DUMPSBOSS.

Google Cloud Associate Data Practitioner (ADP Exam)

Google Associate-Data-Practitioner

Version Demo

Total Demo Questions: 12

Total Premium Questions: 121

Buy Premium PDF

https://dumpsboss.co

support@dumpsboss.co

support@dumpsboss.co
dumpsboss.co

Topic Break Down

Topic	No. of Questions
Topic 1, Ingesting and processing data	33
Topic 2, Storing data	35
Topic 3, Preparing and using data for analysis	32
Topic 4, Maintaining and automating data workloads	21
Total	121

QUESTION NO: 1

Your organization processes clickstream events in a Dataflow streaming pipeline. Events include an event timestamp, but some mobile devices send events up to 20 minutes late. The business requires hourly session metrics to be calculated according to when the events occurred, rather than when Dataflow receives them. What should you do?

A. Configure the pipeline to use processing-time windows and increase the number of workers.

B. Configure the pipeline to use event-time windows with an allowed-lateness period.

C. Write all events directly to BigQuery and run a scheduled query once each day.

D. Configure Pub/Sub to delay delivery of every message by 20 minutes.

ANSWER: B

Explanation:

Configure the pipeline to use event-time windows with an allowed-lateness period. Event time uses the timestamp contained in each event, so hourly aggregations reflect the time at which user activity actually occurred. Allowed lateness permits Dataflow to incorporate events that arrive after an initial window result has been produced, subject to the configured lateness threshold.

Using processing-time windows would group events according to their arrival time and would place delayed mobile events in the wrong hourly interval. Increasing worker capacity can improve throughput but does not correct event-time semantics. A batch pipeline can process delayed data, but it does not meet the requirement to continuously calculate streaming metrics.

QUESTION NO: 2

Your organization uses a BigQuery table that is partitioned by ingestion time. You need to remove data that is older than one year to reduce your organizationâ€™s storage costs. You want to use the most efficient approach while minimizing cost. What should you do?

A. Create a scheduled query that periodically runs an update statement in SQL that sets the â€œdeleted" column to â€œyesâ€□ for data that is more than one year old. Create a view that filters out rows that have been marked deleted.

B. Create a view that filters out rows that are older than one year.

C. Require users to specify a partition filter using the alter table statement in SQL.

D. Set the table partition expiration period to one year using the ALTER TABLE statement in SQL.

ANSWER: D

Explanation:

Set the table partition expiration period to one year using the ALTER TABLE statement in SQL. This configures an automatic retention policy for the ingestion-time partitioned table: BigQuery expires each partition when it reaches the configured age and removes the data from that partition. As a result, data older than one year is deleted without recurring operational work, scheduled jobs, or table scans to identify eligible records. This is the appropriate storage-lifecycle mechanism because the requirement is based directly on partition age, and it ensures that retained data remains limited to the required one-year window. For ingestion-time partitioned tables, partition expiration is calculated from each partition's boundary in UTC. A partition expiration can be configured with SQL by setting the table option `partition_expiration_days`, for example through an `ALTER TABLE ... SET OPTIONS` statement. Automatic partition expiration therefore provides a managed, efficient way to enforce the retention period and reduce the storage occupied by expired data. See the BigQuery documentation on managing partition expiration and the GoogleSQL ALTER TABLE SET OPTIONS reference.

QUESTION NO: 3

You work for a healthcare company. You have a daily ETL pipeline that extracts patient data from a legacy system, transforms it, and loads it into BigQuery for analysis. The pipeline currently runs manually using a shell script. You want to automate this process and add monitoring to ensure pipeline observability and troubleshooting insights. You want one centralized solution, using open-source tooling, without rewriting the ETL code. What should you do?

A. Create a directed acyclic graph (DAG) in Cloud Composer to orchestrate the pipeline daily. Monitor the pipeline's execution using the Apache Airflow web interface and Cloud Monitoring.

B. Configure Cloud Dataflow to implement the ETL pipeline, and use Cloud Scheduler to trigger the Dataflow pipeline daily. Monitor the pipelines execution using the Dataflow job monitoring interface and Cloud Monitoring.

C. Use Cloud Scheduler to trigger a Dataproc job to execute the pipeline daily. Monitor the job's progress using the Dataproc job web interface and Cloud Monitoring.

D. Create a Cloud Run function that runs the pipeline daily. Monitor the functions execution using Cloud Monitoring.

ANSWER: A

Explanation:

Create a directed acyclic graph (DAG) in Cloud Composer to orchestrate the pipeline daily. Monitor the pipeline's execution using the Apache Airflow web interface and Cloud Monitoring. This is the best fit because Cloud Composer is Google Cloud's managed service for Apache Airflow, an open-source workflow orchestration platform. An Airflow DAG can schedule the daily workflow, define its individual steps and dependencies, and preserve a centralized execution history. The existing shell-script-based ETL process can be invoked from an Airflow task, such as a BashOperator, so the organization can automate its established code rather than redesigning the transformation logic.

Cloud Composer also provides the operational visibility needed for a production healthcare data pipeline. The Airflow web interface exposes DAG runs, task status, durations, retries, logs, and dependency views, enabling operators to identify where a failed run occurred. Composer integrates with Google Cloud observability services, allowing logs and metrics to be viewed in Cloud Logging and Cloud Monitoring, where teams can create alerting policies for failures or abnormal behavior. This combines open-source Airflow orchestration with managed infrastructure and centralized monitoring. See the Cloud Composer overview and Cloud Composer monitoring documentation.

QUESTION NO: 4

Your organization has highly sensitive data that gets updated once a day and is stored across multiple datasets in BigQuery. You need to provide a new data analyst access to query specific data in BigQuery while preventing access to sensitive data. What should you do?

A. Grant the data analyst the BigQuery Job User IAM role in the Google Cloud project.

B. Create an authorized view with the limited data in a new dataset, and authorize it to access the required source datasets. Grant the data analyst the BigQuery Data Viewer IAM role on the new dataset and the BigQuery Job User IAM role in the Google Cloud project.

C. Create a new Google Cloud project, and copy the limited data into a BigQuery table. Grant the data analyst the BigQuery Data Owner IAM role in the new Google Cloud project.

D. Grant the data analyst the BigQuery Data Viewer IAM role in the Google Cloud project.

ANSWER: B

Explanation:

Create an authorized view containing only the permitted columns and rows in a separate dataset, authorize that view to read the required source datasets, and grant the analyst BigQuery Data Viewer on the view’s dataset plus BigQuery Job User in the project. An authorized view is designed for this least-privilege access pattern: it lets users query the view’s defined result while BigQuery evaluates the view using its authorization to access the underlying datasets. The analyst therefore receives access to the approved subset of data without being granted direct permissions on the datasets containing sensitive tables. Locating the view in a separate dataset provides a clear permission boundary and makes it easier to manage access as analysts join or leave the team. BigQuery Data Viewer supplies the permission to read the view, and BigQuery Job User

supplies the ability to create query jobs in the project. Because the underlying data changes daily, the view returns results based on the current source data whenever it is queried, without requiring a separate copying workflow. Google documents authorized views as the mechanism for sharing query results without granting access to the source data, and documents BigQuery predefined roles and their appropriate scopes in the IAM reference. See Authorized views and BigQuery IAM access control.

QUESTION NO: 5

Your company's ecommerce website collects product reviews from customers. The reviews are loaded as CSV files daily to a Cloud Storage bucket. The reviews are in multiple languages and need to be translated to Spanish. You need to configure a pipeline that is serverless, efficient, and requires minimal maintenance. What should you do?

A. Load the data into BigQuery using Dataproc. Use Apache Spark to translate the reviews by invoking the Cloud Translation API. Set BigQuery as the sink.U

B. Use a Dataflow templates pipeline to translate the reviews using the Cloud Translation API. Set BigQuery as the sink.

C. Load the data into BigQuery using a Cloud Run function. Use the BigQuery ML create model statement to train a translation model. Use the model to translate the product reviews within BigQuery.

D. Load the data into BigQuery using a Cloud Run function. Create a BigQuery remote function that invokes the Cloud Translation API. Use a scheduled query to translate new reviews.

ANSWER: D

Explanation:

Load the data into BigQuery using a Cloud Run function. Create a BigQuery remote function that invokes the Cloud Translation API. Use a scheduled query to translate new reviews. This design uses managed, serverless services throughout the workflow. A Cloud Run function can be triggered by new objects in the Cloud Storage bucket and can initiate or coordinate loading the daily CSV data into BigQuery. BigQuery then provides a centralized, scalable location for both the incoming reviews and the translated results.

A BigQuery remote function lets SQL invoke application logic hosted in Cloud Run. The Cloud Run service can call Cloud Translation, a fully managed API that translates text without requiring the company to build, train, deploy, or operate a translation model. The scheduled query provides a managed recurring mechanism to identify and process newly loaded reviews, so the translation workflow runs automatically on the required cadence. This approach separates ingestion, translation invocation, and analytical storage while avoiding cluster administration and custom machine-learning operations. BigQuery remote functions are documented at https://cloud.google.com/bigquery/docs/remote-functions, and Cloud Translation capabilities are documented at https://cloud.google.com/translate/docs/overview.

QUESTION NO: 6

Your company has an on-premises file server with 5 TB of data that needs to be migrated to Google Cloud. The network operations team has mandated that you can only use up to 250 Mbps of the total available bandwidth for the migration. You need to perform an online migration to Cloud Storage. What should you do?

A. Use Storage Transfer Service to configure an agent-based transfer. Set the appropriate bandwidth limit for the agent pool.

B. Use the gcloud storage cp command to copy all files from on-premises to Cloud Storage using the --daisy-chain option.

C. Request a Transfer Appliance, copy the data to the appliance, and ship it back to Google Cloud.

D. Use the gcloud storage cp command to copy all files from on-premises to Cloud Storage using the --no-clobber option.

ANSWER: A

Explanation:

Use Storage Transfer Service to configure an agent-based transfer. Set the appropriate bandwidth limit for the agent pool. Storage Transfer Service is designed to move data from on-premises file systems to Cloud Storage over the network. Its agent-based transfer capability uses software agents installed in the on-premises environment to read files and transfer them securely to a Cloud Storage bucket. This directly supports the requirement for an online migration, without needing to physically ship storage hardware.

For an on-premises transfer, agents are organized into an agent pool. Storage Transfer Service lets you configure a bandwidth limit for that pool, expressed in megabits per second. Setting the pool's limit to 250 Mbps ensures the migration does not exceed the network operations team's allowed share of available bandwidth. The limit applies collectively to the agents in the pool, so it remains effective even when multiple agents are used to improve transfer reliability or parallelize file discovery and copying. The transfer can therefore proceed in a controlled manner while normal business network traffic remains protected. Google documents agent pools and their bandwidth controls at Storage Transfer Service agent pools and describes the on-premises file-system workflow at Create agent-based transfers.

QUESTION NO: 7

You have millions of customer feedback records stored in BigQuery. You want to summarize the data by using the large language model (LLM) Gemini. You need to plan and execute this analysis using the most efficient approach. What should you do?

A. Query the BigQuery table from within a Python notebook, use the Gemini API to summarize the data within the notebook, and store the summaries in BigQuery.

B. Use a BigQuery ML model to pre-process the text data, export the results to Cloud Storage, and use the Gemini API to summarize the pre- processed data.

C. Create a BigQuery Cloud resource connection to a remote model in Vertex AI, and use Gemini to summarize the data.

D. Export the raw BigQuery data to a CSV file, upload it to Cloud Storage, and use the Gemini API to summarize the data.

ANSWER: C

Explanation:

Create a BigQuery Cloud resource connection to a remote model in Vertex AI, and use Gemini to summarize the data. This is the most efficient design because it lets BigQuery invoke a Gemini model through a BigQuery ML remote model while the source feedback data remains available for SQL-based processing and analysis. A Cloud resource connection provides the controlled integration between BigQuery and Vertex AI, including the identity used to access the Vertex AI service. After creating the connection and remote model, analysts can use BigQuery SQL to submit text values from table rows to Gemini and write generated summaries into a destination table as part of a scalable, repeatable workflow. This approach is well suited to millions of records because BigQuery can manage set-based query execution and batch-oriented generation workflows without requiring an analyst to manually extract, move, and re-ingest the dataset. It also supports applying normal SQL operations—such as filtering, chunking, grouping, and persisting results—around the generation step. Google documents Gemini integration in BigQuery ML, including remote models and generative AI functions, at Generate text with Gemini models. The required BigQuery connection resource and its access configuration are described in Create a Cloud resource connection.

QUESTION NO: 8

Your organization has several datasets in BigQuery. The datasets need to be shared with your external partners so that they can run SQL queries without needing to copy the data to their own projects. You have organized each partnerâ€™s data in its own BigQuery dataset. Each partner should be able to access only their data. You want to share the data while following Google-recommended practices. What should you do?

A. Use Analytics Hub to create a listing on a private data exchange for each partner dataset. Allow each partner to subscribe to their respective listings.

B. Create a Dataflow job that reads from each BigQuery dataset and pushes the data into a dedicated Pub/Sub topic for each partner. Grant each partner the pubsub. subscriber IAM role.

C. Export the BigQuery data to a Cloud Storage bucket. Grant the partners the storage.objectUser IAM role on the bucket.

D. Grant the partners the bigquery.user IAM role on the BigQuery project.

ANSWER: A

Explanation:

Use Analytics Hub to create a listing on a private data exchange for each partner dataset. Allow each partner to subscribe to their respective listings. Analytics Hub is BigQuery's managed data-sharing service and is designed for securely distributing data across organizational boundaries without requiring consumers to copy the underlying data into their own projects. A provider publishes a listing that points to a shared dataset or other supported resource, and a subscriber is granted access through the private exchange and can query the shared data from BigQuery.

Creating a distinct listing for each partner-specific dataset enforces the required separation of access. Each private listing can be shared only with the intended external partner, supporting least-privilege access management while keeping administration scalable as partners and datasets grow. The provider continues to manage the source data and its updates centrally, while subscribers can analyze the shared data through SQL without an export-and-transfer workflow. This matches Google's recommended approach for sharing BigQuery data with external organizations. For implementation details, see Introduction to Analytics Hub and Manage Analytics Hub listings.

QUESTION NO: 9

You are designing a pipeline to process data files that arrive in Cloud Storage by 3:00 am each day. Data processing is performed in stages, where the output of one stage becomes the input of the next. Each stage takes a long time to run. Occasionally a stage fails, and you have to address

the problem. You need to ensure that the final output is generated as quickly as possible. What should you do?

A. Design a Spark program that runs under Dataproc. Code the program to wait for user input when an error is detected. Rerun the last action after correcting any stage output data errors.

B. Design the pipeline as a set of PTransforms in Dataflow. Restart the pipeline after correcting any stage output data errors.

C. Design the workflow as a Cloud Workflow instance. Code the workflow to jump to a given stage based on an input parameter. Rerun the workflow after correcting any stage output data errors.

D. Design the processing as a directed acyclic graph (DAG) in Cloud Composer. Clear the state of the failed task after correcting any stage output data errors.

ANSWER: D

Explanation:

Designing the processing as a directed acyclic graph (DAG) in Cloud Composer and clearing the state of the failed task after correcting the problem is the appropriate approach. Cloud Composer is Google Cloud's managed Apache Airflow service, which is designed to orchestrate multi-stage workflows with explicit dependencies between tasks. A DAG represents each processing stage as a task and defines the required ordering, so downstream stages run only after their required inputs are available.

When a task fails, Airflow records its state and preserves the states of the rest of the DAG. After the underlying issue or affected stage output has been corrected, an operator can clear the failed task's state. The scheduler then runs that task again and can execute its downstream dependent tasks as needed, rather than rerunning already successful upstream stages. This is especially valuable when stages are long-running because it minimizes unnecessary recomputation and reduces the time to produce the final output. Task retries, dependency management, monitoring, and manual task-state clearing are core operational capabilities of Airflow DAGs in Cloud Composer. See the Cloud Composer documentation for managing DAGs and the Apache Airflow task documentation.

QUESTION NO: 10

Your organization is building a new application on Google Cloud. Several data files will need to be stored in Cloud Storage. Your organization has approved only two specific cloud regions where these data files can reside. You need to determine a Cloud Storage bucket strategy that includes automated high availability. What should you do?

A. Create a dual-region bucket, and upload the files to this bucket.

B. Create a single-region bucket in each of the two regions, and use the gcloud storage command to replicate the data across the buckets in both regions.

C. Create a multi-region bucket, and upload the files to this bucket.

D. Create a single-region bucket in each of the two regions, and use Storage Transfer Service to replicate the data across the buckets in both regions.

ANSWER: A

Explanation:

Create a dual-region bucket, and upload the files to this bucket. A Cloud Storage dual-region bucket is designed for data that must reside in exactly two chosen Google Cloud regions while also requiring built-in availability and redundancy. When creating the bucket, the organization selects a predefined or configurable dual-region configuration that identifies the two permitted regions. Cloud Storage then stores object data redundantly across those regions and manages the replication infrastructure automatically, rather than requiring the application or operations team to copy objects between separate buckets.

This design provides resilient access if infrastructure in one of the selected regions becomes unavailable, while keeping the data-location scope aligned with the organization's regional approval requirement. It also supplies a single bucket namespace and endpoint for applications, simplifying object management and avoiding the consistency, operational scheduling, and failure-handling responsibilities that accompany an application-managed replication design. For stronger cross-region recovery objectives, Cloud Storage also supports turbo replication for eligible dual-region buckets, with a recovery point objective target for newly written objects. Google documents dual-region buckets as a location type that stores data redundantly in two regions and is appropriate when data must be geographically distributed across a specific pair of regions. See Cloud Storage bucket locations and Cloud Storage availability and durability.

QUESTION NO: 11

Your organization sends IoT event data to a Pub/Sub topic. Subscriber applications read and perform transformations on the messages before storing them in the data warehouse. During particularly busy times when more data is being written to the topic, you notice that the subscriber applications are not acknowledging messages within the deadline. You need to modify your pipeline to handle these activity spikes and continue to process the messages. What should you do?

A. B Implement flow control on the subscribers

B. Forward unacknowledged messages to a dead-letter topic.

C. Seek back to the last acknowledged message.

ANSWER: A

Explanation:

Implementing flow control on the subscribers is the appropriate way to handle bursts that exceed the applications' processing capacity. Pub/Sub subscribers can otherwise continue receiving messages faster than transformations and warehouse writes can complete, causing a growing number of outstanding messages. When processing is slow enough, acknowledgement deadlines can expire before acknowledgements are sent, resulting in redelivery and further pressure on the subscriber workload.

Flow control sets limits on the number of outstanding messages and/or their total outstanding bytes. A subscriber pauses pulling additional messages after reaching its configured limit and resumes when previously received messages are acknowledged. This applies backpressure at the consumer, keeping the amount of in-flight work aligned with available CPU, memory, and downstream warehouse throughput. The applications can therefore process each received message reliably and acknowledge it within the deadline while Pub/Sub retains the backlog for later delivery. Google recommends configuring

subscriber flow control to prevent subscriber clients from being overwhelmed; autoscaling subscriber capacity can complement this approach when sustained traffic requires more processing resources.

See Pub/Sub flow control documentation and Google Cloud Pub/Sub subscriber documentation.

QUESTION NO: 12

You are working with a small dataset in Cloud Storage that needs to be transformed and loaded into BigQuery for analysis. The transformation involves simple filtering and aggregation operations. You want to use the most efficient and cost-effective data manipulation approach. What should you do?

A. Use Dataproc to create an Apache Hadoop cluster, perform the ETL process using Apache Spark, and load the results into BigQuery.

B. Use BigQuery's SQL capabilities to load the data from Cloud Storage, transform it, and store the results in a new BigQuery table.

C. Create a Cloud Data Fusion instance and visually design an ETL pipeline that reads data from Cloud Storage, transforms it using built-in transformations, and loads the results into BigQuery.

D. Use Dataflow to perform the ETL process that reads the data from Cloud Storage, transforms it using Apache Beam, and writes the results to BigQuery.

ANSWER: B

Explanation:

Use BigQuery's SQL capabilities to load the data from Cloud Storage, transform it, and store the results in a new BigQuery table. For a small dataset with straightforward filtering and aggregation, BigQuery provides a serverless approach that avoids provisioning, operating, and paying for dedicated processing infrastructure. The data can be made available to SQL through an external table over supported Cloud Storage file formats, or loaded into a native table first. A SQL query can then apply the required filters and aggregate functions, with the results persisted through a `CREATE TABLE AS SELECT` statement or a query destination table. This keeps ingestion, transformation, and analytics in the same managed data warehouse and uses familiar SQL rather than requiring a distributed processing pipeline. BigQuery pricing is based on storage and query processing, and query costs can be managed by selecting only required columns and processing only the small input dataset. Google documents querying Cloud Storage data with external tables at Query Cloud Storage data and creating a table from query results at Write query results.