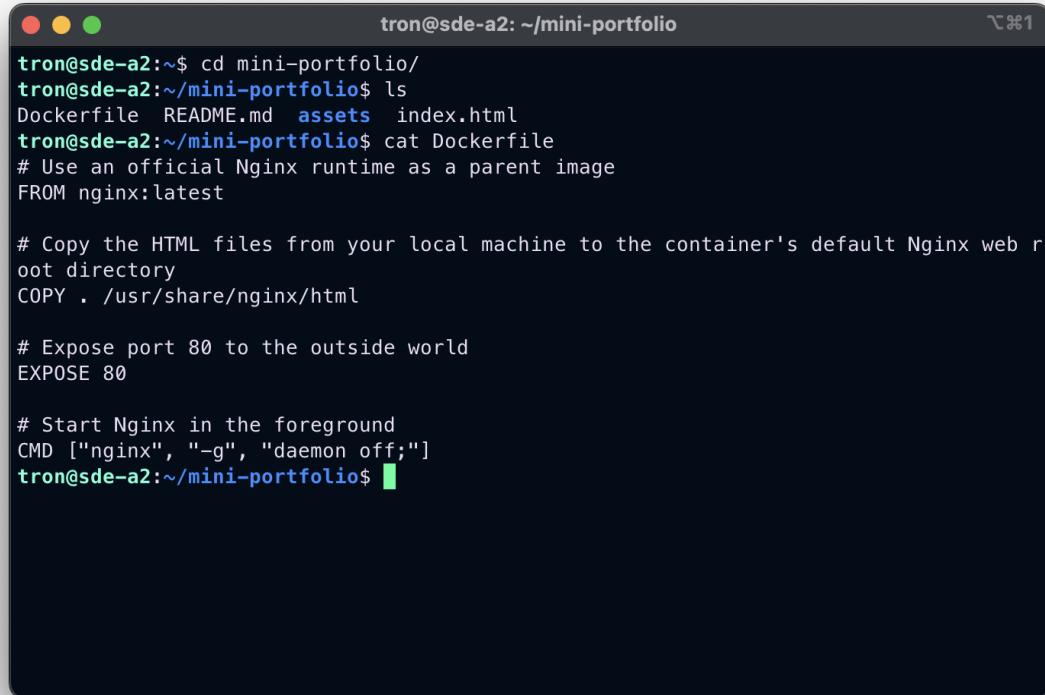
```
# installing engine
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin
docker-compose-plugin docker-compose -y;
```

- Note: Add the user to the docker group so that docker can be used without sudo privileges.

```
Unset  
sudo groupadd docker  
sudo usermod -aG docker $USER  
newgrp docker
```

2. Build a Docker Image:

- Create a directory for your application and navigate to it using `cd` command.
- Write a Dockerfile to define your application's image. For example, create a file named `Dockerfile` with your application configuration (our app config can be seen below).



The screenshot shows a terminal window titled "tron@sde-a2: ~/mini-portfolio". The terminal displays the following content:

```
tron@sde-a2:~$ cd mini-portfolio/  
tron@sde-a2:~/mini-portfolio$ ls  
Dockerfile README.md assets index.html  
tron@sde-a2:~/mini-portfolio$ cat Dockerfile  
# Use an official Nginx runtime as a parent image  
FROM nginx:latest  
  
# Copy the HTML files from your local machine to the container's default Nginx web root directory  
COPY . /usr/share/nginx/html  
  
# Expose port 80 to the outside world  
EXPOSE 80  
  
# Start Nginx in the foreground  
CMD ["nginx", "-g", "daemon off;"]  
tron@sde-a2:~/mini-portfolio$ █
```

- Build the Docker image from the directory containing your Dockerfile:

```
Unset  
docker build -t sde-a2-t2_app .
```

```
tron@sde-a2:~/mini-portfolio$ docker build -t sde-a2-t2_app .
[+] Building 0.2s (7/7) FINISHED                                            docker:default
=> [internal] load .dockerignore                                         0.1s
=> => transferring context: 2B                                           0.0s
=> [internal] load build definition from Dockerfile                   0.1s
=> => transferring dockerfile: 352B                                      0.0s
=> [internal] load metadata for docker.io/library/nginx:latest        0.0s
=> [internal] load build context                                         0.0s
=> => transferring context: 322.33kB                                     0.0s
=> [1/2] FROM docker.io/library/nginx:latest                           0.0s
=> CACHED [2/2] COPY . /usr/share/nginx/html                            0.0s
=> exporting to image                                                 0.0s
=> => exporting layers                                                 0.0s
=> => writing image sha256:39308ccadc843599187b667a4f7c4ae70555155ff4eddc8b7 0.0s
=> => naming to docker.io/library/sde-a2-t2_app                         0.0s
tron@sde-a2:~/mini-portfolio$
```

Unset

```
docker images #to see all the available docker images
```

Unset

```
docker run -d -p 8080:80 sde-a2-t2_app # mapping to 8080 since port 80 is already
being used by systems nginx
```

The screenshot shows a resume website for Bikash Dutta. On the left, there's a sidebar with his profile picture, name (Bikash Dutta), title (Research Grad), and contact information: Email (d22cs05@iitj.ac.in), Phone (+91 98187-57759), and Location (IIT Jodhpur, Karwar, Rajasthan 342037). Below this are social media links for Facebook, LinkedIn, and GitHub. The main content area has a dark background with white text. It features a section titled "About Me" with a short bio: "I'm a research grad student at IIT-Jodhpur. I am working in the domain of artificial intelligence, machine learning, and deep learning. My aim is to solve real-world problems with the help of artificial intelligence." Below this is a section titled "What I'm Doing" with four categories: Python Programming, Machine Learning, Cloud Administration, and Full Stack Development. A callout box highlights "Deep Learning (Started Learning With PyTorch)".

3. Push the Docker image to Google Artifacts Registry:

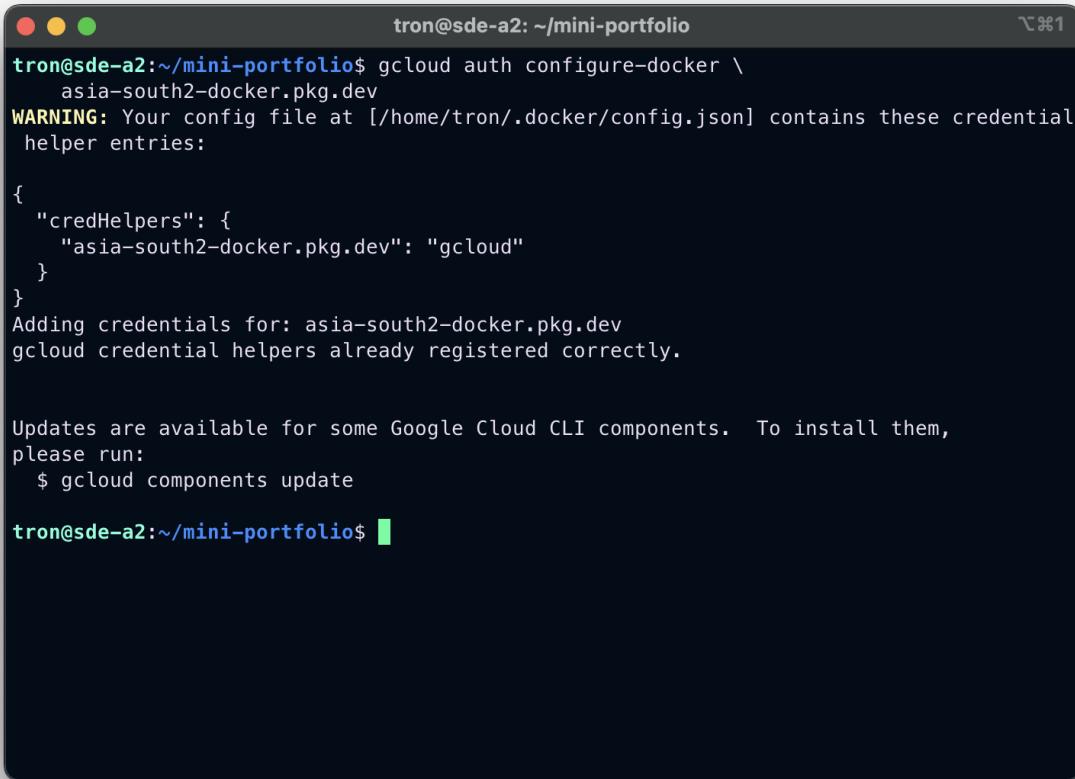
- Go to the artifact registry in your GCP console & create one:
<https://console.cloud.google.com/artifacts>

The screenshot shows the Google Cloud Artifacts Registry interface. At the top, it says "Start your Free Trial with \$300 in credit. Don't worry—you won't be charged if you run out of credits. [Learn more](#)". There are tabs for "DISMISS" and "START FREE". Below this is a search bar with "ARTI" and a dropdown for "SDE-2023". To the right is a "LEARN TUTORIAL" button and a sidebar titled "Recommended for you" with links like "Overview of Artifact Registry", "Quickstart for npm", "Configuring access control", "Pushing and pulling images", "Creating repositories", and "Setting up authentication for Docker". The main content area shows a repository named "sde-a2-docker-registry" with details: Format (Docker), Type (Standard), Location (asia-south2 (Delhi)), Description (sde a2), Labels, and Version policy. A modal window titled "Turn on vulnerability scanning" is open, stating "Your registry is not being monitored for known vulnerabilities. GCP offers automatic vulnerability monitoring of all images pushed or pulled within the last 30 days at a cost of \$0.26 per image." It has "TURN ON" and "LEARN MORE" buttons.

- Open a terminal on your local machine or VM.
- Run the following command to authenticate Docker with your Google Cloud credentials:

Unset

```
gcloud auth configure-docker asia-south2-docker.pkg.dev
```



The screenshot shows a terminal window with a dark background and light-colored text. The title bar says "tron@sde-a2: ~/mini-portfolio". The command entered was "gcloud auth configure-docker \ asia-south2-docker.pkg.dev". The output includes a warning about a config file containing credential helper entries, a JSON object showing the entry, and a message stating that gcloud credential helpers are already registered correctly. It also mentions updates available for Google Cloud CLI components. The prompt at the end is "tron@sde-a2:~/mini-portfolio\$".

```
tron@sde-a2: ~/mini-portfolio
tron@sde-a2:~/mini-portfolio$ gcloud auth configure-docker \
    asia-south2-docker.pkg.dev
WARNING: Your config file at [/home/tron/.docker/config.json] contains these credential
helper entries:

{
  "credHelpers": {
    "asia-south2-docker.pkg.dev": "gcloud"
  }
}
Adding credentials for: asia-south2-docker.pkg.dev
gcloud credential helpers already registered correctly.

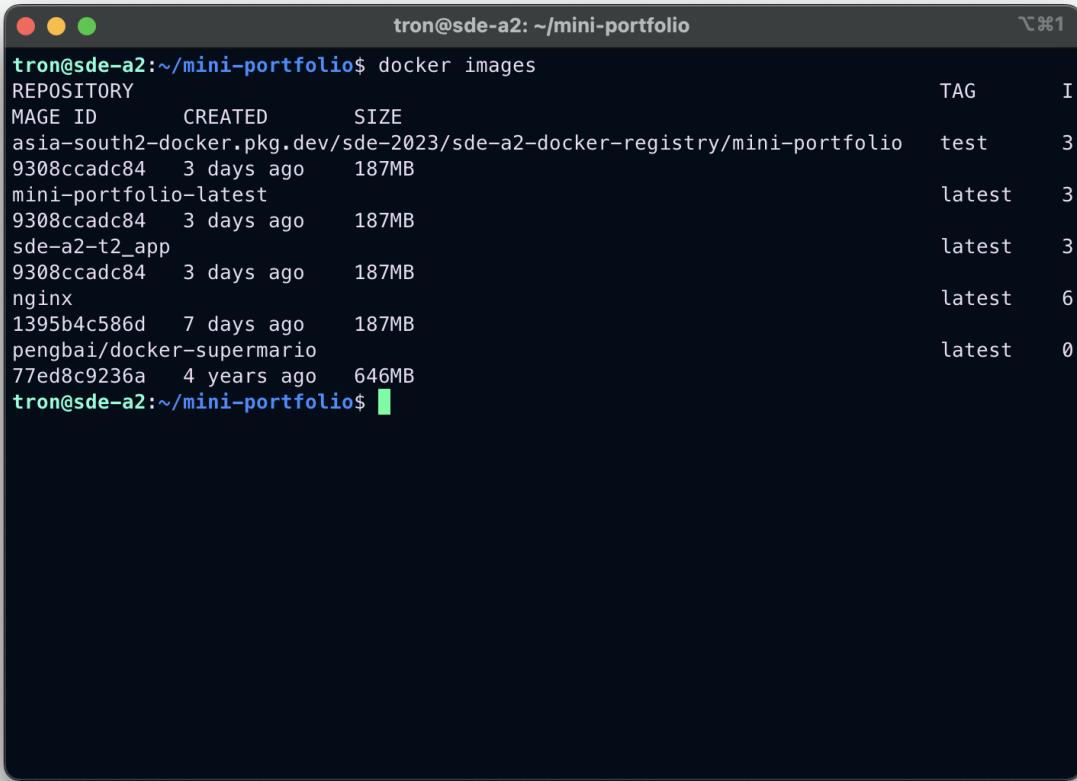
Updates are available for some Google Cloud CLI components. To install them,
please run:
  $ gcloud components update

tron@sde-a2:~/mini-portfolio$
```

- we've already built your Docker image (as mentioned in previous steps). We'll need to tag it with the Google Artifact Registry URL.

Unset

```
# docker tag my-app-image gcr.io/your-project-id/your-region/my-app-image:v1
docker tag mini-portfolio-latest
asia-south2-docker.pkg.dev/sde-2023/sde-a2-docker-registry/mini-portfolio:test
```

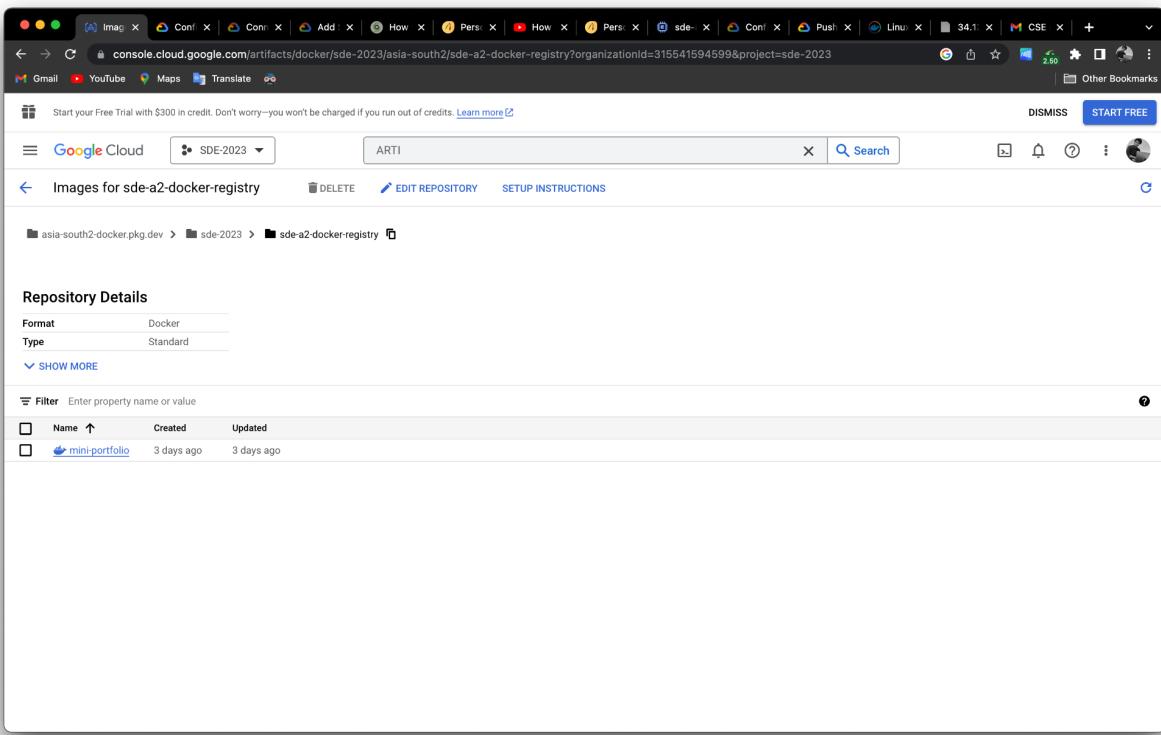
A screenshot of a terminal window titled "tron@sde-a2: ~/mini-portfolio". The window shows the output of the command "docker images". The table lists several Docker images with their repository names, creation dates, sizes, tags, and IDs.

REPOSITORY	CREATED	SIZE	TAG	IMAGE ID
asia-south2-docker.pkg.dev/sde-2023/sde-a2-docker-registry/mini-portfolio	3 days ago	187MB	test	9308ccadc84
mini-portfolio-latest	3 days ago	187MB	latest	9308ccadc84
sde-a2-t2_app	3 days ago	187MB	latest	9308ccadc84
nginx	7 days ago	187MB	latest	1395b4c586d
pengbai/docker-supermario	4 years ago	646MB	latest	77ed8c9236a

- Replace your-project-id with your actual Google Cloud project ID, and your-region with the appropriate region (e.g.,asia-south2).
- Pushing to the registry

Unset

```
#docker push gcr.io/your-project-id/your-region/my-app-image:v1
docker push
asia-south2-docker.pkg.dev/sde-2023/sde-a2-docker-registry/mini-portfolio:test
```

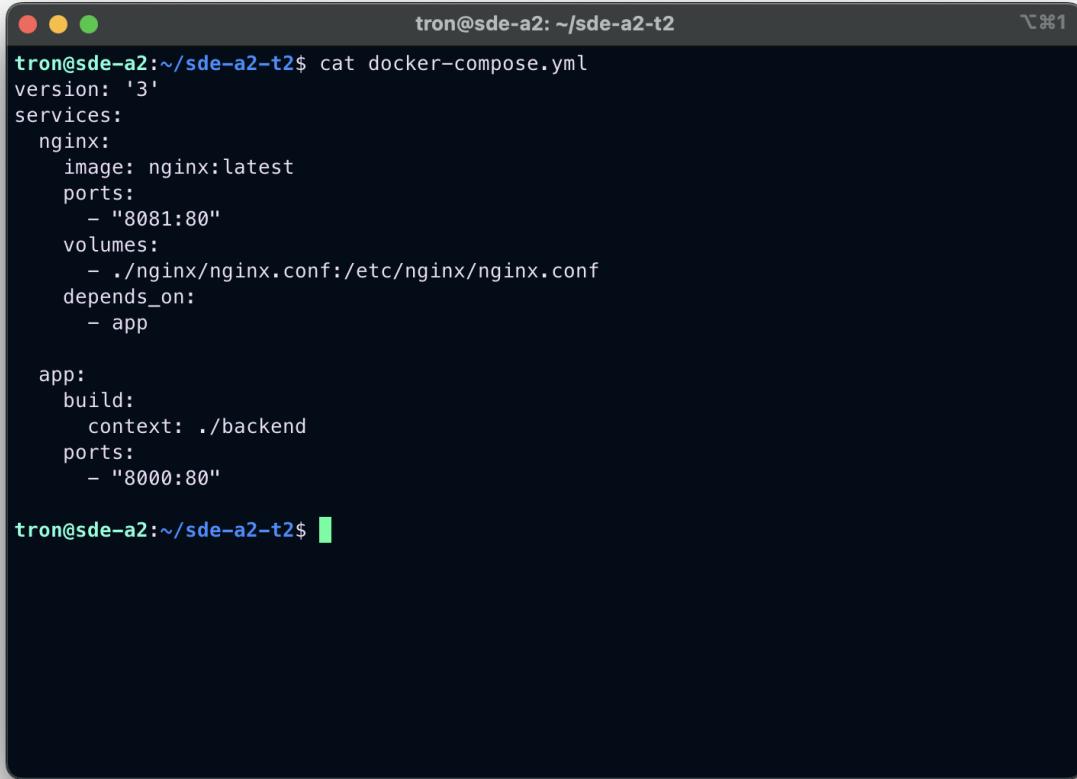


4. Create a Docker Compose File:

- Create a directory for your project and navigate to it in your terminal
- Inside the project directory, create a file named `docker-compose.yml` with the following content

```
Python
version: '3'
services:
  nginx:
    image: nginx:latest
    ports:
      - "8081:80"
    volumes:
      - ./nginx/nginx.conf:/etc/nginx/nginx.conf
    depends_on:
      - app

  app:
    build:
      context: ./backend
    ports:
      - "8000:80"
```



A screenshot of a terminal window titled "tron@sde-a2: ~/sde-a2-t2". The window shows the command "cat docker-compose.yml" being run, displaying a Docker Compose configuration file. The file defines two services: "nginx" and "app". The "nginx" service uses the "nginx:latest" image, listens on port 8081, and proxies requests to the "app" service on port 80. The "app" service is built from the current directory ("./backend") and exposes port 8000. A green cursor is visible at the end of the command line.

```
tron@sde-a2:~/sde-a2-t2$ cat docker-compose.yml
version: '3'
services:
  nginx:
    image: nginx:latest
    ports:
      - "8081:80"
    volumes:
      - ./nginx/nginx.conf:/etc/nginx/nginx.conf
    depends_on:
      - app

  app:
    build:
      context: ./backend
    ports:
      - "8000:80"

tron@sde-a2:~/sde-a2-t2$
```

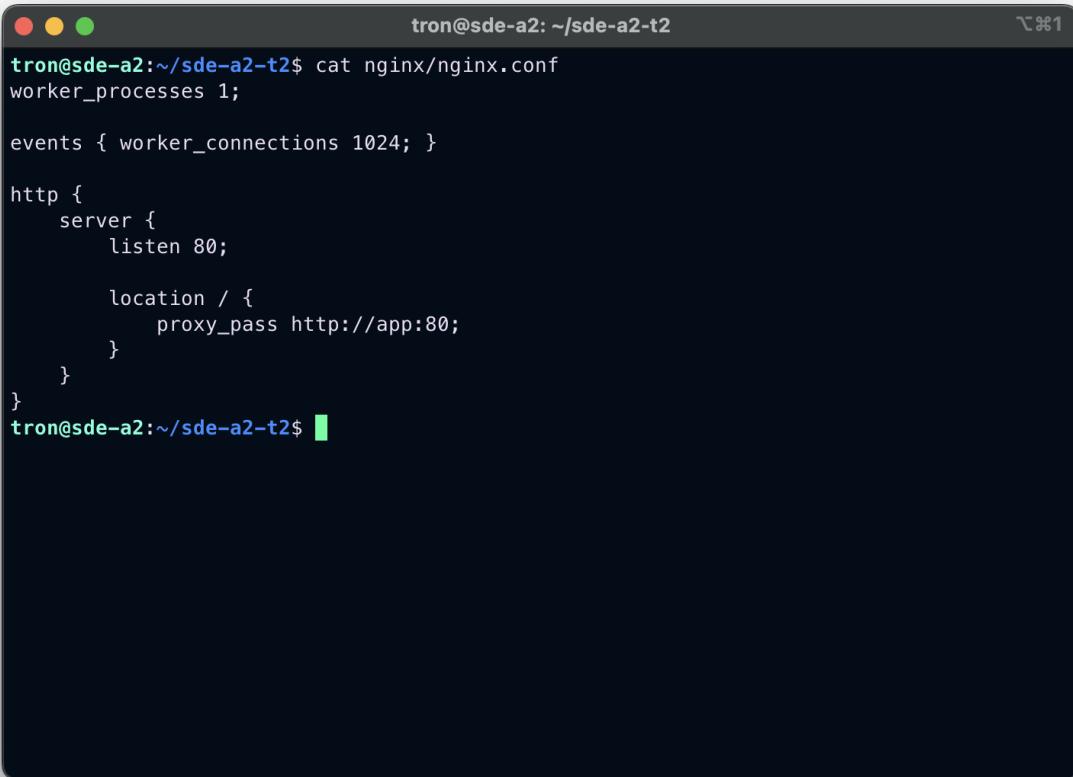
- Create a nginx directory in your project directory and create an nginx.conf file inside it for custom Nginx configuration. Here's a basic example `nginx.conf`.

```
Unset
worker_processes 1;

events { worker_connections 1024; }

http {
    server {
        listen 80;

        location / {
            proxy_pass http://app:80;
        }
    }
}
```



A screenshot of a terminal window titled "tron@sde-a2: ~/sde-a2-t2". The window displays the contents of the file "nginx/nginx.conf". The configuration includes a "worker_processes" directive set to 1, an "events" block with a "worker_connections" value of 1024, and an "http" block containing a "server" block. The "server" block has "listen" set to 80 and a "location" block that uses "proxy_pass" to forward requests to "http://app:80". The terminal prompt "tron@sde-a2: ~/sde-a2-t2\$ " is visible at the bottom.

```
tron@sde-a2:~/sde-a2-t2$ cat nginx/nginx.conf
worker_processes 1;

events { worker_connections 1024; }

http {
    server {
        listen 80;

        location / {
            proxy_pass http://app:80;
        }
    }
}
tron@sde-a2:~/sde-a2-t2$
```

- Create a directory named `backend` in your project directory. This is where we'll create the FastAPI/Flask backend application using Python.

Python

```
# main.py file
from fastapi import FastAPI
import os

worker_id = os.getpid()
app = FastAPI()

@app.get("/")
def root():
    return {"message": "Hello, FastAPI", "worker_number": worker_id}
```

```
Unset  
#backend/requirements.txt  
fastapi  
uvicorn
```

- Inside the backend directory, create a Dockerfile for the FastAPI/Flask application.

```
Unset  
# Use an official Python runtime as a parent image  
FROM python:3.8-slim  
  
# Set the working directory to /app  
WORKDIR /app  
  
# Copy the current directory contents into the container at /app  
COPY . /app  
  
# Install any needed packages specified in requirements.txt  
RUN pip install --no-cache-dir -r requirements.txt  
  
# Make port 80 available to the world outside this container  
EXPOSE 80  
  
# Run app.py when the container launches  
CMD [ "uvicorn", "main:app", "--host", "0.0.0.0", "--port", "80" ]
```

- At the end project will follow the below structure.

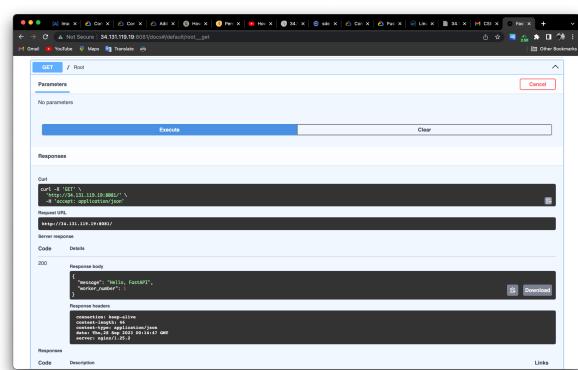
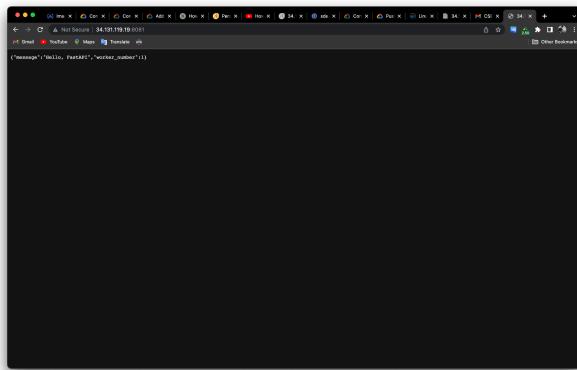
```
Unset  
. .  
|__ backend  
| |__ Dockerfile  
| |__ main.py  
| |__ requirements.txt  
|__ docker-compose.yml  
|__ nginx  
    |__ nginx.conf
```

- Use docker compose to run the application.

Unset

```
docker-compose up -d # -d = run in background
```

```
tron@sde-a2:~/sde-a2-t2$ docker-compose up
Starting sde-a2-t2_app_1 ... done
Starting sde-a2-t2_nginx_1 ... done
Attaching to sde-a2-t2_app_1, sde-a2-t2_nginx_1
app_1    | /docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to p
erform configuration
app_1    | /docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
app_1    | /docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-d
efault.sh
app_1    | 10-listen-on-ipv6-by-default.sh: info: IPv6 listen already enabled
app_1    | /docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envs
h
app_1    | /docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templat
es.sh
app_1    | /docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-process
es.sh
app_1    | /docker-entrypoint.sh: Configuration complete; ready for start up
app_1    | 2023/09/28 00:06:15 [notice] 1#1: using the "epoll" event method
app_1    | 2023/09/28 00:06:15 [notice] 1#1: nginx/1.25.2
app_1    | 2023/09/28 00:06:15 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
app_1    | 2023/09/28 00:06:15 [notice] 1#1: OS: Linux 6.2.0-1014-gcp
app_1    | 2023/09/28 00:06:15 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
app_1    | 2023/09/28 00:06:15 [notice] 1#1: start worker processes
app_1    | 2023/09/28 00:06:15 [notice] 1#1: start worker process 22
app_1    | 2023/09/28 00:06:15 [notice] 1#1: start worker process 23
nginx_1  | /docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to p
erform configuration
nginx_1  | /docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
```



5. Deploy the Multi-Container Application:

- Note: the above application is also a Multi-Container application as it used nginx and python container.
- SSH into your VM as described earlier.
- Navigate to the directory containing your docker-compose.yml file.
- Run the following command to start the multi-container application:

```
Unset
```

```
docker-compose up -d
```

6. Demonstrate Communication Between Containers:

- It is imperative to ensure that the application components are appropriately set to establish communication within the Docker network as defined by Docker Compose. This configuration should enable the visibility of the components when the request is being directed towards the `unicorn` server through `nginx`.

7. Scale the Application Horizontally:

- To scale your application horizontally, adjust the docker-compose.yml file to specify the desired number of replicas for your services.

```
Python
version: '3'
services:
  nginx:
    image: nginx:latest
    ports:
      - "8081:80"
    volumes:
      - ./nginx/nginx-hscale.conf:/etc/nginx/nginx.conf
    depends_on:
      - app

  app:
    build:
      context: ./backend
    deploy:
      replicas: 3 # number of replicas required
```

- Also modifying nginx config file.

```
Unset
worker_processes 1;

events { worker_connections 1024; }

http {
    upstream fastapi{
        server sde-a2-t2_app_1:80;
        server sde-a2-t2_app_2:80;
        server sde-a2-t2_app_3:80;
    }

    server {
        listen 80;

        location / {
            proxy_pass http://fastapi;
        }
    }
}
```

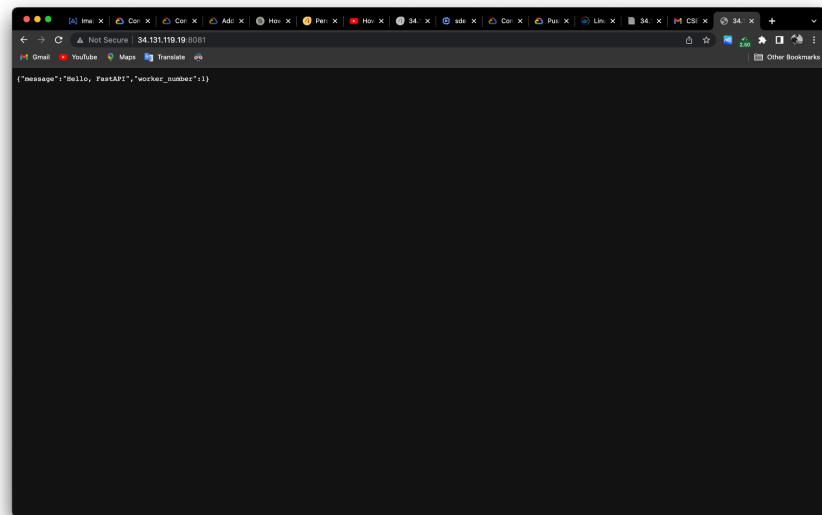
- At the end project will follow the below structure.

```
Unset
.
├── backend
│   ├── Dockerfile
│   ├── main.py
│   └── requirements.txt
├── docker-compose-hscale.yml
├── docker-compose.yml
└── nginx
    ├── nginx-hscale.conf
    └── nginx.conf
```

- Save the file and run the following command to apply the new configuration by using `docker-compose-hscale.yml` file.

```
Unset
docker-compose -f docker-compose-hscale.yml up -d
```

```
tron@sde-a2:~/sde-a2-t2$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
tron@sde-a2:~/sde-a2-t2$ docker-compose -f docker-compose-hscale.yml up -d
Creating network "sde-a2-t2_default" with the default driver
Creating sde-a2-t2_app_1 ... done
Creating sde-a2-t2_app_2 ... done
Creating sde-a2-t2_app_3 ... done
Creating sde-a2-t2_nginx_1 ... done
tron@sde-a2:~/sde-a2-t2$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
a6c88ed9d14 nginx:latest "/docker-entrypoint..." 14 seconds ago Up 13 s
econds 0.0.0.0:8081->80/tcp, :::8081->80/tcp sde-a2-t2_nginx_1
54bcd74f1e6a sde-a2-t2_app "uvicorn main:app --..." 15 seconds ago Up 14 s
econds 80/tcp
6916776f49a9 sde-a2-t2_app "uvicorn main:app --..." 15 seconds ago Up 14 s
econds 80/tcp
6f7f72fbe8f94 sde-a2-t2_app "uvicorn main:app --..." 15 seconds ago Up 14 s
econds 80/tcp
tron@sde-a2:~/sde-a2-t2$
```

A screenshot of a curl command interface. The URL entered is `http://34.131.119.19:8081/`. The response shows a JSON object with the key "message": "Hello, FastAPI" and the key "worker_number": 1. The response code is 200. The response headers include `connection: keep-alive`, `content-length: 14`, `content-type: application/json`, `date: Mon, 20 Mar 2023 00:15:47 GMT`, and `server: nginx/1.25.2`.

Task 3: Container Deployment on GCP (30 points)

1. Create a Google Kubernetes Engine (GKE) cluster.

- Enable GKE API form the console.
- Install the GCP SDK (gcloud): Install the Google Cloud SDK on your machine. This will allow you to interact with GCP services from the command line.
- Authenticate and Set Default Project.

```
Unset
```

```
gcloud auth login  
gcloud config set project PROJECT_ID
```

- Create a GKE Cluster: Use the following command to create a GKE cluster named my-cluster in your specified region and zone.

```
Unset
```

```
gcloud container clusters create sde-a2-cluster --num-nodes=1 --zone  
asia-south2
```

The screenshot shows the Google Cloud Console interface for managing Kubernetes clusters. The URL in the browser is <https://console.cloud.google.com/kubernetes/list/overview?organizationId=315541594599&project=sde-2023>. The page displays a table of existing clusters:

Status	Name	Location	Number of nodes	Total vCPUs	Total memory	Notifications	Labels
Green checkmark	sde-a2	asia-south2	3	6	12 GB	—	⋮

- Configure kubectl: Run the following command to configure kubectl to use the credentials for your new cluster.

```
Unset
```

```
gcloud container clusters get-credentials sde-a2-cluster --zone asia-south2
```

2.Deploy a sample application container (e.g., a basic web app) into the GKE cluster.

- Create a separate directory for the current cluster.
- Create `my-app-deployment.yaml` file.
- We'll be using the pushed docker image in the previous task.

```
Python
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-app
spec:
  replicas: 3
  selector:
    matchLabels:
      app: my-app
  template:
    metadata:
      labels:
        app: my-app
    spec:
      containers:
        - name: my-app-docker
          image:
            asia-south2-docker.pkg.dev/sde-2023/sde-a2-docker-registry/mini-portfolio:test
            # Replace with your custom image name from cloud registry
          ports:
            - containerPort: 80
```

- Create `my-app-service.yaml` file.

```
Python
```

```
apiVersion: v1
kind: Service
metadata:
  name: my-app-service
spec:
  selector:
    app: my-app
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
  type: LoadBalancer
```

- Apply yaml files to the cluster.

```
Unset
```

```
kubectl apply -f my-app-deployment.yaml
kubectl apply -f my-app-service.yaml
```

3. Scale the application by adjusting the number of replicas in the deployment.

- Scale the application by adjusting the number of replicas in the deployment using the below command.

```
Unset
```

```
kubectl scale deployment my-app --replicas=3
```

Note: set replicas to 0 to stop and number greater than zero to scale the application.

```
tron@sde-a2:~/sde-a2-t3$ kubectl scale deployment my-app --replicas=3
deployment.apps/my-app scaled
tron@sde-a2:~/sde-a2-t3$
```

- Use the below command to accessible ip address.

Unset

```
kubectl get svc my-app-service
```

```
tron@sde-a2:~/sde-a2-t3$ kubectl get svc my-app-service
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
my-app-service  LoadBalancer  10.20.8.190  34.131.120.129  80:31693/TCP  26s
tron@sde-a2:~/sde-a2-t3$
```

The screenshot shows a dark-themed resume website. On the left, there's a sidebar with a profile picture of a man, his name 'Bikash Dutta' (with 'Research Grad' below it), and contact information: EMAIL (d22cs05@iitj.ac.in), PHONE (+91 98187-57759), and LOCATION (IIT Jodhpur, Karwar, Rajasthan 342037). Below this are social media links for Facebook, LinkedIn, and GitHub. The main content area has a header 'About Me' with tabs for 'About' (which is active) and 'Resume'. Under 'About Me', there's a short bio: 'I'm a research grad student at IIT-Jodhpur. I am working in the domain of artificial intelligence, machine learning, and deep learning. My aim is to solve real-world problems with the help of artificial intelligence.' Below this is a section titled 'What I'm Doing' with four categories: 'Python Programming', 'Machine Learning', 'Cloud Administration', and 'Full Stack Development'. A note below 'Cloud Administration' says 'Deep Learning (Started Learning With PyTorch)'. The top of the browser window shows multiple tabs open, including 'Not Secure | 34.131.120.129'.

Note: this is running on cluster hence the ip is different then above.

4. Monitor the application's performance and resource utilization using GCP tools.

The screenshot shows the Google Cloud Platform (GCP) dashboard. At the top, there is a banner for a free trial with \$300 in credit. The navigation bar includes links for 'DASHBOARD', 'ACTIVITY', 'RECOMMENDATIONS', and 'CUSTOMIZE'. A search bar is also present.

The main content area is divided into several sections:

- Project info**: Shows project name (SDE-2023), project number (630349215297), and project ID (sde-2023). It also has a link to 'ADD PEOPLE TO THIS PROJECT' and 'Go to project settings'.
- Resources**: Lists services like BigQuery, SQL, Compute Engine, Storage, Cloud Functions, and Cloud Run.
- Trace**: States 'No trace data from the past 7 days' and has a link to 'Get started with Trace'.
- Getting Started**: Provides links to explore APIs, deploy solutions, add logging, monitor errors, deploy apps, take quickstarts, create storage buckets, create functions, install SDKs, and explore tutorials.
- Compute Engine**: A chart showing CPU utilization over time (5:30 AM to 6:15 AM). A callout indicates 'instance/cpu/utilization: 8.14%'. It also has a link to 'Go to Compute Engine'.
- API APIs**: A chart showing requests per second over time. A callout indicates 'Requests: 0.230/s'. It also has a link to 'Go to APIs overview'.
- Google Cloud Platform status**: Reports an issue with Google Kubernetes Engine starting at 2023-09-13 (16:05:13). It includes a link to 'Go to Cloud status dashboard'.
- Billing**: Shows estimated charges for the period Sep 1 – 27, 2023 (USD \$0.00). It includes a link to 'Take a tour of billing' and 'View detailed charges'.
- Monitoring**: Offers options to create a dashboard, set up alerting policies, and create uptime checks. It also includes a link to 'View all dashboards' and 'Go to Monitoring'.
- Error Reporting**: Lists top errors in the last 24 hours, including a probe failure for 'https://10.16.2.4:10250/readyz'. It includes a link to 'Go to Error Reporting'.
- News**: Displays recent news items about Cloud Storage announcements, dynamic query concurrency, and Public Sector Innovation. It includes links to 'Read all news' and 'Read all release notes'.
- Documentation**: Provides links to learn about Compute Engine, Cloud Storage, and Cloud Run.

Start your Free Trial with \$300 in credit. Don't worry—you won't be charged if you run out of credits. [Learn more](#)

DISMISS START FREE

Google Cloud SDE-2023 Search () for resources, docs, products, and more Q Search

Compute Engine sde-a2 EDIT RESET + CREATE MACHINE IMAGE CREATE SIMILAR START / RESUME STOP OPERATIONS

Virtual machines VM instances

MONITOR VM INSTANCES Ops Agent Installed SAVE AS DASHBOARD RESET ZOOM 1 hour 6 hours 1 day 1 week 1 month 6 weeks Custom

METRICS LOGS

Disk Space Utilization Disk Space Usage

Overview CPU Processes Memory Network Summary Packet Mirroring

CLOUD SHELL Terminal (sde-2023) Open Editor Close Reconnect

The connection to your Google Cloud Shell was lost.

```
2023-09-26T21:58:22.990767Z Processing instance: projects/sde-2023/zones/asia-south2-a/instances/sde-a2.
-----Getting output-----
Progress: [=====] 100.0% [1/1] (100.0%) completed; [1/1] (100.0%) succeeded; [0/1] (0.0%) failed;
Instance: projects/sde-2023/zones/asia-south2-a/instances/sde-a2 successfully runs ops-agent. See log file in: ./google_cloud_ops_agent_provisioning/20230926-215822_988791/sde-2023_asia-sou
th2-a_sde-a2.log

SUCCEEDED: [1/1] (100.0%)
FAILED: [0/1] (0.0%)
COMPLETED: [1/1] (100.0%)

d22cs051@cloudshell:~ (sde-2023)$
```

References:

- <https://cloud.google.com/compute/docs>
- <https://cloud.google.com/compute/docs/instances/create-start-instance>
- <https://cloud.google.com/compute/docs/instances/creating-instance-with-custom-machine-type>
- <https://cloud.google.com/compute/docs/connect/add-ssh-keys#metadata>
- <https://cloud.google.com/compute/docs/connect/standard-ssh#openssh-client>
- <https://docs.docker.com/engine/install/ubuntu/>
- <https://docs.docker.com/engine/install/linux-postinstall/>
- <https://docs.docker.com/compose/>
-  How Push docker image to container registry and artifact registry
- <https://cloud.google.com/artifact-registry/docs/access-control>
- <https://cloud.google.com/artifact-registry/docs/docker/pushing-and-pulling#auth>
- <https://cloud.google.com/sdk/docs/install-sdk>
 -  Google Kubernetes Engine: Create GKE Cluster and Deploy Sample Website!!
- <https://cloud.google.com/sdk/docs>
- <https://cloud.google.com/sdk/gcloud/reference/container/clusters>
- <https://cloud.google.com/sdk/gcloud/reference/container/clusters/create>
- <https://cloud.google.com/sdk/gcloud/reference/container/clusters/get-credentials>
- <https://cloud.google.com/sdk/gcloud/reference/container/clusters/delete>