

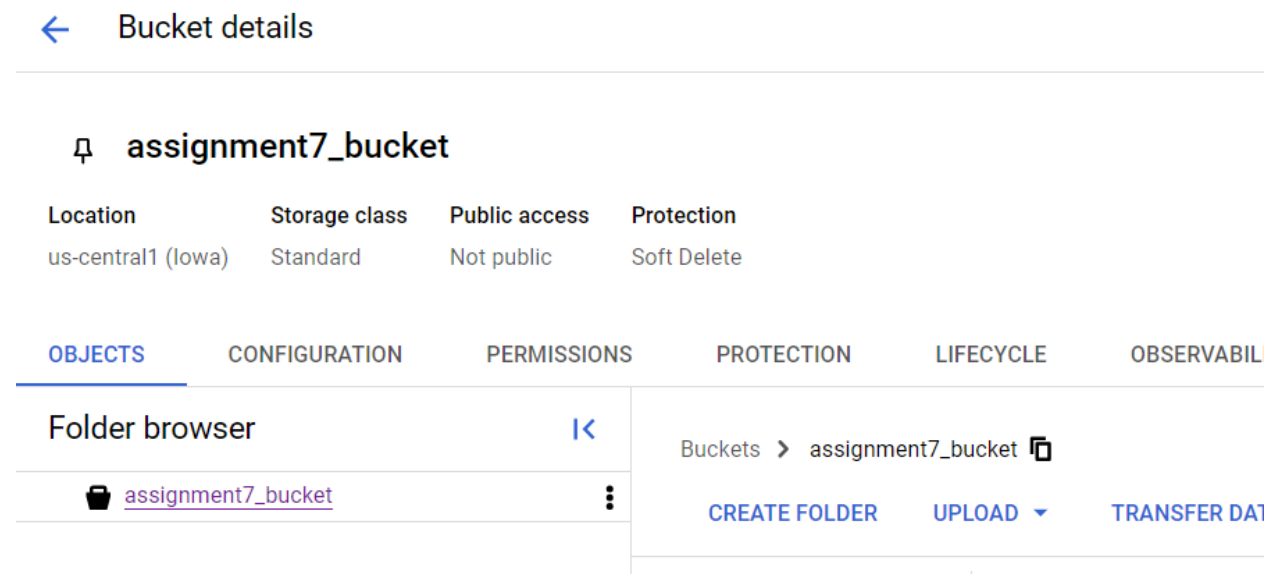
Question: Count the number of lines in a file in GCS in real-time using Google Cloud Functions and Pub Sub.

Submitted by Shramana Sinha, 23f1002703

1. Implementation Details

1.1 Google Cloud Storage Bucket




A bucket named **assignment7_bucket** was created to store input files. Any file uploaded to this bucket triggers the Cloud Function.














Screenshot: The gcs bucket for the assignment

1.2 Pub/Sub Topic and Subscription

A topic named **CountLine** was created to relay messages about new file uploads. A pull subscription named **CountLineSub** was created to allow the subscriber application to receive these messages.

Subscription name	projects/celtic-guru-448518-f8/subscriptions/CountLineSub 
Subscription state	 active
Topic name	projects/celtic-guru-448518-f8/topics/CountLine 

METRICS	DETAILS	MESSAGES
Delivery type 	Pull	
Subscription expiration 	Subscription expires in 31 days if there is no activity.	
Acknowledgement deadline 	10 seconds	
Subscription filter 	—	
Subscription message retention duration 	7 days	
Topic message retention duration 	—	
Retain acknowledged messages 	No	
Exactly once delivery 	Disabled	
Message ordering 	Disabled	
Dead lettering 	Disabled	
Retry policy 	Retry immediately	
Labels	—	

Screenshot: The pub/sub topic and subscription for this assignment

1.3 Google Cloud Function

The Cloud Function was implemented with the following specifications:

- Runtime: Python 3.12
- Trigger: Cloud Storage
- Entry point: `file_upload_trigger`
- Dependencies: google-cloud-pubsub, functions-framework

The function's key responsibilities include:

- Extracting the file name and bucket name from the event data

Python

```
file_name = event.data["name"]
bucket_name = event.data["bucket"]
```

- Creating a structured message with this information

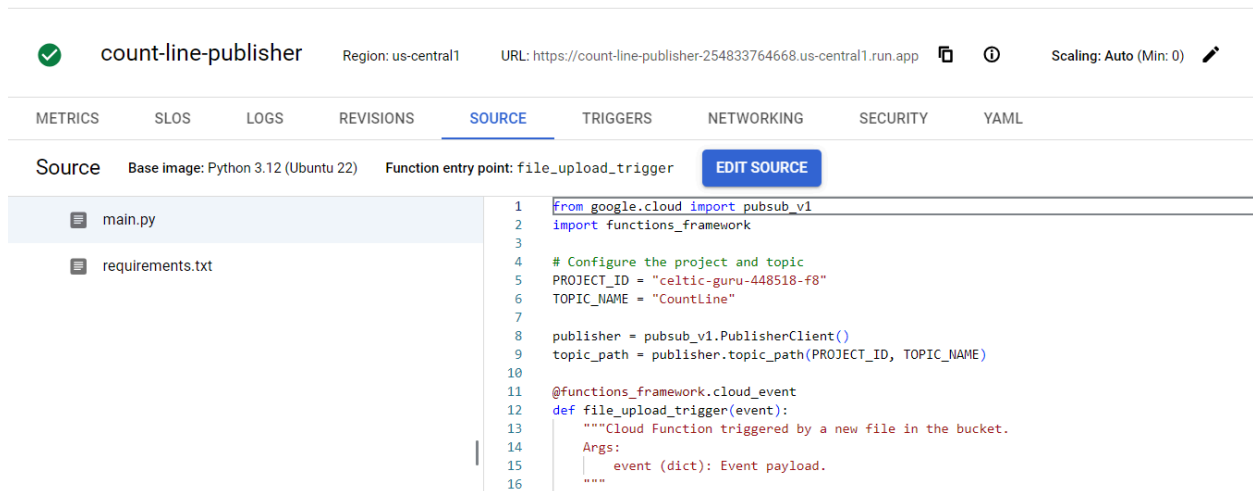
Python

```
message = {"bucket": bucket_name, "file": file_name}
import json
message_data = json.dumps(message).encode("utf-8")
```

- Publishing the message to the Pub/Sub topic

Python

```
future = publisher.publish(topic_path, data=message_data)
```



Screenshot: The google cloud run function for the assignment

1.4 Subscriber Application

The subscriber application was deployed in Google Cloud Shell. Its key responsibilities include:

- Establishing a connection to the Pub/Sub subscription

Python

```
subscriber = pubsub_v1.SubscriberClient()
subscription_path = subscriber.subscription_path(PROJECT_ID,
SUBSCRIPTION_NAME)
streaming_pull_future = subscriber.subscribe(subscription_path,
callback=callback)
```

- Processing incoming messages about file uploads

Python

```
data = json.loads(message.data.decode("utf-8"))
bucket_name = data["bucket"]
file_name = data["file"]
```

- Downloading the referenced files from Cloud Storage

Python

```
bucket = storage_client.bucket(bucket_name)
blob = bucket.blob(file_name)
contents = blob.download_as_string().decode("utf-8")
```

- Counting the number of lines in each file

Python

```
line_count = len(contents.splitlines())
```

- Acknowledging the messages after successful processing

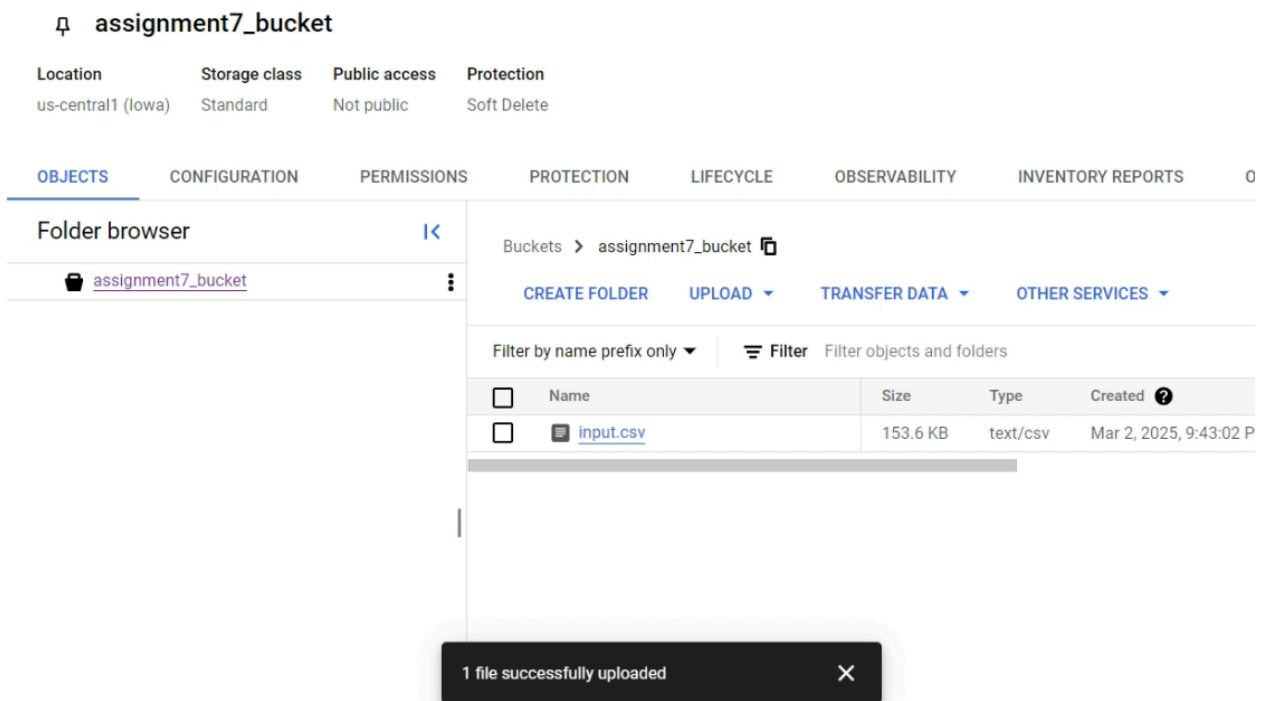
Python

```
message.ack()
```

2. Testing and Validation

This was tested by uploading a file, named `input.csv` which had 2001 lines, to the `assignment7_bucket` bucket. The following observations confirmed the correct operation:

- The Cloud Function was triggered upon file upload.
- The function successfully published a message to the Pub/Sub topic.
- The subscriber received the message with the correct file and bucket information.
- The subscriber successfully downloaded the file and accurately counted the lines.



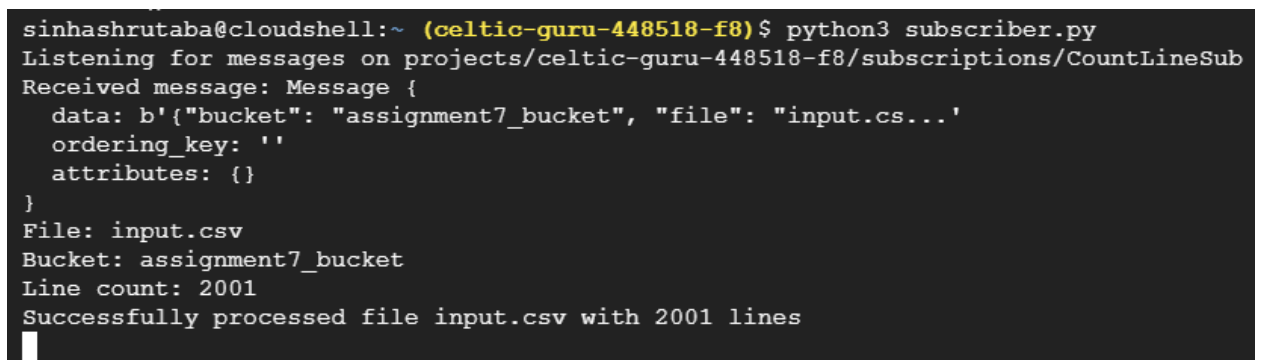
Screenshot: The input file is uploaded

1999	1998	1998	Name1998	City1998	1735-01-01T00:00:00	9999-12-31T00:00:00
2000	1999	1999	Name1999	City1999	1735-01-01T00:00:00	9999-12-31T00:00:00
2001	2000	2000	Name2000	City2000	1735-01-01T00:00:00	9999-12-31T00:00:00

Screenshot: The number of lines in the input file (2001 lines)

>	i	2025-03-02 21:43:03.644 IST	POST	200	130 B	112 ms	APIs-Google; (+https://developers.goo...	https://count-line-publisher-cujvth3w4q-uc.a.run.app/?__GCP_CloudEventsMode=GCS_NOTIFICATION
>	*	2025-03-02 21:43:03.665 IST					Processing file: input.csv from bucket: assignment7_bucket	
>	*	2025-03-02 21:43:03.771 IST					Published message ID: 14127700856253684 for file: input.csv	

Screenshot: The logs for the run function showing the successful message to the Pub/Sub topic



Screenshot: The terminal output showing the successful receiving of the message and counting of the lines in the file