GCP Assignment 1

1) What is the GCP project quota? If necessary, how GCP quota can be increased?

Google Cloud uses quotas to restrict how much of a particular shared Google Cloud resource that you can use. Each quota represents a specific countable resource for example API calls to a particular service, or the number of projects that you can create in GCP and many more things.

GCP Quota can be increased by -

Go to the Quotas page ---> find the quota you want to increase in the Limit name

column ---> Select the checkbox to the left of your quota ---> Click create edit quotas

--->In the quota changes form, enter the increased quota that you want for your

project in the new limit field.

if you are runnig out of GCP quota and need more then you can raise request to increase GCP quota.Most quota increase requests are evaluated by automated systems based on strict criteria, including the availability of resources, the length of time you've used Google Cloud, and other factors. Requests that don't meet specific criteria are denied. Evaluation criteria is not disclosed. In some cases, quota increase requests are escalated to human reviewers, who also follow strict criteria but can consider your unique request.

In some cases you might be asked to pay in advance for some quota increases facility. For example, you might be asked to make a payment if you request more projects that will use paid cloud services. Payment will be happen from your payment details added on the GCP.

Quota increase requests are handled by Cloud Customer Care , who process your request, typically within 2-3 business days. You are sent an email when your request is received, to which you can respond if you have questions or want to provide additional information about your request (for example, if your increase request is urgent). You then get another email after the request is processed, notifying you whether the quota increase was approved or denied. The email provides the effective date of the increase, if applicable.

2) What is the definition of a virtual machine? Virtual machine types are offered in GCP. How to create a virtual computer in Google Cloud Platform.

A virtual machine (VM) is a digital version of a physical computer.

There are 4 different types of VM offered by GCP.

1) General purpose

These machines balance price and performance and are suitable for most workloads including databases,

development and testing environments, web applications, and mobile gaming.

2) Compute-optimized

These machines provide the highest performance per core on Compute Engine and are optimized for compute-intensive workloads, such as high performance computing (HPC), game servers, and latency-sensitive API serving.

3) Memory-optimized

These machines offer the highest memory configurations across VM families with up to 12 TB for a single instance. They are well-suited for memory-intensive workloads such as large in-memory databases like SAP HANA and in-memory data analytics workloads.

4) Accelerator-optimized

These machines are based on the NVIDIA Ampere A100 Tensor Core GPU. With up to 16 GPUs in a single VM, these machines are suitable for demanding workloads like CUDA-enabled machine learning (ML) training and inference, and HPC.

How to create VM in GCP :

1. In the Google Cloud Console, go to the VM instances page.

2. Select your project and click Continue.

3. Click Create instance.

4. Specify a Name for your VM.

5. Optional: Change the Zone for this VM. Compute Engine randomizes the list of

zones within each region to encourage use across multiple zones.

6. Select a Machine configuration for your VM.

7. In the Boot disk section, click Change, and then do the following:

i) Select the Custom Images tab.

ii) To select the image project, click Select a project, and then do the following:

a. Select the project that contains the image.

b. Click Open.

c) In the Image list, click the image that you want to import.

d) Select the type and size of your boot disk.

e) Optional: For advanced configuration options, click Show advanced

configuration.

f) To confirm your boot disk options, click Select.

8. In the Firewall section, to permit HTTP or HTTPS traffic to the VM,

select Allow HTTP traffic or Allow HTTPS traffic.

The Cloud Console adds a network tag to your VM and creates the

corresponding ingress firewall rule that allows all incoming traffic

on tcp:80 (HTTP) or tcp:443 (HTTPS). The network tag associates the firewall

rule with the VM. For more information, see Firewall rules overview in the

Virtual Private Cloud documentation.

9. To create and start the VM, click Create.

3) What is Google Big Query, and how does it work? Replicate certain instances to demonstrate a use case.

Google Big Query is a Cloud Datawarehouse run by Google.   It is capable of analysing terabytes of data in seconds. if you know how to write SQl queries then you already know how to query it.

Google BigQuery Architecture uses column-based storage or columnar storage structure that helps it achieve faster query processing with fewer resources. It is the ma reason how Google BigQuery handles large datasets quantities and delivers excellent speed.

Row-based storage structure is used in Relational Databases where data is stored in rows because it is an efficient way for storing data for transactional Databases. Storing data in columns is efficient for analytical purposes because it needs a faster data reading speed.

Suppose a Database has 1000 records or 1000 columns of data. If we store data in a row-based structure, then querying only 10 rows out of 1000 will take more time as it will read all the 1000 rows to get 10 rows in the query output.

But this is not the case in Google BigQuery's Columnar Database, where all the data is stored in columns instead of rows.

The columnar database will process only 100 columns in the interest of the query, which in turn makes

the overall query processing faster.

Replication use cases

Name	Primary	Replica	Benefits and use cases	More information
Read replica	Cloud SQL instance	Cloud SQL instance	Additional read capacity Analytics target	Creating read replicas Managing read replicas Replication metrics Replication lag Read replica indexes
Cross-region read replica	Cloud SQL instance	Cloud SQL instance	Additional read capacity Analytics target Additional disaster recovery capability Improve read performance Migrate data between regions	Creating read replicas Managing read replicas Replication metrics
External read replica	Cloud SQL instance	MySQL instance external to Cloud SQL	Reduced latency for external connections Analytics target Migration path to other platforms	Configuring external replicas
Replication from an external server	MySQL instance external to Cloud SQL	Cloud SQL for MySQL instance	Migration path to Cloud SQL Data replication to Google Cloud Platform Analytics target	Replicating from an external server

4) What exactly is the Google Cloud SDK? List the numerous Google cloud SDK installation options.

Cloud SDK provides language-specific Cloud Client Libraries supporting each language's natural conventions and styles. This makes it easier for you to interact with Google Cloud APIs in your language of choice

you can install google cloud sdk for mulitple languages

1) google cloud SDk for python

2) google cloud SDk for java

3) google cloud SDk for go

4) google cloud SDk for php

5) google cloud SDk for ruby

6) google cloud SDk for C#

7) google cloud SDk for C++

5) List the many cloud computing deployment models.

There are four cloud deployment models: public, private, community, and hybrid.

1. Public Cloud

The public cloud makes it possible for anybody to access systems and services. The public cloud may be less secure as it is open for everyone. The public cloud is one in which cloud infrastructure services are provided over the internet to the general people or major industry groups.

Advantages of the public cloud model:

Minimal Investment

No setup cost

Infrastructure Management is not required

No maintenance

Dynamic Scalability

2. Private Cloud(internal cloud)

The private cloud deployment model is the exact opposite of the public cloud deployment model. It's a one-on-one environment for a single user (customer). There is no need to share your hardware with anyone else. The distinction between private and public cloud is in how you handle all of the hardware.

Advantages of the private cloud model:

Better Control:

Data Security and Privacy

Supports Legacy Systems

Customization

3.   Hybrid cloud

By bridging the public and private worlds with a layer of proprietary software, hybrid cloud computing gives the best of both worlds. With a hybrid solution, you may host the app in a safe environment while taking advantage of the public cloud's cost savings. Organizations can move data and applications between different clouds using a combination of two or more cloud deployment methods, depending on their needs.

Advantages of the hybrid cloud model:

Flexibility and control:

Cost

Security:

4. Community cloud

It allows systems and services to be accessible by a group of organizations. It is a distributed system that is created by integrating the services of different clouds to address the specific needs of a community, industry, or business.

Advantages of the community cloud model:

Cost Effective

Security

Shared resource

Collaboration and data sharing

6) Describe the Google cloud platform's security features.

24/7/365 operations, device security detection and response from both internal and external threats

Data in-transit encrypted communication to and from Google's public cloud, including layered defense redundancies to protect customers from denial-of-service (DoS) attacks

Identity protection and management through multiple authentication factors

Data at-rest storage security using encryption against unauthorized access and distribution for reliability

An entire hardware infrastructure created, built, controlled, and secured by Google including servers, networking equipment, and security chips

7) What exactly is vertex AI? With some usage scenarios, implement vertex AI.

Vertex AI Workbench is the single environment for data scientists to complete all of

their ML work, from experimentation, to deployment, to managing and monitoring

models. It is a Jupyter-based fully managed, scalable, enterprise-ready compute

infrastructure with security controls and user management capabilities.

USE CASE

Data readiness

Vertex AI supports your data preparation process. You can ingest data from

BigQuery and Cloud Storage and leverage Vertex AI Data Labeling to annotate high_quality training data and improve prediction accuracy
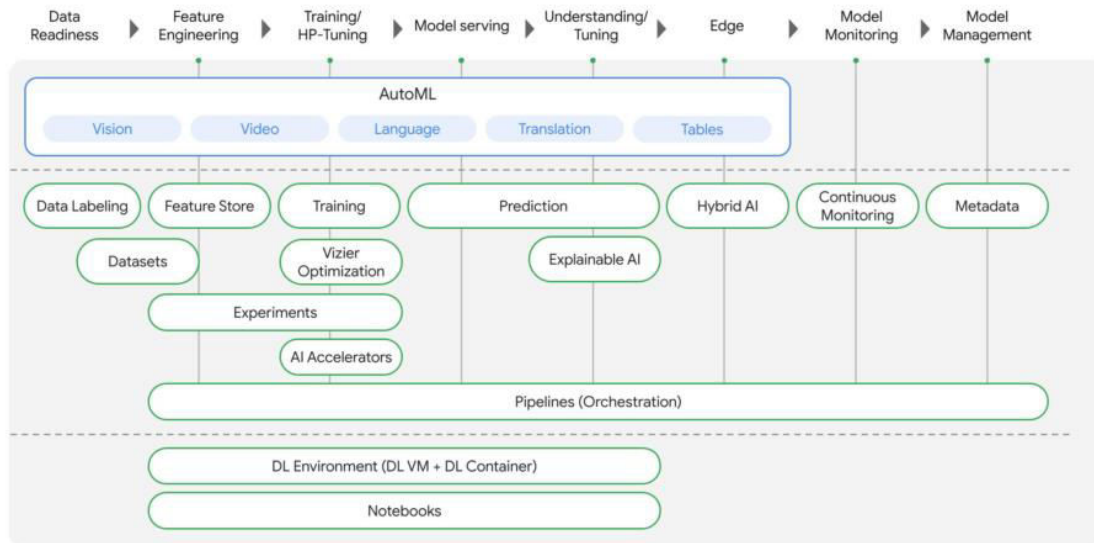
USE CASE

Feature engineering

Use Vertex AI Feature Store, a fully managed rich feature repository, to serve, share, and reuse ML features; Vertex AI Experiments to track, analyze, and discover ML experiments for faster model selection; Vertex AI TensorBoard to visualize ML experiments; and Vertex AI Pipelines to simplify the MLOps process by streamlining the building and running of ML pipelines.

USE CASE

Training and hyperparameter tuning

Build state-of-the-art ML models without code by using AutoML to determine the optimal model architecture for your image, tabular, text, or video-prediction task, or build custom models using Notebooks. Vertex AI Training offers fully managed training services, and Vertex AI Vizier provides optimized hyperparameters for maximum predictive accuracy.

USE CASE

Model serving

Vertex AI Prediction makes it easy to deploy models into production, for online serving via HTTP or batch prediction for bulk scoring. You can deploy custom models built on any framework (including TensorFlow, PyTorch, scikit or XGB) to Vertex AI Prediction, with built-in tooling to track your models' performance.

USE CASE

Model tuning and understanding

Get detailed model evaluation metrics and feature attributions, powered by Vertex Explainable AI. Vertex Explainable AI tells you how important each input feature is to your prediction. Available out of the box in AutoML Forecasting, Vertex AI Prediction, and Vertex AI Workbench.

USE CASE

Edge

Vertex AI Edge Manager (in experimental phase) is designed to facilitate seamless deployment and monitoring of edge inferences and automated processes with flexible APIs, to allow you to distribute AI across your private and public cloud infrastructure, on-premises data centers, and edge devices.

USE CASE

Model monitoring

Continuous monitoring offers easy and proactive monitoring of model performance over time for models deployed in the Vertex AI Prediction service. Continuous monitoring monitors signals for your model's predictive performance and alerts when the signals deviate, diagnose the cause of the deviation, and trigger model-retraining pipelines or collect relevant training data.

USE CASE

Model management

Vertex ML Metadata enables easier auditability and governance by automatically

tracking inputs and outputs to all components in Vertex Pipelines for artifact, lineage,

and execution tracking for your ML workflow. Track custom metadata directly from

your code and query metadata using a Python SDK.

Vertex AI Workbench

Vertex AI Workbench is the single environment for data scientists to complete all of their ML work, from experimentation, to deployment, to managing and monitoring models. It is a Jupyter-based fully managed, scalable, enterprise-ready compute infrastructure with security controls and user management capabilities.

Vertex AI Matching Engine

Massively scalable, low latency, and cost_efficient vector similarity matching service.

Vertex AI Data Labeling

Get highly accurate labels from human labelers for better machine learning models.

Vertex AI Deep Learning Containers

Quickly build and deploy models in a portable and consistent environment for all your AI applications.

Vertex AI Edge Manager

Seamlessly deploy and monitor edge inferences and automated processes with flexible APIs.

Vertex Explainable AI

Understand and build trust in your model predictions with robust, actionable explanations integrated into Vertex AI Prediction, AutoML Tables, and Vertex AI Workbench.

Vertex AI Feature Store

A fully managed rich feature repository for serving, sharing, and reusing ML features.

Vertex ML Metadata

Artifact, lineage, and execution tracking for ML workflows, with an easy-to-use Python SDK.

Vertex AI Model Monitoring

Automated alerts for data drift, concept drift, or other model performance incidents which may require supervision.

Vertex AI Neural Architecture Search

Build new model architectures targeting application-specific needs and optimize your existing model

architectures for latency, memory, and power with this automated service powered by Google's leading AI research.

Vertex AI Pipelines

Build pipelines using TensorFlow Extended and Kubeflow Pipelines, and leverage Google Cloud's managed services to execute scalably and pay per use. Streamline your MLOps with detailed metadata tracking, continuous modeling, and triggered model retraining.

Vertex AI Prediction

Deploy models into production more easily with online serving via HTTP or batch prediction for bulk scoring. Vertex AI Prediction offers a unified framework to deploy custom models trained in TensorFlow, scikit or XGB, as well as BigQuery ML and AutoML models, and on a broad range of machine types and GPUs.

Vertex AI Tensorboard

This visualization and tracking tool for ML experimentation includes model graphs which display images, text, and audio data.

Vertex AI Training

Vertex AI Training provides a set of pre_built algorithms and allows users to bring their custom code to train models. A fully managed training service for users needing greater flexibility and customization or for users running training on-premises or another cloud environment.

Vertex AI Vizier

Optimized hyperparameters for maximum predictive accuracy