GCP Assignment 1

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

Ans - Google Cloud uses project quotas to restrict how much of a particular shared Google Cloud resource that you can use. Quotas are enforced for mainly below reasons:

a) To protect the community of Google Cloud users by preventing unforeseen spikes in usage and overloaded services.
b) To help you manage resources. For example, you can set your own limits on service usage while developing and testing your applications to avoid unexpected bills from using expensive resources.

Quota increase requests are handled by Cloud Customer Care, who process your request, typically within 2-3 business days. To request a higher quota limit by using the Google Cloud Console:

a) Go to the **Quotas** page.
b) On the Quotas page, find the quota you want to increase in the **Limit name** column. You can use the Filter search box to search for your quota.
c) Select the checkbox to the left of your quota.
d) Click **EDIT QUOTAS**. The **Quota changes** form displays.
e) In the **Quota changes** form, enter the increased quota that you want for your project in the New limit field.
f) Complete any additional fields in the form, and then click **DONE**.
g) Click **SUBMIT REQUEST**.

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.

Ans - A virtual machine (VM) is a digital version of a physical computer. Virtual machine software can run programs and operating systems, store data, connect to networks, and do other computing functions, and requires maintenance such as updates and system monitoring.

Virtual machine types that are offered in GCP:

a) The **general-purpose machine** family offers several machine series with the best price-performance ratio for a variety of workloads.
b) The **compute-optimized machine** family has the highest performance per core on Compute Engine and is optimized for compute-intensive workloads.
c) The **memory-optimized machine** family has machine series that are ideal for memory-intensive workloads. This family offers more memory per core than any other machine family, with up to 12 TB of memory.
d) The **accelerator-optimized machine** family is ideal for massively parallelized Compute Unified Device Architecture (CUDA) compute workloads, such as machine learning (ML) and high performance computing (HPC). This family is the optimal choice for workloads that require GPUs.

Create a VM instance from a public image:

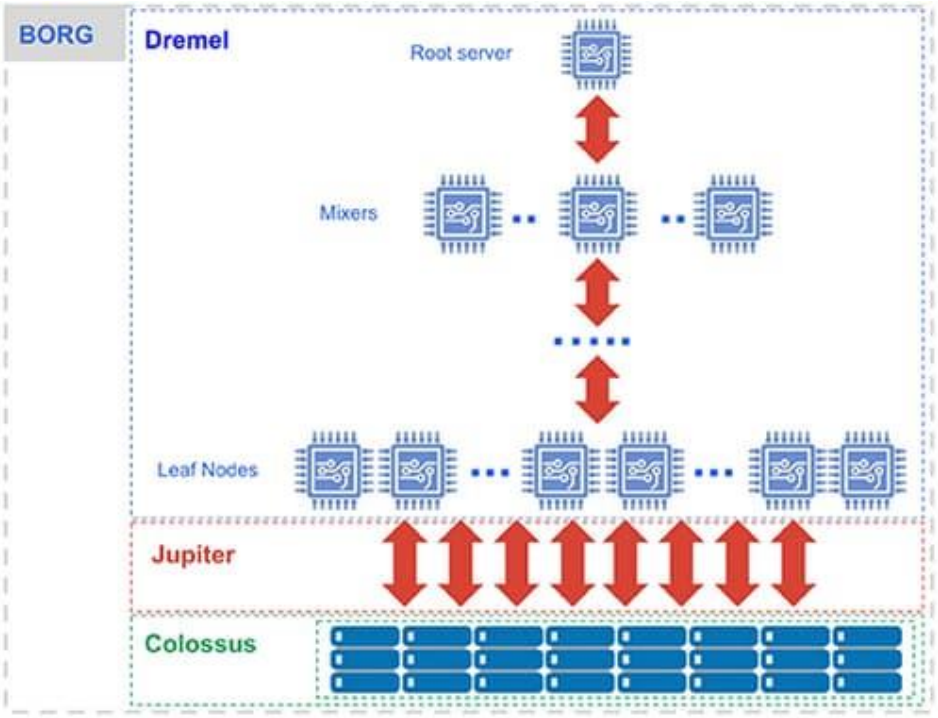
a) In the Google Cloud Console, go to the **VM instances** page.
b) Select your project and click **Continue**.
c) Click **Create instance**.
d) Specify a Name for your VM.
e) Select a **Machine configuration** for your VM.
f) In the **Boot disk** section, click **Change**, and then do the following:
    a. On the Public images tab, choose the following:
       Operating system
       OS version
       Boot disk type
       Boot disk size
    b. To confirm your boot disk options, click Select.
g) In the Firewall section, to permit HTTP or HTTPS traffic to the VM, select Allow HTTP traffic or Allow HTTPS traffic. When you select one of these, Compute Engine adds a network tag to your VM, which associates the firewall rule with the VM. Then, Compute Engine creates the corresponding ingress firewall rule that allows all incoming traffic on tcp:80 (HTTP) or tcp:443 (HTTPS).
h) 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

Ans- BigQuery is a fully managed enterprise data warehouse that helps you manage and analyze your data with built-in features like machine learning, geospatial analysis, and business intelligence. BigQuery's serverless architecture lets you use SQL queries to answer your organization's biggest questions with zero infrastructure management. BigQuery's scalable, distributed analysis engine lets you query terabytes in seconds and petabytes in minutes.

How BigQuery works:

BigQuery’s serverless architecture decouples storage and compute and allows them to scale independently on demand. Under the hood, BigQuery employs a vast set of multi-tenant services driven by low-level Google infrastructure technologies like Dremel, Colossus, Jupiter and Borg.



a) Compute is Dremel, a large multi-tenant cluster that executes SQL queries.

Dremel turns SQL queries into execution trees. The leaves of the tree are called slots and do the heavy lifting of reading data from storage and any necessary computation. The branches of the tree are 'mixers', which perform the aggregation.

Dremel dynamically apportions slots to queries on an as-needed basis, maintaining fairness for concurrent queries from multiple users. A single user can get thousands of slots to run their queries.

b) Storage is Colossus, Google's global storage system.

BigQuery leverages the columnar storage format and compression algorithm to store data in Colossus, optimized for reading large amounts of structured data.

Colossus also handles replication, recovery (when disks crash) and distributed management (so there is no single point of failure). Colossus allows BigQuery users to scale to dozens of petabytes of data stored seamlessly, without paying the penalty of attaching much more expensive compute resources as in traditional data warehouses.

c) Compute and storage talk to each other through the petabit Jupiter network.

In between storage and compute is 'shuffle', which takes advantage of Google's Jupiter network to move data extremely rapidly from one place to another.

d) BigQuery is orchestrated via Borg, Google's precursor to Kubernetes.

The mixers and slots are all run by Borg, which allocates hardware resources.

Use cases for Big Query:

1. Migrating data warehouses to BigQuery

Solve for today's analytics demands and seamlessly scale your business by moving to Google Cloud's modern data warehouse. Streamline your migration path from Netezza, Oracle, Redshift, Teradata, or Snowflake to BigQuery and accelerate your time to insights.

2. Predictive analytics

Predictive analytics helps you predict future outcomes more accurately and discover opportunities in your business. Our smart analytics reference patterns are designed to reduce time-to-value for common analytics use cases with sample code and technical reference guides.

3. Bring any data into BigQuery

Make analytics easier by bringing together data from multiple sources in BigQuery, for seamless analysis. You can upload data files from local sources, Google Drive, or Cloud Storage buckets, take advantage of BigQuery Data Transfer Service (DTS), Data Fusion plug-ins, or leverage Google's industry-leading data integration partnerships. You have ultimate flexibility in how you bring data into your data warehouse.

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

Ans - Google Cloud SDK is a set of tools for accessing Google public cloud platform in a secure way. It contains essential tools for maintaining, managing and monitoring Google Cloud Platform (GCP).

The recommended method for the installation of the Google Cloud SDK.

1. Download the Google Cloud CLI installer.
2. Alternatively, open a PowerShell terminal and run the following PowerShell commands:

```
(New-Object Net.WebClient).DownloadFile("https://dl.google.com/dl/cloudsdk/channels/rapid/GoogleCloudSDKInstaller.exe",
"$env:Temp\GoogleCloudSDKInstaller.exe")

& $env:Temp\GoogleCloudSDKInstaller.exe
```

3. Launch the installer and follow the prompts. The installer is signed by Google LLC.

Other installation options are:

a) Download a versioned archive for a non-interactive installation of a specific version of the gcloud CLI.
b) Use the gcloud CLI Docker image for the latest release (or a specific version) of gcloud CLI.
c) Use a snap package to install the gcloud CLI.

d) For Windows and macOS interactive installations, and all other use cases, run the interactive installer to install the latest release of the gcloud CLI.

5. List the many cloud computing deployment models.

Ans - Different types of cloud computing deployment models are:

A. **Public cloud** - The public cloud is one in which cloud infrastructure services are provided over the internet to the general people or major industry groups. The infrastructure in this cloud model is owned by the entity that delivers the cloud services, not by the consumer.
It is a type of cloud hosting that allows customers and users to easily access systems and services. This form of cloud computing is an excellent example of cloud hosting, in which service providers supply services to a variety of customers. In this arrangement, storage backup and retrieval services are given for free, as a subscription, or on a per-use basis. Example: Google App Engine etc.

B. **Private 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. It is also called the "internal cloud" & it refers to the ability to access systems and services within a given border or organization. The cloud platform is implemented in a cloud-based secure environment that is protected by powerful firewalls and under the supervision of an organization's IT department.

C. **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.

D. **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. The infrastructure of the community could be shared between the organization which has shared concerns or tasks. It is generally managed by a third party or by the combination of one or more organizations in the community.

E. **Multi-cloud** - It's similar to the hybrid cloud deployment approach, which combines public and private cloud resources. Instead of merging private and public clouds, multi-cloud uses many public clouds.

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

Ans - Google cloud platform infrastructure security features include:

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

B) 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
C) Identity protection and management through multiple authentication factors
D) Data at-rest storage security using encryption against unauthorized access and distribution for reliability
E) 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.

Ans - Vertex AI is a managed machine learning platform for deploying and maintaining AI models. It is the single environment for data scientists to complete all of their ML work, from experimentation, to deployment, to managing and monitoring models.

Some use case to take advantage of Vertex AI:

a) 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.

b) 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.

c) 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.

d) 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.

e) 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.

f) 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.

g) 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.

h) 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.