

PYA TILUK

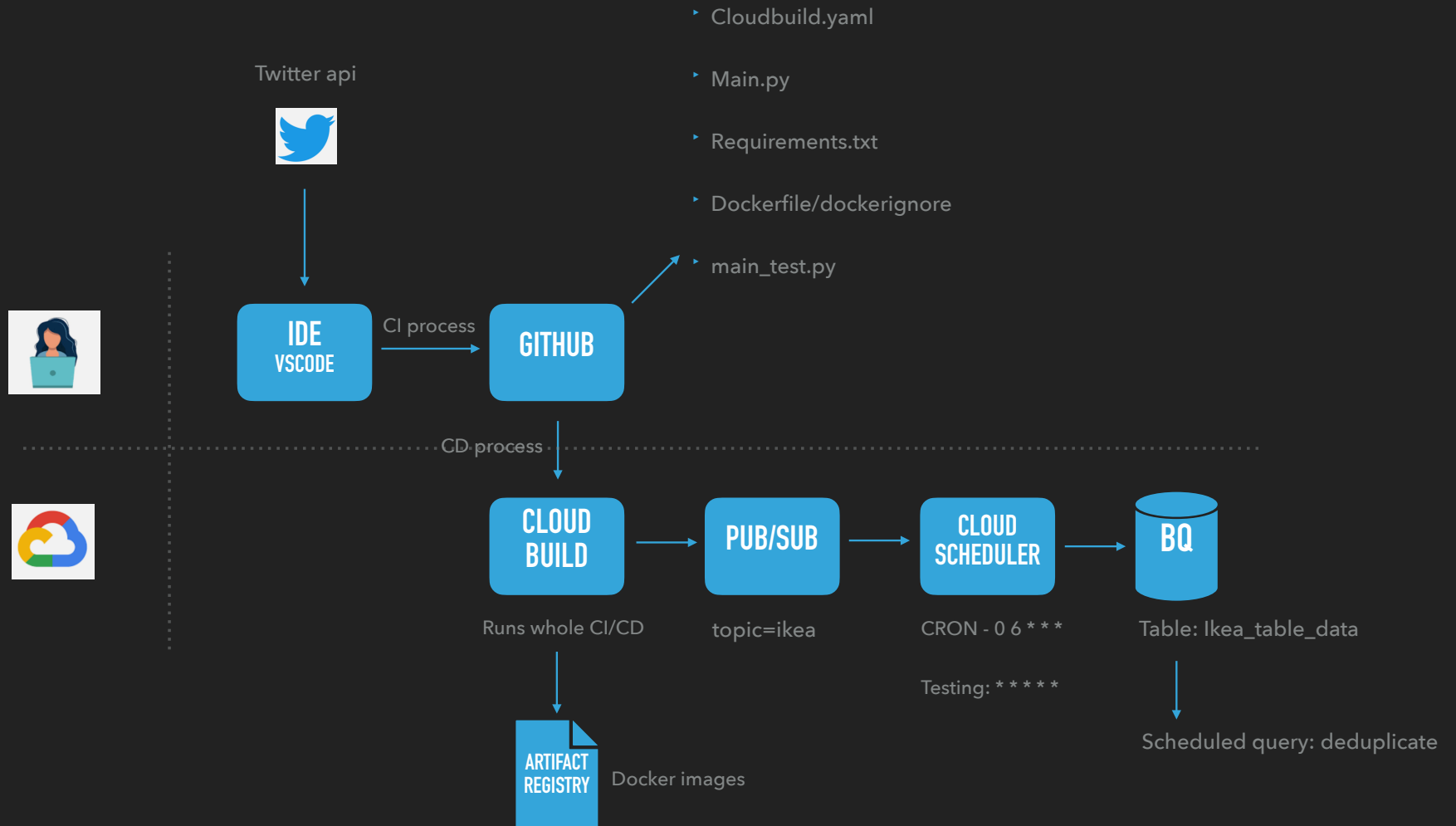
IKEA DATA ENGINEERING ASSIGNMENT

Files available at [GitHub](https://github.com/Pyariksha/tweetsikea.git) :

<https://github.com/Pyariksha/tweetsikea.git>

WORKING SOLUTION USING GCP FOR BATCH PROCESSING

OVERALL FLOW OF SOLUTION



PYTHON MAIN.PY FILE (SEE SHARED GITHUB REPO - TWEETSIKEA)

LIBRARY TO GET TWEETS: TWEETPY (BATCH)

```
EXPLORER  ...  main.py  x  main_test.py  requirements.txt  requirements.txt (Working Tree)

TWEETS...  [?] [?] [?] [?]
> __pycache__  pyte
> .pytest_cache
! cloudbuild.yaml
! ikea_assignment_pya...
main_test.py
main.py
① README.md
requirements.txt

main.py > ...
1  #import key modules
2  import tweepy
3  import pandas as pd
4  from pandas.io import gbq
5  import pandas_gbq
6
7  #load data into bigQuery table from df
8  def bq_load(key, value):
9      """
10     Function loads data into bq from pandas df.
11     """
12     project_name = 'sigma-scheduler-348710'#gcp project name
13     dataset_name = 'ikea'
14     table_name = key
15     value.to_gbq(destination_table='{0}.{1}'.format(dataset_name, table_name), project_id=project_name, if_exists='append')#append tweets to table
16
17 #Bearer Token can be inserted from gcp functions environment variables for twitter api access
18 client = tweepy.Client(bearer_token='AAAAAAAAAAAAAAAAAADpAcAEAAAAAN9RKdHzdbTk126SRndWSkFntZY8%3DdLlih5AeitkKwrr9NitUkJRdyurMgDcBQeTZpawF0caQ6EdK7z4')
19
20 # Get tweets that contain the hashtag #ikea
21 # -is:retweet exclude retweets
22 # lang:en for the tweets in english
23 query = '#ikea -is:retweet lang:en'
24 #search tweets based on hashtag - 3 attributes selected (id, created_at, text)
25 tweets = client.search_recent_tweets(query=query, tweet_fields=['created_at'], max_results=100)
26
27 #define function to get tweets to be run in gcp cloud functions
28 def get_tweets_ikea(tweets):
29     """
30     This function gets the tweets we want from '#ikea' and saves the 3 required attributes to a pandas dataframe.
31     Input params: data and context - default for cloud functions triggered by gcs. param is request for http.
32     """
33     try:
34         list = [] #initialize empty list
35         for tweet in tweets.data:
36             list.append(tweet) #append each tweet to list
37         df = pd.DataFrame(list) #create dataframe from list
38         df = df.drop_duplicates() #drop duplicate tweets in dataframe -- good for batch loads
39         bq_load('ikea_table_data', df) #write to bigquery table
40         #no need for a return statement as the function output is a load run to BigQuery (runs the bq_load function)
41         return df
42     except:
43         raise Exception('Error in function get_tweets_ikea.') #exception handling to separate config/run failures from function code errors
44
45 get_tweets_ikea(tweets) #calls the function as every time gcp runs the function the bq table must update with appended tweets.
```

Main.py script has 2 functions:

- ▶ get_tweets_ikea:

- generates the tweets for Ikea

- runs bq_load

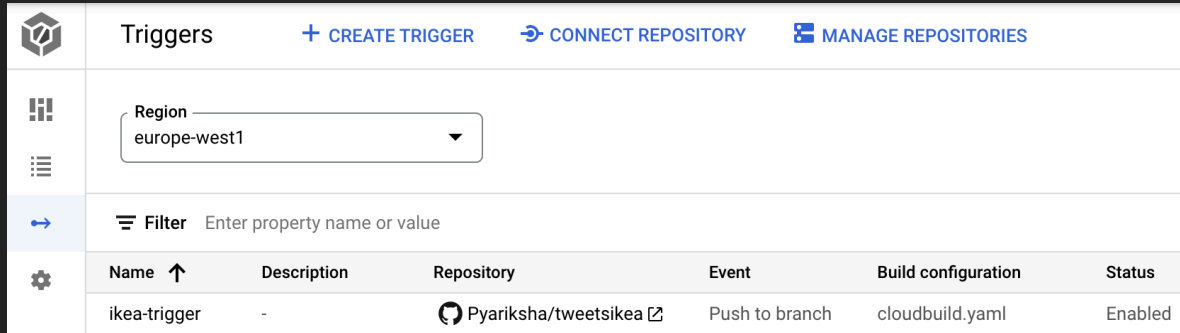
- ▶ bq_load:

- loads data from df to bigQuery table

main_test.py is a unit test using pytest and is run in the CI process.

GCP CLOUD BUILD USED FOR CI/CD

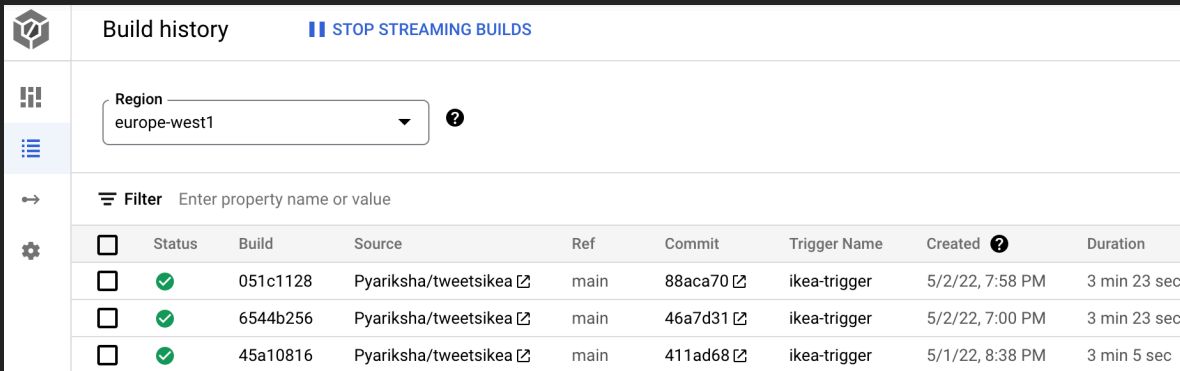
CLOUD BUILD DEVOPS



The screenshot shows the 'Triggers' page in the Google Cloud Build console. At the top, there are links for '+ CREATE TRIGGER', 'CONNECT REPOSITORY', and 'MANAGE REPOSITORIES'. A 'Region' dropdown menu is set to 'europe-west1'. Below this is a 'Filter' input field. A table lists the triggers:

Name	Description	Repository	Event	Build configuration	Status
ikea-trigger	-	Pyariksha/tweetsikea	Push to branch	cloudbuild.yaml	Enabled

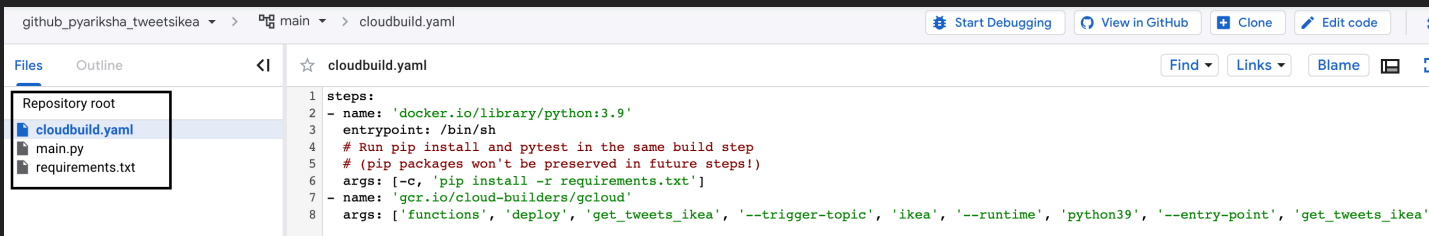
Cloud build



The screenshot shows the 'Build history' page in the Google Cloud Build console. It features a 'Region' dropdown set to 'europe-west1' and a 'Filter' input field. A table lists the build history:

Status	Build	Source	Ref	Commit	Trigger Name	Created	Duration
✓	051c1128	Pyariksha/tweetsikea	main	88aca70	ikea-trigger	5/2/22, 7:58 PM	3 min 23 sec
✓	6544b256	Pyariksha/tweetsikea	main	46a7d31	ikea-trigger	5/2/22, 7:00 PM	3 min 23 sec
✓	45a10816	Pyariksha/tweetsikea	main	411ad68	ikea-trigger	5/1/22, 8:38 PM	3 min 5 sec

Cloud build



The screenshot shows the GitHub repository interface for 'pyariksha_tweetsikea'. The file 'cloudbuild.yaml' is selected, and its content is displayed in the editor. The file contains two build steps: one for installing dependencies and running tests, and another for deploying the application to GCP.

```
1 steps:
2 - name: 'docker.io/library/python:3.9'
3   entrypoint: /bin/sh
4   # Run pip install and pytest in the same build step
5   # (pip packages won't be preserved in future steps!)
6   args: [-c, 'pip install -r requirements.txt']
7 - name: 'gcr.io/cloud-builders/gcloud'
8   args: ['functions', 'deploy', 'get_tweets_ikea', '--trigger-topic', 'ikea', '--runtime', 'python39', '--entry-point', 'get_tweets_ikea']
```







Cloud Build steps:

- ▶ Created cloud build.yaml file with 2 steps:
1.push and 2.deploy function (main.py)
- ▶ Connected GitHub repo - Pyariksha/tweetsikea
- ▶ Included trigger on branch "main" on push/commit

Github repo - cloud build.yaml

EVENT TRIGGER USED

AUTOMATE BATCH RUN: PUB SUB AND CLOUD SCHEDULER

	Topics	+ CREATE TOPIC	 DELETE
	 Filter	Filter topics	
	<input type="checkbox"/>	Topic ID ↑	Encryption key
	<input type="checkbox"/>	ikea	Google-managed
		Topic name	Retention
		projects/sigma-scheduler-348710/topics/ikea	—


Pub sub topic: ikea

Pub Sub topic: ikea

Linked to cloud scheduler job (ikea_batch)


CRON jobs:


- For testing: (* * * * *) / every minute
- Would schedule in production daily for batches





Cloud Scheduler


Jobs


 CREATE JOB


 REFRESH

 EDIT

 COPY


 PAUSE

 RESUME

 DELETE

SCHEDULER JOBS

APP ENGINE CRON JOBS ?



Filter

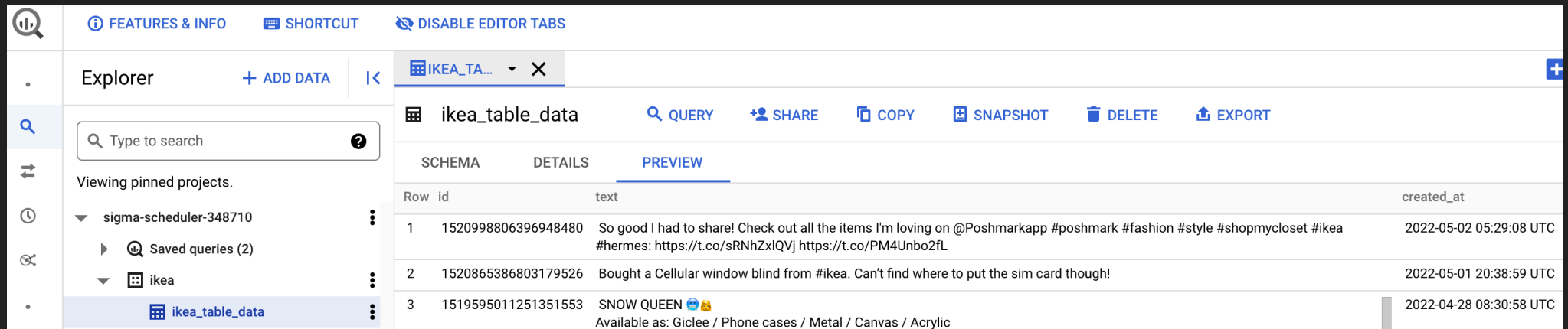
Filter jobs

<input type="checkbox"/>	<input checked="" type="radio"/>	Name ↑	Region	State	Description	Frequency	Target	Last run	Last run result	Next run
<input type="checkbox"/>	<input checked="" type="radio"/>	ikea_batch	europe-west1	Enabled	batch schedule	***** (Europe/Amsterdam)	Topic : projects/sigma-scheduler-348710/topics/ikea	May 2, 2022, 11:57:00 PM	Success	May 2, 2022, 11:58:00 PM

Cloud scheduler job

WRITE OUTPUT - BIG QUERY

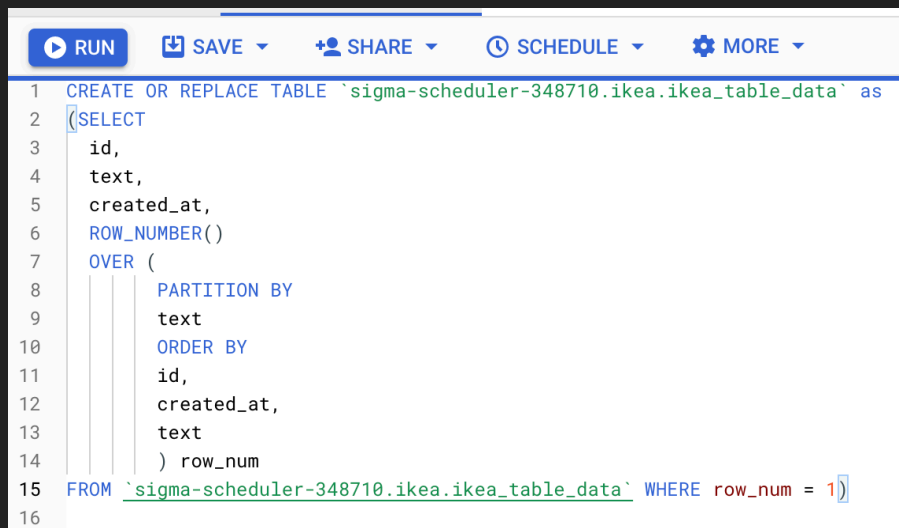
BQ TABLE STORES TWEETS – TRANSFORMATION STEP INCLUDED



The screenshot shows the BigQuery Explorer interface. On the left, the 'Explorer' pane lists pinned projects: 'sigma-scheduler-348710' (containing 'Saved queries (2)' and 'ikea') and 'ikea_table_data'. The main pane shows the 'ikea_table_data' table with tabs for 'SCHEMA', 'DETAILS', and 'PREVIEW'. The 'PREVIEW' tab is active, displaying a table with columns 'id', 'text', and 'created_at'. The table contains three rows of tweet data.

Row	id	text	created_at
1	1520998806396948480	So good I had to share! Check out all the items I'm loving on @Poshmarkapp #poshmark #fashion #style #shopmycloset #ikea #hermes: https://t.co/sRNhZxIQVj https://t.co/PM4Unbo2fL	2022-05-02 05:29:08 UTC
2	1520865386803179526	Bought a Cellular window blind from #ikea. Can't find where to put the sim card though!	2022-05-01 20:38:59 UTC
3	1519595011251351553	SNOW QUEEN 🌨️👑 Available as: Giclee / Phone cases / Metal / Canvas / Acrylic	2022-04-28 08:30:58 UTC

Big query table output



The screenshot shows the BigQuery SQL editor with a query that creates or replaces a table and inserts data from another table, using a window function to deduplicate records.

```
1 CREATE OR REPLACE TABLE `sigma-scheduler-348710.ikea.ikea_table_data` as
2 (SELECT
3   id,
4   text,
5   created_at,
6   ROW_NUMBER()
7   OVER (
8     PARTITION BY
9       text
10    ORDER BY
11      id,
12      created_at,
13      text
14    ) row_num
15 FROM `sigma-scheduler-348710.ikea.ikea_table_data` WHERE row_num = 1)
16
```

Scheduled query: DDL

- ▶ Big query table appended with data as CRON job runs
- ▶ Duplicate tweets in batch are removed via scheduled query