Exam Professional Data Engineer All Questions
View all questions & answers for the Professional Data Engineer exam

Go to Exam

EXAM PROFESSIONAL DATA ENGINEER TOPIC 1 QUESTION 28 DISCUSSION

Actual exam question from Google's Professional Data Engineer
Question #: 28
Topic #: 1
[All Professional Data Engineer Questions]

Your company is performing data preprocessing for a learning algorithm in Google Cloud Dataflow. Numerous data logs are being are being generated during this step, and the team wants to analyze them. Due to the dynamic nature of the campaign, the data is growing exponentially every hour.
The data scientists have written the following code to read the data for a new key features in the logs.

```
BigQueryIO.Read
      .named("ReadLogData")
      .from("clouddataflow-readonly:samples.log_data")
```

You want to improve the performance of this data read. What should you do?

A. Specify the TableReference object in the code.

B. Use .fromQuery operation to read specific fields from the table.

C. Use of both the Google BigQuery TableSchema and TableFieldSchema classes.

D. Call a transform that returns TableRow objects, where each element in the PCollection represents a single row in the table.

Show Suggested Answer

by arthur2385 at *Sept. 2, 2022, 1:45 p.m.*

Comments

Type your comment...

Submit

arthur2385 Highly Voted 2 years, 2 months ago

B BigQueryIO.read.fromQuery() executes a query and then reads the results received after the query execution. Therefore, this function is more time-consuming, given that it requires that a query is first executed (which will incur in the corresponding economic and computational costs).

upvoted 12 times

maxdataengineer Highly Voted 2 years ago

Since we want to be able to analyze data from a new ML feature (column) we only need to check values from that column. By doing a fromQuery(SELECT featueColum FROM table)
we are optimizing costs and performance since we are not checking all columns.

https://cloud.google.com/bigquery/docs/best-practices-costs#avoid_select_

upvoted 7 times

maxdataengineer 2 years ago

The answer is B

upvoted 2 times

cetanx 1 year, 5 months ago

According to Chat GPT, it is also B
In general, if your "primary goal is to reduce the amount of data read and transferred", and the downstream processing mainly focuses on a subset of fields, using .fromQuery to select specific fields would be a good choice.

On the other hand, if you need to simplify downstream processing and optimize resource utilization, transforming data into TableRow objects might be more suitable.

upvoted 3 times

MaxNRG Most Recent 10 months, 3 weeks ago

Selected Answer: B

B as BigQueryIO.read.from() directly reads the whole table from BigQuery.
This function exports the whole table to temporary files in Google Cloud Storage, where it will later be read from.
This requires almost no computation, as it only performs an export job, and later Dataflow reads from GCS (not from BigQuery).
BigQueryIO.read.fromQuery() executes a query and then reads the results received after the query execution. Therefore, this function is more time-consuming, given that it requires that a query is first executed (which will incur in the corresponding economic and computational costs).
https://stackoverflow.com/questions/54413681/bigqueryio-read-vs-fromquery

upvoted 1 times

axantroff 11 months, 2 weeks ago

Selected Answer: B

B works for me

upvoted 1 times

pue_dev_anon 11 months, 2 weeks ago

Selected Answer: B

We are trying to optimize reading each row is not optimal, we want columns

upvoted 1 times

rtcpost 1 year ago

Selected Answer: B

B. Use the .fromQuery operation to read specific fields from the table.

Using the .fromQuery operation allows you to specify the exact fields you need to read from the table, which can significantly improve performance by reducing the amount of data that needs to be processed. This is particularly important when dealing with large and growing datasets.

Option A (specifying the TableReference object) provides information about the table but doesn't inherently improve the performance of reading specific fields.

Option C (using Google BigQuery TableSchema and TableFieldSchema classes) is related to specifying the schema of the data but doesn't directly address improving the performance of reading specific fields.

Option D (calling a transform that returns TableRow objects) is more about how the data is processed after it's read, not how it's initially read from BigQuery.

upvoted 5 times

emmylou 1 year, 1 month ago

When I have a different answer then the "Correct Answer", I run it through AI and it keeps saying ExamTopics is wrong. Is there any way to know if I am going to pass or fail this exam?

upvoted 2 times

axantroff 11 months, 2 weeks ago

AI is just a LLM model, not a silver bullet at all

upvoted 1 times

suku2 1 year, 1 month ago

Selected Answer: B

Since the requirement is to read the data for a *new* key features in the logs, it makes sense to select limited columns, which are required rather than using .from() method which exports the entire BigQuery table.
B makes sense here.

upvoted 1 times

gudguy1a 1 year, 2 months ago

Selected Answer: B

SHOULD be B.
Not quite sure how D is the correct answer (Red herring....?) when you want to improve the query, which is .fromQuery and NOT transform and PCollection....

upvoted 1 times

odiez3 1 year, 3 months ago

Answer Is D, imagine that you dont have permission on BQ AND you cant see the table info or anything else about the table you only aré working whit dataflow the only way Is transform the data using apache beam

upvoted 1 times

Mathew106 1 year, 3 months ago

Selected Answer: B

I have seen people explain why B is not right because it doesn't optimize performance but only cost, which is not true, or because fromQuery is still not performant.

I think it's B because no other option is more performant, even if you claim it's not good.

As for option D, the transform given by the description is already a transform that provides as output a PCollection of TableRow objects. So how would that be any different?

https://beam.apache.org/releases/javadoc/2.1.0/org/apache/beam/sdk/io/gcp/bigquery/BigQueryIO.html

upvoted 1 times

theseawillclaim 1 year, 3 months ago

Why should it be D?
"fromQuery()" allows us to read only the columns we want, I see no point in using a Transform for each row of a "SELECT *", which, moreover, is a bad BQ Practice.

upvoted 1 times

jkh_goh 1 year, 9 months ago

Selected Answer: B

Does BigQuery have a pCollections? I thought it's unique to Apache Beam i.e. Cloud Dataflow

upvoted 1 times

kelvintoys93 1 year, 11 months ago

Guys, how is B the answer? Like all the justifications given here, BigQueryIO.read.fromQuery() is time consuming and the question asked for a better performance solution.

upvoted 4 times

Lestrang 1 year, 9 months ago

That part is the docs trying to explain the side effects of using it, however, the part that is important to us is the fact that it reads from a query. "Read" reads the whole table. If we specify a query we can say select col1 only, which makes it all more efficient.

upvoted 2 times

gcm7 2 years ago

Selected Answer: B

reading only relevant cols

upvoted 6 times

devaid 2 years ago

Selected Answer: D

Answer is D, apparently.

upvoted 1 times

Kowalski 2 years, 1 month ago

Answer is Use .fromQuery operation to read specific fields from the table.

BigQueryIO.read.from() directly reads the whole table from BigQuery. This function exports the whole table to temporary files in Google Cloud Storage, where it will later be read from. This requires almost no computation, as it only performs an export job, and later Dataflow reads from GCS (not from BigQuery).

BigQueryIO.read.fromQuery() executes a query and then reads the results received after the query execution. Therefore, this function is more time-consuming, given that it requires that a query is first executed (which will incur in the corresponding economic and computational costs).

Reference:
https://cloud.google.com/bigquery/docs/best-practices-costs#avoid_select_

upvoted 3 times

Load full discussion...

EXAMTOPICS

Platform

Home

Examtopics PRO

All Exams

Training Courses

© 2024 ExamTopics