

Data Engineering Project Assignment -1

Team Name - Data Foundry

Team Members:

Kiranmayi Thanikonda

Jayita Banerjee

Navya Sri Ambati

Step-1:

Construct a table showing each day for which your pipeline successfully, automatically processed one complete days' worth of sensor readings. It might be a good idea to have logging of some kind for the project so you can gather not only this data, but errors that might occur as you develop your pipeline throughout the term.

Date	Day of Week	Approximate Time of day for your data access	# Sensor Readings	Total Data Saved (KBs)	# Pub/Sub messages published and received
04/07/2025	Monday	12:00 AM	663247	181788.25 KB	663247
04/08/2025	Tuesday	10:00 AM	709902	194718.97 KB	709902
04/09/2025	Wednesday	10:00 AM	729793	200164.08 KB	729793
04/10/2025	Thursday	10:00 AM	688427	188748.67 KB	688427
04/11/2025	Friday	10:00 AM	672597	184510.95 KB	672597
04/12/2025	Saturday	10:00 AM	No Data	No Data	No Data
04/13/2025	Sunday	10:00 AM	No Data	No Data	No Data
04/14/2025	Monday	10:00 AM	744055	173536.76 KB	744055
04/15/2025	Tuesday	10:00 AM	677762	162173.72 KB	677762
04/16/2025	Wednesday	10:00 AM	603623	144467.21 KB	603623
04/17/2025	Thursday	10:00 AM	664401	159022.37 KB	664401

04/18/2025	Friday	10:00 AM	685748	164139.78 KB	685748
04/19/2025	Saturday	10:00 AM			
04/20/2025	Sunday	10:00 AM			

Step-2:

Additionally, include screenshots for the parts C, H and I

1. Output of `crontab -l`: Your scheduled cron jobs.

```

cron_log_2025-04-14.txt publish.py vehicle_data_2025-04-09 vehicle_data_2025-04-17
kthaniko@datafoundry-20250407-185428:~$ crontab -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
0 10 * * * /usr/bin/python3 /home/kthaniko/fetch_bus_data.py >> /home/kthaniko/cron_log_$(date +%Y-%m-%d).txt
2>&1
#3 10 * * * /home/kthaniko/pubsub-env/bin/python /home/kthaniko/publish.py >> /home/kthaniko/cron_log_publisher_
$(date +%F).txt 2>&1
#13 10 * * * /home/kthaniko/publish.sh
5 10 * * * /home/kthaniko/publish.sh
kthaniko@datafoundry-20250407-185428:~$

```

2. systemctl status: This will show the status of your receiver program.

```
• datafoundry-sub
  State: running
  Units: 258 loaded (incl. loaded aliases)
  Jobs: 0 queued
  Failed: 0 units
  Since: Thu 2025-04-17 12:22:29 PDT; 42s ago
  systemd: 252.36-1~deb12u1
  CGroup: /
    └─init.scope
       └─1 /sbin/init
    └─system.slice
       └─cron.service
          └─891 /usr/sbin/cron -f
       └─dbus.service
          └─380 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation
       └─exim4.service
          └─808 /usr/sbin/exim4 -bd -q30m
```

3. VM instance schedule: This will display the schedule settings for your GCP VM instance.

VM instances

Create instanceImport VMCreate scheduleRefresh

Instances

Observability

Instance schedules

Instances schedules

Filter Enter property name or value

<input type="checkbox"/>	Name ↑	Region	Start schedule	Stop schedule	Time zone	Initiation date	Expiration date
<input type="checkbox"/>	vm-schedule-2	us-west1	10:31AM, every day	10:50AM, every day	America/Los_Angeles	Apr 18, 2025, 10:20:00 AM UTC-07:00	Jun 20, 2025, 12:00:00 AM UTC-07:00
<input type="checkbox"/>	vm-schedule1	us-west1	9:50AM, every day	10:30AM, every day	America/Los_Angeles	Apr 17, 2025, 9:49:00 AM UTC-07:00	Jun 20, 2025, 12:00:00 AM UTC-07:00

← Instance schedule details

Delete

✓ vm-schedule1

Description

Region	us-west1
VM Start	9:50AM, every day
VM Stop	10:30AM, every day
Time zone	America/Los_Angeles
Initiation date	Apr 17, 2025, 9:49:00 AM UTC-07:00
End date	Jun 20, 2025, 12:00:00 AM UTC-07:00

Attached instances

Add instances to schedule

Remove instances from schedule

<input type="checkbox"/>	Name ↑	Zone	Creation time	Machine type
<input type="checkbox"/>	datafoundry-20250407-185428	us-west1-a	2025-04-07T11:59:05.195-07:00	e2-medium

← Instance schedule details

Delete

✓ vm-schedule-2

Description

Regionus-west1

VM Start10:31AM, every day

VM Stop10:50AM, every day

Time zoneAmerica/Los_Angeles

Initiation dateApr 18, 2025, 10:20:00 AM UTC-07:00

End dateJun 20, 2025, 12:00:00 AM UTC-07:00

Attached instances

Add instances to schedule

Remove instances from schedule

<input type="checkbox"/>	Name ↑	Zone	Creation time	Machine type
<input type="checkbox"/>	datafoundry-sub	us-west1-a	2025-04-14T11:31:53.447-07:00	e2-medium

GitHub Link:

<https://github.com/kthanikonda/DataEngineering/tree/main/Project/Data%20Gathering%20and%20Transport>

Permissions:

Share "Data Engineering Project Assignment -1"

?

⚙

Add people, groups, and calendar events

People with access

K

Kiranmayi Thanikonda (you)

kthaniko@pdx.edu

Owner

General access

Portland State University

Viewer

Anyone in this group with the link can view

Copy link

Done

