GCP – Serverless Spark

- Dataproc Serverless lets you run Spark batch workloads without requiring you to provision and manage your own cluster.

- Specify workload parameters, and then submit the workload to the Dataproc Serverless service

- The service will run the workload on a managed compute infrastructure, autoscaling resources as needed

- Dataproc Serverless charges apply only to the time when the workload is executing

- You can run the following Spark workload types on the Dataproc Serverless for Spark service:
  - Pyspark
  - Spark SQL
  - Spark R
  - Spark Java/Scala

GCP – Serverless Spark (PHS)

- The Dataproc Persistent History Server (PHS) provides web interfaces to view job history for jobs run on active or deleted Dataproc clusters.

- Runs on a single-node dataproc cluster

- Provides :
  - ✓ Logs
  - ✓ History files
  - ✓ Yarn aggregation logs and metrics

GCP – Serverless Spark AutoScaling

- Dataproc Serverless for Spark can dynamically scale workload resources such as the number of executors to run your workload efficiently

- Below properties can be set at the time of job submit :
  - spark.dynamicAllocation.enabled
  - spark.dynamicAllocation.initialExecutors
  - spark.dynamicAllocation.minExecutors
  - spark.dynamicAllocation.maxExecutors
  - spark.dynamicAllocation.executorAllocationRatio

NO→LATENCY

GCP – Dataproc Serverless

- ✓ Submit Pyspark jobs
- ✓ Autoscaling parameters (Examples provided in submid-command.sh file)
- ✓ Airflow/Cloud Composer
- ✓ Scheduling Serverless pyspark jobs using Cloud Composer
- ✓ **Final Step** : Delete all the resources
- ✓ **Reference Documentation:** https://cloud.google.com/dataproc-serverless/docs