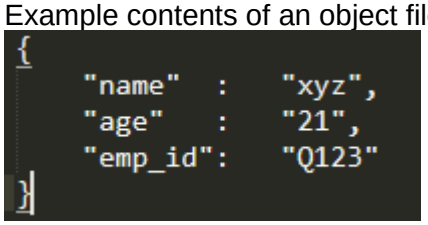
9. Create a Cloud Function which reads the messages from Pub/Sub subscription. The pubsub message contains the bucket name and object path of a json file. The Cloud Function should store the contents of the json file into an appropriate database. Use console. (Explore multiple ways of integrating cloud function with Pub/Sub)

Example contents of an object file(sample.json):



1. Create a bucket and upload the following .json file in it:
   {"name":"xyz","age":"21","emp_id":"Q123"}
2. In your GCP Console Navigation menu, under COMPUTE, click on Cloud Functions.
3. Click on CREATE FUNCTION.
4. Give it a suitable name and set the Trigger as Cloud Pub/Sub. In the dropdown list for Topic, click on 'Create new topic...'
5. Set Runtime as Python 3.7
6. In main.py and requirements.txt, enter the following code and package name(s) respectively:

   main.py

```
from google.cloud import bigquery

def hello_pubsub(event, context):
    bkt = event['attributes']['bkt']
    obj = event['attributes']['obj']

    client = bigquery.Client()
    data_set = client.dataset('av_a1_q9')
    job_conf = bigquery.LoadJobConfig()
    job_conf.autodetect = True
    job_conf.source_format = bigquery.SourceFormat.NEWLINE_DELIMITED_JSON

    uri  = "gs://" + bkt + "/" + obj
    load_job = client.load_table_from_uri(uri, data_set.table("q9table"), job_config=job_conf)
```

   requirements.txt

```
# Function dependencies, for example:
# package>=version
google-cloud-bigquery
```

7. Click on CREATE.
8. Now open the BigQuery console and create a new dataset.
9. Then, create a new table with three columns(name<varchar>, age<int>, emp_id<varchar>).
10. Now, go to Pub/Sub console and open the topic that you created while creating the cloud function.
11. Click on 'PUBLISH MESSAGE' and add the following attributes:
    a) bkt: <your-bucket-name>
    b) obj: <json-file-name>
12. Click on PUBLISH
13. Now open your BigQuery table and perform a select query to check the entrie(s) in the table:
    SELECT * FROM `pe-training.av_a1_q9.q9table` LIMIT 100