Assignment 1, Cloud Computing

Kerimbay Kairuddin

Exercise 1: Understanding Cloud Computing Models

IaaS (Infrastructure as a Service), PaaS (Platform as a Service), and SaaS (Software as a Service) are three fundamental models of cloud computing, each offering different levels of abstraction and control to users.

1) What are the main differences between IaaS, PaaS, and SaaS?

Feature	**IaaS**	**PaaS**	**SaaS**
Abstraction Level	Lowest	Medium	Highest
Control	Highest	Medium	Lowest
Components	Hardware (servers, storage, networking)	Operating system, programming languages, database	Entire application
User Responsibilities	Managing operating systems, applications, and security	Managing applications and data	Accessing and using the application
Examples	AWS, Azure, GCP	Heroku, Google App Engine, AWS Elastic Beanstalk	Salesforce, Microsoft 365, Google Workspace
Ideal For	Businesses with significant IT expertise and flexibility requirements	Developers building web applications or mobile apps	Businesses that need ready-to-use applications

2) Which GCP services fall under each of these models?

IaaS: Compute Engine: virtual machines, Cloud Run: serverless computing**,** Cloud Storage: object storage, Persistent Disk: block storage, Cloud SQL: managed database**,** Cloud Load Balancing, Cloud DNS.

PaaS: App Engine: fully managed platform for web applications, Cloud Functions: serverless functions**,** Cloud Dataflow: data processing pipelines, AI Platform: machine learning platform**,** Cloud Endpoints: API management

SaaS: Google Workspace (Gmail, Docs, Sheets, etc.) collaboration with Google Chat, Google Meet and business applications: Google Cloud Search, Google Analytics.

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

IaaS: A large e-commerce company needs to scale their infrastructure rapidly during peak shopping seasons. They choose IaaS to have full control over their resources and can easily provision additional servers, storage, and networking components as needed.

PaaS: A startup is developing a mobile app and wants to focus on building the application itself without worrying about managing underlying infrastructure. They use PaaS to deploy their app on a scalable platform, handling tasks like server management, operating system updates, and database administration.

SaaS: A small business needs a customer relationship management (CRM) system but doesn't have the IT resources to manage it in-house. They opt for a SaaS CRM solution, which is hosted by the provider and accessible via the internet, allowing them to focus on their core business.

Exercise 2: Exploring Google Cloud Platform's Core Service

1) What is the primary use case of Compute Engine?

Compute Engine's primary use case is to provide scalable and reliable virtual machines (VMs) for running various applications and workloads on Google Cloud Platform (GCP).

2) How does Google Kubernetes Engine (GKE) simplify the management of containerized applications?

GKE simplifies container management by handling infrastructure, scaling, and integration, while ensuring security, supporting blue-green deployments, and working with CI/CD tools.

3) What advantages does Cloud Storage offer for data management?

Scalability: It can easily scale to accommodate growing data needs without requiring significant upfront investment.

Durability: Data is stored redundantly across multiple data centers, ensuring high availability and durability.

Accessibility: Data can be accessed from anywhere in the world with an internet connection.

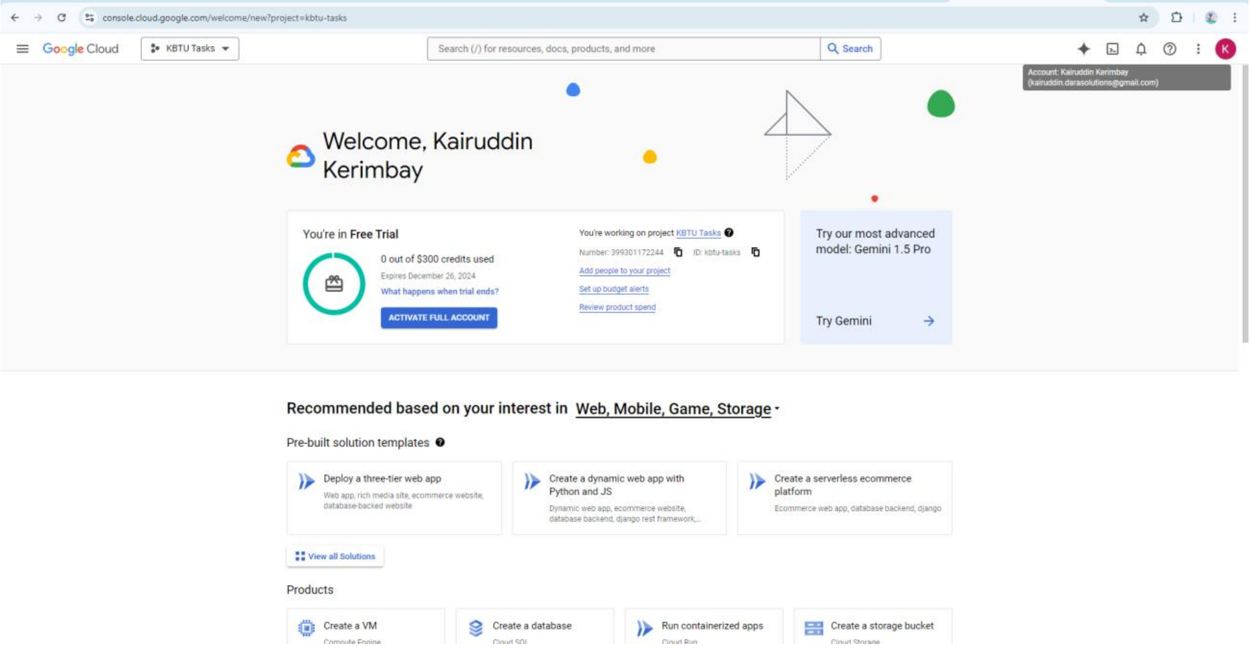
Cost-effective: Cloud Storage can be more cost-effective than traditional on-premises storage solutions, especially for large datasets.

Integration: It integrates seamlessly with other Google Cloud Platform services, making it easy to manage and analyze data.

Security: Cloud Storage provides robust security features, including encryption and access controls.

4) Why would a business choose Big Query for their data analysis needs?

It can handle massive datasets with billions of rows and petabytes of data. There's no need to manage infrastructure, allowing businesses to focus on analysis. Big Query is designed for fast query performance, even on large datasets. It integrates seamlessly with other GCP services, making it easy to work with data from different sources. Pricing is based on the amount of data processed, making it cost-effective for most workloads. Big Query uses a standard SQL dialect, making it easy for analysts to use.



console.cloud.google.com/welcome/new?project=kbtu-tasks

Google Cloud | KBTU Tasks | Search (/) for resources, docs, products, and more | Search

Account: Kairuddin Kerimbay (kairuddin.darasolutions@gmail.com)

Welcome, Kairuddin Kerimbay

You're in **Free Trial**

0 out of $300 credits used

Expires December 26, 2024

What happens when trial ends?

ACTIVATE FULL ACCOUNT

You're working on project KBTU Tasks

Number: 399301172244 ID: kbtu-tasks

Add people to your project

Set up budget alerts

Review product spend

Try our most advanced model: Gemini 1.5 Pro

Try Gemini →

Recommended based on your interest in Web, Mobile, Game, Storage

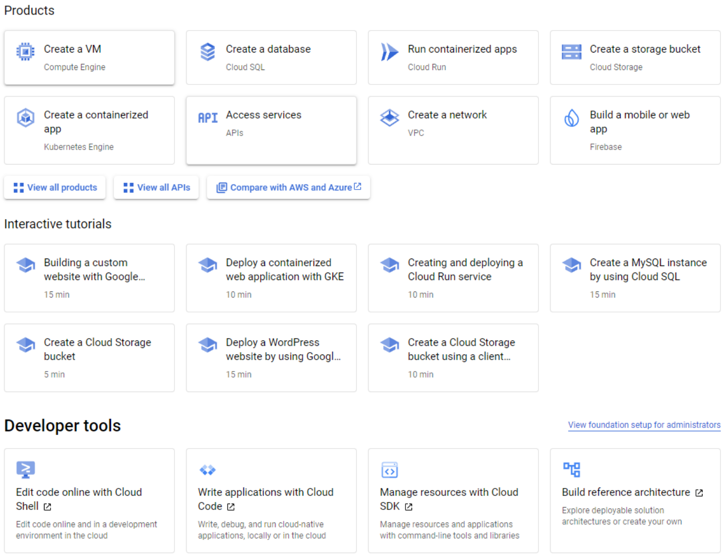
Pre-built solution templates

- Deploy a three-tier web app — Web app, rich media site, ecommerce website, database-backed website
- Create a dynamic web app with Python and JS — Dynamic web app, ecommerce website, database backend, django rest framework,...
- Create a serverless ecommerce platform — Ecommerce web app, database backend, django

View all Solutions

Products

- Create a VM — Compute Engine
- Create a database — Cloud SQL
- Run containerized apps — Cloud Run
- Create a storage bucket — Cloud Storage



Products

- Create a VM — Compute Engine
- Create a database — Cloud SQL
- Run containerized apps — Cloud Run
- Create a storage bucket — Cloud Storage
- Create a containerized app — Kubernetes Engine
- Access services — APIs
- Create a network — VPC
- Build a mobile or web app — Firebase

View all products | View all APIs | Compare with AWS and Azure

Interactive tutorials

- Building a custom website with Google... — 15 min
- Deploy a containerized web application with GKE — 10 min
- Creating and deploying a Cloud Run service — 10 min
- Create a MySQL instance by using Cloud SQL — 15 min
- Create a Cloud Storage bucket — 5 min
- Deploy a WordPress website by using Googl... — 15 min
- Create a Cloud Storage bucket using a client... — 10 min

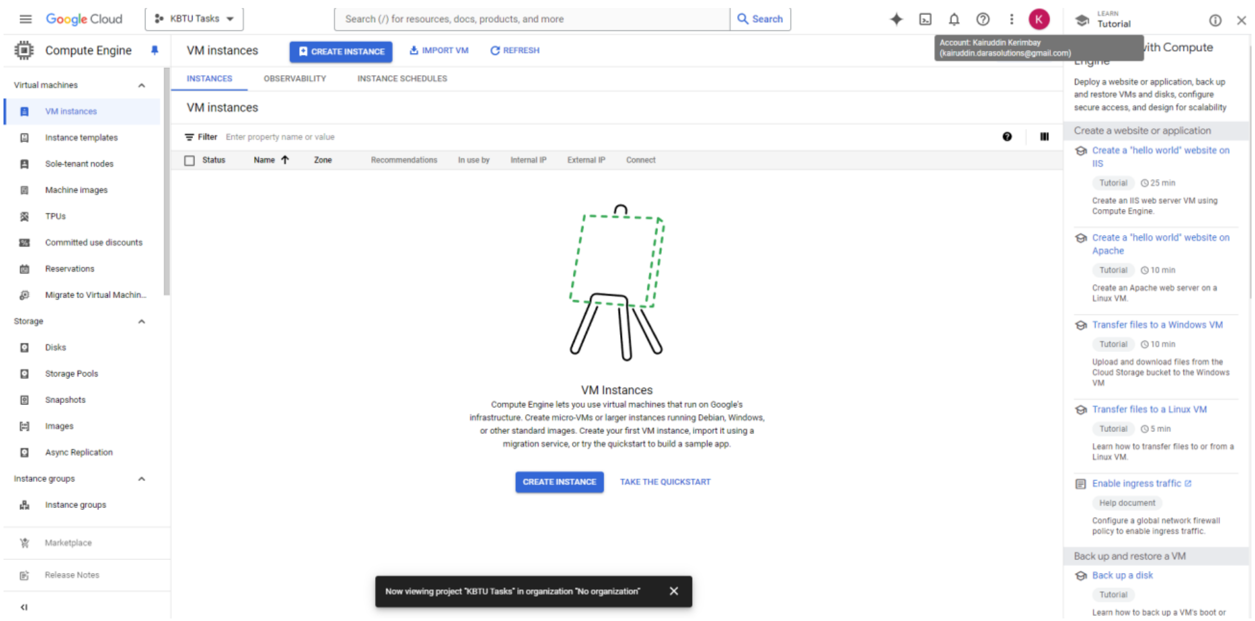
Developer tools

View foundation setup for administrators

- Edit code online with Cloud Shell — Edit code online and in a development environment in the cloud
- Write applications with Cloud Code — Write, debug, and run cloud-native applications, locally or in the cloud
- Manage resources with Cloud SDK — Manage resources and applications with command-line tools and libraries
- Build reference architecture — Explore deployable solution architectures or create your own

Exercise 3: Creating and Managing Virtual Machines with Compute Engine

In the Google Cloud Console, navigate to Compute Engine and create a new VM instance.



Account: Kairuddin Kerimbay (kairuddin.darasolutions@gmail.com)

VM instances

VM Instances

Compute Engine lets you use virtual machines that run on Google's infrastructure. Create micro-VMs or larger instances running Debian, Windows, or other standard images. Create your first VM instance, import it using a migration service, or try the quickstart to build a sample app.

CREATE INSTANCE TAKE THE QUICKSTART

Now viewing project "KBTU Tasks" in organization "No organization"

ith Compute Engine

Deploy a website or application, back up and restore VMs and disks, configure secure access, and design for scalability

Create a website or application

Create a "hello world" website on IIS — Tutorial, 25 min. Create an IIS web server VM using Compute Engine.

Create a "hello world" website on Apache — Tutorial, 10 min. Create an Apache web server on a Linux VM.

Transfer files to a Windows VM — Tutorial, 10 min. Upload and download files from the Cloud Storage bucket to the Windows VM

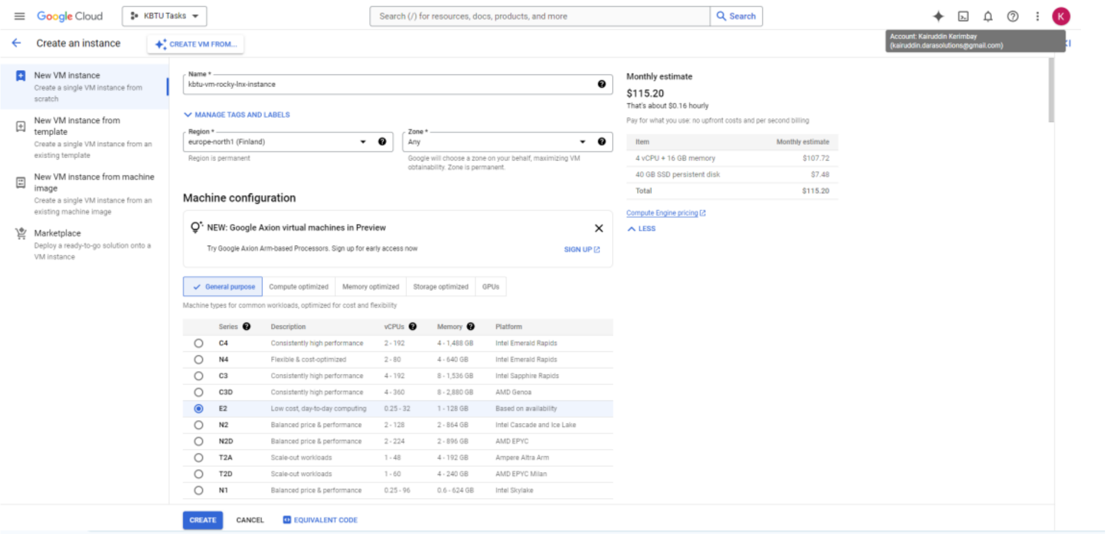
Transfer files to a Linux VM — Tutorial, 5 min. Learn how to transfer files to or from a Linux VM.

Enable ingress traffic — Help document. Configure a global network firewall policy to enable ingress traffic.

Back up and restore a VM

Back up a disk — Tutorial. Learn how to back up a VM's boot or

Configure the VM with specific parameters, such as the machine type, region, and operating system



Account: Kairuddin Kerimbay (kairuddin.darasolutions@gmail.com)

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 *: kbtu-vm-rocky-lnx-instance

Region *: europe-north1 (Finland) — Region is permanent

Zone *: Any — Google will choose a zone on your behalf, maximizing VM obtainability. Zone is permanent.

Machine configuration

NEW: Google Axion virtual machines in Preview

Try Google Axion Arm-based Processors. Sign up for early access now

Machine types for common workloads, optimized for cost and flexibility

	Series	Description	vCPUs	Memory	Platform
○	C4	Consistently high performance	2 - 192	4 - 1,488 GB	Intel Emerald Rapids
○	N4	Flexible & cost-optimized	2 - 80	4 - 640 GB	Intel Emerald Rapids
○	C3	Consistently high performance	4 - 192	8 - 1,536 GB	Intel Sapphire Rapids
○	C3D	Consistently high performance	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

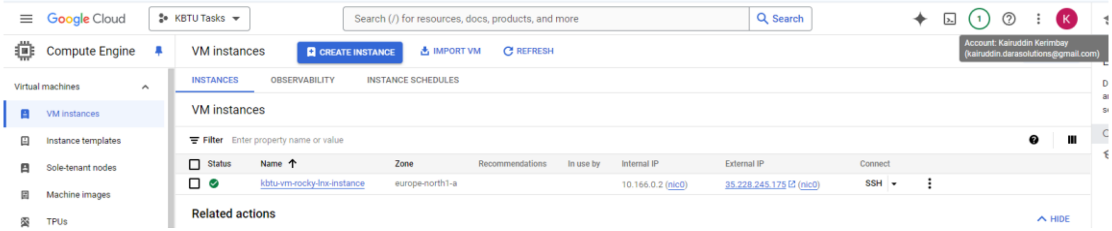
Monthly estimate

$115.20

That's about $0.16 hourly

Pay for what you use: no upfront costs and per second billing

Item	Monthly estimate
4 vCPU + 16 GB memory	$107.72
40 GB SSD persistent disk	$7.48
Total	$115.20



Account: Kairuddin Kerimbay (kairuddin.darasolutions@gmail.com)

VM instances

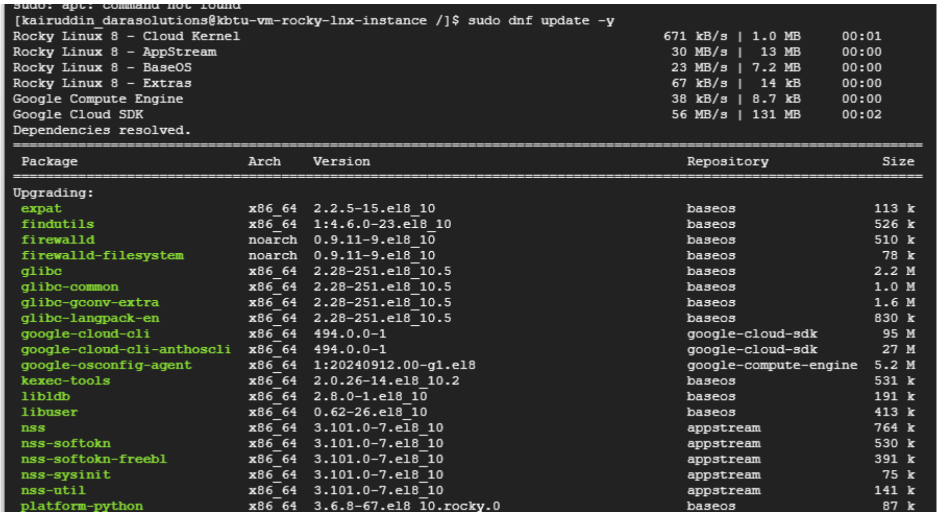
Status	Name	Zone	Recommendations	In use by	Internal IP	External IP	Connect
✓	kbtu-vm-rocky-lnx-instance	europe-north1-a			10.166.0.2 (nic0)	35.228.245.175 (nic0)	SSH

Related actions

Connect to the VM using SSH and install a basic web server (e.g., Apache or Nginx)

Firstly, upgraded system:

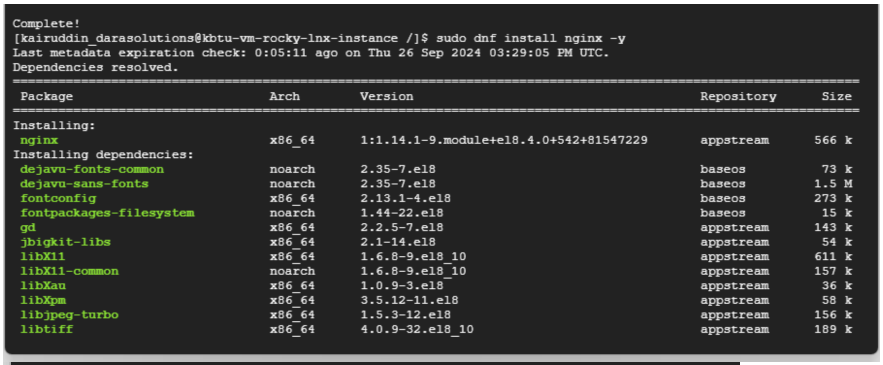
sudo dnf update -y



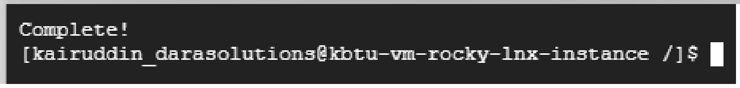
```
sudo: apt: command not found
[kairuddin_darasolutions@kbtu-vm-rocky-lnx-instance /]$ sudo dnf update -y
Rocky Linux 8 - Cloud Kernel                                              671 kB/s | 1.0 MB     00:01
Rocky Linux 8 - AppStream                                                  30 MB/s |  13 MB     00:00
Rocky Linux 8 - BaseOS                                                     23 MB/s | 7.2 MB     00:00
Rocky Linux 8 - Extras                                                     67 kB/s |  14 kB     00:00
Google Compute Engine                                                      38 kB/s | 8.7 kB     00:00
Google Cloud SDK                                                           56 MB/s | 131 MB     00:02
Dependencies resolved.
=========================================================================================================
 Package                      Arch    Version                                 Repository             Size
=========================================================================================================
Upgrading:
 expat                        x86_64  2.2.5-15.el8_10                         baseos                113 k
 findutils                    x86_64  1:4.6.0-23.el8_10                       baseos                526 k
 firewalld                    noarch  0.9.11-9.el8_10                         baseos                510 k
 firewalld-filesystem         noarch  0.9.11-9.el8_10                         baseos                 78 k
 glibc                        x86_64  2.28-251.el8_10.5                       baseos                2.2 M
 glibc-common                 x86_64  2.28-251.el8_10.5                       baseos                1.0 M
 glibc-gconv-extra            x86_64  2.28-251.el8_10.5                       baseos                1.6 M
 glibc-langpack-en            x86_64  2.28-251.el8_10.5                       baseos                830 k
 google-cloud-cli             x86_64  494.0.0-1                               google-cloud-sdk       95 M
 google-cloud-cli-anthoscli   x86_64  494.0.0-1                               google-cloud-sdk       27 M
 google-osconfig-agent        x86_64  1:20240912.00-g1.el8                    google-compute-engine 5.2 M
 kexec-tools                  x86_64  2.0.26-14.el8_10.2                      baseos                531 k
 libldb                       x86_64  2.8.0-1.el8_10                          baseos                191 k
 libuser                      x86_64  0.62-26.el8_10                          baseos                413 k
 nss                          x86_64  3.101.0-7.el8_10                        appstream             764 k
 nss-softokn                  x86_64  3.101.0-7.el8_10                        appstream             530 k
 nss-softokn-freebl           x86_64  3.101.0-7.el8_10                        appstream             391 k
 nss-sysinit                  x86_64  3.101.0-7.el8_10                        appstream              75 k
 nss-util                     x86_64  3.101.0-7.el8_10                        appstream             141 k
 platform-python              x86_64  3.6.8-67.el8_10.rocky.0                 baseos                 87 k
```

Then installed nginx

sudo dnf install nginx -y

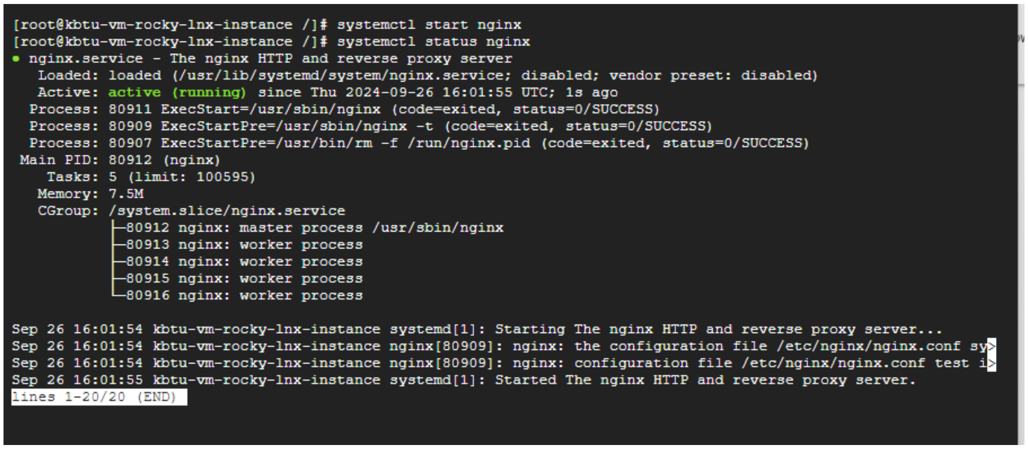


```
Complete!
[kairuddin_darasolutions@kbtu-vm-rocky-lnx-instance /]$ sudo dnf install nginx -y
Last metadata expiration check: 0:05:11 ago on Thu 26 Sep 2024 03:29:05 PM UTC.
Dependencies resolved.
==========================================================================================================
 Package                      Arch        Version                                    Repository      Size
==========================================================================================================
Installing:
 nginx                        x86_64      1:1.14.1-9.module+el8.4.0+542+81547229     appstream      566 k
Installing dependencies:
 dejavu-fonts-common          noarch      2.35-7.el8                                 baseos          73 k
 dejavu-sans-fonts            noarch      2.35-7.el8                                 baseos         1.5 M
 fontconfig                   x86_64      2.13.1-4.el8                               baseos         273 k
 fontpackages-filesystem      noarch      1.44-22.el8                                baseos          15 k
 gd                           x86_64      2.2.5-7.el8                                appstream      143 k
 jbigkit-libs                 x86_64      2.1-14.el8                                 appstream       54 k
 libX11                       x86_64      1.6.8-9.el8_10                             appstream      611 k
 libX11-common                noarch      1.6.8-9.el8_10                             appstream      157 k
 libXau                       x86_64      1.0.9-3.el8                                appstream       36 k
 libXpm                       x86_64      3.5.12-11.el8                              appstream       58 k
 libjpeg-turbo                x86_64      1.5.3-12.el8                               appstream      156 k
 libtiff                      x86_64      4.0.9-32.el8_10                            appstream      189 k
```

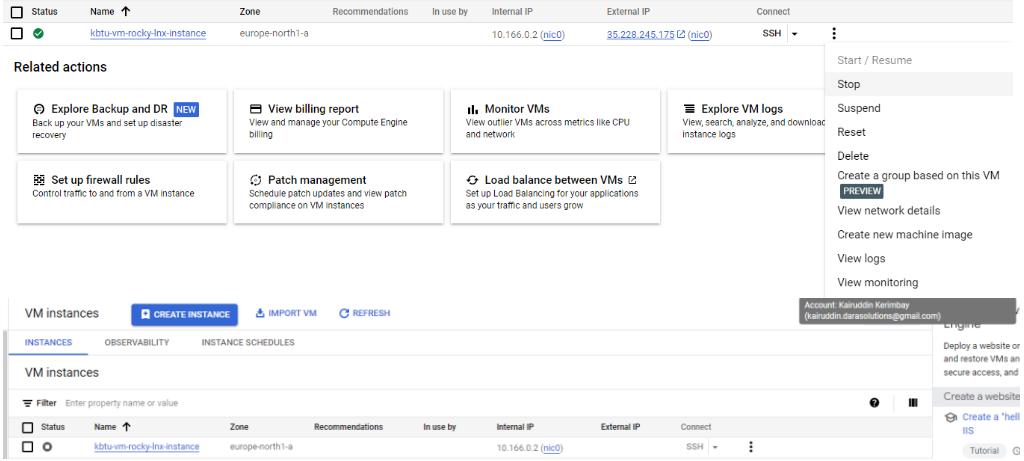


```
Complete!
[kairuddin_darasolutions@kbtu-vm-rocky-lnx-instance /]$
```

Here started nginx



```
[root@kbtu-vm-rocky-lnx-instance /]# systemctl start nginx
[root@kbtu-vm-rocky-lnx-instance /]# systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; disabled; vendor preset: disabled)
   Active: active (running) since Thu 2024-09-26 16:01:55 UTC; 1s ago
  Process: 80911 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
  Process: 80909 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
  Process: 80907 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
 Main PID: 80912 (nginx)
    Tasks: 5 (limit: 100595)
   Memory: 7.5M
   CGroup: /system.slice/nginx.service
           ├─80912 nginx: master process /usr/sbin/nginx
           ├─80913 nginx: worker process
           ├─80914 nginx: worker process
           ├─80915 nginx: worker process
           └─80916 nginx: worker process

Sep 26 16:01:54 kbtu-vm-rocky-lnx-instance systemd[1]: Starting The nginx HTTP and reverse proxy server...
Sep 26 16:01:54 kbtu-vm-rocky-lnx-instance nginx[80909]: nginx: the configuration file /etc/nginx/nginx.conf sy>
Sep 26 16:01:54 kbtu-vm-rocky-lnx-instance nginx[80909]: nginx: configuration file /etc/nginx/nginx.conf test i>
Sep 26 16:01:55 kbtu-vm-rocky-lnx-instance systemd[1]: Started The nginx HTTP and reverse proxy server.
lines 1-20/20 (END)
```



□	Status	Name ↑	Zone	Recommendations	In use by	Internal IP	External IP	Connect
□	✓	kbtu-vm-rocky-lnx-instance	europe-north1-a			10.166.0.2 (nic0)	35.228.245.175 (nic0)	SSH

Related actions

Explore Backup and DR NEW
Back up your VMs and set up disaster recovery

View billing report
View and manage your Compute Engine billing

Monitor VMs
View outlier VMs across metrics like CPU and network

Explore VM logs
View, search, analyze, and download instance logs

Set up firewall rules
Control traffic to and from a VM instance

Patch management
Schedule patch updates and view patch compliance on VM instances

Load balance between VMs
Set up Load Balancing for your applications as your traffic and users grow

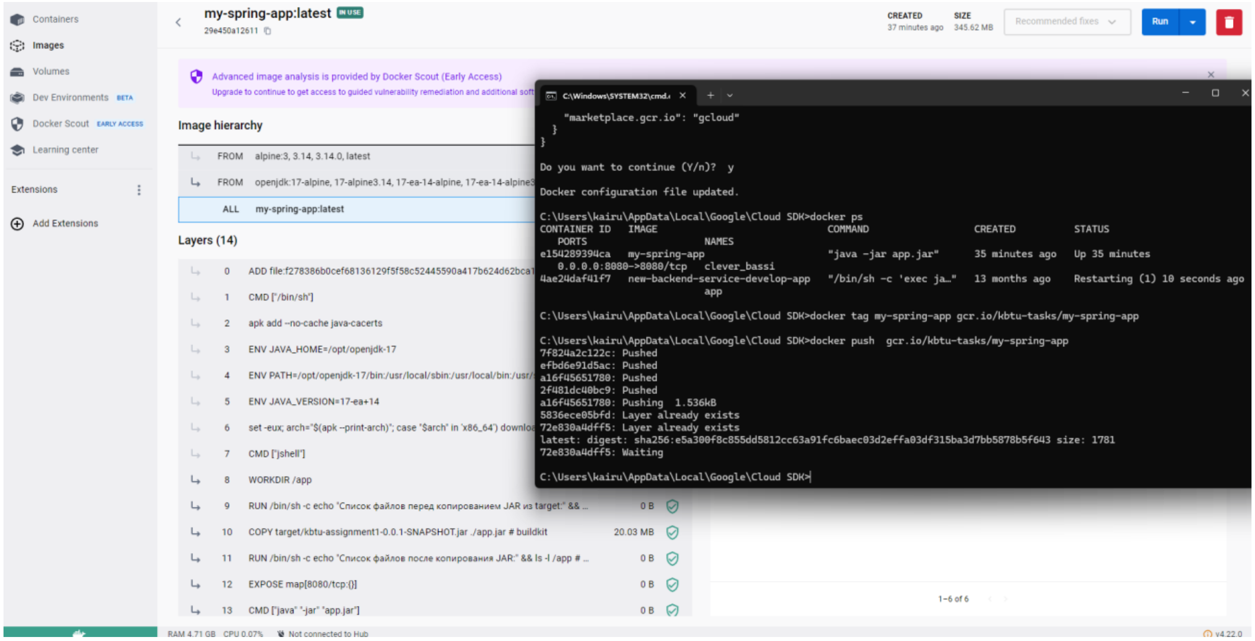
VM instances

□	Status	Name ↑	Zone	Recommendations	In use by	Internal IP	External IP	Connect
□	○	kbtu-vm-rocky-lnx-instance	europe-north1-a			10.166.0.2 (nic0)		SSH

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

Commands used:

gcloud auth configure-docker
docker tag my-spring-app gcr.io/kbtu-tasks/my-spring-app
docker push gcr.io/kbtu-tasks/my-spring-app



my-spring-app:latest
29e450a12611

CREATED 37 minutes ago SIZE 345.62 MB

Advanced image analysis is provided by Docker Scout (Early Access)
Upgrade to continue to get access to guided vulnerability remediation and additional soft

Image hierarchy

FROM	alpine:3, 3.14, 3.14.0, latest
FROM	openjdk:17-alpine, 17-alpine3.14, 17-ea-14-alpine, 17-ea-14-alpine3
ALL	my-spring-app:latest

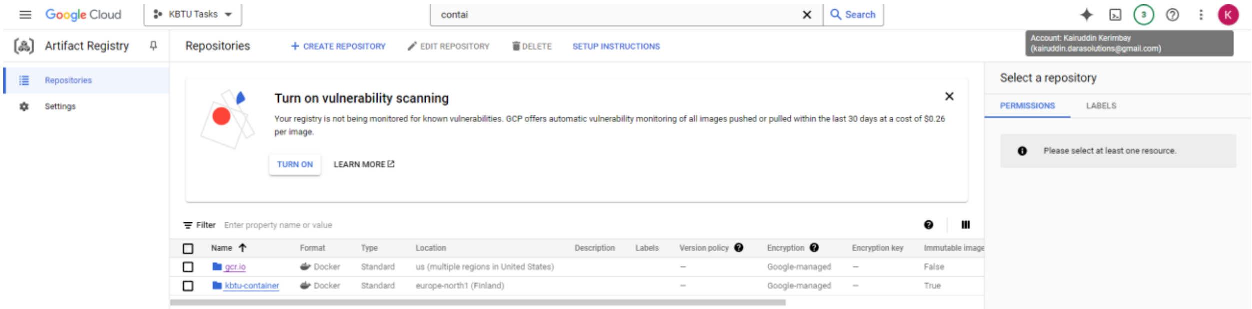
Layers (14)

0	ADD file:f278386b0cef68136129f5f58c52445590a417b624d62bca1	
1	CMD ["/bin/sh"]	
2	apk add --no-cache java-cacerts	
3	ENV JAVA_HOME=/opt/openjdk-17	
4	ENV PATH=/opt/openjdk-17/bin:/usr/local/sbin:/usr/local/bin:/usr/	
5	ENV JAVA_VERSION=17-ea+14	
6	set -eux; arch="$(apk --print-arch)"; case "$arch" in 'x86_64') downlo	
7	CMD ["jshell"]	
8	WORKDIR /app	
9	RUN /bin/sh -c echo "Список файлов перед копированием JAR из target:" && ...	0 B
10	COPY target/kbtu-assignment1-0.0.1-SNAPSHOT.jar ./app.jar # buildkit	20.03 MB
11	RUN /bin/sh -c echo "Список файлов после копирования JAR:" && ls -l /app # ...	0 B
12	EXPOSE map[8080/tcp:{}]	0 B
13	CMD ["java" "-jar" "app.jar"]	0 B

1–6 of 6

C:\Windows\SYSTEM32\cmd.

```
    "marketplace.gcr.io": "gcloud"
  }
}

Do you want to continue (Y/n)?  y

Docker configuration file updated.

C:\Users\kairu\AppData\Local\Google\Cloud SDK>docker ps
CONTAINER ID   IMAGE                            COMMAND                  CREATED          STATUS
   PORTS                    NAMES
e154289394ca   my-spring-app                    "java -jar app.jar"      35 minutes ago   Up 35 minutes
   0.0.0.0:8080->8080/tcp   clever_bassi
4ae24daf41f7   new-backend-service-develop-app   "/bin/sh -c 'exec ja…"   13 months ago    Restarting (1) 10 seconds ago
                            app

C:\Users\kairu\AppData\Local\Google\Cloud SDK>docker tag my-spring-app gcr.io/kbtu-tasks/my-spring-app

C:\Users\kairu\AppData\Local\Google\Cloud SDK>docker push  gcr.io/kbtu-tasks/my-spring-app
7f824a2c122c: Pushed
efbd6e91d5ac: Pushed
a16f45651780: Pushed
2f481dc40bc9: Pushed
a16f45651780: Pushing  1.536kB
5836ece05bfd: Layer already exists
72e830a4dff5: Layer already exists
latest: digest: sha256:e5a300f8c855dd5812cc63a91fc6baec03d2effa03df315ba3d7bb5878b5f643 size: 1781
72e830a4dff5: Waiting

C:\Users\kairu\AppData\Local\Google\Cloud SDK>
```

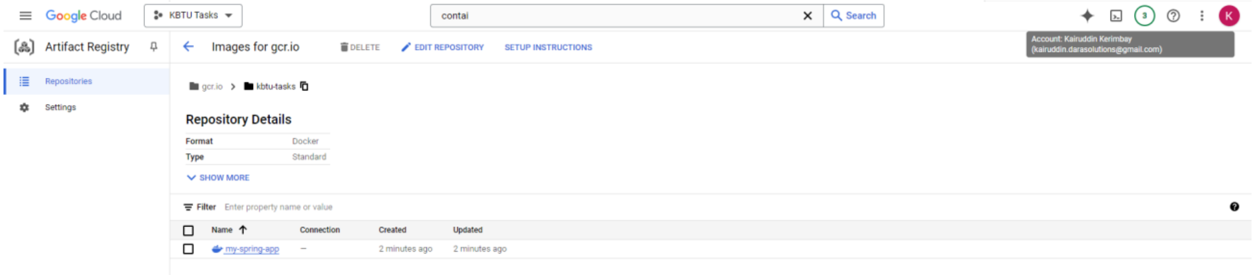


Turn on vulnerability scanning

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.

Please select at least one resource.

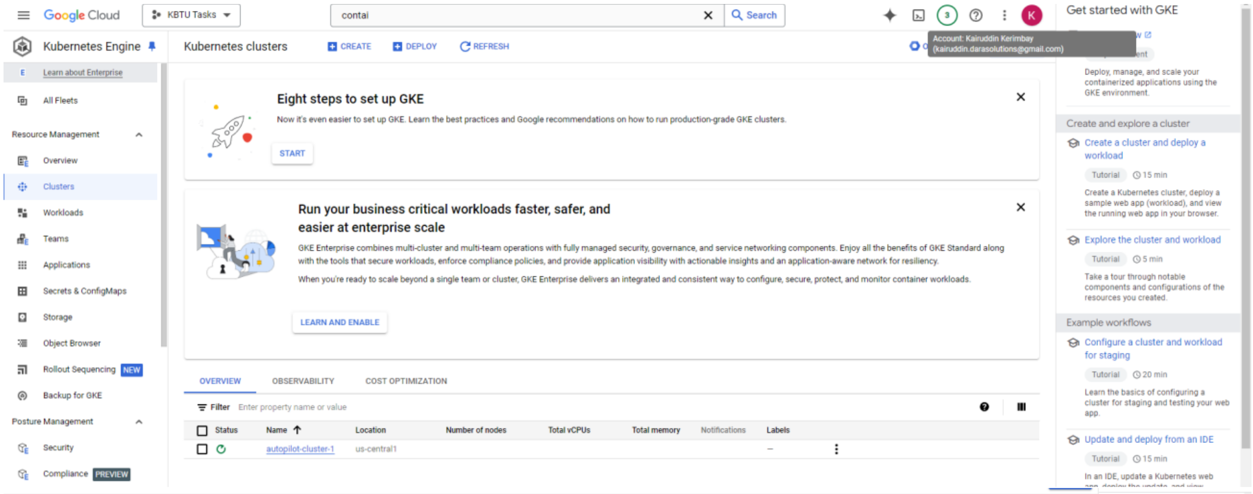
	Name ↑	Format	Type	Location	Description	Labels	Version policy	Encryption	Encryption key	Immutable image
□	gcr.io	Docker	Standard	us (multiple regions in United States)			–	Google-managed	–	False
□	kbtu-container	Docker	Standard	europe-north1 (Finland)			–	Google-managed	–	True



Repository Details

Format	Docker
Type	Standard

	Name ↑	Connection	Created	Updated
□	my-spring-app	–	2 minutes ago	2 minutes ago



Eight steps to set up GKE

Now it's even easier to set up GKE. Learn the best practices and Google recommendations on how to run production-grade GKE clusters.

Run your business critical workloads faster, safer, and easier at enterprise scale

GKE Enterprise combines multi-cluster and multi-team operations with fully managed security, governance, and service networking components. Enjoy all the benefits of GKE Standard along with the tools that secure workloads, enforce compliance policies, and provide application visibility with actionable insights and an application-aware network for resiliency.

When you're ready to scale beyond a single team or cluster, GKE Enterprise delivers an integrated and consistent way to configure, secure, protect, and monitor container workloads.

	Status	Name ↑	Location	Number of nodes	Total vCPUs	Total memory	Notifications	Labels
□	✓	autopilot-cluster-1	us-central1					–

Get started with GKE

Deploy, manage, and scale your containerized applications using the GKE environment.

Create and explore a cluster

Create a cluster and deploy a workload

Tutorial 15 min

Create a Kubernetes cluster, deploy a sample web app (workload), and view the running web app in your browser.

Explore the cluster and workload

Tutorial 5 min

Take a tour through notable components and configurations of the resources you created.

Example workflows

Configure a cluster and workload for staging

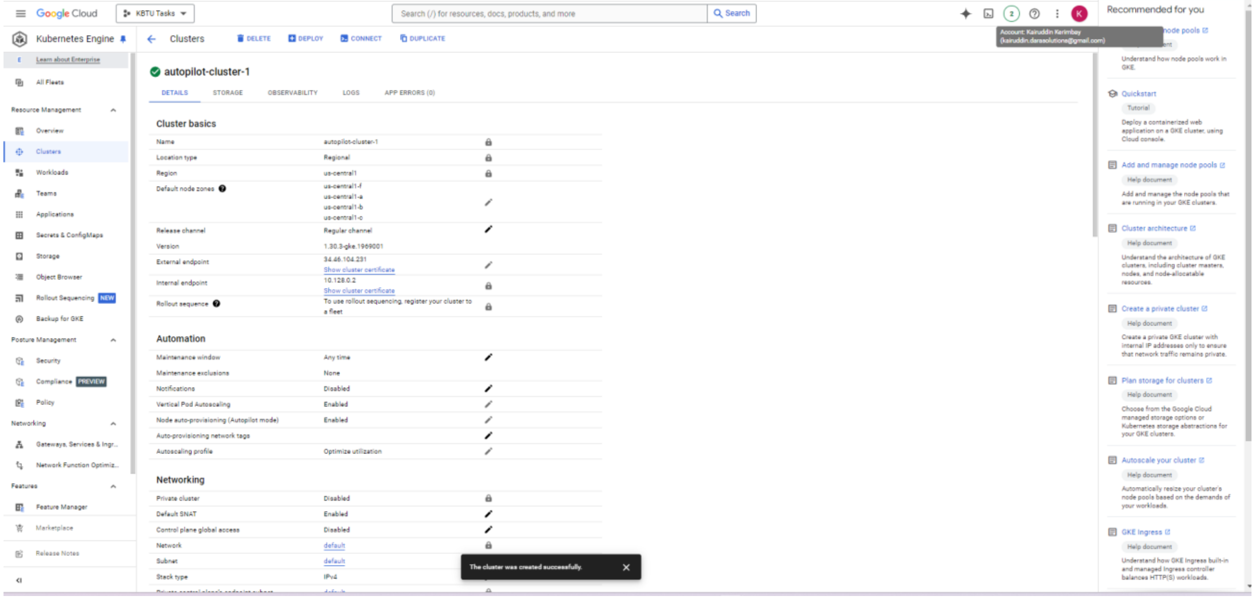
Tutorial 20 min

Learn the basics of configuring a cluster for staging and testing your web app.

Update and deploy from an IDE

Tutorial 15 min

In an IDE, update a Kubernetes web



autopilot-cluster-1

Cluster basics

Name	autopilot-cluster-1
Location type	Regional
Region	us-central1
Default node zones	us-central1-f us-central1-a us-central1-b us-central1-c
Release channel	Regular channel
Version	1.30.3-gke.1969001
External endpoint	34.46.104.231 Show cluster certificate
Internal endpoint	10.128.0.2 Show cluster certificate
Rollout sequence	To use rollout sequencing, register your cluster to a fleet

Automation

Maintenance window	Any time
Maintenance exclusions	None
Notifications	Disabled
Vertical Pod Autoscaling	Enabled
Node auto-provisioning (Autopilot mode)	Enabled
Auto-provisioning network tags	
Autoscaling profile	Optimize utilization

Networking

Private cluster	Disabled
Default SNAT	Enabled
Control plane global access	Disabled
Network	default
Subnet	default
Stack type	IPv4

The cluster was created successfully.

Recommended for you

node pools

Understand how node pools work in GKE.

Quickstart

Tutorial

Deploy a containerized web application on a GKE cluster, using Cloud console.

Add and manage node pools

Help document

Add and manage the node pools that are running in your GKE clusters.

Cluster architecture

Help document

Understand the architecture of GKE clusters, including cluster masters, nodes, and node-allocatable resources.

Create a private cluster

Help document

Create a private GKE cluster with internal IP addresses only to ensure that network traffic remains private.

Plan storage for clusters

Help document

Choose from the Google Cloud managed storage options or Kubernetes storage abstractions for your GKE clusters.

Autoscale your cluster

Help document

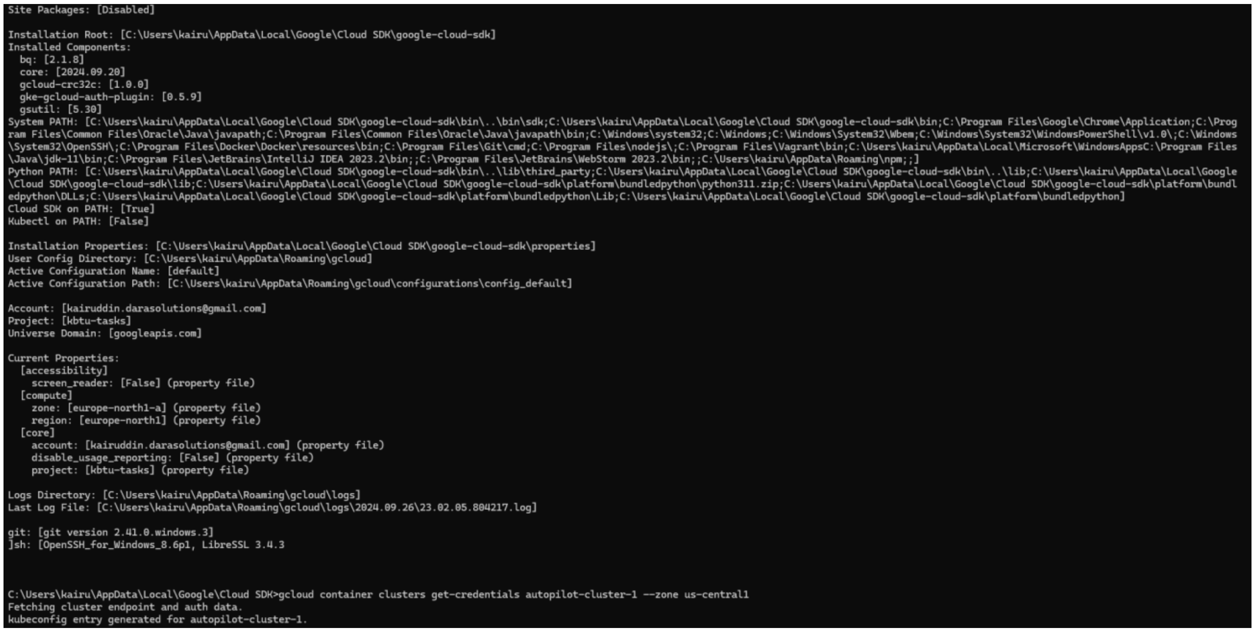
Automatically resize your cluster's node pools based on the demands of your workloads.

GKE Ingress

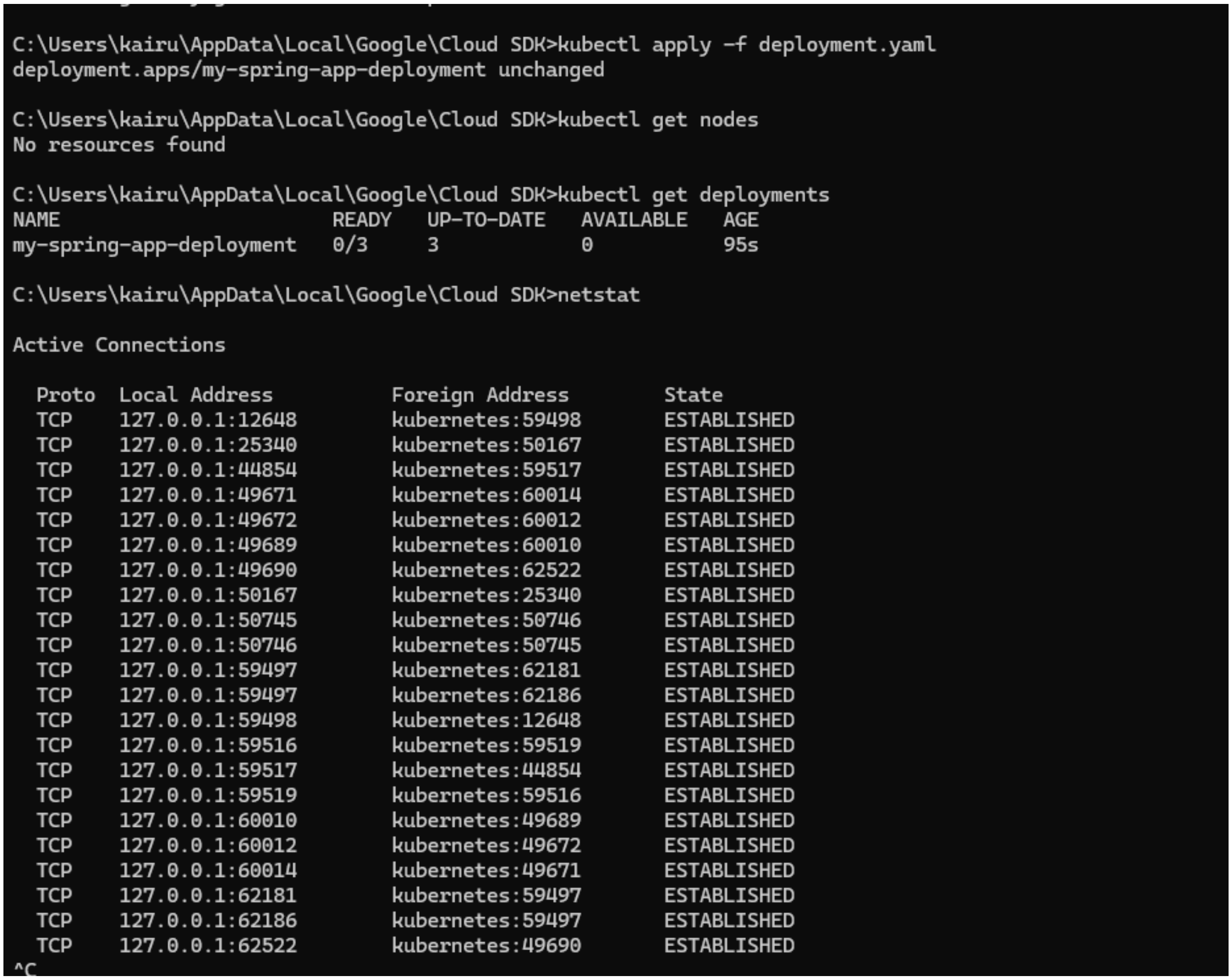
Help document

Understand how GKE Ingress built-in and managed Ingress controller balances HTTP(S) workloads.

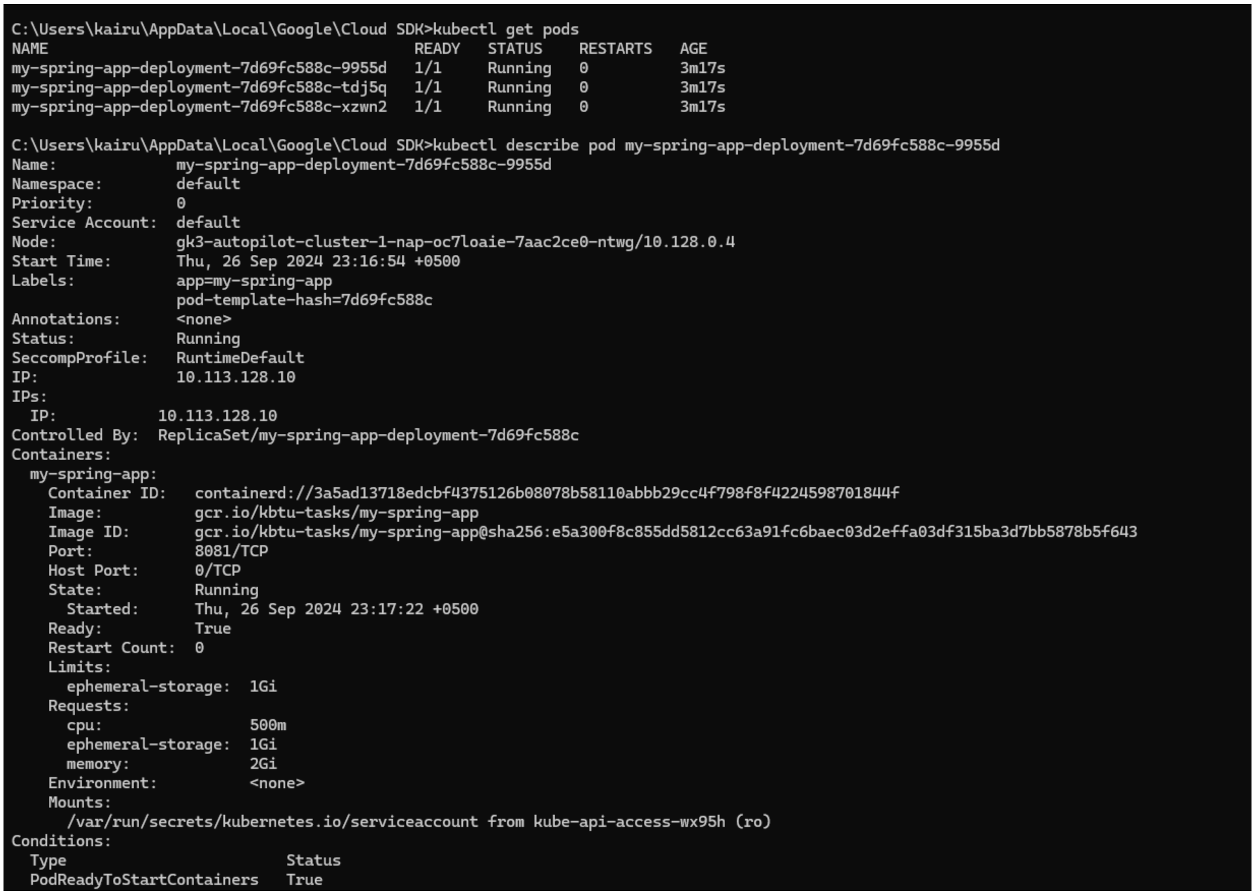
Create a GKE cluster in Google Cloud Console.



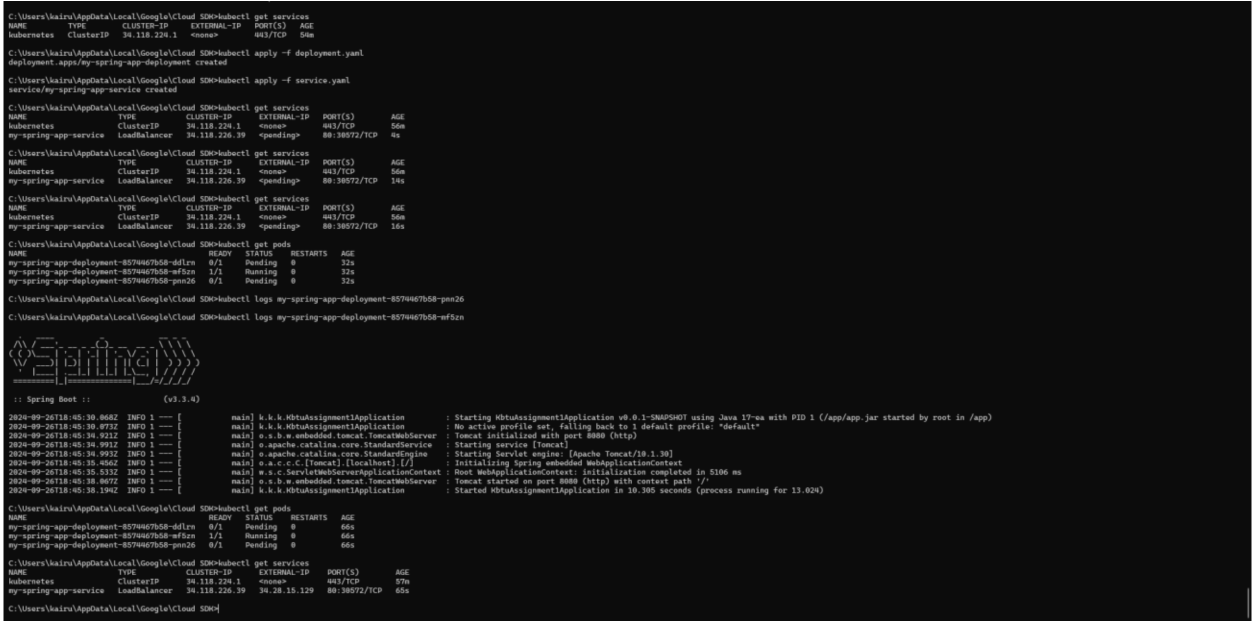
```
Site Packages: [Disabled]

Installation Root: [C:\Users\kairu\AppData\Local\Google\Cloud SDK\google-cloud-sdk]
Installed Components:
  bq: [2.1.8]
  core: [2024.09.20]
  gcloud-crc32c: [1.0.0]
  gke-gcloud-auth-plugin: [0.5.9]
  gsutil: [5.30]
System PATH: [C:\Users\kairu\AppData\Local\Google\Cloud SDK\google-cloud-sdk\bin\..\bin\sdk;C:\Users\kairu\AppData\Local\Google\Cloud SDK\google-cloud-sdk\bin;C:\Program Files\Google\Chrome\Application;C:\Prog
ram Files\Common Files\Oracle\Java\javapath;C:\Program Files\Common Files\Oracle\Java\javapath\bin;C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Windows
\System32\OpenSSH\;C:\Program Files\Docker\Docker\resources\bin;C:\Program Files\Git\cmd;C:\Program Files\nodejs\;C:\Program Files\Vagrant\bin;C:\Users\kairu\AppData\Local\Microsoft\WindowsAppsC:\Program Files
\Java\jdk-11\bin;C:\Program Files\JetBrains\IntelliJ IDEA 2023.2\bin;;C:\Program Files\JetBrains\WebStorm 2023.2\bin;;C:\Users\kairu\AppData\Roaming\npm;;]
Python PATH: [C:\Users\kairu\AppData\Local\Google\Cloud SDK\google-cloud-sdk\bin\..\lib\third_party;C:\Users\kairu\AppData\Local\Google\Cloud SDK\google-cloud-sdk\bin\..\lib;C:\Users\kairu\AppData\Local\Google
\Cloud SDK\google-cloud-sdk\lib;C:\Users\kairu\AppData\Local\Google\Cloud SDK\google-cloud-sdk\platform\bundledpython\python311.zip;C:\Users\kairu\AppData\Local\Google\Cloud SDK\google-cloud-sdk\platform\bundl
edpython\DLLs;C:\Users\kairu\AppData\Local\Google\Cloud SDK\google-cloud-sdk\platform\bundledpython\Lib;C:\Users\kairu\AppData\Local\Google\Cloud SDK\google-cloud-sdk\platform\bundledpython]
Cloud SDK on PATH: [True]
Kubectl on PATH: [False]

Installation Properties: [C:\Users\kairu\AppData\Local\Google\Cloud SDK\google-cloud-sdk\properties]
User Config Directory: [C:\Users\kairu\AppData\Roaming\gcloud]
Active Configuration Name: [default]
Active Configuration Path: [C:\Users\kairu\AppData\Roaming\gcloud\configurations\config_default]

Account: [kairuddin.darasolutions@gmail.com]
Project: [kbtu-tasks]
Universe Domain: [googleapis.com]

Current Properties:
  [accessibility]
    screen_reader: [False] (property file)
  [compute]
    zone: [europe-north1-a] (property file)
    region: [europe-north1] (property file)
  [core]
    account: [kairuddin.darasolutions@gmail.com] (property file)
    disable_usage_reporting: [False] (property file)
    project: [kbtu-tasks] (property file)

Logs Directory: [C:\Users\kairu\AppData\Roaming\gcloud\logs]
Last Log File: [C:\Users\kairu\AppData\Roaming\gcloud\logs\2024.09.26\23.02.05.804217.log]

git: [git version 2.41.0.windows.3]
]sh: [OpenSSH_for_Windows_8.6p1, LibreSSL 3.4.3



C:\Users\kairu\AppData\Local\Google\Cloud SDK>gcloud container clusters get-credentials autopilot-cluster-1 --zone us-central1
Fetching cluster endpoint and auth data.
kubeconfig entry generated for autopilot-cluster-1.
```



```
C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl apply -f deployment.yaml
deployment.apps/my-spring-app-deployment unchanged

C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl get nodes
No resources found

C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl get deployments
NAME                       READY   UP-TO-DATE   AVAILABLE   AGE
my-spring-app-deployment   0/3     3            0           95s

C:\Users\kairu\AppData\Local\Google\Cloud SDK>netstat

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    127.0.0.1:12648        kubernetes:59498       ESTABLISHED
  TCP    127.0.0.1:25340        kubernetes:50167       ESTABLISHED
  TCP    127.0.0.1:44854        kubernetes:59517       ESTABLISHED
  TCP    127.0.0.1:49671        kubernetes:60014       ESTABLISHED
  TCP    127.0.0.1:49672        kubernetes:60012       ESTABLISHED
  TCP    127.0.0.1:49689        kubernetes:60010       ESTABLISHED
  TCP    127.0.0.1:49690        kubernetes:62522       ESTABLISHED
  TCP    127.0.0.1:50167        kubernetes:25340       ESTABLISHED
  TCP    127.0.0.1:50745        kubernetes:50746       ESTABLISHED
  TCP    127.0.0.1:50746        kubernetes:50745       ESTABLISHED
  TCP    127.0.0.1:59497        kubernetes:62181       ESTABLISHED
  TCP    127.0.0.1:59497        kubernetes:62186       ESTABLISHED
  TCP    127.0.0.1:59498        kubernetes:12648       ESTABLISHED
  TCP    127.0.0.1:59516        kubernetes:59519       ESTABLISHED
  TCP    127.0.0.1:59517        kubernetes:44854       ESTABLISHED
  TCP    127.0.0.1:59519        kubernetes:59516       ESTABLISHED
  TCP    127.0.0.1:60010        kubernetes:49689       ESTABLISHED
  TCP    127.0.0.1:60012        kubernetes:49672       ESTABLISHED
  TCP    127.0.0.1:60014        kubernetes:49671       ESTABLISHED
  TCP    127.0.0.1:62181        kubernetes:59497       ESTABLISHED
  TCP    127.0.0.1:62186        kubernetes:59497       ESTABLISHED
  TCP    127.0.0.1:62522        kubernetes:49690       ESTABLISHED
^C
```

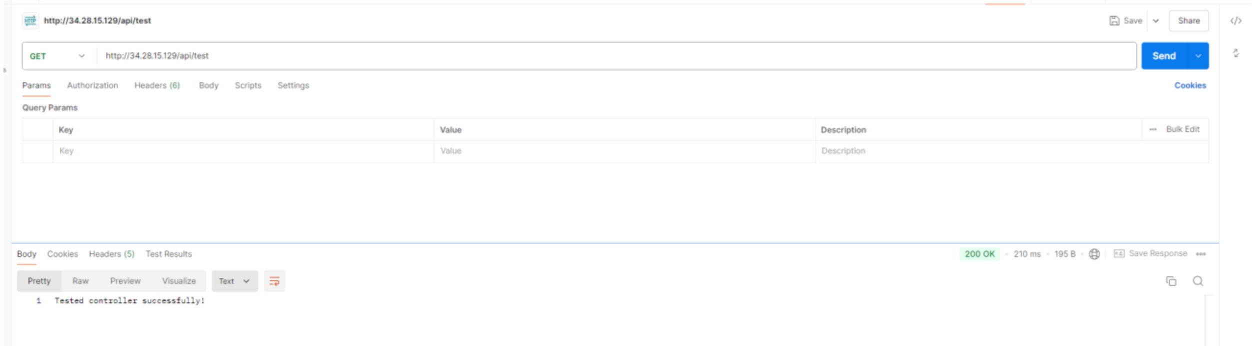


```
C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl get pods
NAME                                     READY   STATUS    RESTARTS   AGE
my-spring-app-deployment-7d69fc588c-9955d   1/1     Running   0          3m17s
my-spring-app-deployment-7d69fc588c-tdj5q   1/1     Running   0          3m17s
my-spring-app-deployment-7d69fc588c-xzwn2   1/1     Running   0          3m17s

C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl describe pod my-spring-app-deployment-7d69fc588c-9955d
Name:             my-spring-app-deployment-7d69fc588c-9955d
Namespace:        default
Priority:         0
Service Account:  default
Node:             gk3-autopilot-cluster-1-nap-oc7loaie-7aac2ce0-ntwg/10.128.0.4
Start Time:       Thu, 26 Sep 2024 23:16:54 +0500
Labels:           app=my-spring-app
                  pod-template-hash=7d69fc588c
Annotations:      <none>
Status:           Running
SeccompProfile:   RuntimeDefault
IP:               10.113.128.10
IPs:
  IP:           10.113.128.10
Controlled By:  ReplicaSet/my-spring-app-deployment-7d69fc588c
Containers:
  my-spring-app:
    Container ID:   containerd://3a5ad13718edcbf4375126b08078b58110abbb29cc4f798f8f4224598701844f
    Image:          gcr.io/kbtu-tasks/my-spring-app
    Image ID:       gcr.io/kbtu-tasks/my-spring-app@sha256:e5a300f8c855dd5812cc63a91fc6baec03d2effa03df315ba3d7bb5878b5f643
    Port:           8081/TCP
    Host Port:      0/TCP
    State:          Running
      Started:      Thu, 26 Sep 2024 23:17:22 +0500
    Ready:          True
    Restart Count:  0
    Limits:
      ephemeral-storage:  1Gi
    Requests:
      cpu:                500m
      ephemeral-storage:  1Gi
      memory:             2Gi
    Environment:          <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-wx95h (ro)
Conditions:
  Type                        Status
  PodReadyToStartContainers   True
```



```
C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl get services
NAME         TYPE        CLUSTER-IP    EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP   34.118.224.1   <none>        443/TCP   54m

C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl apply -f deployment.yaml
deployment.apps/my-spring-app-deployment created

C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl apply -f service.yaml
service/my-spring-app-service created

C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl get services
NAME                    TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
kubernetes              ClusterIP      34.118.224.1    <none>        443/TCP        56m
my-spring-app-service   LoadBalancer   34.118.226.39   <pending>     80:30572/TCP   4s

C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl get services
NAME                    TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
kubernetes              ClusterIP      34.118.224.1    <none>        443/TCP        56m
my-spring-app-service   LoadBalancer   34.118.226.39   <pending>     80:30572/TCP   14s

C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl get services
NAME                    TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
kubernetes              ClusterIP      34.118.224.1    <none>        443/TCP        56m
my-spring-app-service   LoadBalancer   34.118.226.39   <pending>     80:30572/TCP   16s

C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl get pods
NAME                                        READY   STATUS    RESTARTS   AGE
my-spring-app-deployment-8574467b58-ddlrn   0/1     Pending   0          32s
my-spring-app-deployment-8574467b58-mf5zn   1/1     Running   0          32s
my-spring-app-deployment-8574467b58-pnn26   0/1     Pending   0          32s

C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl logs my-spring-app-deployment-8574467b58-pnn26

C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl logs my-spring-app-deployment-8574467b58-mf5zn

  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/

 :: Spring Boot ::                (v3.3.4)

2024-09-26T18:45:30.068Z  INFO 1 --- [           main] k.k.k.KbtuAssignment1Application         : Starting KbtuAssignment1Application v0.0.1-SNAPSHOT using Java 17-ea with PID 1 (/app/app.jar started by root in /app)
2024-09-26T18:45:30.073Z  INFO 1 --- [           main] k.k.k.KbtuAssignment1Application         : No active profile set, falling back to 1 default profile: "default"
2024-09-26T18:45:34.921Z  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port 8080 (http)
2024-09-26T18:45:34.991Z  INFO 1 --- [           main] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2024-09-26T18:45:34.993Z  INFO 1 --- [           main] o.apache.catalina.core.StandardEngine    : Starting Servlet engine: [Apache Tomcat/10.1.30]
2024-09-26T18:45:35.456Z  INFO 1 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationContext
2024-09-26T18:45:35.533Z  INFO 1 --- [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 5106 ms
2024-09-26T18:45:38.067Z  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port 8080 (http) with context path '/'
2024-09-26T18:45:38.194Z  INFO 1 --- [           main] k.k.k.KbtuAssignment1Application         : Started KbtuAssignment1Application in 10.305 seconds (process running for 13.024)

C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl get pods
NAME                                        READY   STATUS    RESTARTS   AGE
my-spring-app-deployment-8574467b58-ddlrn   0/1     Pending   0          66s
my-spring-app-deployment-8574467b58-mf5zn   1/1     Running   0          66s
my-spring-app-deployment-8574467b58-pnn26   0/1     Pending   0          66s

C:\Users\kairu\AppData\Local\Google\Cloud SDK>kubectl get services
NAME                    TYPE           CLUSTER-IP      EXTERNAL-IP    PORT(S)        AGE
kubernetes              ClusterIP      34.118.224.1    <none>         443/TCP        57m
my-spring-app-service   LoadBalancer   34.118.226.39   34.28.15.129   80:30572/TCP   65s

C:\Users\kairu\AppData\Local\Google\Cloud SDK>
```

How did you verify that your application was successfully deployed and accessible?



http://34.28.15.129/api/test

GET http://34.28.15.129/api/test Send

Query Params

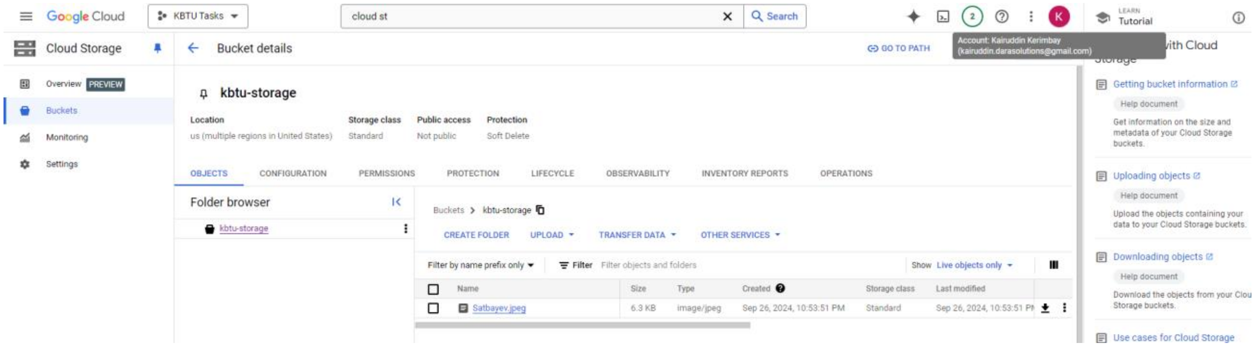
Key	Value	Description
Key	Value	Description

200 OK · 210 ms · 195 B

```
1  Tested controller successfully!
```

CONGRATS 🎉🎉🎉

Exercise 5: Storing and Accessing Data in Google Cloud Storage



Bucket details

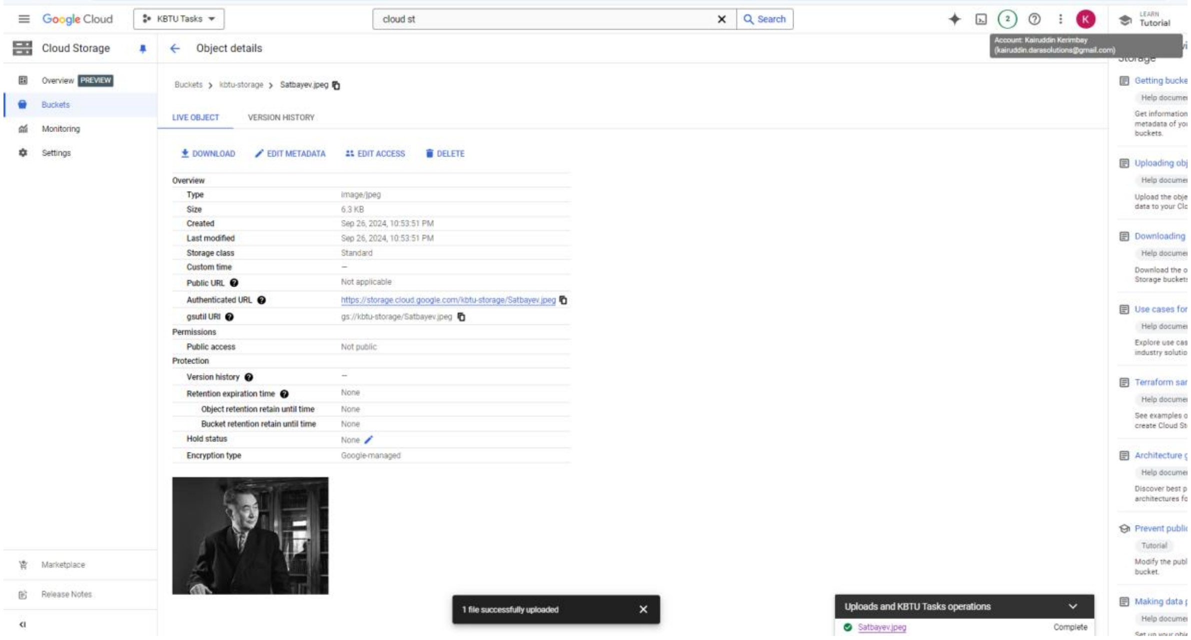
kbtu-storage

Location	Storage class	Public access	Protection
us (multiple regions in United States)	Standard	Not public	Soft Delete

Buckets > kbtu-storage

Filter by name prefix only ▾ Filter Filter objects and folders Show Live objects only ▾

□	Name	Size	Type	Created	Storage class	Last modified
□	Satbayev.jpeg	6.3 KB	image/jpeg	Sep 26, 2024, 10:53:51 PM	Standard	Sep 26, 2024, 10:53:51 PI



Object details

Buckets > kbtu-storage > Satbayev.jpeg

Overview	
Type	image/jpeg
Size	6.3 KB
Created	Sep 26, 2024, 10:53:51 PM
Last modified	Sep 26, 2024, 10:53:51 PM
Storage class	Standard
Custom time	—
Public URL	Not applicable
Authenticated URL	https://storage.cloud.google.com/kbtu-storage/Satbayev.jpeg
gsutil URI	gs://kbtu-storage/Satbayev.jpeg
Permissions	
Public access	Not public
Protection	
Version history	—
Retention expiration time	None
Object retention retain until time	None
Bucket retention retain until time	None
Hold status	None
Encryption type	Google-managed

1 file successfully uploaded

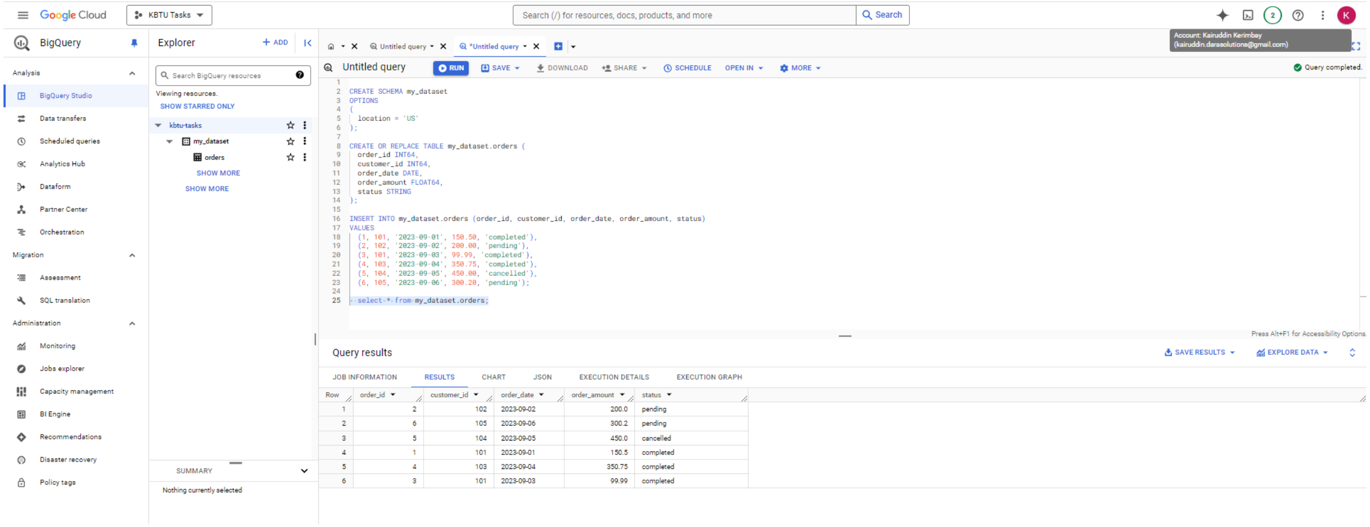
Uploads and KBTU Tasks operations

Satbayev.jpeg Complete

Exercise 6: Analyzing Data with BigQuery

```
CREATE SCHEMA my_dataset
OPTIONS
(
  location = 'US'
);

CREATE OR REPLACE TABLE my_dataset.orders (
  order_id INT64,
  customer_id INT64,
  order_date DATE,
  order_amount FLOAT64,
  status STRING
);

INSERT INTO my_dataset.orders (order_id, customer_id, order_date, order_amount, status)
VALUES
  (1, 101, '2023-09-01', 150.50, 'completed'),
  (2, 102, '2023-09-02', 200.00, 'pending'),
  (3, 101, '2023-09-03', 99.99, 'completed'),
  (4, 103, '2023-09-04', 350.75, 'completed'),
  (5, 104, '2023-09-05', 450.00, 'cancelled'),
  (6, 105, '2023-09-06', 300.20, 'pending');
```

```
SELECT * from my_dataset.orders;


SELECT * FROM my_dataset.orders WHERE status = 'completed';

SELECT status, COUNT(*) AS order_count FROM my_dataset.orders
GROUP BY status
ORDER BY order_count DESC;
```



```

CREATE SCHEMA my_dataset
OPTIONS
(
  location = 'US'
);

CREATE OR REPLACE TABLE my_dataset.orders (
  order_id INT64,
  customer_id INT64,
  order_date DATE,
  order_amount FLOAT64,
  status STRING
);

INSERT INTO my_dataset.orders (order_id, customer_id, order_date, order_amount, status)
VALUES
  (1, 101, '2023-09-01', 150.50, 'completed'),
  (2, 102, '2023-09-02', 200.00, 'pending'),
  (3, 101, '2023-09-03', 99.99, 'completed'),
  (4, 103, '2023-09-04', 350.75, 'completed'),
  (5, 104, '2023-09-05', 450.00, 'cancelled'),
  (6, 105, '2023-09-06', 300.20, 'pending');

  select * from my_dataset.orders;
```

Row	order_id	customer_id	order_date	order_amount	status
1	2	102	2023-09-02	200.0	pending
2	6	105	2023-09-06	300.2	pending
3	5	104	2023-09-05	450.0	cancelled
4	1	101	2023-09-01	150.5	completed
5	4	103	2023-09-04	350.75	completed
6	3	101	2023-09-03	99.99	completed