Free Questions for Professional-Data-Engineer by certsinside

Shared by Ochoa on 24-05-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

You need to look at BigQuery data from a specific table multiple times a day. The underlying table you are querying is several petabytes in size, but you want to filter your data and provide simple aggregations to downstream users. You want to run queries faster and get up-to-date insights quicker. What should you do?

Options:

A- Run a scheduled query to pull the necessary data at specific intervals daily.

B- Create a materialized view based off of the query being run.

C- Use a cached query to accelerate time to results.

D- Limit the query columns being pulled in the final result.

Answer:

B

Explanation:

Materialized views are precomputed views that periodically cache the results of a query for increased performance and efficiency. BigQuery leverages precomputed results from materialized views and whenever possible reads only changes from the base tables to compute up-to-date results. Materialized views can significantly improve the performance of workloads that have the characteristic of common and repeated queries. Materialized views can also optimize queries with high computation cost and small dataset results, such as filtering and aggregating large tables. Materialized views are refreshed automatically when the base tables change, so they always return fresh data. Materialized views can also be used by the BigQuery optimizer to process queries to the base tables, if any part of the query can be resolved by querying the materialized view.Reference:

Introduction to materialized views

Create materialized views

BigQuery Materialized View Simplified: Steps to Create and 3 Best Practices

Materialized view in Bigquery

Question 2

Question Type: **MultipleChoice**

You have designed an Apache Beam processing pipeline that reads from a Pub/Sub topic. The topic has a message retention duration of one day, and writes to a Cloud Storage bucket. You need to select a bucket location and processing strategy to prevent data loss in case of a regional outage with an RPO of 15 minutes. What should you do?

Options:

A- 1 Use a regional Cloud Storage bucket
2 Monitor Dataflow metrics with Cloud Monitoring to determine when an outage occurs
3 Seek the subscription back in time by one day to recover the acknowledged messages
4 Start the Dataflow job in a secondary region and write in a bucket in the same region

B- 1 Use a multi-regional Cloud Storage bucket
2 Monitor Dataflow metrics with Cloud Monitoring to determine when an outage occurs
3 Seek the subscription back in time by 60 minutes to recover the acknowledged messages
4 Start the Dataflow job in a secondary region

C- 1. Use a dual-region Cloud Storage bucket.
2. Monitor Dataflow metrics with Cloud Monitoring to determine when an outage occurs
3 Seek the subscription back in time by 15 minutes to recover the acknowledged messages
4 Start the Dataflow job in a secondary region

D- 1. Use a dual-region Cloud Storage bucket with turbo replication enabled
2 Monitor Dataflow metrics with Cloud Monitoring to determine when an outage occurs
3 Seek the subscription back in time by 60 minutes to recover the acknowledged messages
4 Start the Dataflow job in a secondary region.

Answer:

C

Explanation:

A dual-region Cloud Storage bucket is a type of bucket that stores data redundantly across two regions within the same continent. This provides higher availability and durability than a regional bucket, which stores data in a single region. A dual-region bucket also provides lower latency and higher throughput than a multi-regional bucket, which stores data across multiple regions within a continent or across continents. A dual-region bucket with turbo replication enabled is a premium option that offers even faster replication across regions, but it is more expensive and not necessary for this scenario.

By using a dual-region Cloud Storage bucket, you can ensure that your data is protected from regional outages, and that you can access it from either region with low latency and high performance. You can also monitor the Dataflow metrics with Cloud Monitoring to determine when an outage occurs, and seek the subscription back in time by 15 minutes to recover the acknowledged messages. Seeking a subscription allows you to replay the messages from a Pub/Sub topic that were published within the message retention duration, which is one day in this case. By seeking the subscription back in time by 15 minutes, you can meet the RPO of 15 minutes, which means the maximum amount of data loss that is acceptable for your business. You can then start the Dataflow job in a secondary region and write to the same dual-region bucket, which will resume the processing of the messages and prevent data loss.

Option A is not a good solution, as using a regional Cloud Storage bucket does not provide any redundancy or protection from regional outages. If the region where the bucket is located experiences an outage, you will not be able to access your data or write new data to the bucket. Seeking the subscription back in time by one day is also unnecessary and inefficient, as it will replay all the messages from the past day, even though you only need to recover the messages from the past 15 minutes.

Option B is not a good solution, as using a multi-regional Cloud Storage bucket does not provide the best performance or cost-efficiency for this scenario. A multi-regional bucket stores data across multiple regions within a continent or across continents, which provides higher availability and durability than a dual-region bucket, but also higher latency and lower throughput. A multi-regional bucket is more suitable for serving data to a global audience, not for processing data with Dataflow within a single continent. Seeking the subscription back in time by 60 minutes is also unnecessary and inefficient, as it will replay more messages than needed to meet the RPO of 15

minutes.

Option D is not a good solution, as using a dual-region Cloud Storage bucket with turbo replication enabled does not provide any additional benefit for this scenario, but only increases the cost. Turbo replication is a premium option that offers faster replication across regions, but it is not required to meet the RPO of 15 minutes. Seeking the subscription back in time by 60 minutes is also unnecessary and inefficient, as it will replay more messages than needed to meet the RPO of 15 minutes.Reference:Storage locations | Cloud Storage | Google Cloud,Dataflow metrics | Cloud Dataflow | Google Cloud,Seeking a subscription | Cloud Pub/Sub | Google Cloud,Recovery point objective (RPO) | Acronis.

Question 3

Question Type: MultipleChoice

You have 100 GB of data stored in a BigQuery table. This data is outdated and will only be accessed one or two times a year for analytics with SQL. For backup purposes, you want to store this data to be immutable for 3 years. You want to minimize storage costs. What should you do?

Options:

A- 1 Create a BigQuery table clone.

2. Query the clone when you need to perform analytics.

B- 1 Create a BigQuery table snapshot.
2 Restore the snapshot when you need to perform analytics.

C- 1. Perform a BigQuery export to a Cloud Storage bucket with archive storage class.
2 Enable versionmg on the bucket.
3. Create a BigQuery external table on the exported files.

D- 1 Perform a BigQuery export to a Cloud Storage bucket with archive storage class.
2 Set a locked retention policy on the bucket.
3. Create a BigQuery external table on the exported files.

Answer:

D

Explanation:

This option will allow you to store the data in a low-cost storage option, as the archive storage class has the lowest price per GB among the Cloud Storage classes. It will also ensure that the data is immutable for 3 years, as the locked retention policy prevents the deletion or overwriting of the data until the retention period expires. You can still query the data using SQL by creating a BigQuery external table that references the exported files in the Cloud Storage bucket. Option A is incorrect because creating a BigQuery table clone will not reduce the storage costs, as the clone will have the same size and storage class as the original table. Option B is incorrect because creating a BigQuery table snapshot will also not reduce the storage costs, as the snapshot will have the same size and storage class as the original table. Option C is incorrect because enabling versioning on the bucket will not make the data immutable, as the versions can still be deleted or overwritten by anyone with the appropriate permissions. It will also increase the storage costs, as each version of the

file will be charged separately.Reference:

Exporting table data | BigQuery | Google Cloud

Storage classes | Cloud Storage | Google Cloud

Retention policies and retention periods | Cloud Storage | Google Cloud

Federated queries | BigQuery | Google Cloud

Question 4

Question Type: MultipleChoice

You work for a large ecommerce company. You store your customers order data in Bigtable. You have a garbage collection policy set to delete the data after 30 days and the number of versions is set to 1. When the data analysts run a query to report total customer spending, the analysts sometimes see customer data that is older than 30 days. You need to ensure that the analysts do not see customer data older than 30 days while minimizing cost and overhead. What should you do?

Options:

A- Set the expiring values of the column families to 30 days and set the number of versions to 2.

B- Use a timestamp range filter in the query to fetch the customer's data for a specific range.

C- Set the expiring values of the column families to 29 days and keep the number of versions to 1.

D- Schedule a job daily to scan the data in the table and delete data older than 30 days.

Answer:

B

Explanation:

By using a timestamp range filter in the query, you can ensure that the analysts only see the customer data that is within the desired time range, regardless of the garbage collection policy1. This option is the most cost-effective and simple way to avoid fetching data that is marked for deletion by garbage collection, as it does not require changing the existing policy or creating additional jobs.You can use the Bigtable client libraries or the cbt CLI to apply a timestamp range filter to your read requests2.

Option A is not effective, as it increases the number of versions to 2, which may cause more data to be retained and increase the storage costs. Option C is not reliable, as it reduces the expiring values to 29 days, which may not match the actual data arrival and usage patterns. Option D is not efficient, as it requires scheduling a job daily to scan and delete the data, which may incur additional overhead and complexity.Moreover, none of these options guarantee that the data older than 30 days will be immediately deleted, as garbage collection is an asynchronous process that can take up to a week to remove the data3.Reference:

1: Filters | Cloud Bigtable Documentation | Google Cloud

2: Read data | Cloud Bigtable Documentation | Google Cloud

3: Garbage collection overview | Cloud Bigtable Documentation | Google Cloud

Question 5

Question Type: MultipleChoice

You are configuring networking for a Dataflow job. The data pipeline uses custom container images with the libraries that are required for the transformation logic preinstalled. The data pipeline reads the data from Cloud Storage and writes the data to BigQuery. You need to ensure cost-effective and secure communication between the pipeline and Google APIs and services. What should you do?

Options:

A- Leave external IP addresses assigned to worker VMs while enforcing firewall rules.

B- Disable external IP addresses and establish a Private Service Connect endpoint IP address.

C- Disable external IP addresses from worker VMs and enable Private Google Access.

D- Enable Cloud NAT to provide outbound internet connectivity while enforcing firewall rules.

Answer:

C

Explanation:

Private Google Access allows VMs without external IP addresses to communicate with Google APIs and services over internal routes. This reduces the cost and increases the security of the data pipeline. Custom container images can be stored in Container Registry, which supports Private Google Access. Dataflow supports Private Google Access for both batch and streaming jobs.Reference:

Private Google Access overview

Using Private Google Access and Cloud NAT

Using custom containers with Dataflow

Question 6

Question Type: MultipleChoice

You are troubleshooting your Dataflow pipeline that processes data from Cloud Storage to BigQuery. You have discovered that the Dataflow worker nodes cannot communicate with one another Your networking team relies on Google Cloud network tags to define firewall rules You need to identify the issue while following Google-recommended networking security practices. What should you do?

Options:

A- Determine whether your Dataflow pipeline has a custom network tag set.

B- Determine whether there is a firewall rule set to allow traffic on TCP ports 12345 and 12346 for the Dataflow network tag.

C- Determine whether your Dataflow pipeline is deployed with the external IP address option enabled.

D- Determine whether there is a firewall rule set to allow traffic on TCP ports 12345 and 12346 on the subnet used by Dataflow workers.

Answer:

B

Explanation:

Dataflow worker nodes need to communicate with each other and with the Dataflow service on TCP ports 12345 and 12346. These ports are used for data shuffling and streaming engine communication. By default, Dataflow assigns a network tag called dataflow to the worker nodes, and creates a firewall rule that allows traffic on these ports for the dataflow network tag. However, if you use a custom network tag for your Dataflow pipeline, you need to create a firewall rule that allows traffic on these ports for your custom network tag. Otherwise, the worker nodes will not be able to communicate with each other and the Dataflow service, and the pipeline will fail.

Therefore, the best way to identify the issue is to determine whether there is a firewall rule set to allow traffic on TCP ports 12345 and 12346 for the Dataflow network tag. If there is no such firewall rule, or if the firewall rule does not match the network tag used by your Dataflow pipeline, you need to create or update the firewall rule accordingly.

Option A is not a good solution, as determining whether your Dataflow pipeline has a custom network tag set does not tell you whether there is a firewall rule that allows traffic on the required ports for that network tag. You need to check the firewall rule as well.

Option C is not a good solution, as determining whether your Dataflow pipeline is deployed with the external IP address option enabled does not tell you whether there is a firewall rule that allows traffic on the required ports for the Dataflow network tag. The external IP address option determines whether the worker nodes can access resources on the public internet, but it does not affect the internal communication between the worker nodes and the Dataflow service.

Option D is not a good solution, as determining whether there is a firewall rule set to allow traffic on TCP ports 12345 and 12346 on the subnet used by Dataflow workers does not tell you whether the firewall rule applies to the Dataflow network tag. The firewall rule should be based on the network tag, not the subnet, as the network tag is more specific and secure.Reference:Dataflow network tags | Cloud Dataflow | Google Cloud,Dataflow firewall rules | Cloud Dataflow | Google Cloud,Dataflow network configuration | Cloud Dataflow | Google Cloud,Dataflow Streaming Engine | Cloud Dataflow | Google Cloud.

Question 7

Question Type: **MultipleChoice**

You use a dataset in BigQuery for analysis. You want to provide third-party companies with access to the same dataset. You need to keep the costs of data sharing low and ensure that the data is current. What should you do?

Options:

A- Use Analytics Hub to control data access, and provide third party companies with access to the dataset

B- Create a Dataflow job that reads the data in frequent time intervals and writes it to the relevant BigQuery dataset or Cloud Storage bucket for third-party companies to use.

C- Use Cloud Scheduler to export the data on a regular basis to Cloud Storage, and provide third-party companies with access to the bucket.

D- Create a separate dataset in BigQuery that contains the relevant data to share, and provide third-party companies with access to the new dataset.

Answer:

A

Explanation:

Analytics Hub is a service that allows you to securely share and discover data assets across your organization and with external partners. You can use Analytics Hub to create and manage data assets, such as BigQuery datasets, views, and queries, and control who can access them. You can also browse and use data assets that others have shared with you. By using Analytics Hub, you can keep the costs of data sharing low and ensure that the data is current, as the data assets are not copied or moved, but rather referenced from their original sources.

Question 8

Question Type: MultipleChoice

You work for a large ecommerce company. You are using Pub/Sub to ingest the clickstream data to Google Cloud for analytics. You observe that when a new subscriber connects to an existing topic to analyze data, they are unable to subscribe to older data for an upcoming yearly sale event in two months, you need a solution that, once implemented, will enable any new subscriber to read the last 30 days of dat

a. What should you do?

Options:

A- Create a new topic, and publish the last 30 days of data each time a new subscriber connects to an existing topic.

B- Set the topic retention policy to 30 days.

C- Set the subscriber retention policy to 30 days.

D- Ask the source system to re-push the data to Pub/Sub, and subscribe to it.

Answer:

B

Explanation:

By setting the topic retention policy to 30 days, you can ensure that any new subscriber can access the messages that were published to the topic within the last 30 days1.This feature allows you to replay previously acknowledged messages or initialize new subscribers with historical data2.You can configure the topic retention policy by using the Cloud Console, the gcloud command-line tool, or the Pub/Sub API1.

Option A is not efficient, as it requires creating a new topic and duplicating the data for each new subscriber, which would increase the storage costs and complexity.Option C is not effective, as it only affects the unacknowledged messages in a subscription, and does not allow new subscribers to access older data3. Option D is not feasible, as it depends on the source system's ability and willingness to re-push the data, and it may cause data duplication or inconsistency.Reference:

1: Create a topic | Cloud Pub/Sub Documentation | Google Cloud

2: Replay and purge messages with seek | Cloud Pub/Sub Documentation | Google Cloud

3: When is a PubSub Subscription considered to be inactive?

Question 9

Question Type: MultipleChoice

You are running a streaming pipeline with Dataflow and are using hopping windows to group the data as the data arrives. You noticed that some data is arriving late but is not being marked as late data, which is resulting in inaccurate aggregations downstream. You need to find a solution that allows you to capture the late data in the appropriate window. What should you do?

Options:

A- Change your windowing function to session windows to define your windows based on certain activity.

B- Change your windowing function to tumbling windows to avoid overlapping window periods.

C- Expand your hopping window so that the late data has more time to arrive within the grouping.

D- Use watermarks to define the expected data arrival window Allow late data as it arrives.

Answer:

D

Explanation:

Watermarks are a way of tracking the progress of time in a streaming pipeline. They are used to determine when a window can be closed and the results emitted. Watermarks can be either event-time based or processing-time based. Event-time watermarks track the progress of time based on the timestamps of the data elements, while processing-time watermarks track the progress of time based on the system clock. Event-time watermarks are more accurate, but they require the data source to provide reliable timestamps. Processing-time watermarks are simpler, but they can be affected by system delays or backlogs.

By using watermarks, you can define the expected data arrival window for each windowing function. You can also specify how to handle late data, which is data that arrives after the watermark has passed. You can either discard late data, or allow late data and update the results as new data arrives. Allowing late data requires you to use triggers to control when the results are emitted.

In this case, using watermarks and allowing late data is the best solution to capture the late data in the appropriate window. Changing the windowing function to session windows or tumbling windows will not solve the problem of late data, as they still rely on watermarks to determine when to close the windows. Expanding the hopping window might reduce the amount of late data, but it will also change the semantics of the windowing function and the results.

Streaming pipelines | Cloud Dataflow | Google Cloud

Windowing | Apache Beam

Question 10

Question Type: **MultipleChoice**

You have terabytes of customer behavioral data streaming from Google Analytics into BigQuery daily Your customers' information, such as their preferences, is hosted on a Cloud SQL for MySQL database Your CRM database is hosted on a Cloud SQL for PostgreSQL instance. The marketing team wants to use your customers' information from the two databases and the customer behavioral data to create marketing campaigns for yearly active customers. You need to ensure that the marketing team can run the campaigns over 100 times a day on typical days and up to 300 during sales. At the same time you want to keep the load on the Cloud SQL databases to a minimum. What should you do?

Options:

A- Create BigQuery connections to both Cloud SQL databases Use BigQuery federated queries on the two databases and the Google Analytics data on BigQuery to run these queries.

B- Create streams in Datastream to replicate the required tables from both Cloud SQL databases to BigQuery for these queries.

C- Create a Dataproc cluster with Trino to establish connections to both Cloud SQL databases and BigQuery, to execute the queries.

D- Create a job on Apache Spark with Dataproc Serverless to query both Cloud SQL databases and the Google Analytics data on BigQuery for these queries.

Answer:

B

Explanation:

Datastream is a serverless Change Data Capture (CDC) and replication service that allows you to stream data changes from Oracle and MySQL databases to Google Cloud services such as BigQuery, Cloud Storage, Cloud SQL, and Pub/Sub. Datastream captures and delivers database changes in real-time, with minimal impact on the source database performance. Datastream also preserves the schema and data types of the source database, and automatically creates and updates the corresponding tables in BigQuery.

By using Datastream, you can replicate the required tables from both Cloud SQL databases to BigQuery, and keep them in sync with the source databases. This way, you can reduce the load on the Cloud SQL databases, as the marketing team can run their queries on the

BigQuery tables instead of the Cloud SQL tables. You can also leverage the scalability and performance of BigQuery to query the customer behavioral data from Google Analytics and the customer information from the replicated tables. You can run the queries as frequently as needed, without worrying about the impact on the Cloud SQL databases.

Option A is not a good solution, as BigQuery federated queries allow you to query external data sources such as Cloud SQL databases, but they do not reduce the load on the source databases. In fact, federated queries may increase the load on the source databases, as they need to execute the query statements on the external data sources and return the results to BigQuery. Federated queries also have some limitations, such as data type mappings, quotas, and performance issues.

Option C is not a good solution, as creating a Dataproc cluster with Trino would require more resources and management overhead than using Datastream. Trino is a distributed SQL query engine that can connect to multiple data sources, such as Cloud SQL and BigQuery, and execute queries across them. However, Trino requires a Dataproc cluster to run, which means you need to provision, configure, and monitor the cluster nodes. You also need to install and configure the Trino connector for Cloud SQL and BigQuery, and write the queries in Trino SQL dialect. Moreover, Trino does not replicate or sync the data from Cloud SQL to BigQuery, so the load on the Cloud SQL databases would still be high.

Option D is not a good solution, as creating a job on Apache Spark with Dataproc Serverless would require more coding and processing power than using Datastream. Apache Spark is a distributed data processing framework that can read and write data from various sources, such as Cloud SQL and BigQuery, and perform complex transformations and analytics on them. Dataproc Serverless is a serverless Spark service that allows you to run Spark jobs without managing clusters. However, Spark requires you to write code in Python, Scala, Java, or R, and use the Spark connector for Cloud SQL and BigQuery to access the data sources. Spark also does not replicate or sync the data from Cloud SQL to BigQuery, so the load on the Cloud SQL databases would still be high.Reference:Datastream overview | Datastream | Google Cloud,Datastream concepts | Datastream | Google Cloud,Datastream quickstart | Datastream | Google Cloud,Introduction to federated queries | BigQuery | Google Cloud,Trino overview | Dataproc Documentation | Google Cloud,Dataproc Serverless overview | Dataproc Documentation | Google Cloud,Apache Spark overview | Dataproc Documentation | Google Cloud.

Question 11

Question Type: MultipleChoice

You work for an airline and you need to store weather data in a BigQuery table Weather data will be used as input to a machine learning model. The model only uses the last 30 days of weather dat

a. You want to avoid storing unnecessary data and minimize costs. What should you do?

Options:

A- Create a BigQuery table where each record has an ingestion timestamp Run a scheduled query to delete all the rows with an ingestion timestamp older than 30 days.

B- Create a BigQuery table partitioned by ingestion time Set up partition expiration to 30 days.

C- Create a BigQuery table partitioned by datetime value of the weather date Set up partition expiration to 30 days.

D- Create a BigQuery table with a datetime column for the day the weather data refers to. Run a scheduled query to delete rows with a datetime value older than 30 days.

Answer:

B

Explanation:

Partitioning a table by ingestion time means that the data is divided into partitions based on the time when the data was loaded into the table. This allows you to delete or archive old data by setting a partition expiration policy. You can specify the number of days to keep the data in each partition, and BigQuery automatically deletes the data when it expires. This way, you can avoid storing unnecessary data and minimize costs.

Question 12

Question Type: MultipleChoice

You are designing a data mesh on Google Cloud with multiple distinct data engineering teams building data products. The typical data curation design pattern consists of landing files in Cloud Storage, transforming raw data in Cloud Storage and BigQuery datasets. and storing the final curated data product in BigQuery datasets You need to configure Dataplex to ensure that each team can access only the assets needed to build their data products. You also need to ensure that teams can easily share the curated data product. What should you do?

Options:

A- 1 Create a single Dataplex virtual lake and create a single zone to contain landing, raw. and curated data.
2 Provide each data engineering team access to the virtual lake.

B- 1 Create a single Dataplex virtual lake and create a single zone to contain landing, raw. and curated data. 2 Build separate assets for each data product within the zone.
3. Assign permissions to the data engineering teams at the zone level.

C- 1 Create a Dataplex virtual lake for each data product, and create a single zone to contain landing, raw, and curated data.
2. Provide the data engineering teams with full access to the virtual lake assigned to their data product.

D- 1 Create a Dataplex virtual lake for each data product, and create multiple zones for landing, raw. and curated data.
2. Provide the data engineering teams with full access to the virtual lake assigned to their data product.

Answer:

D

Explanation:

This option is the best way to configure Dataplex for a data mesh architecture, as it allows each data engineering team to have full ownership and control over their data products, while also enabling easy discovery and sharing of the curated data across the organization12.By creating a Dataplex virtual lake for each data product, you can isolate the data assets and resources for each domain, and avoid conflicts and dependencies between different teams3.By creating multiple zones for landing, raw, and curated data, you can enforce different security and governance policies for each stage of the data curation process, and ensure that only authorized users can access the data assets45. By providing the data engineering teams with full access to the virtual lake assigned to their data product, you can empower them to manage and monitor their data products, and leverage the Dataplex features such as tagging, quality, and lineage.

Option A is not suitable, as it creates a single point of failure and a bottleneck for the data mesh, and does not allow for fine-grained access control and governance for different data products2.Option B is also not suitable, as it does not isolate the data assets and resources for each data product, and assigns permissions at the zone level, which may not reflect the different roles and responsibilities of the data engineering teams34.Option C is better than option A and B, but it does not create multiple zones for landing, raw, and curated data, which may compromise the security and quality of the data products5.Reference:

1: Building a data mesh on Google Cloud using BigQuery and Dataplex | Google Cloud Blog

2: Data Mesh - 7 Effective Practices to Get Started - Confluent

3: Best practices | Dataplex | Google Cloud

4: Secure your lake | Dataplex | Google Cloud

5: Zones | Dataplex | Google Cloud

[6]: Managing a Data Mesh with Dataplex -- ROI Training