EXAMTOPICS

- Expert Verified, Online, **Free.**

☰ MENU   🔍

← **Google Discussions**

Exam Professional Machine Learning Engineer All Questions

View all questions & answers for the Professional Machine Learning Engineer exam

Go to Exam

📄 **EXAM PROFESSIONAL MACHINE LEARNING ENGINEER TOPIC 1 QUESTION 35 DISCUSSIO..**

Actual exam question from Google's Professional Machine Learning Engineer

Question #: 35

Topic #: 1

[All Professional Machine Learning Engineer Questions]

You are developing a Kubeflow pipeline on Google Kubernetes Engine. The first step in the pipeline is to issue a query against BigQuery. You plan to use the results of that query as the input to the next step in your pipeline. You want to achieve this in the easiest way possible. What should you do?

A. Use the BigQuery console to execute your query, and then save the query results into a new BigQuery table.

B. Write a Python script that uses the BigQuery API to execute queries against BigQuery. Execute this script as the first step in your Kubeflow pipeline.

C. Use the Kubeflow Pipelines domain-specific language to create a custom component that uses the Python BigQuery client library to execute queries.

D. Locate the Kubeflow Pipelines repository on GitHub. Find the BigQuery Query Component, copy that component's URL, and use it to load the component into your pipeline. Use the component to execute queries against BigQuery.

Show Suggested Answer

by 👤 chohan at *June 18, 2021, 2:09 p.m.*

Comments

Type your comment...

maartenalexander `Highly Voted 👍` 3 years, 4 months ago

D. Kubeflow pipelines have different types of components, ranging from low- to high-level. They have a ComponentStore that allows you to access prebuilt functionality from GitHub.

👍 ↩ 🚩 upvoted 21 times

---

gcp2021go 3 years, 2 months ago

agree, links: https://github.com/kubeflow/pipelines/blob/master/components/gcp/bigquery/query/sample.ipynb; https://v0-5.kubeflow.org/docs/pipelines/reusable-components/

👍 ↩ 🚩 upvoted 6 times

NamitSehgal `Highly Voted 👍` 2 years, 9 months ago

Selected Answer: D

Not sure what is the reason behind putting A as it is manual and manual steps can not be part of automation. I would say Answer is D as it just require a clone of the component from github. Using a Python and import bigquery component may sounds good too, but ask was what is easiest. It depends how word "easy" is taken by individuals but definitely not A.

👍 ↩ 🚩 upvoted 6 times

---

taksan `Most Recent ⊘` 2 months ago

Selected Answer: D

D is the correct answer, as reusing an existing component is the most streamlined way to interact with BigQuery.

👍 ↩ 🚩 upvoted 1 times

---

nktyagi 2 months, 2 weeks ago

Selected Answer: B

much simpler to just write a couple of lines of python

👍 ↩ 🚩 upvoted 1 times

---

desertlotus1211 21 hours, 53 minutes ago

Writing a Python script using the BigQuery API is possible, but it's more complex than using an existing component. It requires more development effort and doesn't take advantage of the pre-built components available in Kubeflow.

👍 ↩ 🚩 upvoted 1 times

---

jsalvasoler 2 months, 2 weeks ago

Selected Answer: B

Clearly B

👍 ↩ 🚩 upvoted 1 times

PhilipKoku 4 months, 2 weeks ago

Selected Answer: B

B) Python API

👍 ↩ 🚩 upvoted 3 times

Amabo 5 months, 2 weeks ago

from kfp.components import load_component_from_url

bigquery_query_op =
load_component_from_url('https://raw.githubusercontent.com/kubeflow/pipelines/master/components/gcp/bigquery/query/component.yaml')

def my_pipeline():
query_result = bigquery_query_op(
project_id='my-project',
query='SELECT * FROM my_dataset.my_table'
)
Use the query_result as input to the next step in the pipeline

👍 ↩ 🚩 upvoted 2 times

---

fragkris 10 months, 3 weeks ago

Selected Answer: B

Im going "against the flow" and chosing B. It just sounds a lot easier option than D.

👍 ↩ 🚩 upvoted 3 times

---

friedi 1 year, 4 months ago

Selected Answer: B

Very confused as to why D is the correct answer. To me it seems a) much simpler to just write a couple of lines of python (https://cloud.google.com/bigquery/docs/reference/libraries#client-libraries-install-python) and b) the documentation for the

(https://cloud.google.com/bigquery/docs/reference/libraries#client-libraries-install-python) and b) the documentation for the BigQuery reusable component (https://v0-5.kubeflow.org/docs/pipelines/reusable-components/) states that the data is written to Google Cloud Storage, which means we have to write the fetching logic in the next pipeline step, going against the "as simple as possible" requirement. Would be interested to hear why I am wrong.

👍 ↩ 🏳 upvoted 2 times

⊟ 👤 **friedi** 1 year, 4 months ago

Actually, the problem statement even says that the query result has to be used as input to the next step, meaning with answer D) we would have to download the results before passing them to the next step. Additionally, we would have to handle potentially existing files in Google Cloud Storage if the pipeline is either executed multiple times or even in parallel. (I will die on this hill 😄).

👍 ↩ 🏳 upvoted 2 times

⊟ 👤 **tavva_prudhvi** 11 months, 3 weeks ago

Yup, you raised valid points. Depending on your specific requirements and familiarity with Python, writing a custom script using the BigQuery API (Option B) can be a simpler and more flexible approach.

With Option B, you can write a Python script that uses the BigQuery API to execute queries against BigQuery and fetch the data directly into your pipeline. This way, you can process the data as needed and pass it to the next step in the pipeline without the need to fetch it from Google Cloud Storage.

While using the reusable BigQuery Query Component (Option D) provides a pre-built solution, it does require additional steps to fetch the data from Google Cloud Storage for the next step in the pipeline, which might not be the simplest approach.

👍 ↩ 🏳 upvoted 2 times

⊟ 👤 **M25** 1 year, 5 months ago

Selected Answer: D

Went with D

👍 ↩ 🏳 upvoted 1 times

⊟ 👤 **Mohamed_Mossad** 2 years, 3 months ago

Selected Answer: D

https://linuxtut.com/en/f4771efee37658c083cc/

👍 ↩ 🏳 upvoted 1 times

⊟ 👤 **Mohamed_Mossad** 2 years, 3 months ago

answer between C,D but above link has an article which uses a ready .yml file for bigquery component on official kubeflow pipelines repo

👍 ↩ 🏳 upvoted 1 times

⊟ 👤 **David_ml** 2 years, 5 months ago

Selected Answer: D

Answer is D.

👍 ↩ 🏳 upvoted 2 times

⊟ 👤 **donchoripan** 2 years, 6 months ago

A. it says the easiest way possible so it sounds like just running the query on the console should be enogh. It doesn't says that the data will need to be uploaded again anytime soon, so we can asume that its just a one time query to be run.

👍 ↩ 🏳 upvoted 1 times

⊟ 👤 **David_ml** 2 years, 5 months ago

A is wrong. Answer is D. It's a pipeline which means you will run it multiple times? Do you always want to make the query manually each time you run your pipeline?

👍 ↩ 🏳 upvoted 3 times

⊟ 👤 **xiaoF** 2 years, 8 months ago

D is good.

👍 ↩ 🏳 upvoted 2 times

⊟ 👤 **aepos** 2 years, 10 months ago

The result of D is just the path to the Cloud Storage where the result is stored not the data itself. So the input to the next step is this path, where you still have to load the data? So i would guess B. Can anyone explain if i am wrong?

👍 ↩ 🏳 upvoted 2 times

⊟ 👤 **kaike_reis** 2 years, 11 months ago

D. The easiest way possible in developer's world: copy code from stackoverflow or github hahaha. Jokes a part, I think D is the correct. (A) is manual, so you have to do always. (B) could be, but is not the easiest one because you need to write a script for this. (C) uses Kubeflow intern solution, but you need to work to create a custom component. (D) is the (C) solution, but easier using a component created previously to do the job.

👍 ↩ 🏳 upvoted 2 times

⊟ 👤 **celia20200410** 3 years, 3 months ago
ans: c
https://medium.com/google-cloud/using-bigquery-and-bigquery-ml-from-kubeflow-pipelines-991a2fa4bea8
https://cloud.google.com/architecture/architecture-for-mlops-using-tfx-kubeflow-pipelines-and-cloud-build#kubeflow-piplines-components
Kubeflow Pipelines, a containerized task can invoke other services such as BigQuery jobs, AI Platform (distributed) training jobs, and Dataflow jobs.

👍 ↩ 🏳 upvoted 1 times

⊟ 👤 **raviperi** 3 years, 1 month ago
why create a custom component when a big query's reusable component is already present. Answer is D.

👍 ↩ 🏳 upvoted 6 times

Load full discussion...

Start Learning for free