GCP Assignment 3

1) What distinguishes the Google Cloud Machine Learning Engine from others?

The Google Cloud ML Engine is a hosted platform to run machine learning training jobs and predictions at scale. The service treats these two processes (training and predictions) independently. It is possible to use Google Cloud ML Engine just to train a complex model by leveraging the GPU and TPU infrastructure. The outcome from this step — a fully-trained machine learning model — can be hosted in other environments including on-prem infrastructure and public cloud. The service can also be used to deploy a model that is trained in external environments. Cloud ML Engine automates all resource provisioning and monitoring for running the jobs. It can also manage the lifecycle of deployed models and their versions.

Apart from training and hosting, Cloud ML Engine can also perform hyperparameter tuning that influences the accuracy of predictions. Without automated hyperparameter tuning, data scientists will have to experiment with multiple values while evaluating the accuracy of the results.

Whether Google Cloud Platform is less expensive than its competitors depends a lot on which type of workloads you want to run and how you run them. It offers discounts based on what it calls sustained use, meaning situations where customers keep a workload running for extended periods.

This distinguishes the Google Cloud Machine Learning Engine from others.

2) What are GCP's cloud storage libraries and tools?

- Google Cloud Platform Console, which performs fundamental tasks on objects and buckets.

- Cloud Storage Client Libraries, which give programming backing to different languages including Java, Ruby, and Python.

- GustilCommand-line Tool, which gives a command-line interface to the cloud storage.

3) Define the Google Cloud pricing model.

While working with Google Cloud Platform, the client is charged based on compute instance, network use, and storage by Google Compute Engine. At that point, the expense of storage is charged based on the measure of information that you store.

4) The Google Compute Engine API has a variety of authentication mechanisms.

Google Cloud APIs use the OAuth 2.0 protocol for authenticating both user accounts and service accounts. The OAuth 2.0 authentication process determines both the principal and the application.

Most Google Cloud APIs also support anonymous access to public data using API keys. However, API keys only identify the application, not the principal. When using API keys, the principal must be authenticated by other means.

Google Cloud APIs support multiple authentication flows for different runtime environments. For the best developer experience, we recommend using Google Cloud Client Libraries with Google Cloud APIs. They use Google-provided authentication libraries that support a variety of authentication flows and runtime environments.

5) Create Google Cloud service accounts and explain how to use them.

A service account is a special type of Google account intended to represent a non-human user that needs to authenticate and be authorized to access data in Google APIs. Typically, service accounts are used in scenarios such as: Running workloads on virtual machines (VMs).

1. Sign in to the Google API Console.

2. Open the Credentials page. If prompted, select the project that has the Android Management API enabled.

3. Click **Create credentials** > **Service account key**.

4. From the dropdown menu, select **New service account**. Enter a name for your service account.

5. Select your preferred key type and click **Create**. Your new public/private key pair is generated and downloaded to your machine and is the only copy of this key. You are responsible for storing it securely.

6. Open the IAM page. If prompted, select the project that has the Android Management API enabled.

7. Click **Add**.

8. Add the service account you've just created as member and select the role **Android Management User**.

9. Click **Save**.

10. (Optional, but highly recommended) Add additional project owners by granting the Owner role to existing project members.

6) How to make a Google Cloud Storage project?
Cloud Storage

1. Select Browser in the lefthand menu.

2. Click CREATE BUCKET.

3. Enter a unique bucket name. Names must start and end with a letter or number.

4. Pick a storage class. Storage class affects performance, cost, and location.

5. Pick a location to store your data.

6. Click Create.