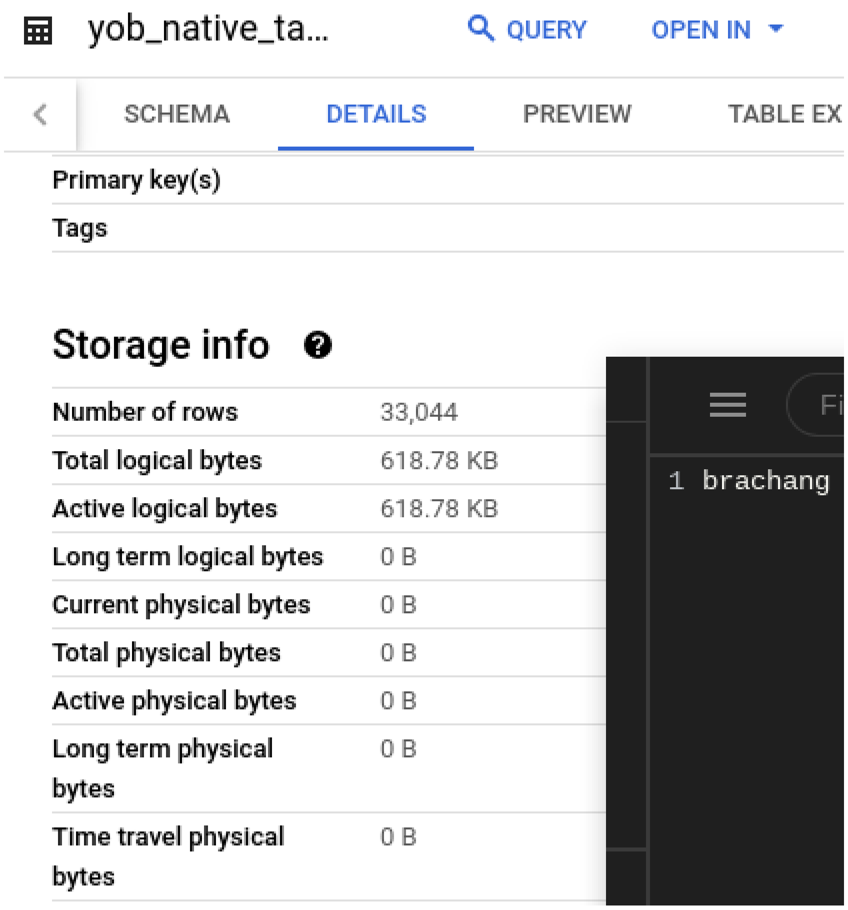
Bradley Chang

9.1g: BigQuery, BigLake

3. Create dataset

Take a screenshot of the table's details that includes the number of rows in the table.



Primary key(s)
Tags

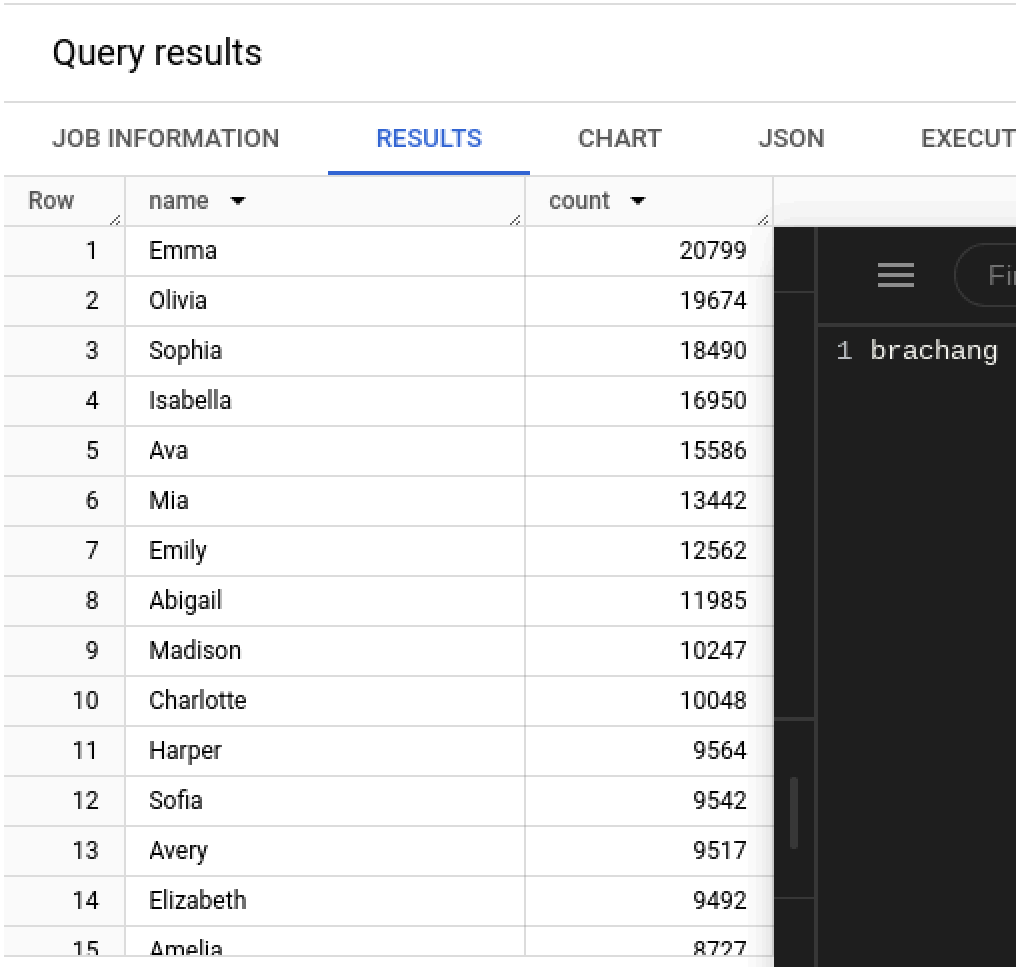
Storage info

Number of rows	33,044
Total logical bytes	618.78 KB
Active logical bytes	618.78 KB
Long term logical bytes	0 B
Current physical bytes	0 B
Total physical bytes	0 B
Active physical bytes	0 B
Long term physical bytes	0 B
Time travel physical bytes	0 B

```
1 brachang
```

4. Query data

Screenshot the query results and include it in your lab notebook



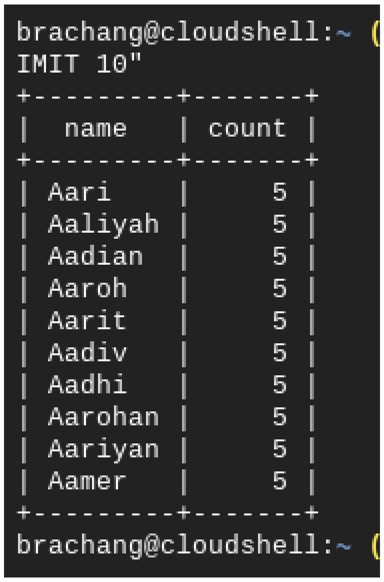
Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUT

Row	name	count
1	Emma	20799
2	Olivia	19674
3	Sophia	18490
4	Isabella	16950
5	Ava	15586
6	Mia	13442
7	Emily	12562
8	Abigail	11985
9	Madison	10247
10	Charlotte	10048
11	Harper	9564
12	Sofia	9542
13	Avery	9517
14	Elizabeth	9492
15	Amelia	8727

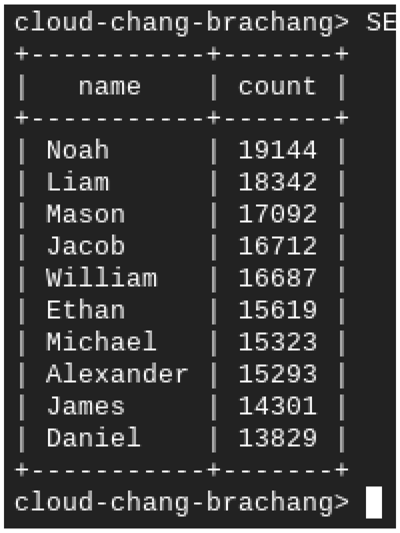
```
1 brachang
```

Screenshot your results and include it in your lab notebook



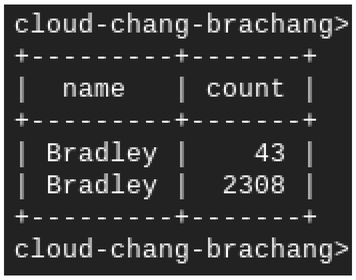
```
brachang@cloudshell:~ (
IMIT 10"
+---------+-------+
|  name   | count |
+---------+-------+
| Aari    |     5 |
| Aaliyah |     5 |
| Aadian  |     5 |
| Aaroh   |     5 |
| Aarit   |     5 |
| Aadiv   |     5 |
| Aadhi   |     5 |
| Aarohan |     5 |
| Aariyan |     5 |
| Aamer   |     5 |
+---------+-------+
brachang@cloudshell:~ (
```

Screenshot your results and include it in your lab notebook



```
cloud-chang-brachang> SE
+-----------+-------+
|   name    | count |
+-----------+-------+
| Noah      | 19144 |
| Liam      | 18342 |
| Mason     | 17092 |
| Jacob     | 16712 |
| William   | 16687 |
| Ethan     | 15619 |
| Michael   | 15323 |
| Alexander | 15293 |
| James     | 14301 |
| Daniel    | 13829 |
+-----------+-------+
cloud-chang-brachang>
```

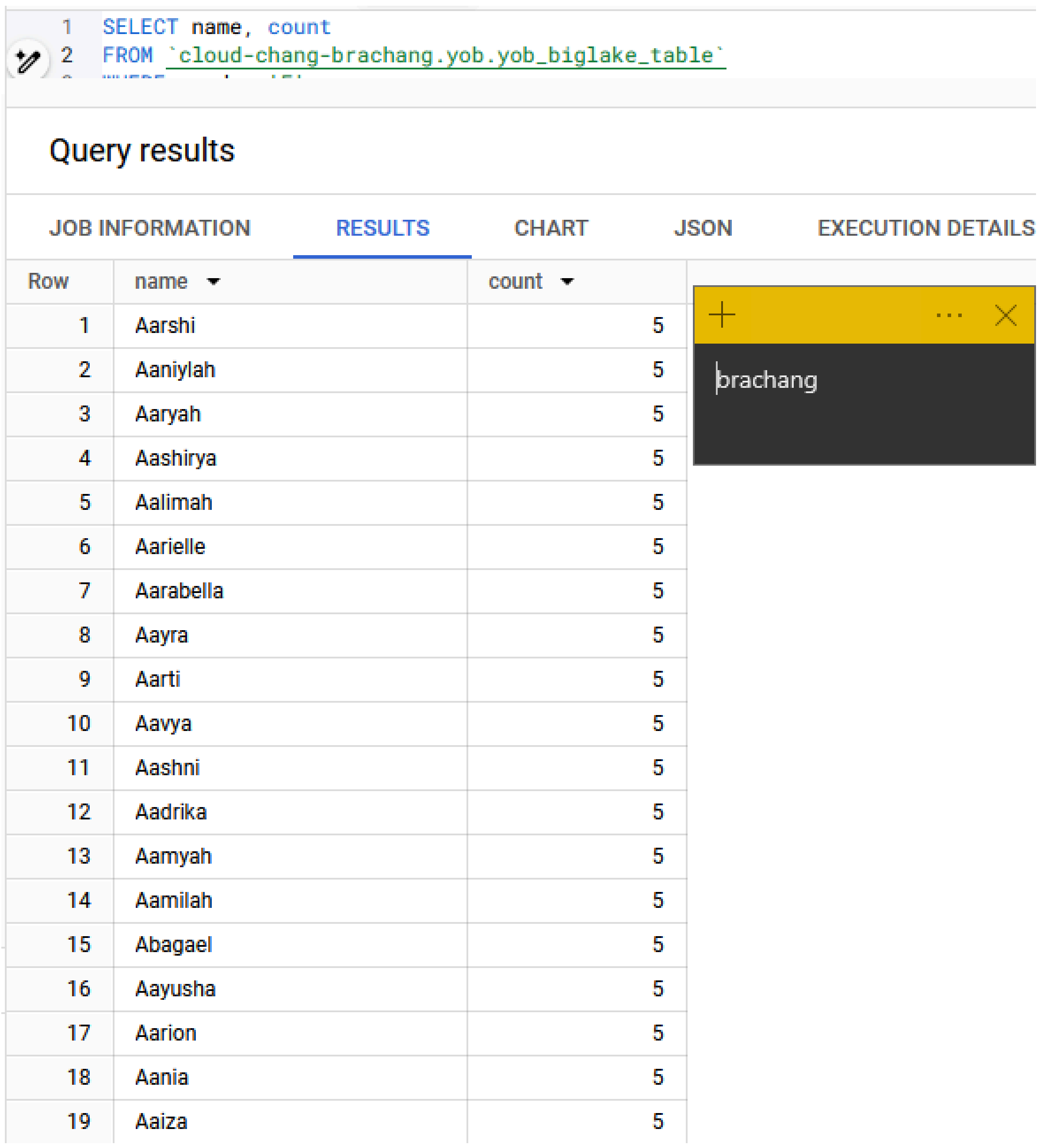
Screenshot your results and include it in your lab notebook



```
cloud-chang-brachang>
+---------+-------+
|  name   | count |
+---------+-------+
| Bradley |    43 |
| Bradley |  2308 |
+---------+-------+
cloud-chang-brachang>
```

9. Query data

Screenshot the query results and include it in your lab notebook



```
SELECT name, count
FROM `cloud-chang-brachang.yob.yob_biglake_table`
```

Query results

Row	name	count
1	Aarshi	5
2	Aaniylah	5
3	Aaryah	5
4	Aashirya	5
5	Aalimah	5
6	Aarielle	5
7	Aarabella	5
8	Aayra	5
9	Aarti	5
10	Aavya	5
11	Aashni	5
12	Aadrika	5
13	Aamyah	5
14	Aamilah	5
15	Abagael	5
16	Aayusha	5
17	Aarion	5
18	Aania	5
19	Aaiza	5

brachang

9.2g: Jupyter Notebooks

3. BigQuery query

How much less data does this query process compared to the size of the table?

The original query was about to process 21.94 gb.

The modified query only processes 3.05 gb

How many twins were born during this time range?

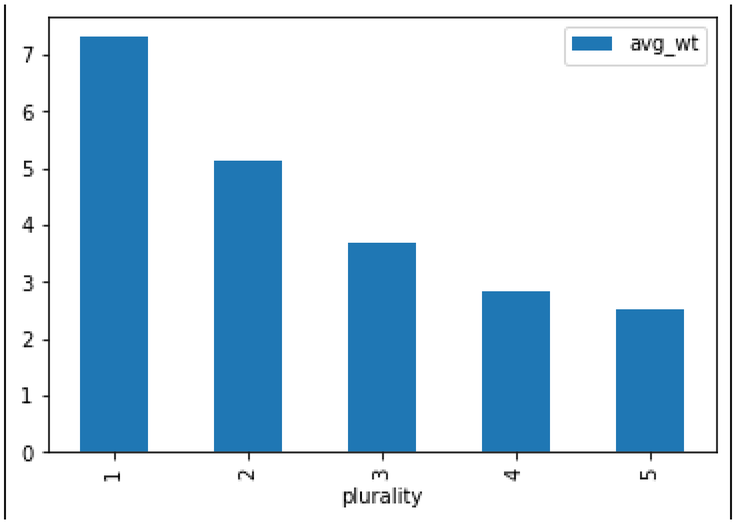
125233

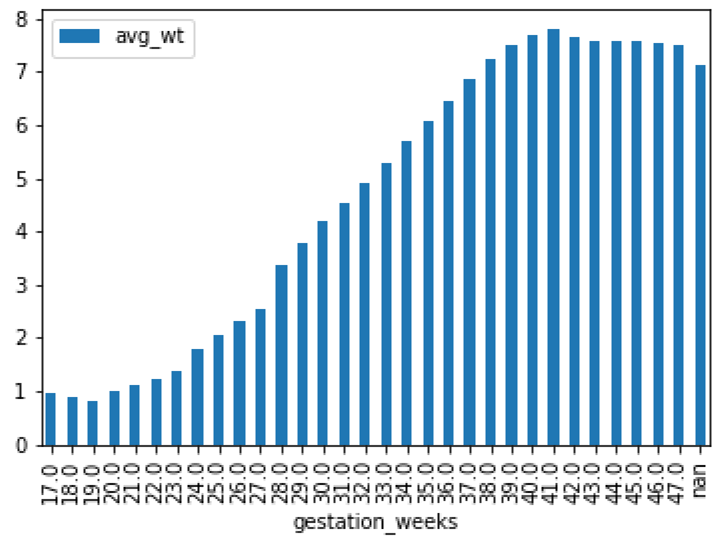
How much lighter on average are they compared to single babies?

Twins had an average weight of 5.17 pounds compared to single babies with an average of 7.35 pounds

6. Run queries

Show the plots generated for the two most important features for your lab notebook

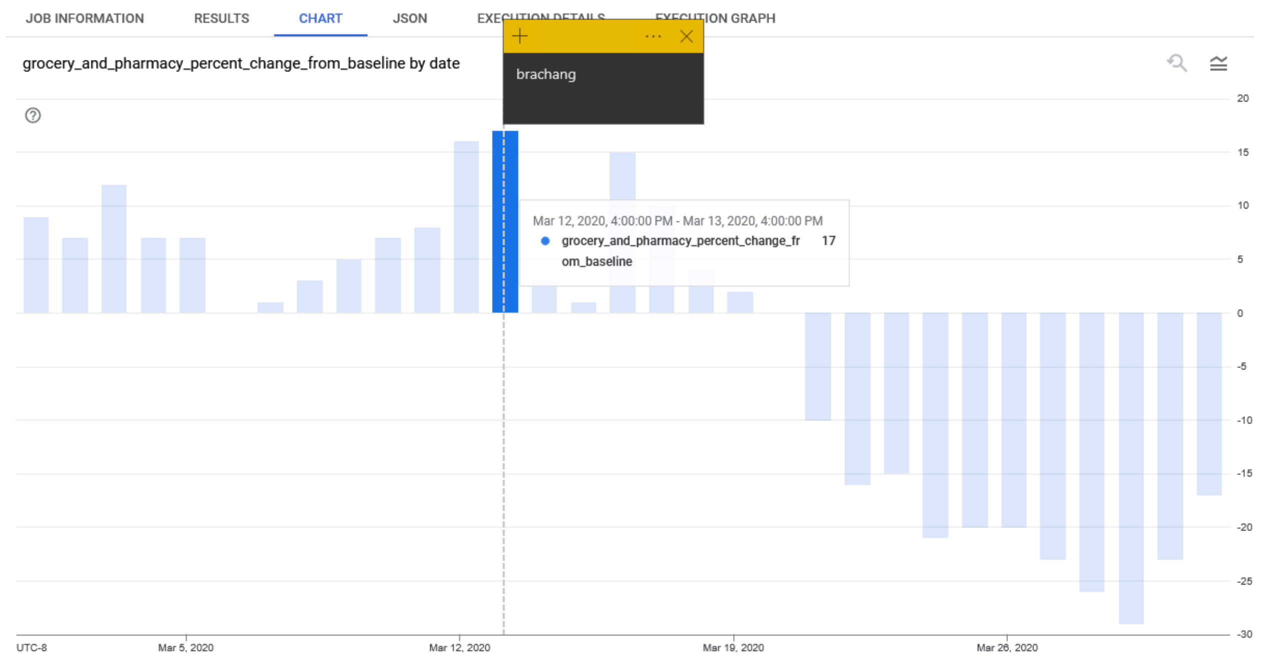




8. Mobility

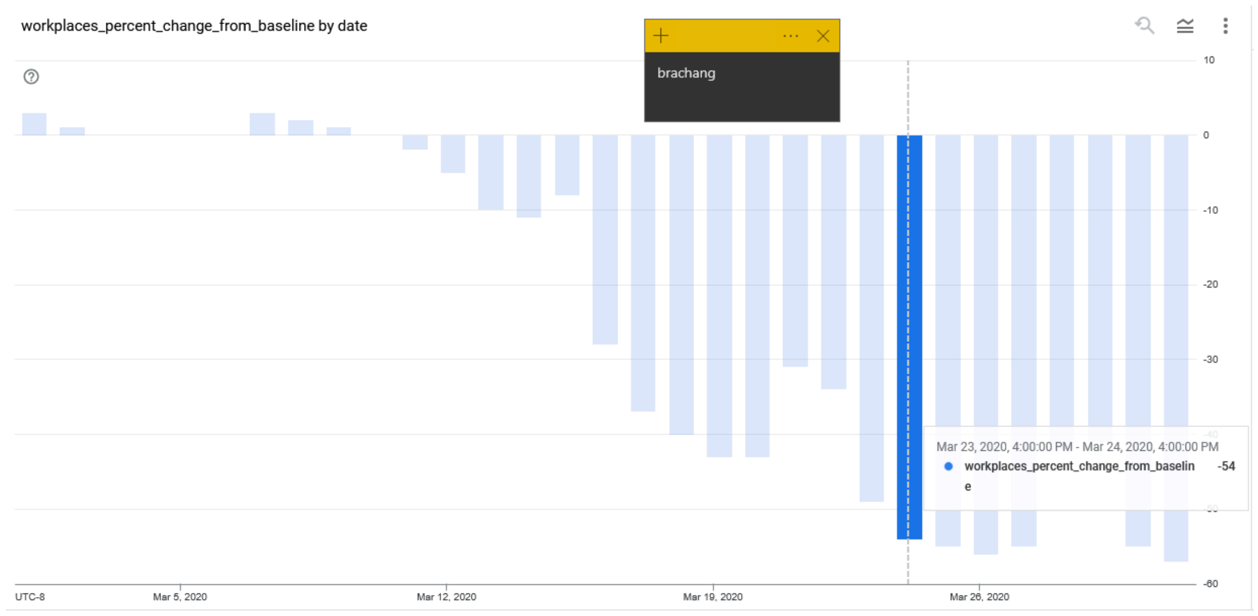
What day saw the largest spike in trips to grocery and pharmacy stores?

Largest positive spike was on Mar 12 2020


JOB INFORMATION
RESULTS
CHART
JSON
EXECUTION DETAILS
EXECUTION GRAPH
grocery_and_pharmacy_percent_change_from_baseline by date
brachang
Mar 12, 2020, 4:00:00 PM - Mar 13, 2020, 4:00:00 PM
grocery_and_pharmacy_percent_change_from_baseline 17
UTC-8
Mar 5, 2020
Mar 12, 2020
Mar 19, 2020
Mar 26, 2020

On the day the stay-at-home order took effect (3/23/2020), what was the total impact on workplace trips?

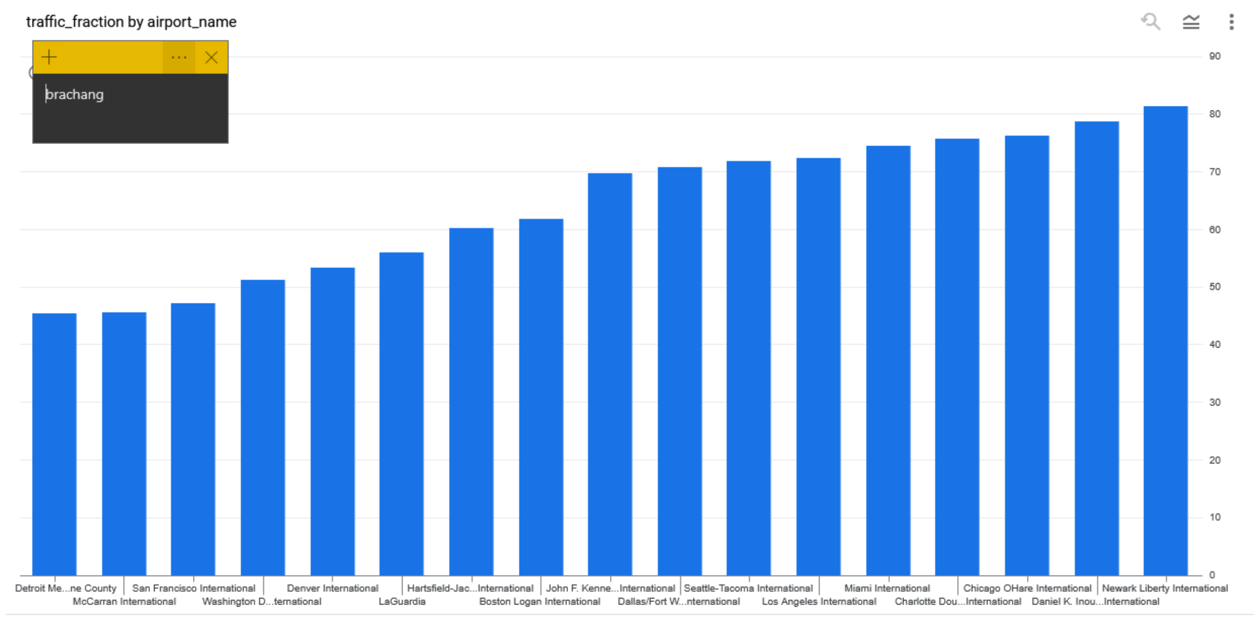
-54% from baseline



9. Airport traffic

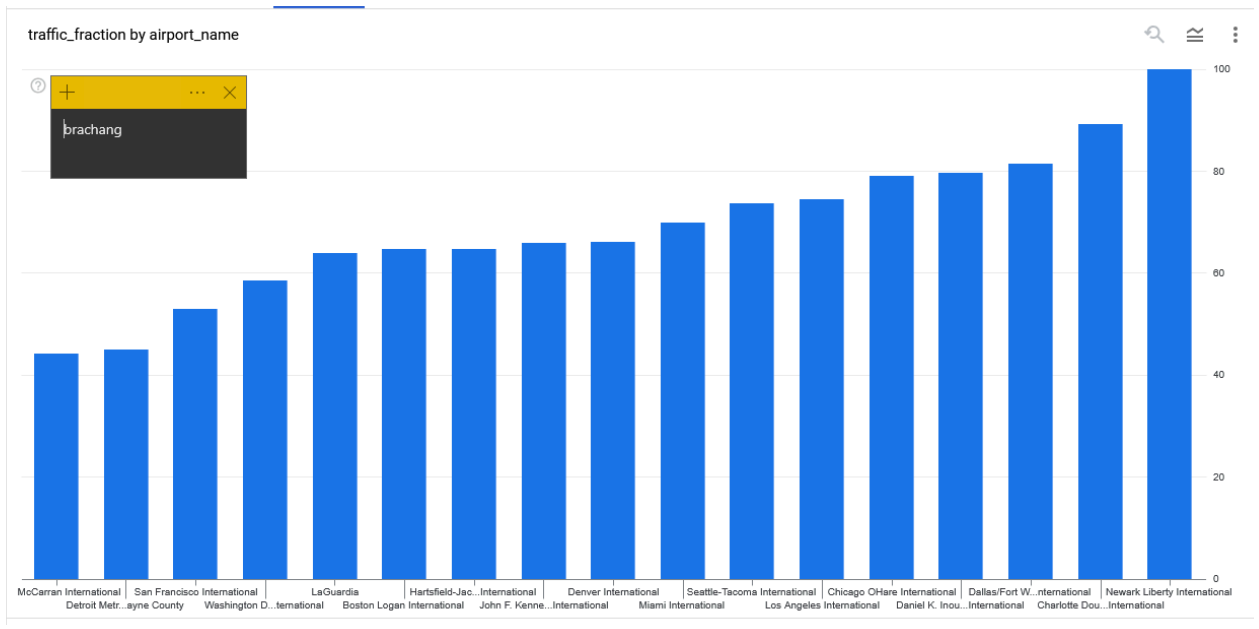
Which three airports were impacted the most in April 2020 (the month when lockdowns became widespread)?

Detroit, McCarran International, and San Francisco International



Run the query again using the month of August 2020. Which three airports were impacted the most?

The same airports Detroit, McCarran International, and San Francisco International



What table and columns identify the place name, the starting date, and the number of excess deaths from COVID-19?

excess_deaths table. Columns are country, start_date, excess_deaths

What table and columns identify the date, county, and deaths from COVID-19?

Table is us_counties. Columns are date, county, and deaths.

What table and columns identify the date, state, and confirmed cases of COVID-19?

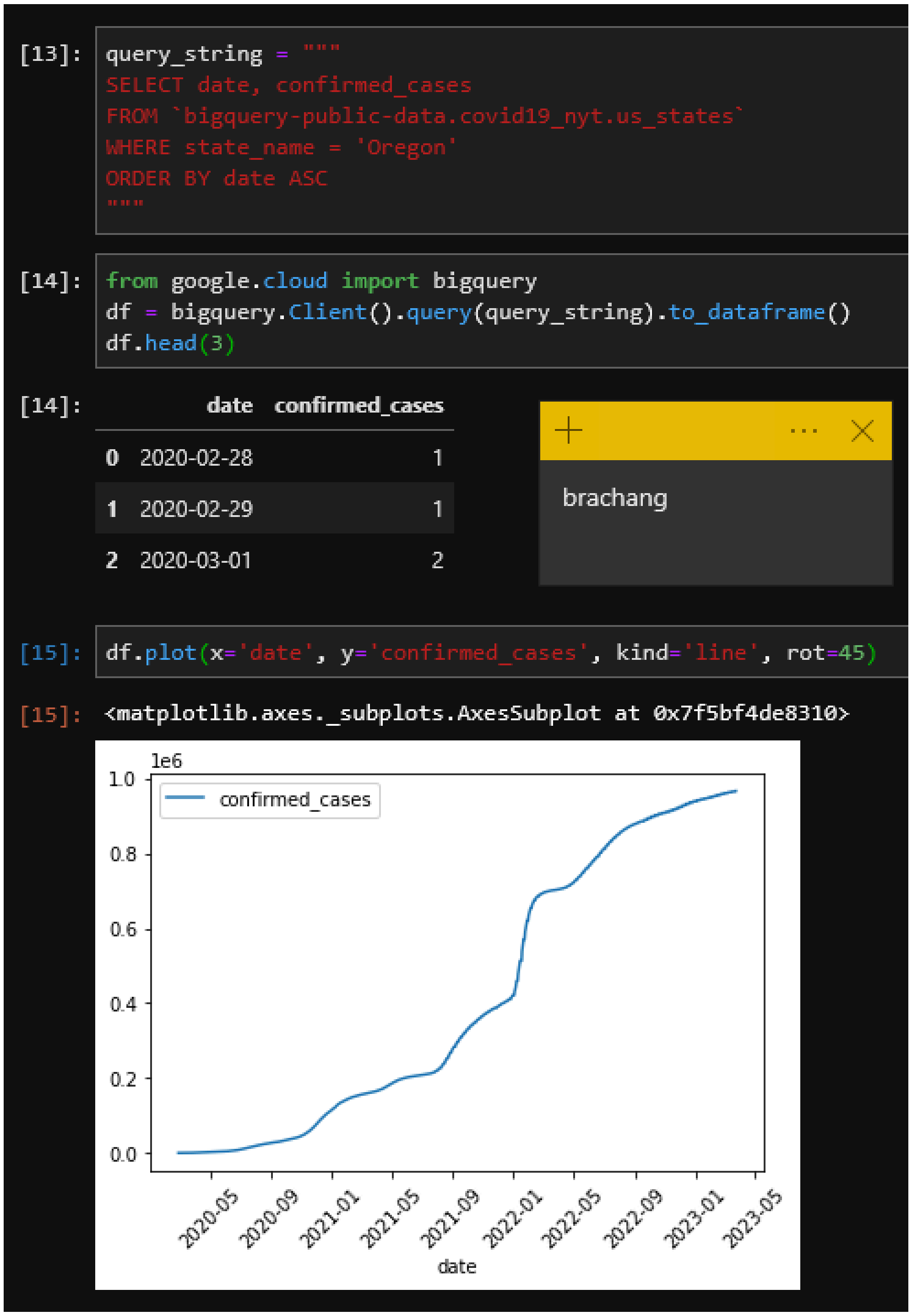
Table is us_dates. Columns are date, state_name, and confirmed cases.

What table and columns identify a county code and the percentage of its residents that report they always wear masks?

Table is mask_use_by_county. Columns are county_fips_code, and always.

11. Run example queries

Show a screenshot of the plot and the code used to generate it for your lab notebook



```
[13]: query_string = """
      SELECT date, confirmed_cases
      FROM `bigquery-public-data.covid19_nyt.us_states`
      WHERE state_name = 'Oregon'
      ORDER BY date ASC
      """
```

```
[14]: from google.cloud import bigquery
      df = bigquery.Client().query(query_string).to_dataframe()
      df.head(3)
```

[14]:

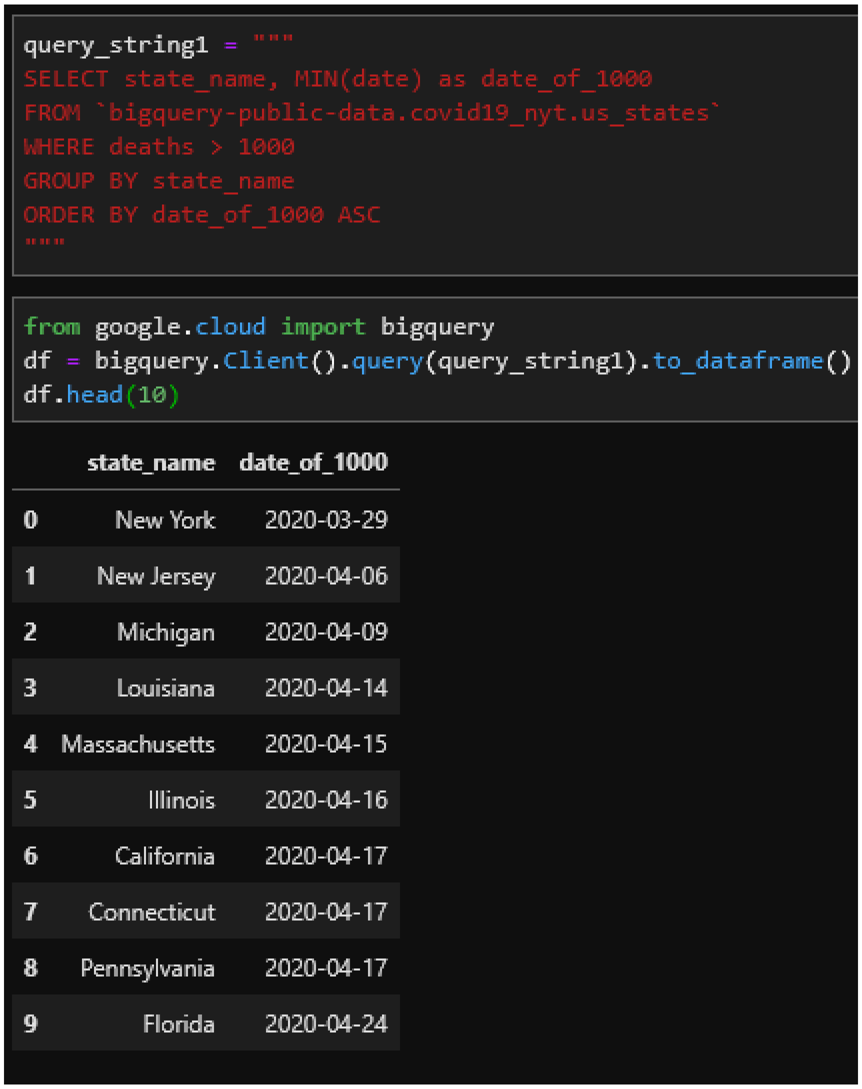
	date	confirmed_cases
0	2020-02-28	1
1	2020-02-29	1
2	2020-03-01	2

brachang

```
[15]: df.plot(x='date', y='confirmed_cases', kind='line', rot=45)
```

```
[15]: <matplotlib.axes._subplots.AxesSubplot at 0x7f5bf4de8310>
```

From within your Jupyter notebook, run the query and write code that shows the first 10 states that reached 1000 deaths from COVID-19. Take a screenshot for your lab notebook.

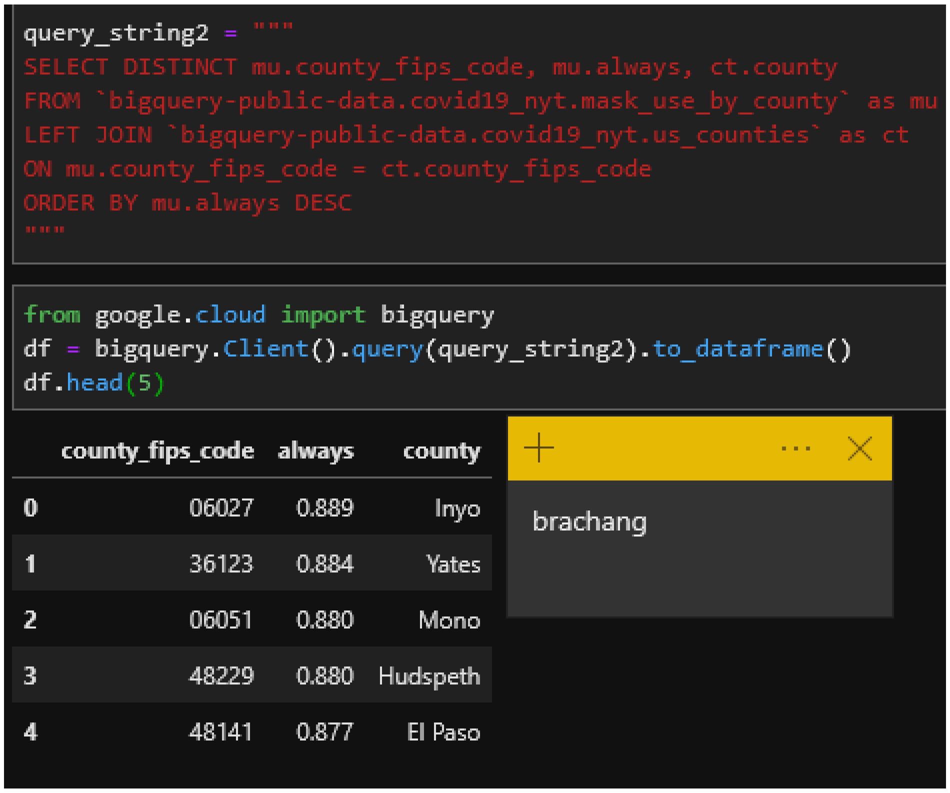


```
query_string1 = """
SELECT state_name, MIN(date) as date_of_1000
FROM `bigquery-public-data.covid19_nyt.us_states`
WHERE deaths > 1000
GROUP BY state_name
ORDER BY date_of_1000 ASC
"""
```

```
from google.cloud import bigquery
df = bigquery.Client().query(query_string1).to_dataframe()
df.head(10)
```

	state_name	date_of_1000
0	New York	2020-03-29
1	New Jersey	2020-04-06
2	Michigan	2020-04-09
3	Louisiana	2020-04-14
4	Massachusetts	2020-04-15
5	Illinois	2020-04-16
6	California	2020-04-17
7	Connecticut	2020-04-17
8	Pennsylvania	2020-04-17
9	Florida	2020-04-24

Take a screenshot for your lab notebook of the Top 5 counties and the states they are located in.


```
query_string2 = """
SELECT DISTINCT mu.county_fips_code, mu.always, ct.county
FROM `bigquery-public-data.covid19_nyt.mask_use_by_county` as mu
LEFT JOIN `bigquery-public-data.covid19_nyt.us_counties` as ct
ON mu.county_fips_code = ct.county_fips_code
ORDER BY mu.always DESC
"""
```

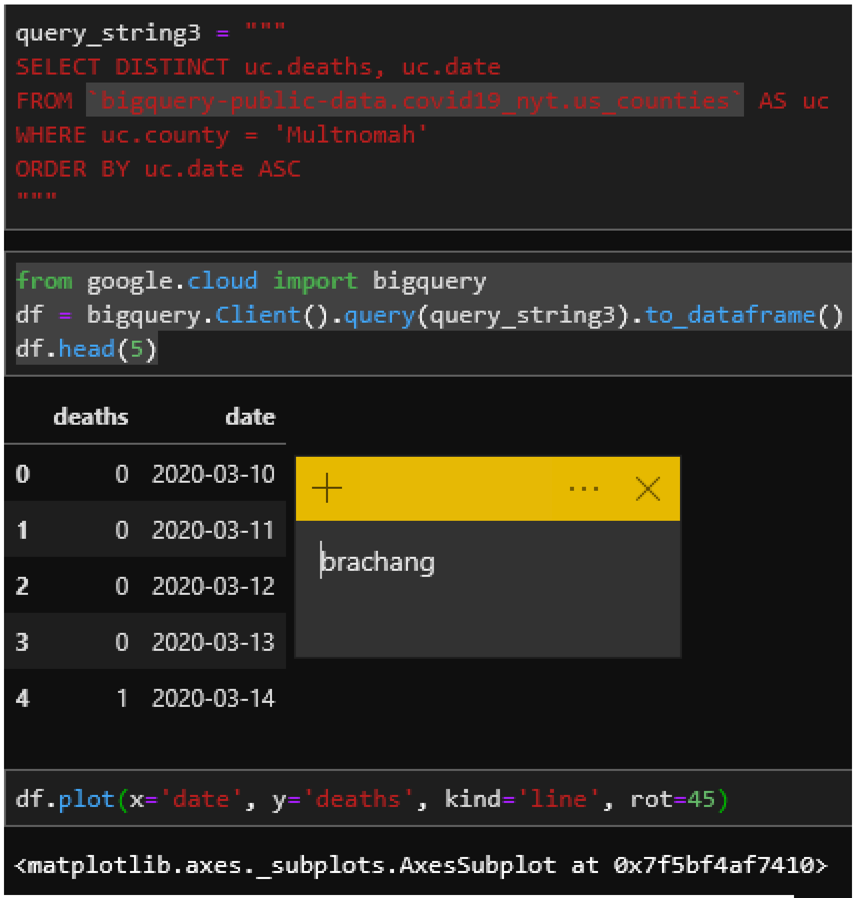
```
from google.cloud import bigquery
df = bigquery.Client().query(query_string2).to_dataframe()
df.head(5)
```

	county_fips_code	always	county
0	06027	0.889	Inyo
1	36123	0.884	Yates
2	06051	0.880	Mono
3	48229	0.880	Hudspeth
4	48141	0.877	El Paso

brachang

12. Write queries

Plot the results and take a screenshot for your lab notebook.



```
query_string3 = """
SELECT DISTINCT uc.deaths, uc.date
FROM `bigquery-public-data.covid19_nyt.us_counties` AS uc
WHERE uc.county = 'Multnomah'
ORDER BY uc.date ASC
"""
```

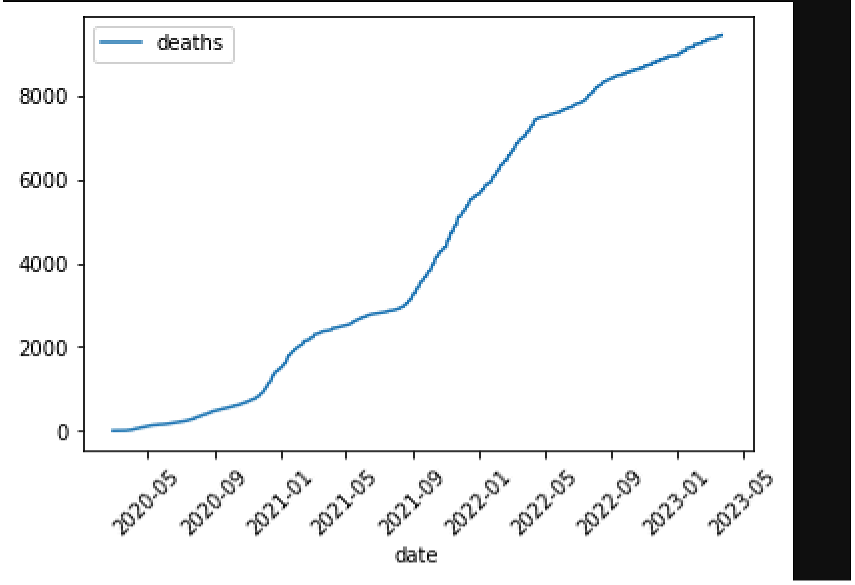
```
from google.cloud import bigquery
df = bigquery.Client().query(query_string3).to_dataframe()
df.head(5)
```

	deaths	date
0	0	2020-03-10
1	0	2020-03-11
2	0	2020-03-12
3	0	2020-03-13
4	1	2020-03-14

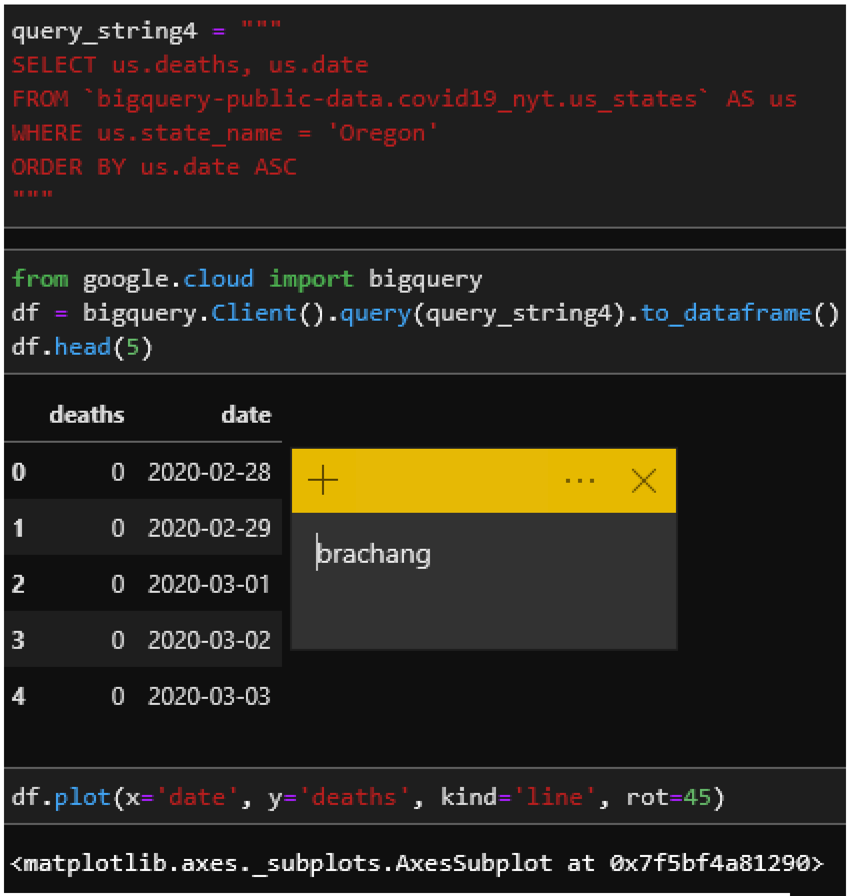
brachang

```
df.plot(x='date', y='deaths', kind='line', rot=45)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5bf4af7410>
```



Plot the results and take a screenshot for your lab notebook.



```
query_string4 = """
SELECT us.deaths, us.date
FROM `bigquery-public-data.covid19_nyt.us_states` AS us
WHERE us.state_name = 'Oregon'
ORDER BY us.date ASC
"""
```

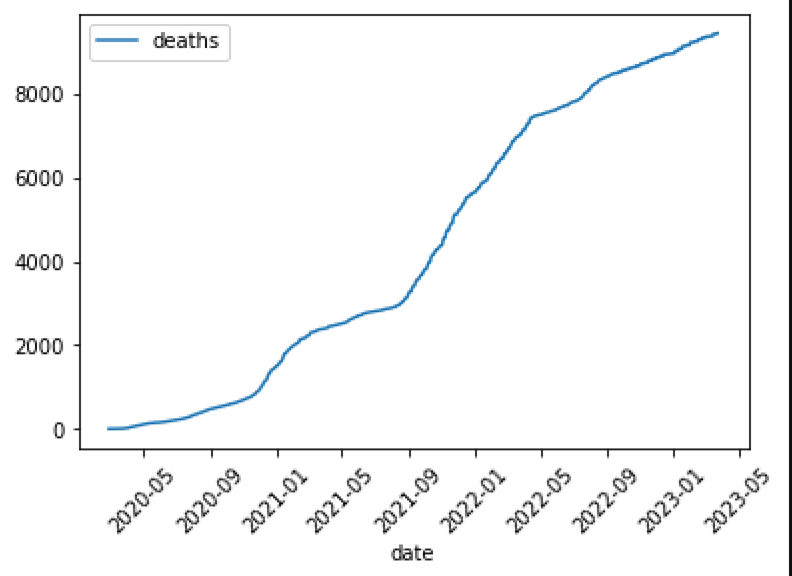
```
from google.cloud import bigquery
df = bigquery.Client().query(query_string4).to_dataframe()
df.head(5)
```

	deaths	date
0	0	2020-02-28
1	0	2020-02-29
2	0	2020-03-01
3	0	2020-03-02
4	0	2020-03-03

brachang

```
df.plot(x='date', y='deaths', kind='line', rot=45)
```

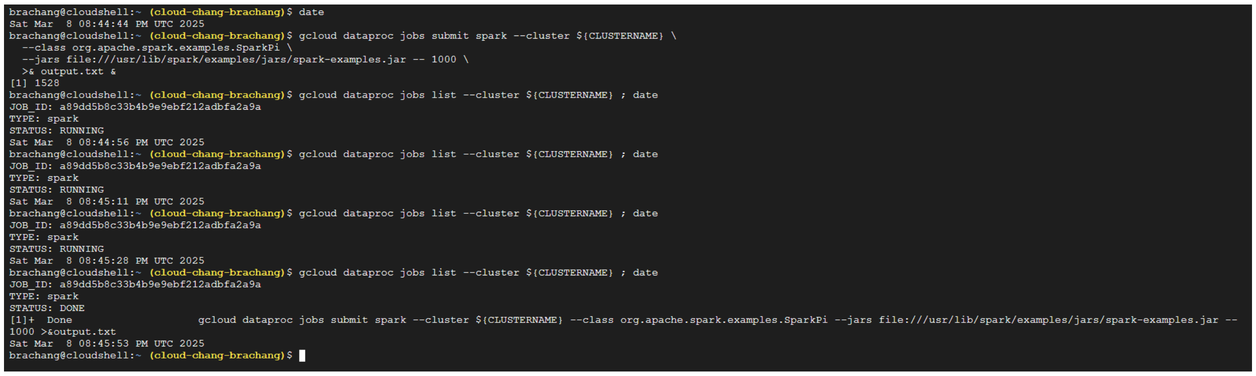
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5bf4a81290>
```



9.3g: Dataproc

6. Run computation

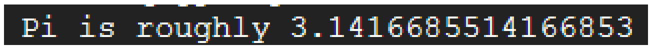
How long did the job take to execute?



```
brachang@cloudshell:~ (cloud-chang-brachang)$ date
Sat Mar  8 08:44:44 PM UTC 2025
brachang@cloudshell:~ (cloud-chang-brachang)$ gcloud dataproc jobs submit spark --cluster ${CLUSTERNAME} \
  --class org.apache.spark.examples.SparkPi \
  --jars file:///usr/lib/spark/examples/jars/spark-examples.jar -- 1000 \
  >& output.txt &
[1] 1528
brachang@cloudshell:~ (cloud-chang-brachang)$ gcloud dataproc jobs list --cluster ${CLUSTERNAME} ; date
JOB_ID: a89dd5b8c33b4b9e9ebf212adbfa2a9a
TYPE: spark
STATUS: RUNNING
Sat Mar  8 08:44:56 PM UTC 2025
brachang@cloudshell:~ (cloud-chang-brachang)$ gcloud dataproc jobs list --cluster ${CLUSTERNAME} ; date
JOB_ID: a89dd5b8c33b4b9e9ebf212adbfa2a9a
TYPE: spark
STATUS: RUNNING
Sat Mar  8 08:45:11 PM UTC 2025
brachang@cloudshell:~ (cloud-chang-brachang)$ gcloud dataproc jobs list --cluster ${CLUSTERNAME} ; date
JOB_ID: a89dd5b8c33b4b9e9ebf212adbfa2a9a
TYPE: spark
STATUS: RUNNING
Sat Mar  8 08:45:28 PM UTC 2025
brachang@cloudshell:~ (cloud-chang-brachang)$ gcloud dataproc jobs list --cluster ${CLUSTERNAME} ; date
JOB_ID: a89dd5b8c33b4b9e9ebf212adbfa2a9a
TYPE: spark
STATUS: DONE
[1]+  Done                    gcloud dataproc jobs submit spark --cluster ${CLUSTERNAME} --class org.apache.spark.examples.SparkPi --jars file:///usr/lib/spark/examples/jars/spark-examples.jar --
1000 >&output.txt
Sat Mar  8 08:45:53 PM UTC 2025
brachang@cloudshell:~ (cloud-chang-brachang)$ 
```

About a minute

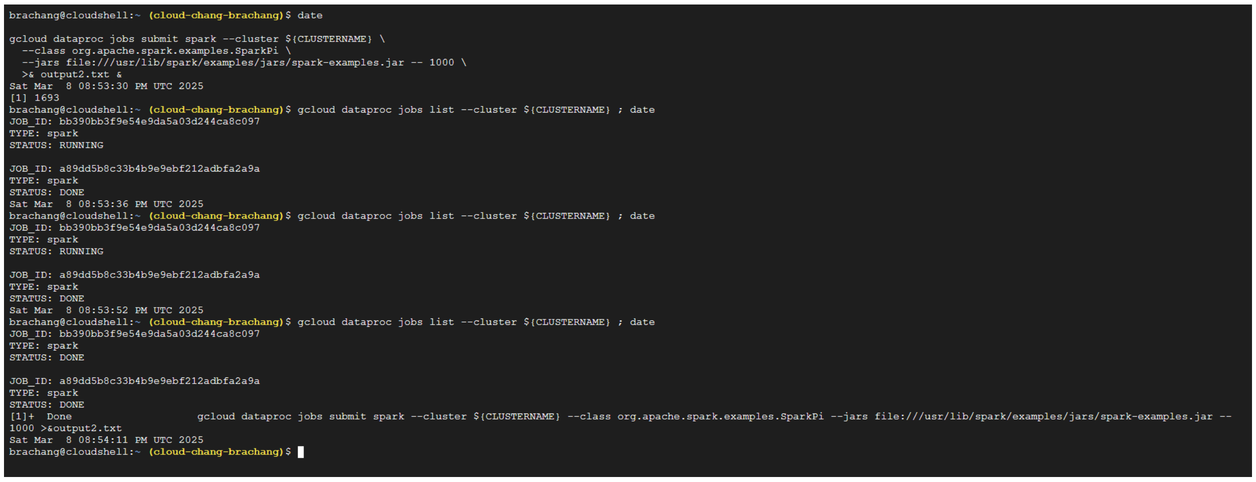
Examine output.txt and show the estimate of π calculated.



```
Pi is roughly 3.1416685514166853
```

8. Run computation again

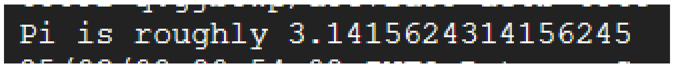
How long did the job take to execute? How much faster did it take?



```
brachang@cloudshell:~ (cloud-chang-brachang)$ date

gcloud dataproc jobs submit spark --cluster ${CLUSTERNAME} \
  --class org.apache.spark.examples.SparkPi \
  --jars file:///usr/lib/spark/examples/jars/spark-examples.jar -- 1000 \
  >& output2.txt &
Sat Mar  8 08:53:30 PM UTC 2025
[1] 1693
brachang@cloudshell:~ (cloud-chang-brachang)$ gcloud dataproc jobs list --cluster ${CLUSTERNAME} ; date
JOB_ID: bb390bb3f9e54e9da5a03d244ca8c097
TYPE: spark
STATUS: RUNNING

JOB_ID: a89dd5b8c33b4b9e9ebf212adbfa2a9a
TYPE: spark
STATUS: DONE
Sat Mar  8 08:53:36 PM UTC 2025
brachang@cloudshell:~ (cloud-chang-brachang)$ gcloud dataproc jobs list --cluster ${CLUSTERNAME} ; date
JOB_ID: bb390bb3f9e54e9da5a03d244ca8c097
TYPE: spark
STATUS: RUNNING

JOB_ID: a89dd5b8c33b4b9e9ebf212adbfa2a9a
TYPE: spark
STATUS: DONE
Sat Mar  8 08:53:52 PM UTC 2025
brachang@cloudshell:~ (cloud-chang-brachang)$ gcloud dataproc jobs list --cluster ${CLUSTERNAME} ; date
JOB_ID: bb390bb3f9e54e9da5a03d244ca8c097
TYPE: spark
STATUS: DONE

JOB_ID: a89dd5b8c33b4b9e9ebf212adbfa2a9a
TYPE: spark
STATUS: DONE
[1]+  Done                    gcloud dataproc jobs submit spark --cluster ${CLUSTERNAME} --class org.apache.spark.examples.SparkPi --jars file:///usr/lib/spark/examples/jars/spark-examples.jar --
1000 >&output2.txt
Sat Mar  8 08:54:11 PM UTC 2025
brachang@cloudshell:~ (cloud-chang-brachang)$ 
```

About 40 secs

Examine output2.txt and show the estimate of π calculated.



```
Pi is roughly 3.1415624314156245
```

9.4g: Dataflow

3. Beam code

Where is the input taken from by default?

parser.add_argument('--input',

default='../javahelp/src/main/java/com/google/cloud/training/dataanalyst/javahelp/', help='Input directory')

Where does the output go by default?

parser.add_argument('--output_prefix', default='/tmp/output', help='Output prefix')

Examine both the getPackages() function and the splitPackageName() function. What operation does the 'PackageUse()' transform implement?

packages = getPackages(line, keyword)

Look up Beam's CombinePerKey. What operation does the TotalUse operation implement?

CombinePerKey "Identifies sets of values associated with the same key in the input PCollection, then applies a CombineFn to condense those sets to single values." So TotalUse takes a set of values with the same key and sums them up.

Which operations correspond to a "Map"?

| 'GetImports' >> beam.FlatMap(lambda line: startsWith(line, keyword))

| 'PackageUse' >> beam.FlatMap(lambda line: packageUse(line, keyword))

Which operation corresponds to a "Shuffle-Reduce"?

| 'TotalUse' >> beam.CombinePerKey(sum)

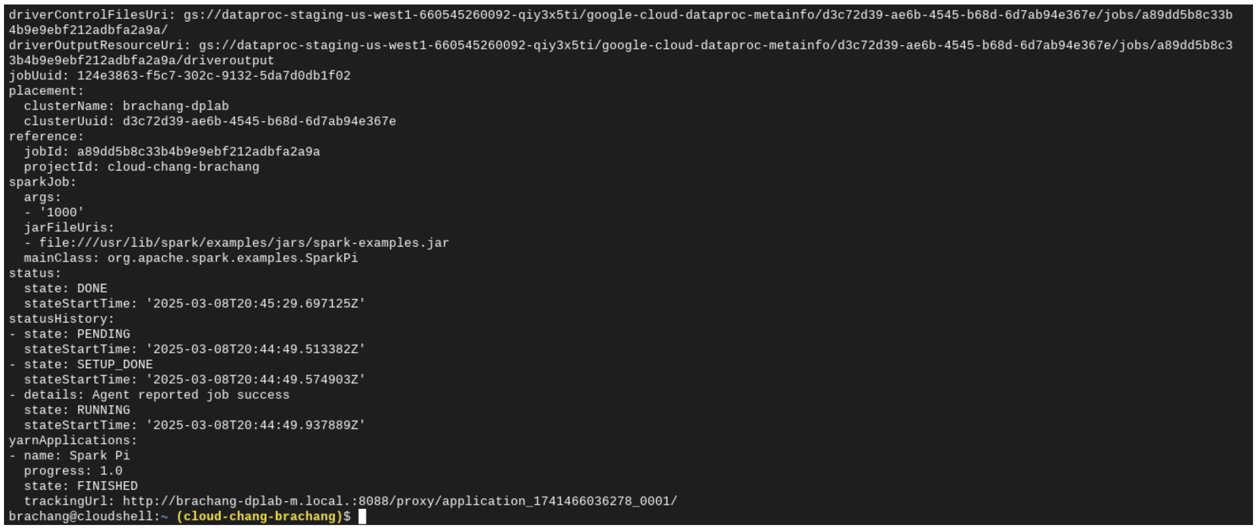
Which operation corresponds to a "Reduce"?

| 'Top_5' >> beam.transforms.combiners.Top.Of(5, key=lambda kv: kv[1])

| 'write' >> beam.io.WriteToText(output_prefix)

4. Run pipeline locally

Take a screenshot of its contents



```
driverControlFilesUri: gs://dataproc-staging-us-west1-660545260092-qiy3x5ti/google-cloud-dataproc-metainfo/d3c72d39-ae6b-4545-b68d-6d7ab94e367e/jobs/a89dd5b8c33b
4b9e9ebf212adbfa2a9a/
driverOutputResourceUri: gs://dataproc-staging-us-west1-660545260092-qiy3x5ti/google-cloud-dataproc-metainfo/d3c72d39-ae6b-4545-b68d-6d7ab94e367e/jobs/a89dd5b8c3
3b4b9e9ebf212adbfa2a9a/driveroutput
jobUuid: 124e3863-f5c7-302c-9132-5da7d0db1f02
placement:
  clusterName: brachang-dplab
  clusterUuid: d3c72d39-ae6b-4545-b68d-6d7ab94e367e
reference:
  jobId: a89dd5b8c33b4b9e9ebf212adbfa2a9a
  projectId: cloud-chang-brachang
sparkJob:
  args:
  - '1000'
  jarFileUris:
  - file:///usr/lib/spark/examples/jars/spark-examples.jar
  mainClass: org.apache.spark.examples.SparkPi
status:
  state: DONE
  stateStartTime: '2025-03-08T20:45:29.697125Z'
statusHistory:
- state: PENDING
  stateStartTime: '2025-03-08T20:44:49.513382Z'
- state: SETUP_DONE
  stateStartTime: '2025-03-08T20:44:49.574903Z'
- details: Agent reported job success
  state: RUNNING
  stateStartTime: '2025-03-08T20:44:49.937889Z'
yarnApplications:
- name: Spark Pi
  progress: 1.0
  state: FINISHED
  trackingUrl: http://brachang-dplab-m.local.:8088/proxy/application_1741466036278_0001/
brachang@cloudshell:~ (cloud-chang-brachang)$
```

Explain what the data in this output file corresponds to based on your understanding of the program.

Listing all the jobs

5. Dataflow Lab #2 (Word count)

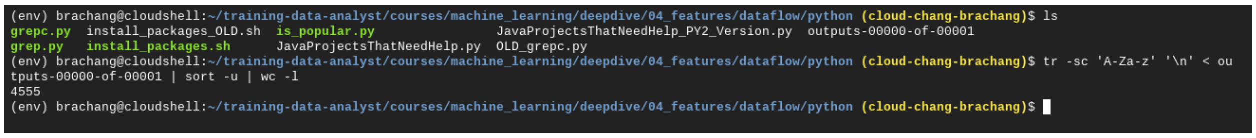
What are the names of the stages in the pipeline?

Split, PairWithOne, GroupAndSum

Describe what each stage does.

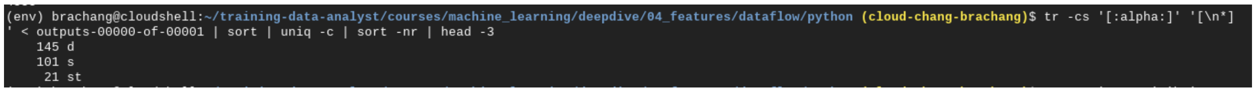
6. Run code locally

Use wc with an appropriate flag to determine the number of different words in King Lear.



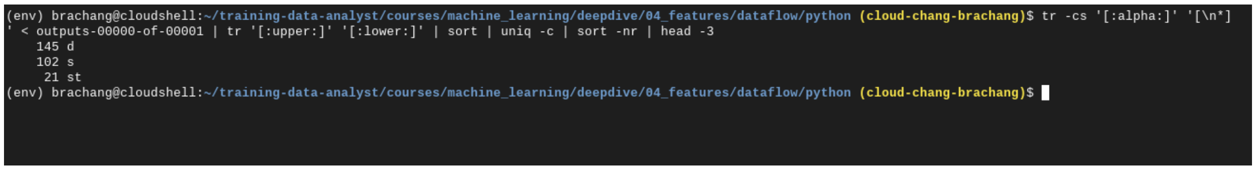
```
(env) brachang@cloudshell:~/training-data-analyst/courses/machine_learning/deepdive/04_features/dataflow/python (cloud-chang-brachang)$ ls
grepc.py  install_packages_OLD.sh  is_popular.py               JavaProjectsThatNeedHelp_PY2_Version.py  outputs-00000-of-00001
grep.py   install_packages.sh      JavaProjectsThatNeedHelp.py  OLD_grepc.py
(env) brachang@cloudshell:~/training-data-analyst/courses/machine_learning/deepdive/04_features/dataflow/python (cloud-chang-brachang)$ tr -sc 'A-Za-z' '\n' < ou
tputs-00000-of-00001 | sort -u | wc -l
4555
(env) brachang@cloudshell:~/training-data-analyst/courses/machine_learning/deepdive/04_features/dataflow/python (cloud-chang-brachang)$
```

Use sort with appropriate flags to perform a numeric sort on the key field containing the count for each word in descending order. Pipe the output into head to show the top 3 words in King Lear and the number of times they appear



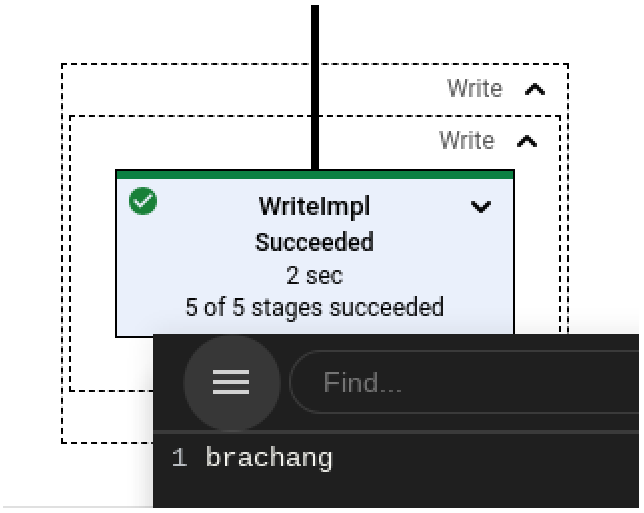
```
(env) brachang@cloudshell:~/training-data-analyst/courses/machine_learning/deepdive/04_features/dataflow/python (cloud-chang-brachang)$ tr -cs '[:alpha:]' '[\n*]
' < outputs-00000-of-00001 | sort | uniq -c | sort -nr | head -3
    145 d
    101 s
     21 st
```

Use the previous method to show the top 3 words in King Lear, case-insensitive, and the number of times they appear.

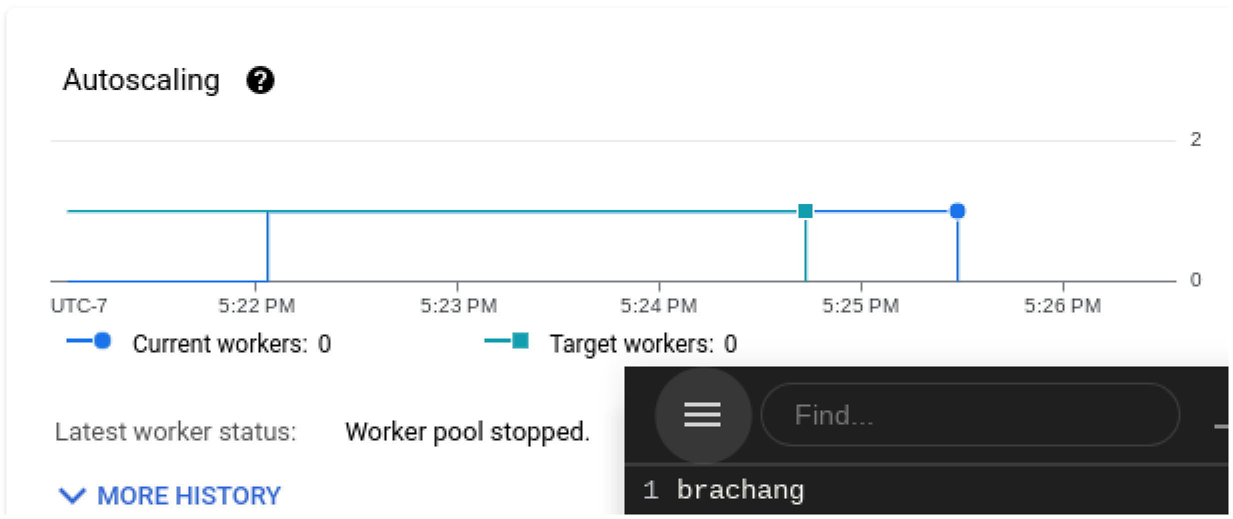


```
(env) brachang@cloudshell:~/training-data-analyst/courses/machine_learning/deepdive/04_features/dataflow/python (cloud-chang-brachang)$ tr -cs '[:alpha:]' '[\n*]
' < outputs-00000-of-00001 | tr '[:upper:]' '[:lower:]' | sort | uniq -c | sort -nr | head -3
    145 d
    102 s
     21 st
(env) brachang@cloudshell:~/training-data-analyst/courses/machine_learning/deepdive/04_features/dataflow/python (cloud-chang-brachang)$
```

The part of the job graph that has taken the longest time to complete.



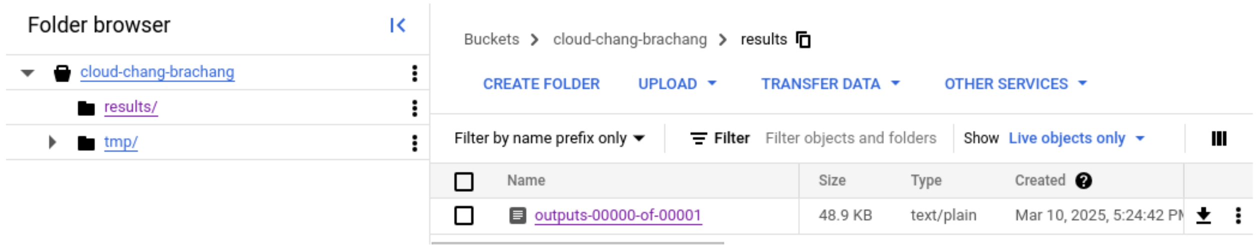
The autoscaling graph showing when the worker was created and stopped.



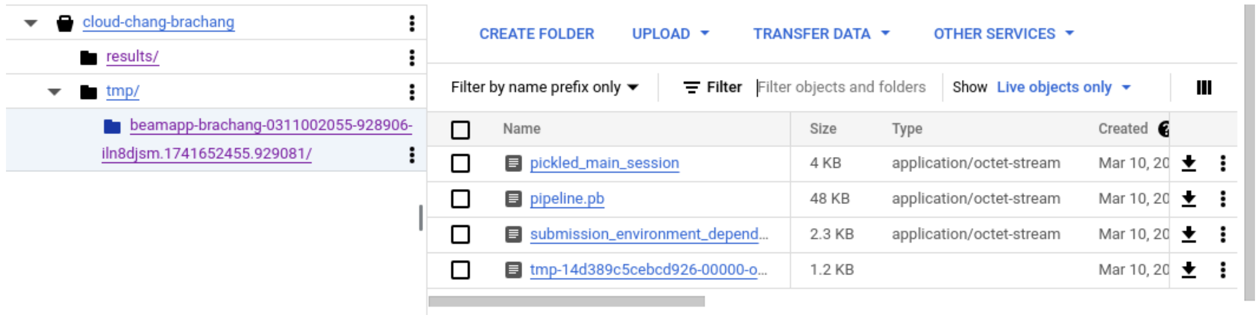
Latest worker status: Worker pool stopped.

```
1 brachang
```

Examine the output directory in Cloud Storage. How many files has the final write stage in the pipeline created?



Name	Size	Type	Created
outputs-00000-of-00001	48.9 KB	text/plain	Mar 10, 2025, 5:24:42 PM

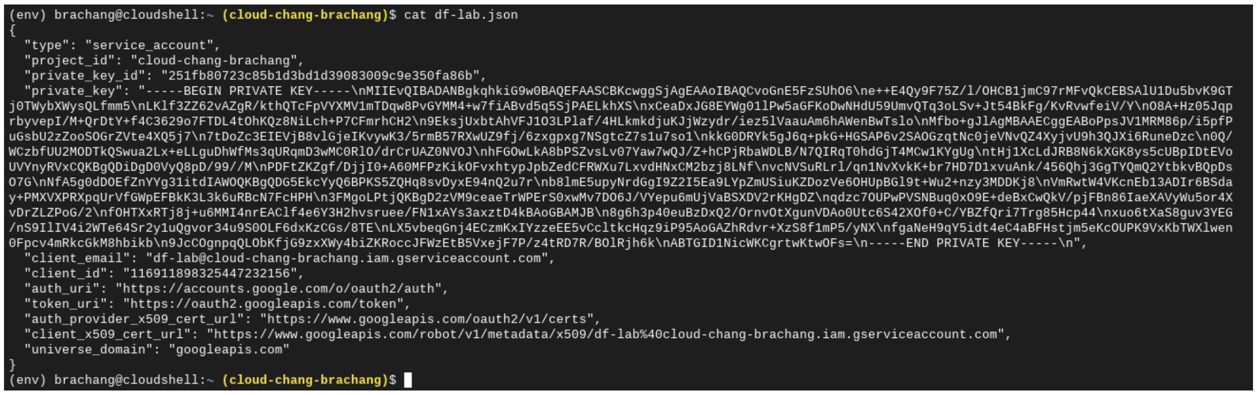

cloud-chang-brachang
results/
tmp/
beamapp-brachang-0311002055-928906-iln8djsm.1741652455.929081/
CREATE FOLDER
UPLOAD
TRANSFER DATA
OTHER SERVICES
Filter by name prefix only
Filter
Filter objects and folders
Show Live objects only

Name	Size	Type	Created
pickled_main_session	4 KB	application/octet-stream	Mar 10, 20
pipeline.pb	48 KB	application/octet-stream	Mar 10, 20
submission_environment_depend...	2.3 KB	application/octet-stream	Mar 10, 20
tmp-14d389c5cebcd926-00000-o...	1.2 KB		Mar 10, 20

Created 5 files including the output file in results

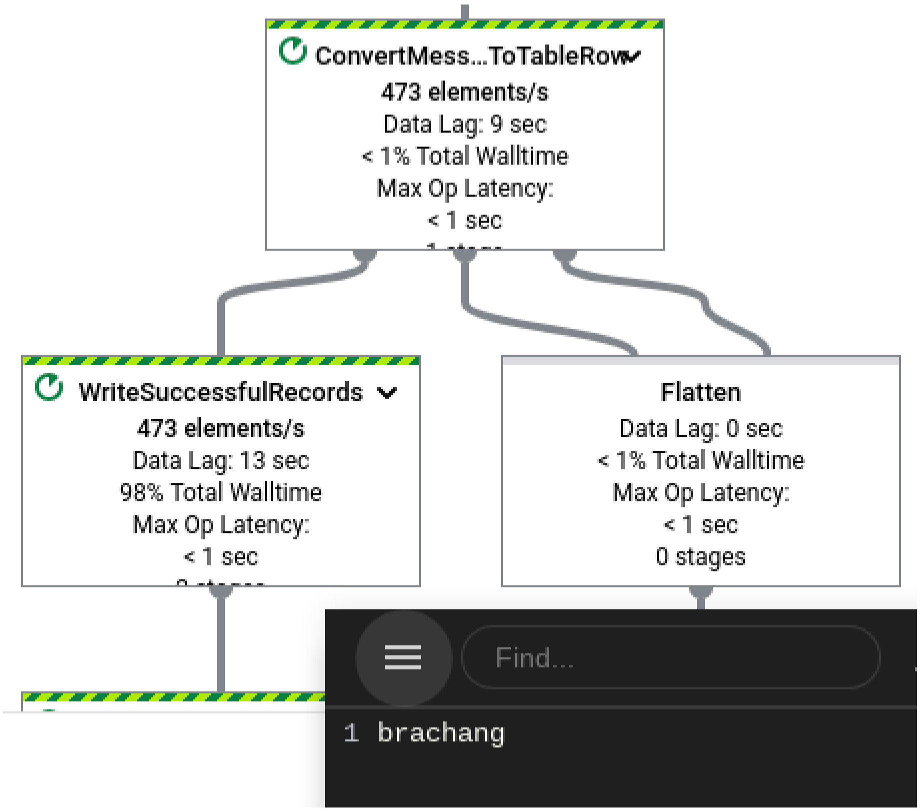
12. View raw data from PubSub

Take a screenshot listing the different fields of this object.



```
(env) brachang@cloudshell:~ (cloud-chang-brachang)$ cat df-lab.json
{
  "type": "service_account",
  "project_id": "cloud-chang-brachang",
  "private_key_id": "251fb80723c85b1d3bd1d39083009c9e350fa86b",
  "private_key": "-----BEGIN PRIVATE KEY-----\nMIIEvQIBADANBgkqhkiG9w0BAQEFAASCBKcwggSjAgEAAoIBAQCvoGnE5FzSUhO6\ne++E4Qy9F75Z/l/OHCB1jmC97rMFvQkCEBSAlU1Du5bvK9GT
j0TWybXWysQLfmm5\nLKlf3ZZ62vAZgR/kthQTcFpVYXMV1mTDqw8PvGYMM4+w7fiABvd5q5SjPAELkhXS\nxCeaDxJG8EYWg01lPw5aGFKoDwNHdU59UmvQTq3oLSv+Jt54BkFg/KvRvwfeiV/Y\nO8A+Hz05Jqp
rbyvepI/M+QrDtY+f4C3629o7FTDL4tOhKQz8NiLch+P7CFmrhCH2\n9EksjUxbtAhVFJ1O3LPlaf/4HLkmkdjuKJjWzydr/iez5lVaauAm6hAWenBwTslo\nMfbo+gJlAgMBAAECggEABoPpsJV1MRM86p/i5pfP
uGsbU2zZooSOGrZVte4XQ5j7\n7tDoZc3EIEVjB8vlGjeIKvywK3/5rmB57RXwUZ9fj/6zxgpxg7NSgtcZ7s1u7so1\nkkG0DRYk5gJ6q+pkG+HGSAP6v2SAOGzqtNc0jeVNvQZ4XyjvU9h3QJXi6RuneDzc\n0Q/
WCzbfUU2MODTkQSwua2Lx+eLLguDhWfMs3qURqmD3wMC0RlO/drCrUAZ0NVOJ\nhFGOwLkA8bPSZvsLv07Yaw7wQJ/Z+hCPjRbaWDLB/N7QIRqT0hdGjT4MCw1KYgUg\ntHj1XcLdJRB8N6kXGK8ys5cUBpIDtEVo
UVYnyRVxCQKBgQDiDgD0VyQ8pD/99//M\nPDFtZKZgf/DjjI0+A60MFPzKikOFvxhtypJpbZedCFRWXu7LxvdHNxCM2bzj8LNf\nvcNVSuRLrl/qn1NvXvkK+br7HD7D1xvuAnk/456Qhj3GgTYQmQ2YtbkvBQpDs
O7G\nNfA5g0dDOEfZnYYg31itdIAWOQKBgQDG5EkcYyQ6BPKS5ZQHq8svDyxE94nQ2u7r\nb8lmE5upyNrdGgI9Z2I5Ea9LYpZmUSiuKZDozVe6OHUpBGl9t+Wu2+nzy3MDDKj8\nVmRwtW4VKcnEb13ADIr6BSda
y+PMXVXPRXpqUrVfGWpEFBkK3L3k6uRBcN7FcHPH\n3FMgoLPtjQKBgD2zVM9ceaeTrWPErS0xwMv7DO6J/VYepu6mUjVaBSXDV2rKHgDZ\nqdzc7OUPwPVSNBuq0xO9E+deBxCwQkV/pjFBn86IaeXAVyWu5or4X
vDrZLZPoG/2\nfOHTXxRTj8j+u6MMI4nrEAClf4e6Y3H2hvsruee/FN1xAYs3axztD4kBAoGBAMJB\n8g6h3p40euBzDxQ2/OrnvOtXgunVDAo0Utc6S42XOf0+C/YBZfQri7Trg85Hcp44\nxuo6tXaS8guv3YEG
/nS9IlIV4i2WTe64Sr2y1uQgvor34u9S0OLF6dxKzCGs/8TE\nLX5vbeqGnj4ECzmKxIYzzeEE5vCcltkcHqz9iP95AoGAZhRdvr+XzS8f1mP5/yNX\nfgaNeH9qY5idt4eC4aBFHstjm5eKcOUPK9VxKbTWXlwen
0Fpcv4mRkcGkM8hbikb\n9JcCOgnpqQLObKfjG9zxXWy4biZKRoccJFWzEtB5VxejF7P/z4tRD7R/BOlRjh6k\nABTGID1NicWKCgrtwKtwOFs=\n-----END PRIVATE KEY-----\n",
  "client_email": "df-lab@cloud-chang-brachang.iam.gserviceaccount.com",
  "client_id": "116911898325447232156",
  "auth_uri": "https://accounts.google.com/o/oauth2/auth",
  "token_uri": "https://oauth2.googleapis.com/token",
  "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs",
  "client_x509_cert_url": "https://www.googleapis.com/robot/v1/metadata/x509/df-lab%40cloud-chang-brachang.iam.gserviceaccount.com",
  "universe_domain": "googleapis.com"
}
(env) brachang@cloudshell:~ (cloud-chang-brachang)$
```

14. Run Dataflow job from template



ConvertMess...ToTableRow
473 elements/s
Data Lag: 9 sec
< 1% Total Walltime
Max Op Latency:
< 1 sec

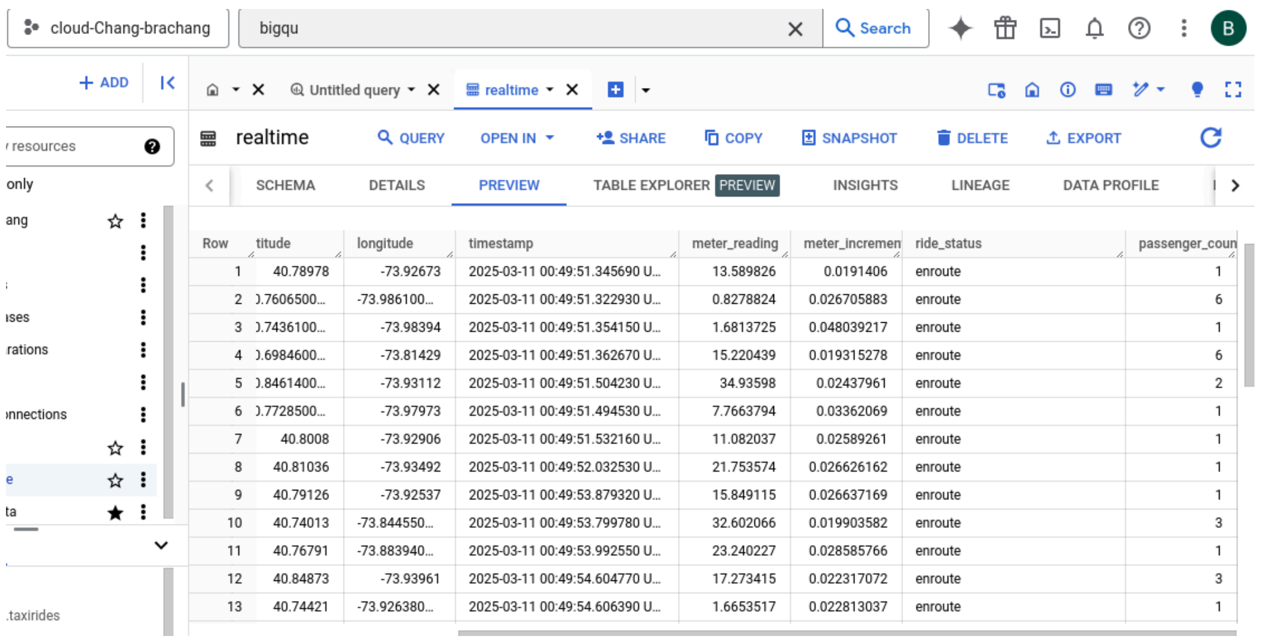
WriteSuccessfulRecords
473 elements/s
Data Lag: 13 sec
98% Total Walltime
Max Op Latency:
< 1 sec

Flatten
Data Lag: 0 sec
< 1% Total Walltime
Max Op Latency:
< 1 sec
0 stages

brachang

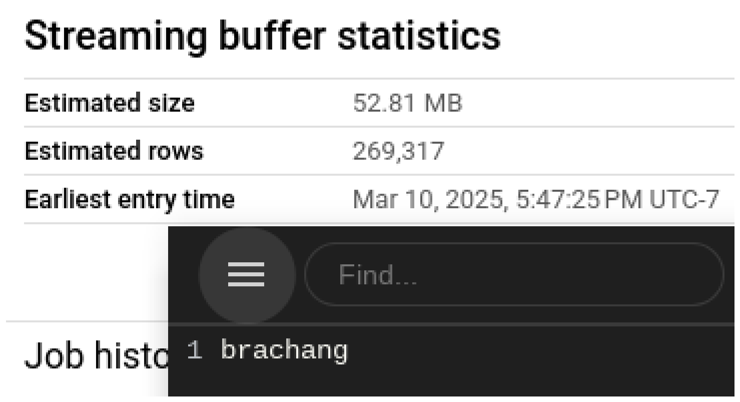
15. Query data in BigQuery

Take a screenshot showing the number of passengers and the amount paid for the first ride



Row	titude	longitude	timestamp	meter_reading	meter_incremen	ride_status	passenger_coun
1	40.78978	-73.92673	2025-03-11 00:49:51.345690 U...	13.589826	0.0191406	enroute	1
2	).7606500...	-73.986100...	2025-03-11 00:49:51.322930 U...	0.8278824	0.026705883	enroute	6
3	).7436100...	-73.98394	2025-03-11 00:49:51.354150 U...	1.6813725	0.048039217	enroute	1
4	).6984600...	-73.81429	2025-03-11 00:49:51.362670 U...	15.220439	0.019315278	enroute	6
5	).8461400...	-73.93112	2025-03-11 00:49:51.504230 U...	34.93598	0.02437961	enroute	2
6	).7728500...	-73.97973	2025-03-11 00:49:51.494530 U...	7.7663794	0.03362069	enroute	1
7	40.8008	-73.92906	2025-03-11 00:49:51.532160 U...	11.082037	0.02589261	enroute	1
8	40.81036	-73.93492	2025-03-11 00:49:52.032530 U...	21.753574	0.026626162	enroute	1
9	40.79126	-73.92537	2025-03-11 00:49:53.879320 U...	15.849115	0.026637169	enroute	1
10	40.74013	-73.844550...	2025-03-11 00:49:53.799780 U...	32.602066	0.019903582	enroute	3
11	40.76791	-73.883940...	2025-03-11 00:49:53.992550 U...	23.240227	0.028585766	enroute	1
12	40.84873	-73.93961	2025-03-11 00:49:54.604770 U...	17.273415	0.022317072	enroute	3
13	40.74421	-73.926380...	2025-03-11 00:49:54.606390 U...	1.6653517	0.022813037	enroute	1

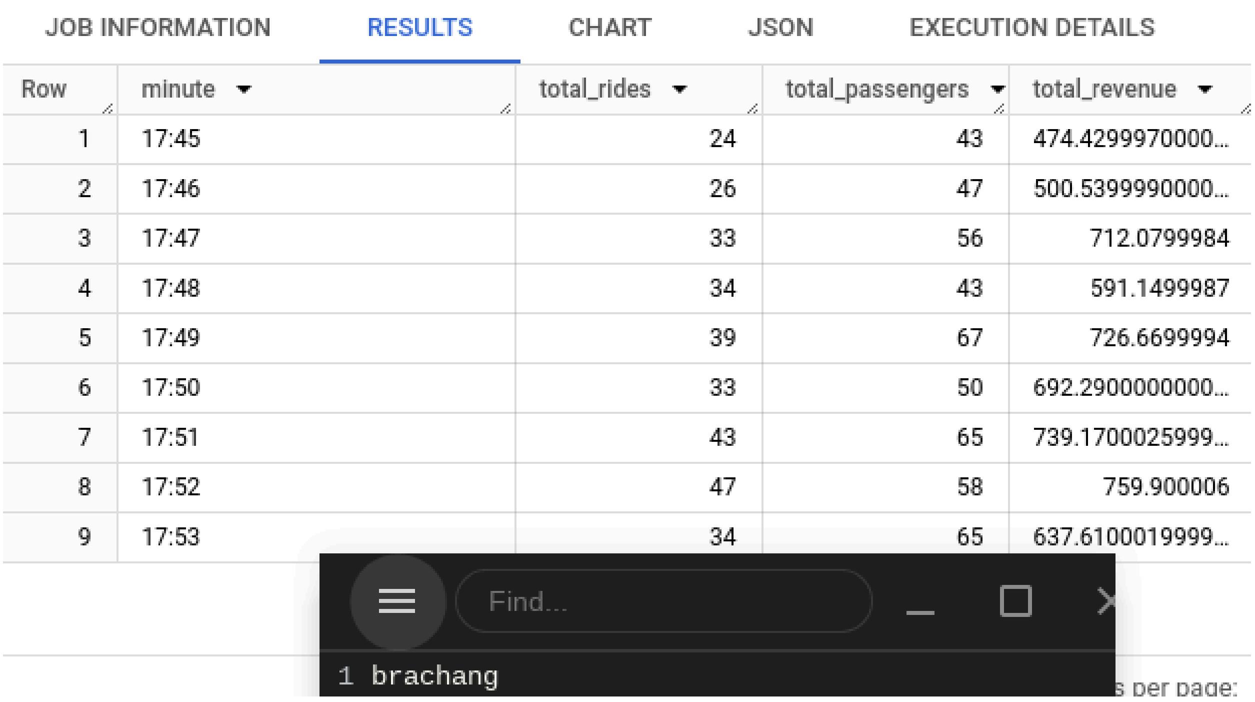
Take a screenshot showing the estimated number of rows in the table.



Streaming buffer statistics

Estimated size	52.81 MB
Estimated rows	269,317
Earliest entry time	Mar 10, 2025, 5:47:25 PM UTC-7

Take a screenshot showing the per-minute number of rides, passengers, and revenue for the data collected



JOB INFORMATION RESULTS CHART JSON EXECUTION DETAILS

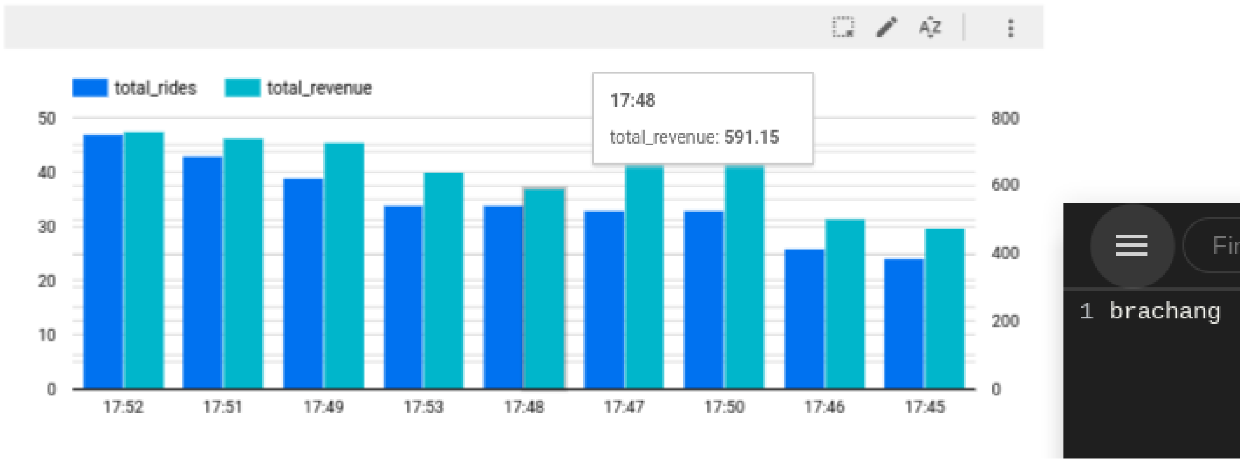
Row	minute	total_rides	total_passengers	total_revenue
1	17:45	24	43	474.4299970000...
2	17:46	26	47	500.5399990000...
3	17:47	33	56	712.0799984
4	17:48	34	43	591.1499987
5	17:49	39	67	726.6699994
6	17:50	33	50	692.2900000000...
7	17:51	43	65	739.1700025999...
8	17:52	47	58	759.900006
9	17:53	34	65	637.6100019999...

```
1 brachang
```

s per page:

16. Data visualization

Take a screenshot showing the plot for your data for your lab notebook



```
1 brachang
```