Data Engineering 2 Project Report – June 2021

Google Cloud Platform workflow

Bina Khatnani (11015617)

Motivation

Google cloud platform (GCP) is one of the innovative tools used widely for designing big data pipelines and performing analytics. Services like Big Query (Cloud warehousing), Cloud storage, Google cloud dataflow (real time batch processing), Pub Sub (continuous data streaming) and Data Studio (Data Visualization) are available in one platform. GCP gives us access to develop data pipelines with massive amount of data and perform visualization without need of integrating with other market tools.

Introduction

With the help of GCP API services like cloud storage, Pub/Subtopic, dataflow, big query and data studio created a continuous data pipeline using public dataset international education. Furthermore, the insights of the dataset are queried and visualized using interesting charts and graphical panels available within the data studio.

Dataset

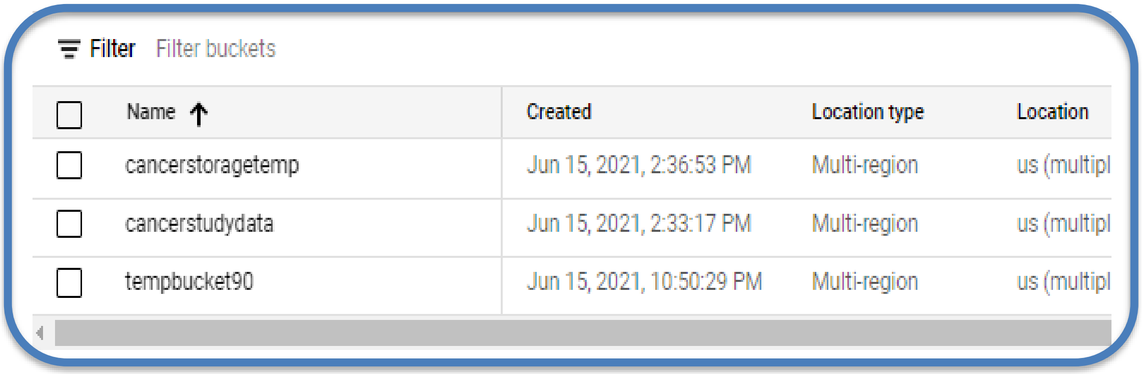
International Education is the GCP public dataset sourced from the world bank. The dataset combines key education statistics from a variety of sources to provide a look at global literacy, spending and access. The variables defined in the dataset are as below.

Variables	Data Type	Discription
country_name	STRING	Name of the Country
country_code	STRING	Code of the country
indicator_name	STRING	Factors associated with education status in the country
indicator_code	STRING	Unique code indicators for unemployment%, Gender wise education, Population, and overall GDP of the country
value	FLOAT	Values determining indicating factors in percentage or Number
year	INTEGER	Indicating the values in particular year

Application Design

The platform consists of 4 main components working in tandem to receive and store data regularly. The four components are **cloud storage, Dataflow from Pub/Sub to Big query, big query,** and **data studio.**

- **Cloud Storage:** Cloud storage gives user a paradigm to store the data in bucket.
  In the current context a bucket is created for storage of data extracted from public dataset and for storing temporary files. The cloud storage bucket is used to transfer the data to Big query.

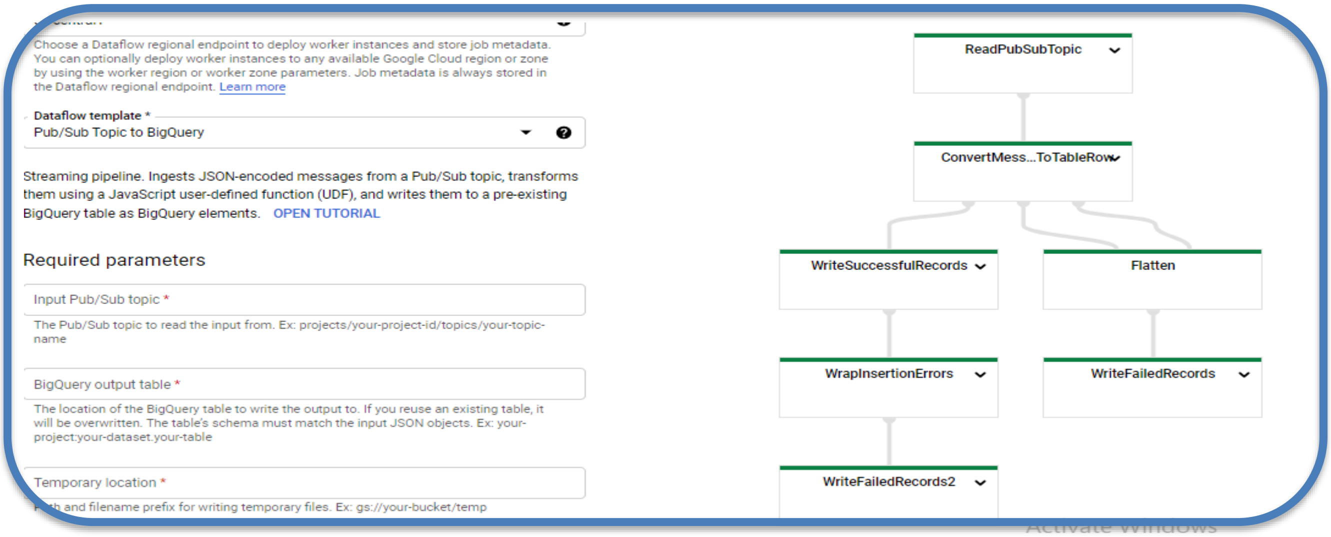


Filter Filter buckets

□	Name ↑	Created	Location type	Location
□	cancerstoragetemp	Jun 15, 2021, 2:36:53 PM	Multi-region	us (multipl
□	cancerstudydata	Jun 15, 2021, 2:33:17 PM	Multi-region	us (multipl
□	tempbucket90	Jun 15, 2021, 10:50:29 PM	Multi-region	us (multipl

Bucket repository in cloud storage

- **Dataflow from Pub/Sub to Big query:** Dataflow provides a platform to create jobs either by streaming data continuously or in batches. In the current context a dataflow is created for transferring data from cloud storage to big query. The pub/sub gives us an infrastructure to transfer the new incoming data on periodical basis by publishing the message manually in topic or by creating a python script for publishing the message automatically.



Choose a Dataflow regional endpoint to deploy worker instances and store job metadata. You can optionally deploy worker instances to any available Google Cloud region or zone by using the worker region or worker zone parameters. Job metadata is always stored in the Dataflow regional endpoint. Learn more

Dataflow template *
Pub/Sub Topic to BigQuery

Streaming pipeline. Ingests JSON-encoded messages from a Pub/Sub topic, transforms them using a JavaScript user-defined function (UDF), and writes them to a pre-existing BigQuery table as BigQuery elements. OPEN TUTORIAL

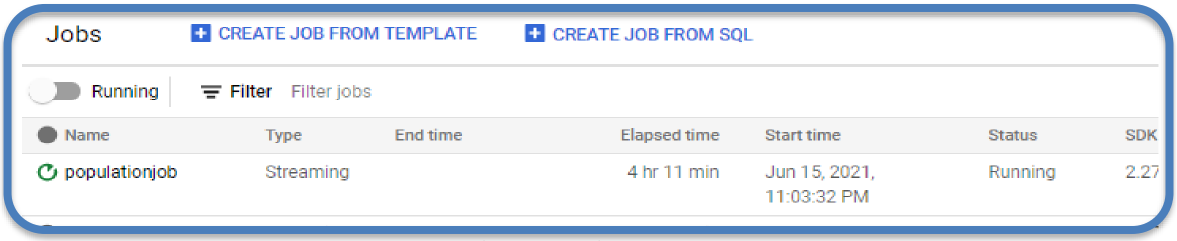
Required parameters

Input Pub/Sub topic *
The Pub/Sub topic to read the input from. Ex: projects/your-project-id/topics/your-topic-name

BigQuery output table *
The location of the BigQuery table to write the output to. If you reuse an existing table, it will be overwritten. The table's schema must match the input JSON objects. Ex: your-project:your-dataset.your-table

Temporary location *
Path and filename prefix for writing temporary files. Ex: gs://your-bucket/temp

Workflow created for transfer of data from pub/sub to big query table.

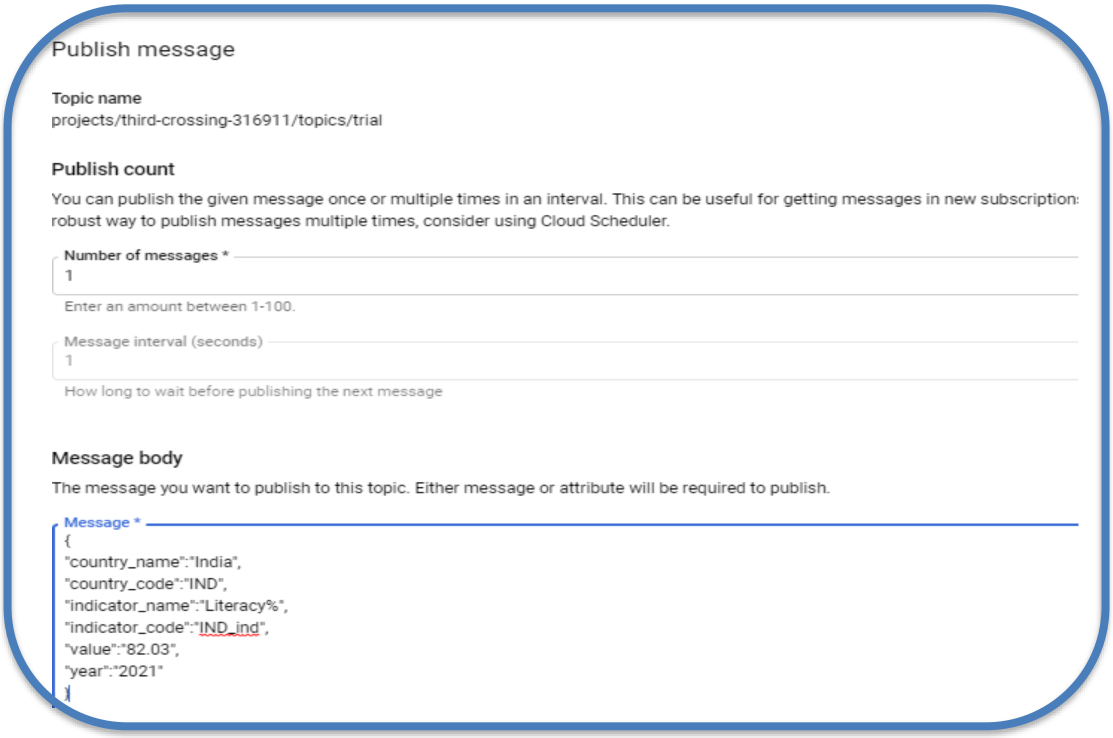


Jobs CREATE JOB FROM TEMPLATE CREATE JOB FROM SQL

Running Filter Filter jobs

Name	Type	End time	Elapsed time	Start time	Status	SDK
populationjob	Streaming		4 hr 11 min	Jun 15, 2021, 11:03:32 PM	Running	2.27

Status of the Dataflow



Publish message

Topic name
projects/third-crossing-316911/topics/trial

Publish count

You can publish the given message once or multiple times in an interval. This can be useful for getting messages in new subscription:
robust way to publish messages multiple times, consider using Cloud Scheduler.

Number of messages *
1
Enter an amount between 1-100.

Message interval (seconds)
1
How long to wait before publishing the next message

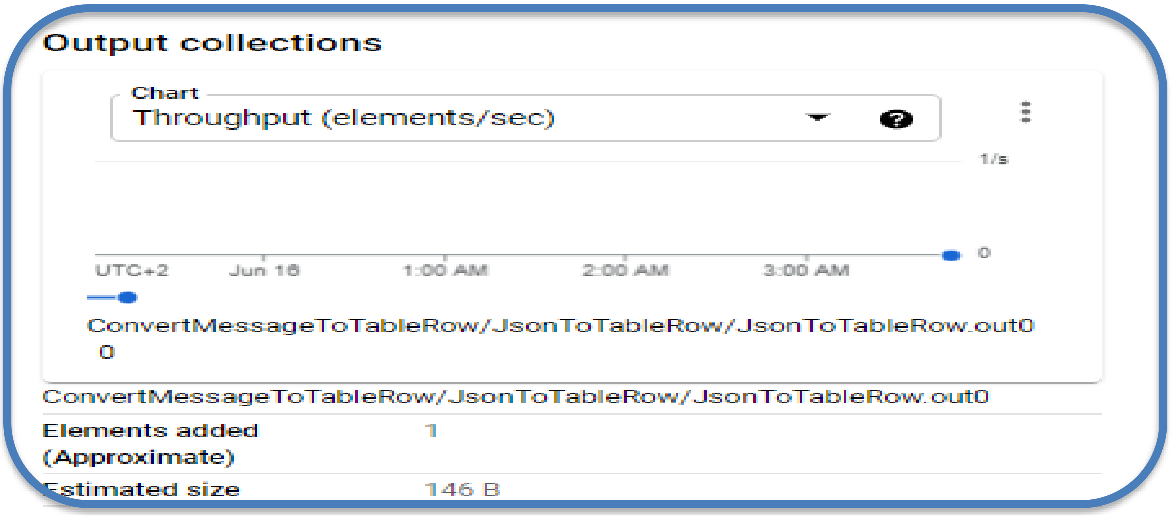
Message body

The message you want to publish to this topic. Either message or attribute will be required to publish.

Message *

```
{
"country_name":"India",
"country_code":"IND",
"indicator_name":"Literacy%",
"indicator_code":"IND_ind",
"value":"82.03",
"year":"2021"
}
```

When a message is published on investigating **Convert Message to Table row** Bucket displays the elements (Number of objects send in JSON Format) added and received by the bucket.

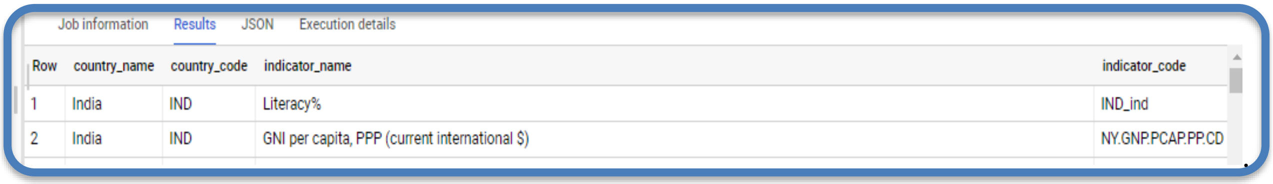


Output collections

Chart
Throughput (elements/sec)

ConvertMessageToTableRow/JsonToTableRow/JsonToTableRow.out0

Elements added (Approximate)	1
Estimated size	146 B



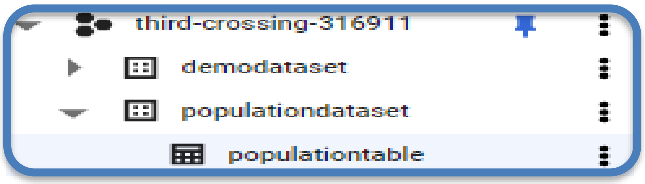
Job information Results JSON Execution details

Row	country_name	country_code	indicator_name	indicator_code
1	India	IND	Literacy%	IND_ind
2	India	IND	GNI per capita, PPP (current international $)	NY.GNP.PCAP.PP.CD

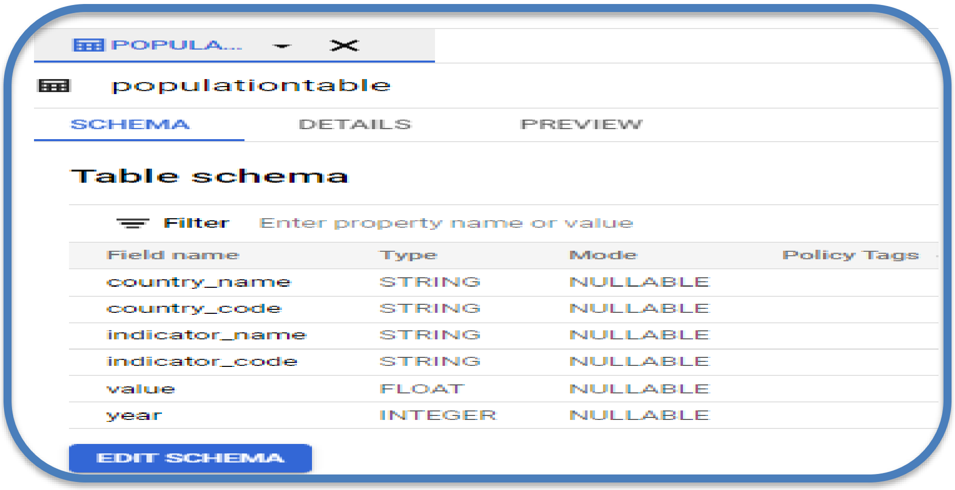
Checking the results in Big Query

- **Big query:** Big Query gives user a provision to received transferred data in the table format by establishing dataflow from different sources like cloud storage, pub/sub, Hive, Kafka, Text files

(JSON, CSV etc). In current context big query is receiving data as per defined schema of the data received in the table. To use big query as a data storage a user must create dataset and the table, this table reference is provided as a parameter to the dataflow.



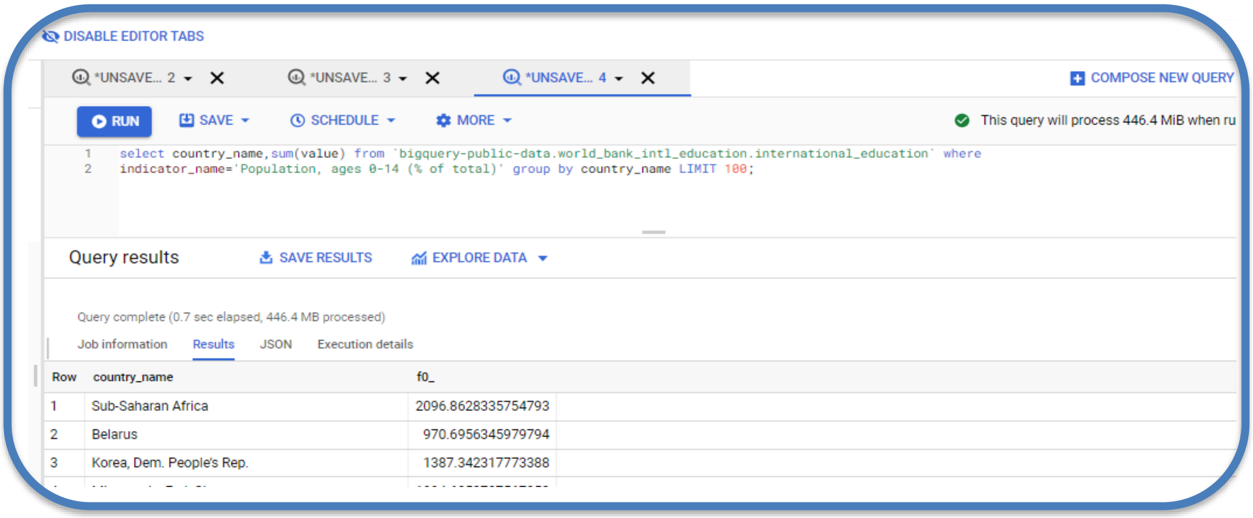
Dataset and Table



Field name	Type	Mode	Policy Tags
country_name	STRING	NULLABLE	
country_code	STRING	NULLABLE	
indicator_name	STRING	NULLABLE	
indicator_code	STRING	NULLABLE	
value	FLOAT	NULLABLE	
year	INTEGER	NULLABLE	

Schema of the table being created

- **Data Studio:** Data studio is used to visualize data that is queried from the table results produced from big query. The dataset in the project used is an educational statistic and the queries that are used to visualize the data is in relevance to the population percent% with respect to the different factors the country's education is associated with.



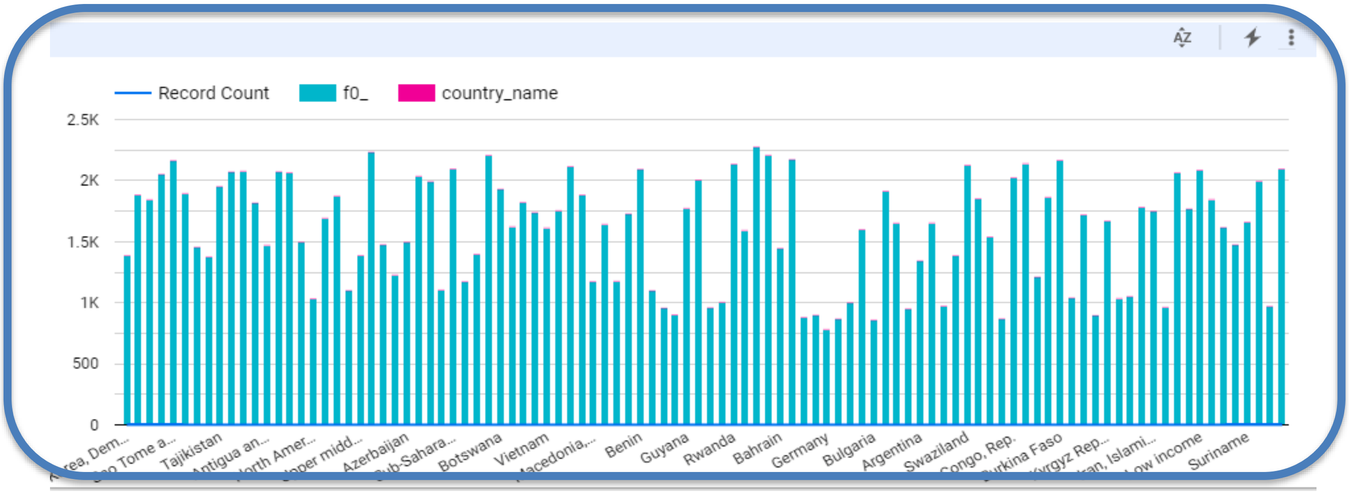
This query will process 446.4 MiB when ru

```
select country_name,sum(value) from `bigquery-public-data.world_bank_intl_education.international_education` where
indicator_name='Population, ages 0-14 (% of total)' group by country_name LIMIT 100;
```

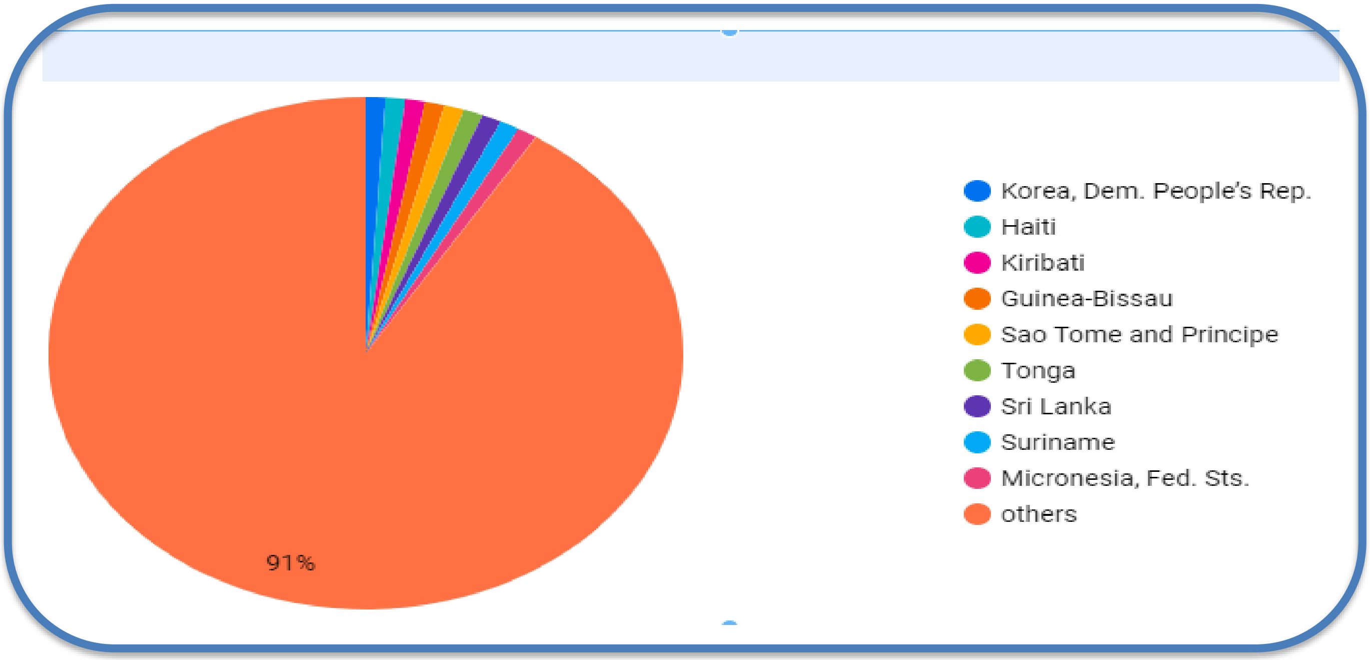
Query complete (0.7 sec elapsed, 446.4 MB processed)

Row	country_name	f0_
1	Sub-Saharan Africa	2096.8628335754793
2	Belarus	970.6956345979794
3	Korea, Dem. People's Rep.	1387.342317773388

Query Results from the dataset.Table

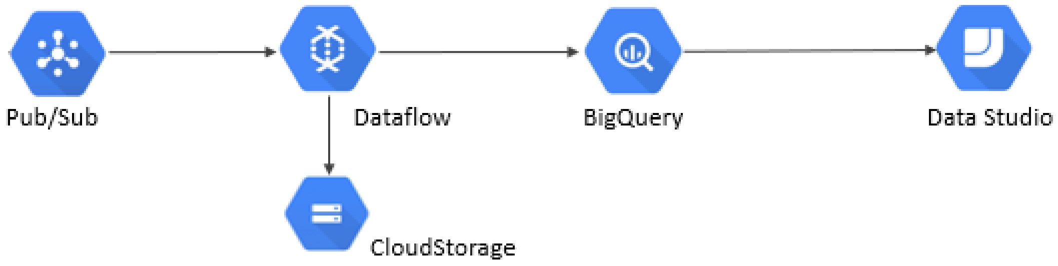


Visualization of the education statistics of different countries



Visulizing the education stats affected due to population age(%) <14 years using Pie chart

Current Architecture:



Alternative Approach using Python Script and Google Cloud shell

a) Create a bucket using GCP Shell using `gsutil mb gs://csvtestbucketde`
b) Download the necessary python libraries using `pip3 install google-cloud-bigquery –upgrade.`
c) Create dataset `bq mk --dataset:` delta-geode-316210: csvtestdataset
d) Create table and define the schema using bq mk -t csvtestdataset.csvtable id:INTEGER,first_name:STRING,last_name:STRING,email:STRING,gender:STRING,ip_address:STRING
e) Create a folder and place all the files as per the attached data engineering folder.
f) Change the directory to the folder using cd Folder_Name.
g) Locate the files in the folder directory using dir (for windows) command in google cloud console shell.
h) Load the contents of python file in console shell using type Main.py (Refer Attach folder)
i) Load the Contents in google shell for Env.yaml , Requirements.txt and .gcloudignore
j) env.yaml will generate the deployment specific configurations like name of the bucket, dataset ,table and service account location automatically by using the python os functions.
k) requirement.txt is used to generate necessary imports for the deployment.
l) .gcloudignore file is created so that the deployments like csv and yaml will not be deployed in GCP permanently.
m) After loading the contents of the file use the command `gcloud beta functions deploy csv_loader` on shell for creating a cloud function `csv_loader`
n) Using the function `gsutil cp testdata.csv gs://csvtestbucketde/` the contents of the csv are loaded in bucket.
o) Using `gcloud functions logs read function` load the data into the table.
p) `Use bq query 'select * from csvtestdataset.csvtable` for generating the query.
q) Create visualization in big query.

References:

Google Cloud Platform, https://cloud.google.com/docs

https://rickt.org/2018/10/22/poc-automated-insert-of-csv-data-into-bigquery-via-gcs-bucket-python