Exercise 1: Understanding Cloud Computing Models

1. Objective: Explore different cloud computing models and understand their key differences.
2. Steps:
   - Create a table comparing these models in terms of control, flexibility, and use cases.
   - Identify examples of services offered by Google Cloud Platform (GCP) under each model.

	IaaS	PaaS	SaaS
Control	Over the servers, storage, networking, resources like virtual machines and virtual storage	Over tools for applications	No control
Flexibility	The most flexible	Medium	The least flexible
Use case	Web Hosting, Backup, Virtual machines and servers	The least flexible Application development and deployment, API management, Big Data analytics	Email and collaboration tools, CRM, HR management systems
Services offered by Google Cloud Platform	Google Compute Engine, Google Cloud Storage, Virtual Private Cloud, Persistent Disk	Google App Engine, Google BigQuery, Cloud Run	Google Maps Platform, Google Workspace, Google Analytics

3. **Questions**:
   - What are the main differences between IaaS, PaaS, and SaaS?
     Main differences between these 3 models are that **IaaS** stands for Infrastructure as a Service, **PaaS** stands for Platform as a Service which provides tools for building and managing software applications and **SaaS** which gives Software as a Service.

   - Which GCP services fall under each of these models?
     There are GCP services for each model:
     IaaS: Google Compute Engine, Google Cloud Storage, Virtual Private Cloud, Persistent Disk.
     PaaS: Google App Engine, Google BigQuery, Cloud Run.
     SaaS: Google Maps Platform, Google Workspace, Google Analytics.

   - Provide a real-world example where each cloud service model might be the most appropriate choice.

Iaas: For example, if a growing company wants to build a custom web application and needs full control over the server environment for security, performance, and specific software configurations, they will choose IaaS to manage virtual machines, storage, and networking. In our world Netflix uses AWS's IaaS model to build its own customized infrastructure for streaming video content to millions of users globally, handling massive spikes in demand.
PaaS: For example, if a startup is developing a mobile application and needs a quick and cost-effective way to deploy the app, with a focus on writing code without worrying about managing servers, runtime environments, or security, they will definitely need PaaS.
SaaS: For example, if a large enterprise needs to improve team collaboration, file sharing, and productivity without managing software installations, updates, or servers, they need ready-to-use software that can be accessed from anywhere.

Exercise 2: Exploring Google Cloud Platform's Core Services

1. Objective: Get acquainted with the core services provided by Google Cloud Platform.
2. Explore and describe the purpose of the following core services. For each service, identify a potential use case in a business scenario:
   Compute Engine creates and runs virtual machines on Google Cloud Platform.
   Google Kubernetes Engine (GKE) builds and manages container-based applications, powered by the open-source Kubernetes technology.
   App Engine is a managed platform for building and deploying applications without worrying about infrastructure management.
   Cloud Storage is a managed service for storing unstructured data.
   BigQuery stores and analyzes large amounts of data easily and allows to run fast SQL queries.
3. **Questions**:
   - What is the primary use case of Compute Engine?
     Primary use case of Compute Engine is to run virtual machines (VMs) on Google's infrastructure. It provides scalable computing for various workloads.

   - How does Google Kubernetes Engine (GKE) simplify the management of containerized applications?
     GKE simplifies the deployment, management, and scaling of containerized applications by providing an automated, managed environment as auto-scaling, automatic cluster management, etc. for Kubernetes.

   - What advantages does Cloud Storage offer for data management?
     Cloud Storage offers scalability, durability and availability, global accessibility for data management, especially for large-scale, unstructured data.

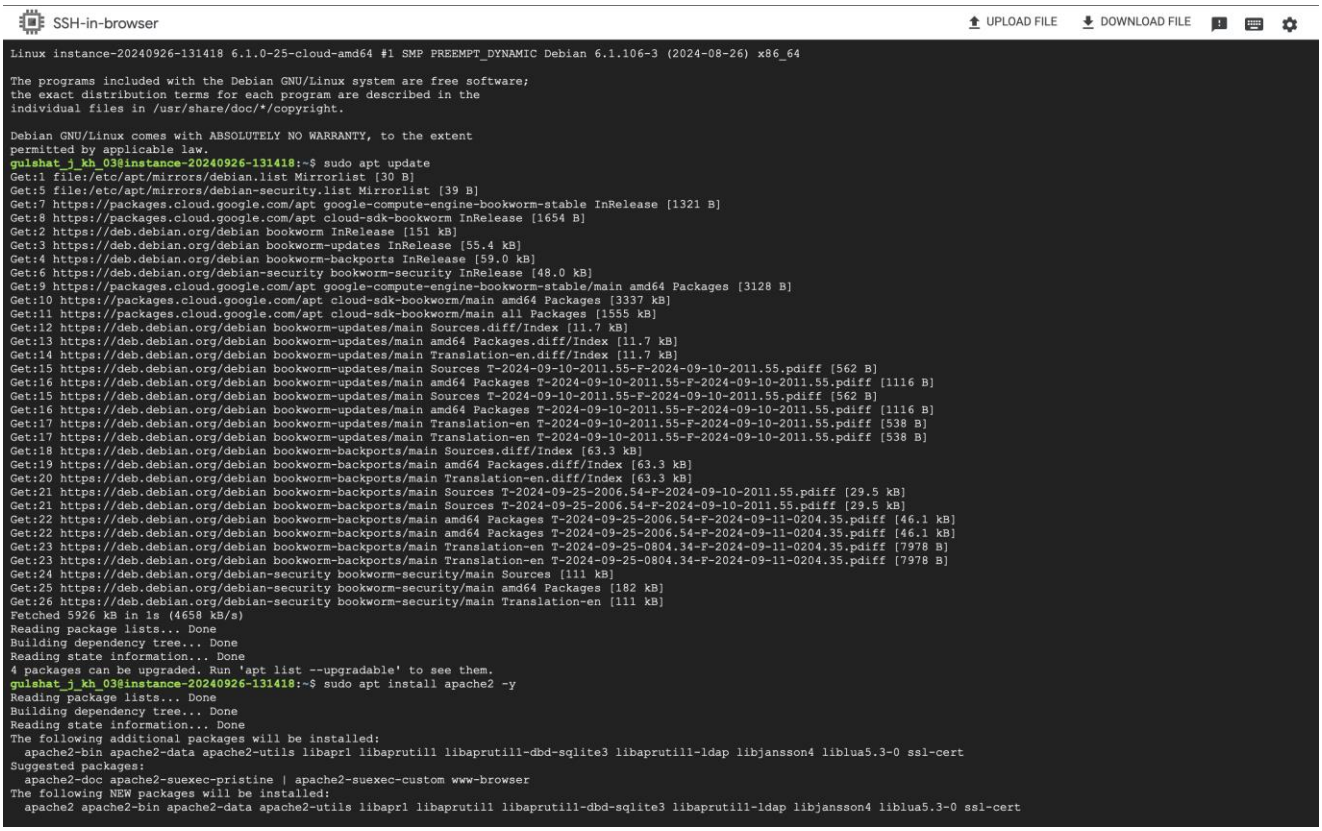
   - Why would a business choose BigQuery for their data analysis needs?

BigQuery is a fully managed, serverless data warehouse designed for fast, scalable analytics. BigQuery can handle large-scale data analysis tasks, supports real-time data ingestion and querying, offers built-in machine learning (BigQuery ML), geospatial analysis, and integration with other tools.

Exercise 3: Creating and Managing Virtual Machines with Compute Engine

1. Objective: Learn how to create, manage, and interact with virtual machines (VMs) using Compute Engine.
2. **Questions:**
   - What steps did you follow to create the VM?
     I went to Cloud Engine in the navigation menu, then clicked VM instances -> Create instance. It showed a modal window with fields to fill information about name, region, zone, machine type, operating system. After filling fields, I chose a button "Create". Then created VM appeared in VM instance page, where I clicked "SSH", after that it opened SSH terminal window.

   - How did you connect to the VM, and what commands did you use to install the web server?
     In opened SSH terminal window I run these commands to install a web server Apache on my VM:


SSH-in-browser
UPLOAD FILE
DOWNLOAD FILE
Linux instance-20240926-131418 6.1.0-25-cloud-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.106-3 (2024-08-26) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
gulshat_j_kh_03@instance-20240926-131418:~$ sudo apt update
Get:1 file:/etc/apt/mirrors/debian.list Mirrorlist [30 B]
Get:5 file:/etc/apt/mirrors/debian-security.list Mirrorlist [39 B]
Get:7 https://packages.cloud.google.com/apt google-compute-engine-bookworm-stable InRelease [1321 B]
Get:8 https://packages.cloud.google.com/apt cloud-sdk-bookworm InRelease [1654 B]
Get:2 https://deb.debian.org/debian bookworm InRelease [151 kB]
Get:3 https://deb.debian.org/debian bookworm-updates InRelease [55.4 kB]
Get:4 https://deb.debian.org/debian bookworm-backports InRelease [59.0 kB]
Get:6 https://deb.debian.org/debian-security bookworm-security InRelease [48.0 kB]
Get:9 https://packages.cloud.google.com/apt google-compute-engine-bookworm-stable/main amd64 Packages [3128 B]
Get:10 https://packages.cloud.google.com/apt cloud-sdk-bookworm/main amd64 Packages [3337 kB]
Get:11 https://packages.cloud.google.com/apt cloud-sdk-bookworm/main all Packages [1555 kB]
Get:12 https://deb.debian.org/debian bookworm-updates/main Sources.diff/Index [11.7 kB]
Get:13 https://deb.debian.org/debian bookworm-updates/main amd64 Packages.diff/Index [11.7 kB]
Get:14 https://deb.debian.org/debian bookworm-updates/main Translation-en.diff/Index [11.7 kB]
Get:15 https://deb.debian.org/debian bookworm-updates/main Sources T-2024-09-10-2011.55-F-2024-09-10-2011.55.pdiff [562 B]
Get:16 https://deb.debian.org/debian bookworm-updates/main amd64 Packages T-2024-09-10-2011.55-F-2024-09-10-2011.55.pdiff [1116 B]
Get:15 https://deb.debian.org/debian bookworm-updates/main Sources T-2024-09-10-2011.55-F-2024-09-10-2011.55.pdiff [562 B]
Get:16 https://deb.debian.org/debian bookworm-updates/main amd64 Packages T-2024-09-10-2011.55-F-2024-09-10-2011.55.pdiff [1116 B]
Get:17 https://deb.debian.org/debian bookworm-updates/main Translation-en T-2024-09-10-2011.55-F-2024-09-10-2011.55.pdiff [538 B]
Get:17 https://deb.debian.org/debian bookworm-updates/main Translation-en T-2024-09-10-2011.55-F-2024-09-10-2011.55.pdiff [538 B]
Get:18 https://deb.debian.org/debian bookworm-backports/main Sources.diff/Index [63.3 kB]
Get:19 https://deb.debian.org/debian bookworm-backports/main amd64 Packages.diff/Index [63.3 kB]
Get:20 https://deb.debian.org/debian bookworm-backports/main Translation-en.diff/Index [63.3 kB]
Get:21 https://deb.debian.org/debian bookworm-backports/main Sources T-2024-09-25-2006.54-F-2024-09-10-2011.55.pdiff [29.5 kB]
Get:21 https://deb.debian.org/debian bookworm-backports/main Sources T-2024-09-25-2006.54-F-2024-09-10-2011.55.pdiff [29.5 kB]
Get:22 https://deb.debian.org/debian bookworm-backports/main amd64 Packages T-2024-09-25-2006.54-F-2024-09-11-0204.35.pdiff [46.1 kB]
Get:22 https://deb.debian.org/debian bookworm-backports/main amd64 Packages T-2024-09-25-2006.54-F-2024-09-11-0204.35.pdiff [46.1 kB]
Get:23 https://deb.debian.org/debian bookworm-backports/main Translation-en T-2024-09-25-0804.34-F-2024-09-11-0204.35.pdiff [7978 B]
Get:23 https://deb.debian.org/debian bookworm-backports/main Translation-en T-2024-09-25-0804.34-F-2024-09-11-0204.35.pdiff [7978 B]
Get:24 https://deb.debian.org/debian-security bookworm-security/main Sources [111 kB]
Get:25 https://deb.debian.org/debian-security bookworm-security/main amd64 Packages [182 kB]
Get:26 https://deb.debian.org/debian-security bookworm-security/main Translation-en [111 kB]
Fetched 5926 kB in 1s (4658 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
4 packages can be upgraded. Run 'apt list --upgradable' to see them.
gulshat_j_kh_03@instance-20240926-131418:~$ sudo apt install apache2 -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libjansson4 liblua5.3-0 ssl-cert
Suggested packages:
apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser
The following NEW packages will be installed:
apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libjansson4 liblua5.3-0 ssl-cert

SSH-in-browser

UPLOAD FILE DOWNLOAD FILE

```
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libjansson4 liblua5.3-0 ssl-cert
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libjansson4 liblua5.3-0 ssl-cert
0 upgraded, 11 newly installed, 0 to remove and 4 not upgraded.
Need to get 2378 kB of archives.
After this operation, 8468 kB of additional disk space will be used.
Get:1 file:/etc/apt/mirrors/debian.list Mirrorlist [30 B]
Get:2 https://deb.debian.org/debian bookworm/main amd64 libapr1 amd64 1.7.2-3 [102 kB]
Get:3 https://deb.debian.org/debian bookworm/main amd64 libaprutil1 amd64 1.6.3-1 [87.8 kB]
Get:4 https://deb.debian.org/debian bookworm/main amd64 libaprutil1-dbd-sqlite3 amd64 1.6.3-1 [13.6 kB]
Get:5 https://deb.debian.org/debian bookworm/main amd64 libaprutil1-ldap amd64 1.6.3-1 [11.8 kB]
Get:6 https://deb.debian.org/debian bookworm/main amd64 libjansson4 amd64 2.14-2 [40.8 kB]
Get:7 https://deb.debian.org/debian bookworm/main amd64 liblua5.3-0 amd64 5.3.6-2 [123 kB]
Get:8 https://deb.debian.org/debian bookworm/main amd64 apache2-bin amd64 2.4.62-1~deb12u1 [1385 kB]
Get:9 https://deb.debian.org/debian bookworm/main amd64 apache2-data all 2.4.62-1~deb12u1 [160 kB]
Get:10 https://deb.debian.org/debian bookworm/main amd64 apache2-utils amd64 2.4.62-1~deb12u1 [210 kB]
Get:11 https://deb.debian.org/debian bookworm/main amd64 apache2 amd64 2.4.62-1~deb12u1 [223 kB]
Get:12 https://deb.debian.org/debian bookworm/main amd64 ssl-cert all 1.1.2 [21.1 kB]
Fetched 2378 kB in 0s (13.0 MB/s)
Preconfiguring packages ...
Selecting previously unselected package libapr1:amd64.
(Reading database ... 69885 files and directories currently installed.)
Preparing to unpack .../00-libapr1_1.7.2-3_amd64.deb ...
Unpacking libapr1:amd64 (1.7.2-3) ...
Selecting previously unselected package libaprutil1:amd64.
Preparing to unpack .../01-libaprutil1_1.6.3-1_amd64.deb ...
Unpacking libaprutil1:amd64 (1.6.3-1) ...
Selecting previously unselected package libaprutil1-dbd-sqlite3:amd64.
Preparing to unpack .../02-libaprutil1-dbd-sqlite3_1.6.3-1_amd64.deb ...
Unpacking libaprutil1-dbd-sqlite3:amd64 (1.6.3-1) ...
Selecting previously unselected package libaprutil1-ldap:amd64.
Preparing to unpack .../03-libaprutil1-ldap_1.6.3-1_amd64.deb ...
Unpacking libaprutil1-ldap:amd64 (1.6.3-1) ...
Selecting previously unselected package libjansson4:amd64.
Preparing to unpack .../04-libjansson4_2.14-2_amd64.deb ...
Unpacking libjansson4:amd64 (2.14-2) ...
Selecting previously unselected package liblua5.3-0:amd64.
Preparing to unpack .../05-liblua5.3-0_5.3.6-2_amd64.deb ...
Unpacking liblua5.3-0:amd64 (5.3.6-2) ...
Selecting previously unselected package apache2-bin.
Preparing to unpack .../06-apache2-bin_2.4.62-1~deb12u1_amd64.deb ...
Unpacking apache2-bin (2.4.62-1~deb12u1) ...
Selecting previously unselected package apache2-data.
Preparing to unpack .../07-apache2-data_2.4.62-1~deb12u1_all.deb ...
Unpacking apache2-data (2.4.62-1~deb12u1) ...
Selecting previously unselected package apache2-utils.
Preparing to unpack .../08-apache2-utils_2.4.62-1~deb12u1_amd64.deb ...
Unpacking apache2-utils (2.4.62-1~deb12u1) ...
Selecting previously unselected package apache2.
Preparing to unpack .../09-apache2_2.4.62-1~deb12u1_amd64.deb ...
Unpacking apache2 (2.4.62-1~deb12u1) ...
Selecting previously unselected package ssl-cert.
Preparing to unpack .../10-ssl-cert_1.1.2_all.deb ...
Unpacking ssl-cert (1.1.2) ...
```

SSH-in-browser

UPLOAD FILE DOWNLOAD FILE

```
Setting up libapr1:amd64 (1.7.2-3) ...
Setting up libjansson4:amd64 (2.14-2) ...
Setting up ssl-cert (1.1.2) ...
Setting up liblua5.3-0:amd64 (5.3.6-2) ...
Setting up apache2-data (2.4.62-1~deb12u1) ...
Setting up libaprutil1:amd64 (1.6.3-1) ...
Setting up libaprutil1-ldap:amd64 (1.6.3-1) ...
Setting up libaprutil1-dbd-sqlite3:amd64 (1.6.3-1) ...
Setting up apache2-utils (2.4.62-1~deb12u1) ...
Setting up apache2-bin (2.4.62-1~deb12u1) ...
Setting up apache2 (2.4.62-1~deb12u1) ...
Enabling module mpm_event.
Enabling module authz_core.
Enabling module authz_host.
Enabling module authn_core.
Enabling module auth_basic.
Enabling module access_compat.
Enabling module authn_file.
Enabling module authz_user.
Enabling module alias.
Enabling module dir.
Enabling module autoindex.
Enabling module env.
Enabling module mime.
Enabling module negotiation.
Enabling module setenvif.
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service → /lib/systemd/system/apache-htcacheclean.service.
Processing triggers for man-db (2.11.2-2) ...
Processing triggers for libc-bin (2.36-9+deb12u8) ...
gulshat_j_kh_03@instance-20240926-131418:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Thu 2024-09-26 13:27:05 UTC; 52s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 1900 (apache2)
      Tasks: 55 (limit: 4682)
     Memory: 10.9M
        CPU: 54ms
     CGroup: /system.slice/apache2.service
             ├─1900 /usr/sbin/apache2 -k start
             ├─1901 /usr/sbin/apache2 -k start
             └─1902 /usr/sbin/apache2 -k start

Sep 26 13:27:05 instance-20240926-131418 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Sep 26 13:27:05 instance-20240926-131418 systemd[1]: Started apache2.service - The Apache HTTP Server.
```

- What happens to the VM and its data when it is stopped versus when it is deleted?
  STOP: The virtual machine will shut down, and its status changes to TERMINATED
  DELETE: The VM will be deleted and cannot be recovered. The status of the VM will be removed from the list of instances.

Exercise 4: Deploying a Containerized Application on Google Kubernetes Engine (GKE)

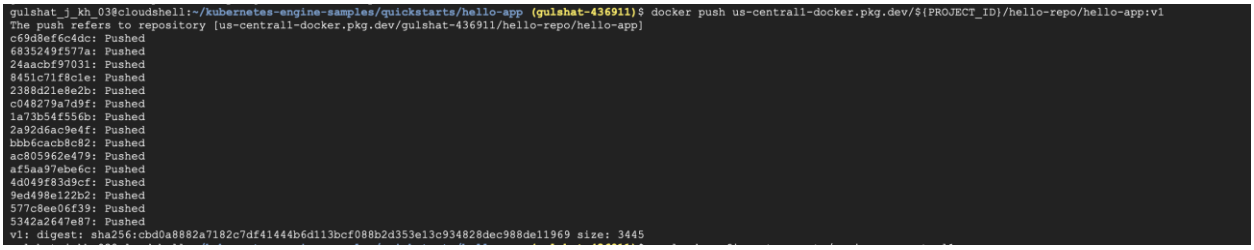
1. Objective: Understand how to deploy and manage containerized applications using Google Kubernetes Engine.

1) First, I created a repository named hello-repo.

2) Web application called ”hello-app” I got from GitHub, then built and tagged the Docker image for this app: https://github.com/GoogleCloudPlatform/kubernetes-engine-samples/tree/main/quickstarts/hello-app.

3) Added IAM policy bindings to my account:

4) Pushed the Docker image to the repository hello-repo:



5) Created a GKE cluster to run hello-app:

```
gulshat_j_kh_03@cloudshell:~/kubernetes-engine-samples/quickstarts/hello-app (gulshat-436911)$  gcloud config set compute/region us-central1
WARNING: Property validation for compute/region was skipped.
Updated property [compute/region].
gulshat_j_kh_03@cloudshell:~/kubernetes-engine-samples/quickstarts/hello-app (gulshat-436911)$  gcloud container clusters create-auto hello-cluster
Note: The Kubelet readonly port (10255) is now deprecated. Please update your workloads to use the recommended alternatives. See https://cloud.google.com/kubernetes-engine/docs/how-to/disable-kubelet-readonly-port for ways to check usage and for migration instructions.
Creating cluster hello-cluster in us-central1... Cluster is being health-checked (master is healthy)...done.
Created [https://container.googleapis.com/v1/projects/gulshat-436911/zones/us-central1/clusters/hello-cluster].
To inspect the contents of your cluster, go to: https://console.cloud.google.com/kubernetes/workload_/gcloud/us-central1/hello-cluster?project=gulshat-436911
kubeconfig entry generated for hello-cluster.
NAME: hello-cluster
LOCATION: us-central1
MASTER_VERSION: 1.30.3-gke.1969001
MASTER_IP: 34.30.187.60
MACHINE_TYPE: e2-small
NODE_VERSION: 1.30.3-gke.1969001
NUM_NODES: 3
STATUS: RUNNING
```

6) Created a Kubernetes Deployment for hello-app Docker image:

```
gulshat_j_kh_03@cloudshell:~/kubernetes-engine-samples/quickstarts/hello-app (gulshat-436911)$ kubectl create deployment hello-app --image=us-central1-docker.pkg.dev/${PROJECT_ID}/hello-repo/hello-app:v1
Warning: autopilot-default-resources-mutator:Autopilot updated Deployment default/hello-app: defaulted unspecified 'cpu' resource for containers [hello-app] (see http://g.co/gke/autopilot-defaults).
deployment.apps/hello-app created
```

7) Set the scales and create a HorizontalPodAutoscaler for deployment:

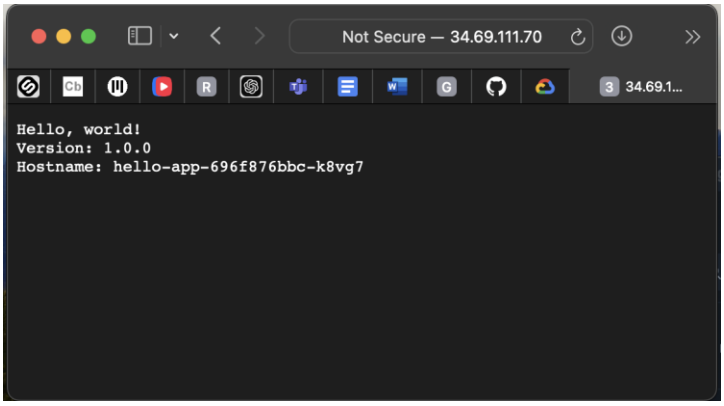
```
gulshat_j_kh_03@cloudshell:~/kubernetes-engine-samples/quickstarts/hello-app (gulshat-436911)$ kubectl scale deployment hello-app --replicas=3
deployment.apps/hello-app scaled
gulshat_j_kh_03@cloudshell:~/kubernetes-engine-samples/quickstarts/hello-app (gulshat-436911)$ kubectl autoscale deployment hello-app --cpu-percent=80 --min=1 --max=5
horizontalpodautoscaler.autoscaling/hello-app autoscaled
```

8) Checked pods:

```
gulshat_j_kh_03@cloudshell:~/kubernetes-engine-samples/quickstarts/hello-app (gulshat-436911)$ kubectl get pods
NAME                         READY   STATUS    RESTARTS   AGE
hello-app-696f876bbc-4kwfw   1/1     Running   0          2m8s
hello-app-696f876bbc-k8vg7   1/1     Running   0          2m8s
hello-app-696f876bbc-n25n7   1/1     Running   0          2m21s
```

9) Checked external-IP: 34.69.111.70 which means that the "hello-app" Pods are exposed to the internet through a Kubernetes Service by this IP-address:

```
gulshat_j_kh_03@cloudshell:~/kubernetes-engine-samples/quickstarts/hello-app (gulshat-436911)$ kubectl expose deployment hello-app --name=hello-app-service --type=LoadBalancer --port 80 --target-port 8080
service/hello-app-service exposed
```



2. **Questions**:

- How did you create and push the Docker container to GCR?
  I got dockerfile from github repository, built docker image by command 'docker build' (step 2). Pushing images from GCR requires the service account to have permissions to read or write to GCR. This is controlled via IAM (step 3).

- What steps were involved in setting up the GKE cluster?
  After those steps, stored a docker container in Artifact Registry (step 4). So, Docker image is stored in Artifact Registry, then created a GKE cluster to run hello-app (step

5). Created a Kubernetes Deployment for hello-app Docker image by scaling and setting limitans to Deployment pods (step 6,7,8).

- How did you verify that your application was successfully deployed and accessible?
  I used the 'kubectl expose' command to generate a Kubernetes Service for the hello-app deployment and got IP address by checking service details (step 9). Then pasted this address in browser and verified that hello-app was successfully depoyed on the Internet.

Exercise 5: Storing and Accessing Data in Google Cloud Storage.

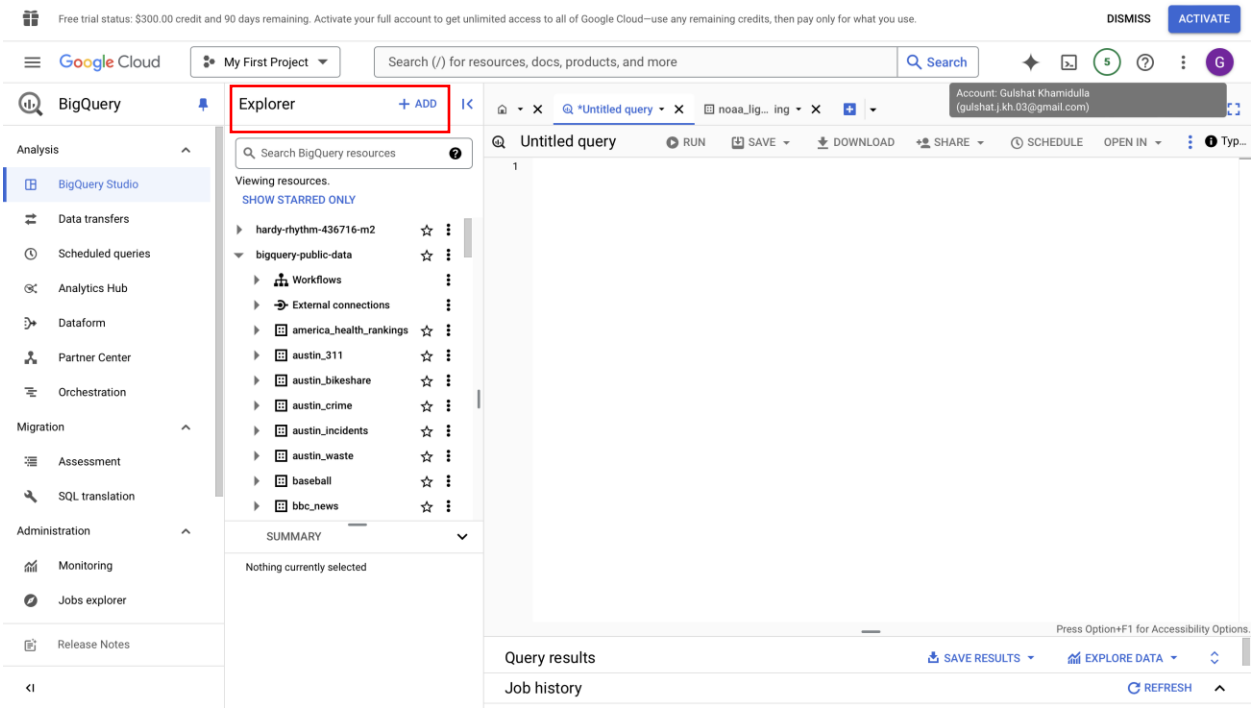
1. Objective: Learn how to store, manage, and access data using Google Cloud Storage.
2. **Questions**:
   - How do you create a Cloud Storage bucket, and what options are available during setup?
     To begin using GCS, I first went to the Google Cloud Console. After that, I clicked the "Create Bucket" button. During the setup, I provided a unique name for my bucket and selected a storage location, region.

   - What are the differences between setting a bucket to public versus private?
     With **public access** any user on the internet can view the files in the bucket if given the URL. While with **private access** only authenticated users or specific service accounts with granted permissions can access the bucket and we can control who has access to the bucket by assigning roles to specific users.
   - How can you manage access permissions for individual files in a bucket?
     By setting bucket to Fine-grained access control, we can manage permissions for individual files.

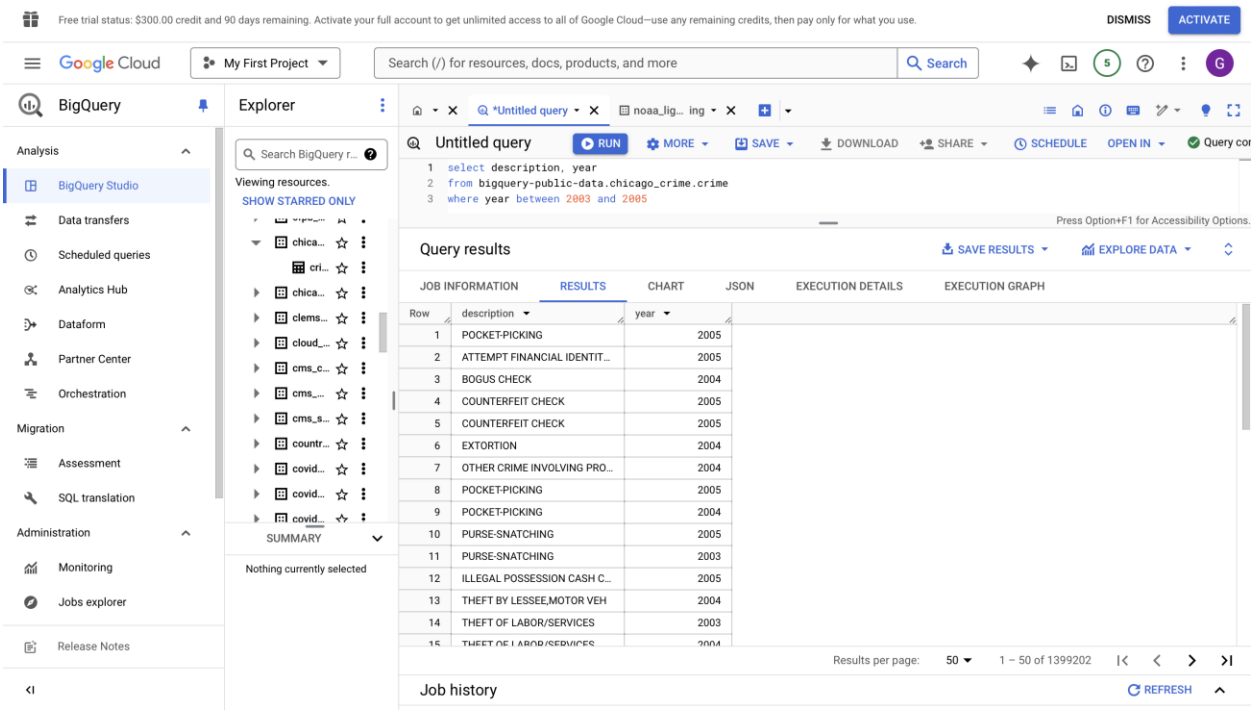
Exercise 6: Analyzing Data with BigQuery.

1. Objective: Perform data analysis tasks using BigQuery.
2. **Questions**:
   - What steps did you take to create a dataset and table in BigQuery?

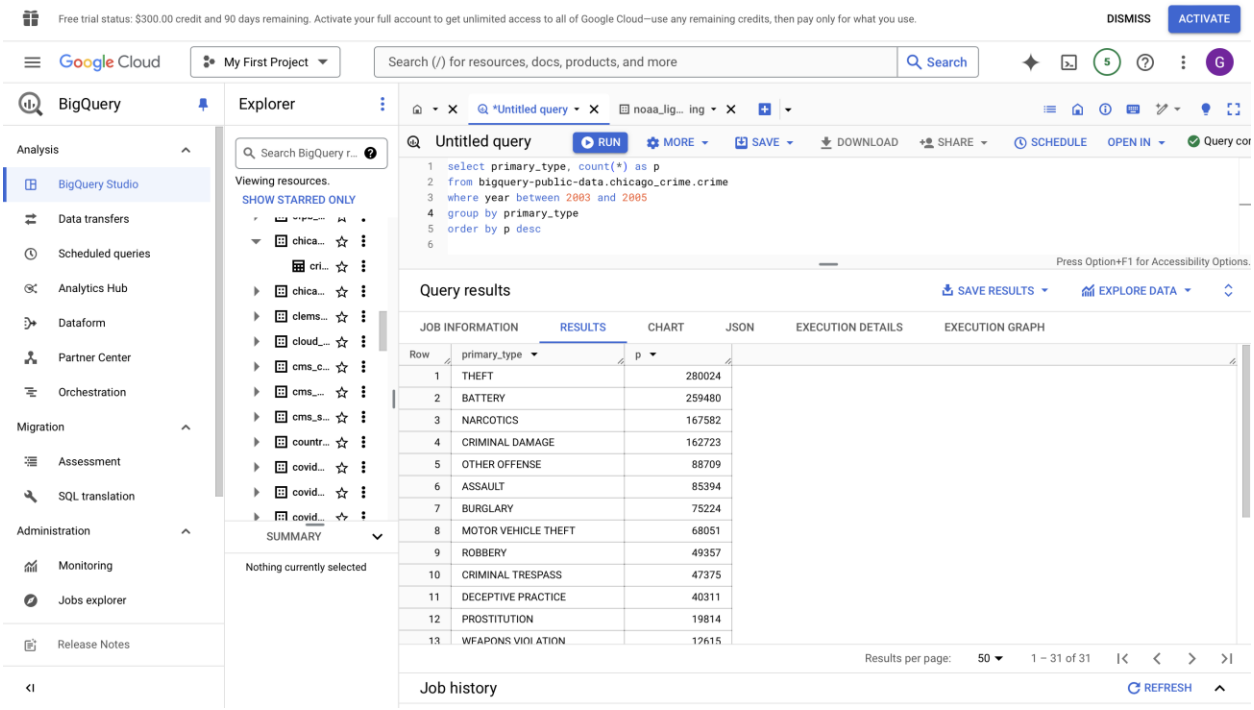
   I went to Navigation Menu -> BigQuery -> Explorer -> Add. Then searched "Public datasets" and chose one.



- How did you write and execute SQL queries in BigQuery?
  In Query tab wrote and run some sql queries:
  1. This query shows what crimes were committed in 2003-2005 years.



  2. Next query shows the most submitted crime description in 2003-2005 years. As written in result, the most submitted crime is theft.



- What insights were you able to derive from the data analysis?

ActuIly I could derive any information I wanted. For example, I was able to sort crimes by location, years, coordinates.