# Index

[**Index**](#_aqmhas6swn46) **1**

[**Week 1**](#_lmn4xz2kygkn) **3**

[Docker run for windows](#_elp0kb5zhc72) 3

[Running Postgres with Docker](#_tuopdu4khw5f) 3

[Error with Pgcli](#_5royb3m49mag) 4

[Problems while connecting to Postgres in Python](#_y1evrcypfhb0) 5

[Docker-compose - Error translating host name to address](#_7vh3v4remj0w) 6

[pgAdmin](#_u8c0k8on8fdg) 7

[SQL](#_1o1nnovzvp1j) 7

[Terraform](#_4jyw1x6aflvt) 7

[GCP](#_peujx2qr4r66) 7

[**Virtual Machine**](#_kefvw5280wff) **9**

[**Week 2**](#_j6086ct72gfi) **10**

[Airflow - I’ve got this error: google.auth.exceptions.DefaultCredentialsError: File /.google/credentials/google\_credentials.json was not found.](#_u4rc5ws2q5ad) 10

[Installing python libraries in airflow](#_o6kh4n6kp0o2) 11

[**Week 3**](#_r3ubooami7lx) **11**

[**Week 4**](#_u3mfeppzfpyy) **12**

[I am having problems with columns datatype while running DBT/BigQuery](#_1iixhn1fzx6c) 12

[When running your first dbt model, if it fails with an error: 404 Not found: Dataset was not found in location US](#_s3k9o4z6b9bv) 12

[When executing dbt run after fact\_trips.sql has been created, the task failed with error:](#_j5o2f488tyjd) 13

[Why my Fact\_trips only contains a few days of data?](#_gqeckghjh0tu) 13

[BigQuery returns an error when I try to run the dm\_monthly\_zone\_revenue.sql model.](#_p6dcleiuuri) 13

[Error thrown by format\_to\_parquet\_task when converting fhv\_tripdata\_2020-01.csv using Airflow](#_r14r9met8o2j) 13

[Why do we need the Staging dataset?](#_ssdqky8b1360) 14

[My main branch on dbt suddenly changed to read-only... How do I change it back while working on DBT, the branch of the project](#_6kiwa3l61b4w) 14

[**Week 5**](#_46wg59yfccku) **15**

[RuntimeError: Java gateway process exited before sending its port number](#_e0nv1vejsx6w) 15

[Module Not Found Error in Jupyter Notebook .](#_3bt81nhjdjqo) 15

[ModuleNotFoundError: No module named 'py4j'` while executing `import pyspark](#_tnp7y7zaito1) 16

[Exception: Jupyter command `jupyter-notebook` not found.](#_vqkuo1kaimmj) 16

[Error java.io.FileNotFoundException](#_4tfmz0el5djo) 17

[Which type of SQL is used in Spark? Postgres? MySQL? SQL Server?](#_rvyjbusobc5x) 17

[The spark viewer on localhost:4040 was not showing the current run](#_4q3iclcanyjy) 18

[java.lang.NoSuchMethodError: sun.nio.ch.DirectBuffer.cleaner()Lsun/misc/Cleaner Error during repartition call (conda pyspark installation)](#_tlerxp55nkrh) 18

[**Week 6**](#_a6r752g4lijr) **18**

[Could not start docker image “control-center” from the docker-compose.yaml file.](#_oppe61qf4o7l) 18

[Module “kafka” not found when trying to run producer.py](#_aquns8ex863j) 19

[Error importing cimpl dll when running avro examples](#_86oebqa7c90) 19

[ModuleNotFoundError: No module named 'avro'](#_hnizzwrvxtwc) 20

[WGet error /workaround](#_cs74r1fos11b) 20

[**Project**](#_ou8k2ojlt9qg) **20**

[Does anyone know nice and relatively large datasets?](#_1hspwy1mtnp1) 20

[Spark fails when reading from BigQuery and using `.show()` on `SELECT` quer](#_crvxqjae41q)y 20

# 

# 

# Week 1

## Docker run for windows

You may have such an error:

| $ docker run -it ubuntu bash  the input device is not a TTY. If you are using mintty, try prefixing the command with 'winpty'' |
| --- |

Solution:

Use **winpty** before docker command ([source](https://stackoverflow.com/a/49965690))

| $ winpty docker run -it ubuntu bash  You also can make an alias:  echo "alias docker='winpty docker'" >> ~/.bashrc OR  echo "alias docker='winpty docker'" >> ~/.bash\_profile |
| --- |

## You already have Postgres

If you already have postgres that you installed without Docker, use a different port, e.g. 5431:

docker run -it \

-e POSTGRES\_USER="root" \

-e POSTGRES\_PASSWORD="root" \

-e POSTGRES\_DB="ny\_taxi" \

-v $(pwd)/ny\_taxi\_postgres\_data:/var/lib/postgresql/data \

-p 5431:5432 \

postgres:13

And use it when connecting with pgcli:

pgcli -h localhost -p 5431 -U root -d ny\_taxi

## **Running Postgres with Docker**

Permission issue on /var/lib/postgresql/data

| $ docker run -it\  -e POSTGRES\_USER="root" \  -e POSTGRES\_PASSWORD="admin" \  -e POSTGRES\_DB="ny\_taxi" \  -v "/mnt/path/to/ny\_taxi\_postgres\_data":"/var/lib/postgresql/data" \  -p 5432:5432 \  postgres:13  # Output  The files belonging to this database system will be owned by user "postgres".  This user must also own the server process.  The database cluster will be initialized with locale "en\_US.utf8".  The default database encoding has accordingly been set to "UTF8".  The default text search configuration will be set to "english".  Data page checksums are disabled.  fixing permissions on existing directory /var/lib/postgresql/data ... initdb: error: could not change permissions of directory "/var/lib/postgresql/data": Operation not permitted |
| --- |

To work around this issue, we can create a local docker volume and map it to postgres data directory /var/lib/postgresql/data which has permission issue.

| $ docker volume create --name dtc\_postgres\_volume\_local -d local  $ docker run -it\  -e POSTGRES\_USER="root" \  -e POSTGRES\_PASSWORD="admin" \  -e POSTGRES\_DB="ny\_taxi" \  -v dtc\_postgres\_volume\_local:/var/lib/postgresql/data \  -p 5432:5432 \  postgres:13 |
| --- |

## Mounting volumes with Docker on Windows

Mapping volumes on Windows could be tricky. The way it was done in the course video doesn’t work for everyone.

If you have spaces in the path, move your data to some folder without spaces. E.g. if your code is in “C:/Users/Alexey Grigorev/git/…”, move it to “C:/git/…”

Try replacing the “-v” part with one of the following options:

* -v /c:/some/path/ny\_taxi\_postgres\_data:/var/lib/postgresql/data
* -v //c:/some/path/ny\_taxi\_postgres\_data:/var/lib/postgresql/data
* -v /c/some/path/ny\_taxi\_postgres\_data:/var/lib/postgresql/data
* -v //c/some/path/ny\_taxi\_postgres\_data:/var/lib/postgresql/data

Try adding quotes:

* -v "/c:/some/path/ny\_taxi\_postgres\_data:/var/lib/postgresql/data"
* -v "//c:/some/path/ny\_taxi\_postgres\_data:/var/lib/postgresql/data"
* -v "/c/some/path/ny\_taxi\_postgres\_data:/var/lib/postgresql/data"
* -v "//c/some/path/ny\_taxi\_postgres\_data:/var/lib/postgresql/data"

Important: note how the quotes are placed.

If none of these options work, you can use a volume name instead of the path:

* -v ny\_taxi\_postgres\_data:/var/lib/postgresql/data

## **Error with Pgcli**

I get this error

pgcli -h localhost -p 5432 -U root -d ny\_taxi

Traceback (most recent call last):

File "/opt/anaconda3/bin/pgcli", line 8, in <module>

sys.exit(cli())

File "/opt/anaconda3/lib/python3.9/site-packages/click/core.py", line 1128, in \_\_call\_\_

return self.main(\*args, \*\*kwargs)

File "/opt/anaconda3/lib/python3.9/sitYe-packages/click/core.py", line 1053, in main

rv = self.invoke(ctx)

File "/opt/anaconda3/lib/python3.9/site-packages/click/core.py", line 1395, in invoke

return ctx.invoke(self.callback, \*\*ctx.params)

File "/opt/anaconda3/lib/python3.9/site-packages/click/core.py", line 754, in invoke

return \_\_callback(\*args, \*\*kwargs)

File "/opt/anaconda3/lib/python3.9/site-packages/pgcli/main.py", line 880, in cli

os.makedirs(config\_dir)

File "/opt/anaconda3/lib/python3.9/os.py", line 225, in makedirs

mkdir(name, mode)

PermissionError: [Errno 13] Permission denied: '/Users/vray/.config/pgcli'

Make sure you install pgcli without sudo. The recommended approach is to use conda/anaconda to make sure your system python is not affected.

**Pgcli- Password error**

If your Bash prompt is stuck on the password command for postgres- Try using windows terminal or terminal in VS code.

Possible error while connecting to postgres:

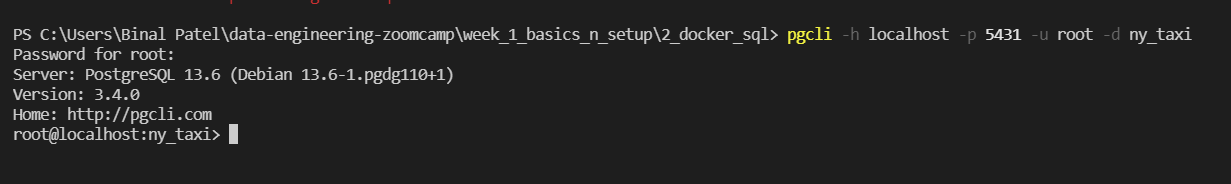
*FATAL: password authentication failed for user "root"*



Try changing the port on the host to 5431

docker run -it -e POSTGRES\_USER="root" -e POSTGRES\_PASSWORD="root" -e POSTGRES\_DB="ny\_taxi" -v d:/Github/Data-Engineering-Zoomcamp/01\_basics\_n\_setup/2\_docker\_sql/ny\_taxi\_postgres\_data:/var/lib/postgresql/data -p 5431:5432 postgres:13

This will connect you to postgres.



## **Problems while connecting to Postgres in Python**

~\anaconda3\lib\site-packages\psycopg2\\_\_init\_\_.py in connect(dsn, connection\_factory, cursor\_factory, \*\*kwargs)

120

121 dsn = \_ext.make\_dsn(dsn, \*\*kwargs)

--> 122 conn = \_connect(dsn, connection\_factory=connection\_factory, \*\*kwasync)

123 if cursor\_factory is not None:

124 conn.cursor\_factory = cursor\_factory

OperationalError: (psycopg2.OperationalError) connection to server at "localhost" (::1), port 5432 failed: FATAL: database "ny\_taxi" does not exist

Make sure postgres is running. You can check that by running `docker ps`

Error ingesting data using python script

curl: (6) Could not resolve host: output.csv

for mac users - os.system(f"curl {url} —-output {csv\_name}")

## **Docker-compose - Error translating host name to address**

Couldn’t translate host name to address

## 

## 

Make sure postgres database is running.  
  
​​Use the command to start containers in detached mode: docker-compose up -d

(data-engineering-zoomcamp) hw % docker compose up -d

[+] Running 2/2

⠿ Container pg-admin Started 0.6s

⠿ Container pg-database Started

To view the containers use: docker ps.

(data-engineering-zoomcamp) hw % docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

faf05090972e postgres:13 "docker-entrypoint.s…" 39 seconds ago Up 37 seconds 0.0.0.0:5432->5432/tcp pg-database

6344dcecd58f dpage/pgadmin4 "/entrypoint.sh" 39 seconds ago Up 37 seconds 443/tcp, 0.0.0.0:8080->80/tcp pg-admin  
hw

To view logs for a container: docker logs <containerid>

(data-engineering-zoomcamp) hw % docker logs faf05090972e

PostgreSQL Database directory appears to contain a database; Skipping initialization

2022-01-25 05:58:45.948 UTC [1] LOG: starting PostgreSQL 13.5 (Debian 13.5-1.pgdg110+1) on aarch64-unknown-linux-gnu, compiled by gcc (Debian 10.2.1-6) 10.2.1 20210110, 64-bit

2022-01-25 05:58:45.948 UTC [1] LOG: listening on IPv4 address "0.0.0.0", port 5432

2022-01-25 05:58:45.948 UTC [1] LOG: listening on IPv6 address "::", port 5432

2022-01-25 05:58:45.954 UTC [1] LOG: listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"

2022-01-25 05:58:45.984 UTC [28] LOG: database system was interrupted; last known up at 2022-01-24 17:48:35 UTC

2022-01-25 05:58:48.581 UTC [28] LOG: database system was not properly shut down; automatic recovery in progress

2022-01-25 05:58:48.602 UTC [28] LOG: redo starts at 0/872A5910

2022-01-25 05:59:33.726 UTC [28] LOG: invalid record length at 0/98A3C160: wanted 24, got 0

2022-01-25 05:59:33.726 UTC [28] LOG: redo done at 0/98A3C128

2022-01-25 05:59:48.051 UTC [1] LOG: database system is ready to accept connections

If docker ps doesn’t show pgdatabase running,  
  
Run docker ps -a

This should show all containers, either running or stopped.

Get the container id for pgdatabase-1, and run docker log <container\_id>

## pgAdmin

pgAdmin has a new version. Create server dialog may not appear. Try using register-> server instead.

## SQL

## Terraform

## GCP

**Trying to initialize gcloud sdk:**

It asked me to create a project. This should be done from the cloud console. So maybe we don’t need this FAQ.

WARNING: Project creation failed: HttpError accessing <https://cloudresourcemanager.googleapis.com/v1/projects?alt=json>: response: <{'vary': 'Origin, X-Origin, Referer', 'content-type': 'application/json; charset=UTF-8', 'content-encoding': 'gzip', 'date': 'Mon, 24 Jan 2022 19:29:12 GMT', 'server': 'ESF', 'cache-control': 'private', 'x-xss-protection': '0', 'x-frame-options': 'SAMEORIGIN', 'x-content-type-options': 'nosniff', 'server-timing': 'gfet4t7; dur=189', 'alt-svc': 'h3=":443"; ma=2592000,h3-29=":443"; ma=2592000,h3-Q050=":443"; ma=2592000,h3-Q046=":443"; ma=2592000,h3-Q043=":443"; ma=2592000,quic=":443"; ma=2592000; v="46,43"', 'transfer-encoding': 'chunked', 'status': 409}>, content <{

"error": {

"code": 409,

"message": "Requested entity already exists",

"status": "ALREADY\_EXISTS"

}

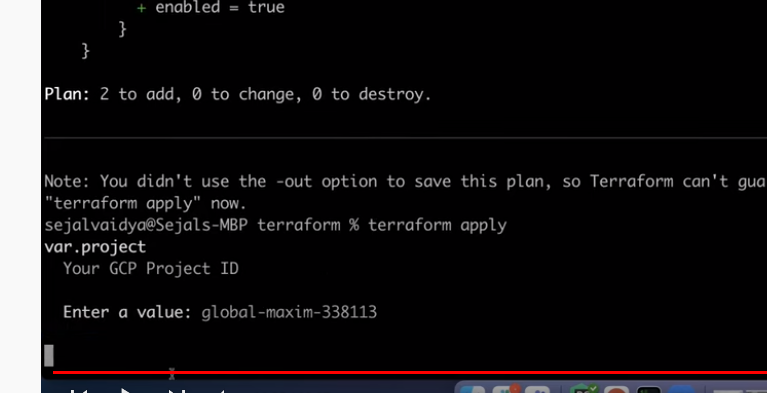
}

From Stackoverflow: <https://stackoverflow.com/questions/52561383/gcloud-cli-cannot-create-project-the-project-id-you-specified-is-already-in-us?rq=1>

Project IDs are unique across all projects. That means if *any* user *ever* had a project with that ID, you cannot use it. testproject is pretty common, so it's not surprising it's already taken.

**Error: googleapi:**

If you receive the error: “Error 403: The project to be billed is associated with an absent billing account., accountDisabled” It is most likely because you did not enter **YOUR** project ID. The snip below is from video 1.3.2.



The value you enter here will be unique to each student. You can find this value on your GCP Dashboard when you login.

# Virtual Machine

I am trying to create a directory but it won't let me do it

User1@DESKTOP-PD6UM8A MINGW64 /

$ mkdir .ssh

mkdir: cannot create directory ‘.ssh’: Permission denied

**Answers**

You should do it in your home directory. Should be your home (~)

Local. But it seems you're trying to do it in the root folder (/). Should be your home (~)

Video 1.4.1

<https://www.youtube.com/watch?v=ae-CV2KfoN0&list=PL3MmuxUbc_hJed7dXYoJw8DoCuVHhGEQb>

**Error while saving the file in VM via VS Code**

**Failed to save '<file>': Unable to write file 'vscode-remote://ssh-remote+de-zoomcamp/home/<user>/data\_engineering\_course/week\_2/airflow/dags/<file>' (NoPermissions (FileSystemError): Error: EACCES: permission denied, open '/home/<user>/data\_engineering\_course/week\_2/airflow/dags/<file>')**

You need to change the owner of the files you are trying to edit via VS Code. You can run the following command to change the ownership.

sudo chown -R <user> <path to your directory>

**VM connection request timeout**

Question: I connected to my VM perfectly fine last week (ssh) but when I tried again this week, the connection request keeps timing out.

Answer: Start your VM. Once the VM is running, copy its External IP and paste that into your config file within the ~/.ssh folder.

cd ~/.ssh

code config ← this opens the config file in VSCode

# **Week 2**

## **Airflow -** I’ve got this error: **google.auth.exceptions.DefaultCredentialsError: File /.google/credentials/google\_credentials.json was not found.**

Change the path of the *google\_credentials* mounting in the docker-compose file to an absolute one. For example in Ubuntu,

Instead of this: **/.google/credentials/:/.google/credentials:ro**

Use this: **/home/<username>/.google/credentials/:/.google/credentials**

I got the error below when I was running download\_dataset\_task:

\*\*\* Log file does not exist: /opt/airflow/logs/taxi\_zone\_dag/download\_dataset\_task/2022-02-02T09:39:17.124318+00:00/6.log

\*\*\* Fetching from: http://:8793/log/taxi\_zone\_dag/download\_dataset\_task/2022-02-02T09:39:17.124318+00:00/6.log

\*\*\* Failed to fetch log file from worker. Request URL missing either an 'http://' or 'https://' protocol.

I resolved it by running:

docker-compose down -v --rmi all --remove-orphans

After that, remove the following line from my codes:

From datetime import time

And then, restart docker-compose again:

docker-compose up

## Installing python libraries in airflow

Under this section of the docker-compose.yaml file, find the

\_PIP\_ADDITIONAL\_REQUIREMENTS:

build:

context: .

dockerfile: ./Dockerfile

environment:

\_PIP\_ADDITIONAL\_REQUIREMENTS:${\_PIP\_ADDITIONAL\_REQUIREMENTS:-}

E.g

\_PIP\_ADDITIONAL\_REQUIREMENTS:${\_PIP\_ADDITIONAL\_REQUIREMENTS:- pyspark}

See here from documentation: <https://airflow.apache.org/docs/docker-stack/entrypoint.html#installing-additional-requirements>

## Airflow won’t update the DAG / It keeps returning errors even though I supposedly installed additional Python libraries

Make sure that you update your Airflow image to a more recent one. Inside your Dockerfile, modify the *FROM apache/airflow:2.2.3* to any of the more recent images available in the official Airflow Docker repository, available at <https://hub.docker.com/r/apache/airflow/tags>

## Airflow web login issue on docker:

I was unable to log onto my linux instance of airflow with the web password until I modified the config file in docker\_compose.yaml from:

\_AIRFLOW\_WWW\_USER\_CREATE: 'true'

\_AIRFLOW\_WWW\_USER\_USERNAME: ${\_AIRFLOW\_WWW\_USER\_USERNAME:-airflow}

\_AIRFLOW\_WWW\_USER\_PASSWORD: ${\_AIRFLOW\_WWW\_USER\_PASSWORD:-airflow}

to :

\_AIRFLOW\_WWW\_USER\_CREATE=True

\_AIRFLOW\_WWW\_USER\_USERNAME=airflow

\_AIRFLOW\_WWW\_USER\_PASSWORD=airflow

# Week 3

Q: Docker-compose takes infinitely long to install zip unzip packages for linux, which are required to unpack datasets

A:

1 solution) Add -Y flag, so that apt-get automatically agrees to install additional packages

2) Use python ZipFile package, which is included in all modern python distributions

# **Week 4**

## **I am having problems with columns datatype while running DBT/BigQuery**

**R:** If you don’t define the column format while converting from csv to parquet Python will “choose” based on the first rows.

**Solution:** Defined the schema while running web\_to\_gcp.py pipeline.

Sebastian adapted the script:

<https://github.com/sebastian2296/data-engineering-zoomcamp/blob/main/week_4_analytics_engineering/web_to_gcs.py>

**Same ERROR - When running dbt run for fact\_trips.sql, the task failed with error:**

“Parquet column 'ehail\_fee' has type DOUBLE which does not match the target cpp\_type INT64”

**Reason:** Parquet files has their own schema. Some parquet files for green data has records with decimals in ehail\_fee column.

There are some possible fixes:

* Drop ehail\_feel column since it is not really used. For instance when creating a partitioned table from the external table in BigQuery

SELECT \* EXCEPT (ehail\_fee) FROM…

* Modify stg\_green\_tripdata.sql model using this line cast(0 as numeric) as ehail\_fee.
* Modify Airflow dag to make the conversion and avoid the error.

pv.read\_csv(src\_file, convert\_options=pv.ConvertOptions(column\_types = {'ehail\_fee': 'float64'}))

## **When running your first dbt model, if it fails with an error: 404 Not found: Dataset was not found in location US**

**R:** Go to BigQuery, and check the location of BOTH

1. The source dataset (trips\_data\_all), and
2. The schema you’re trying to write to (name should be dbt\_<first initial><last name>)

Likely, your source data will be in your region, but the write location will be a multi-regional location (US in this example). Delete these datasets, and recreate them with your specified region and the correct naming format.

Additionally please see this post of Sandy:

<https://learningdataengineering540969211.wordpress.com/dbt-cloud-and-bigquery-an-effort-to-try-and-resolve-location-issues/>

## **When executing dbt run after fact\_trips.sql has been created, the task failed with error:**

**R:** “Access Denied: BigQuery BigQuery: Permission denied while globbing file pattern.”

Fixed by adding the Viewer role to the service account in use in BigQuery.

## **Why my Fact\_trips only contains a few days of data?**

Make sure you use dbt run –var ‘is\_test\_tun: false’ or dbt build –var ‘is\_test\_tun: false’

## **BigQuery returns an error when I try to run the dm\_monthly\_zone\_revenue.sql model.**

R: After the second SELECT, change this line:

date\_trunc('month', pickup\_datetime) as revenue\_month,

To this line:

date\_trunc(pickup\_datetime, month) as revenue\_month,

Make sure that “month” isn’t surrounded by quotes!

## **Error thrown by format\_to\_parquet\_task when converting fhv\_tripdata\_2020-01.csv using Airflow**

R: This conversion is needed for the question 3 of homework, in order to process files for fhv data. The error is:

pyarrow.lib.ArrowInvalid: CSV parse error: Expected 7 columns, got 1: B02765

Cause: Some random line breaks in this particular file.

Fixed by opening a bash in the container executing the dag and manually running the following command that deletes all \n not preceded by \r.

perl -i -pe 's/(?<!\r)\n/\1/g' fhv\_tripdata\_2020-01.csv

After that, clear the failed task in Airflow to force re-execution.

## **Why do we need the Staging dataset?**

Vic created three different datasets in the videos.. dbt\_<name> was used for development and you used a **production** dataset for the production environment. What was the use for the **staging** dataset?

**R:** Staging, as the name suggests, is like an intermediate between the raw datasets and the fact and dim tables, which are the finished product, so to speak. You'll notice that the dataset in staging are materialised as views and not tables.

Vic didn't use it for the project, you just need to create production and dbt\_name + trips\_data\_all that you had already.

## My main branch on dbt suddenly changed to read-only... How do I change it back while working on DBT, the branch of the project

**R:** Since you are on the main branch, it doesn't allow you to change. Just create a new branch to keep going

## DBT Docs Served but Not Accessible via Browser

Try removing the “network: host” line in docker-compose.

# **Week 5**

## **RuntimeError: Java gateway process exited before sending its port number**

After installing all including pyspark (and it is successfully imported), but then running this script on the jupyter notebook

import pyspark

from pyspark.sql import SparkSession

spark = SparkSession.builder \

.master("local[\*]") \

.appName('test') \

.getOrCreate()

df = spark.read \

.option("header", "true") \

.csv('taxi+\_zone\_lookup.csv')

df.show()

it gives the error:

RuntimeError: Java gateway process exited before sending its port number

The solution (for me) was:

pip install findspark on the command line and then add import findspark

findspark.init() to the top of the script.

Another possible solution is:

Check that pyspark is pointing to the correct location. Run pyspark.\_\_file\_\_. It should be list /home/<your user name>/spark/spark-3.0.3-bin-hadoop3.2/python/pyspark/\_\_init\_\_.py if you followed the videos. If it is pointing to your python site-packages remove the pyspark directory there and check that you have added the correct exports to you .bashrc file and that there are not any other exports which might supersede the ones provided in the course content.

## **Module Not Found Error in Jupyter Notebook .**

Even after installing pyspark correctly on linux machine (VM ) as per course instructions, faced a module not found error in jupyter notebook .

The solution which worked for me(use following in jupyter notebook) :

!pip install findspark

Import findspark

findspark.init()

Thereafter , import pyspark and create spark context as usual

None of the solutions above worked for me till I ran !pip3 install pyspark instead !pip install pyspark.

## **ModuleNotFoundError: No module named 'py4j'` while executing `import pyspark`**

Make sure that the version under `${SPARK\_HOME}/python/lib/` matches the filename of py4j or you will encounter `ModuleNotFoundError: No module named 'py4j'` while executing `import pyspark`. For instance, if the file under `${SPARK\_HOME}/python/lib/` was `py4j-0.10.9.3-src.zip`.

Then the export PYTHONPATH statement above should be changed to `export PYTHONPATH="${SPARK\_HOME}/python/lib/py4j-0.10.9.3-src.zip:$PYTHONPATH"` appropriately.

## **Exception: Jupyter command `jupyter-notebook` not found.**

Even after we have exported our paths correctly you may find that even though Jupyter is installed you might not have Jupyter Notebook for one reason or another. Full instructions are found [here](https://learningdataengineering540969211.wordpress.com/2022/02/24/week-5-de-zoomcamp-5-2-1-installing-spark-on-linux/) (for my walkthrough) or [here](https://speedysense.com/install-jupyter-notebook-on-ubuntu-20-04/) (where I got the original instructions from) but are included below. These instructions include setting up a virtual environment (handy if you are on your own machine doing this and not a VM):

Full steps:

1. Update and upgrade packages:
   1. sudo apt update && sudo apt -y upgrade
2. Install Python:
   1. sudo apt install python3-pip python3-dev
3. Install Python virtualenv:
   1. sudo -H pip3 install --upgrade pip
   2. sudo -H pip3 install virtualenv
4. Create a Python Virtual Environment:
   1. mkdir notebook
   2. cd notebook
   3. virtualenv jupyterenv
   4. source jupyterenv/bin/activate
5. Install Jupyter Notebook:
   1. pip install jupyter
6. Run Jupyter Notebook:
   1. jupyter notebook

## **Error java.io.FileNotFoundException**

Code executed:

df = spark.read.parquet(pq\_path)

… some operations on df …

df.write.parquet(pq\_path, mode="overwrite")

java.io.FileNotFoundException: File file:/home/xxx/code/data/pq/fhvhv/2021/02/part-00021-523f9ad5-14af-4332-9434-bdcb0831f2b7-c000.snappy.parquet does not exist

The problem is that Sparks performs lazy transformations, so the actual action that trigger the job is df.write, which does delete the parquet files that is trying to read (mode=”overwrite”)

Solution: Write to a different directory

df.write.parquet(pq\_path\_temp, mode="overwrite")

## **Which type of SQL is used in Spark? Postgres? MySQL? SQL Server?**

Actually Spark SQL is one independent “type” of SQL - Spark SQL.

The several SQL providers are very similar:

***SELECT [attributes]***

***FROM [table]***

***WHERE [filter]***

***GROUP BY [grouping attributes]***

***HAVING [filtering the groups]***

***ORDER BY [attribute to order]***

***(INNER/FULL/LEFT/RIGHT) JOIN [table2]***

***ON [attributes table joining table2] (...)***

What differs the most between several SQL providers are built-in functions.

For Built-in Spark SQL function check this link:  
<https://spark.apache.org/docs/latest/api/sql/index.html>

Extra information on SPARK SQL :

<https://databricks.com/glossary/what-is-spark-sql#:~:text=Spark%20SQL%20is%20a%20Spark,on%20existing%20deployments%20and%20data>.

## **The spark viewer on localhost:4040 was not showing the current run**

Solution: I had two notebooks running, and the one I wanted to look at had opened a port on localhost:4041.

If port is in use, then Spark uses next. It can be even 4044. You can run   
spark.sparkContext.uiWebUrl

and result will be some like  
'http://172.19.10.61:4041'

## **java.lang.NoSuchMethodError: sun.nio.ch.DirectBuffer.cleaner()Lsun/misc/Cleaner Error during repartition call (conda pyspark installation)**

Solution: replace Java Developer Kit 11 with Java Developer Kit 8.

**RuntimeError: Java gateway process exited before sending its port number**

Shows java\_home is not set on the notebook log

<https://sparkbyexamples.com/pyspark/pyspark-exception-java-gateway-process-exited-before-sending-the-driver-its-port-number/>

# **Week 6**

## **Could not start docker image “control-center” from the docker-compose.yaml file.**

On Mac OSX 12.2.1 (Monterey) I could not start the kafka control center. I opened Docker Desktop and saw docker images still running from week 4, which I did not see when I typed “docker ps.” I deleted them in docker desktop and then had no problem starting up the kafka environment.

## **Module “kafka” not found when trying to run producer.py**

Solution from Alexei: create a virtual environment and run requirements.txt and the python files in that environment.

To create a virtual env and install packages (run only once)

python -m venv env

source env/bin/activate

pip install -r requirements.txt

To activate it (you'll need to run it every time you need the virtual env):

source env/bin/activate

To deactivate it:

deactivate

This works on MacOS, Linux and Windows - but for Windows the path is slightly different (it's env/Scripts/activate)

Also the virtual environment should be created only to run the python file. Docker images should first all be up and running.

## **Error importing cimpl dll when running avro examples**

ImportError: DLL load failed while importing cimpl: The specified module could not be found

... you may have to load librdkafka-5d2e2910.dll in the code. Add this before importing avro:

from ctypes import CDLL

CDLL("C:\\Users\\YOUR\_USER\_NAME\\anaconda3\\envs\\dtcde\\Lib\\site-packages\\confluent\_kafka.libs\librdkafka-5d2e2910.dll")

It seems that the error may occur depending on the OS and python version installed.

ALTERNATIVE:

ImportError: DLL load failed while importing cimpl

SOLUTION: $env:CONDA\_DLL\_SEARCH\_MODIFICATION\_ENABLE=1 in Powershell.

You need to set this DLL manually in Conda Env.

Source: <https://githubhot.com/repo/confluentinc/confluent-kafka-python/issues/1186?page=2>

## **ModuleNotFoundError: No module named 'avro'**

SOLUTION: pip install confluent-kafka[avro].

For some reason, Conda also doesn't include this when installing confluent-kafka via pip.

More sources on Anaconda and confluent-kafka issues:

* <https://github.com/confluentinc/confluent-kafka-python/issues/590>
* <https://github.com/confluentinc/confluent-kafka-python/issues/1221>
* <https://stackoverflow.com/questions/69085157/cannot-import-producer-from-confluent-kafka>

**Error while running python3 stream.py worker**

If you get an error while running the command

“python3 stream.py worker”

Run pip uninstall kafka-python

Then run

Pip install kafka-python==1.4.6

## WGet error /workaround

Installing Hadoop Binary Files and Spark using Wget:

For anyone having issues installing hadoop binaries on windows and facing a wget gets a WGET is not recognized as an internal or external command

you can use: python -m wget instead of wget

**Got Error message Negsignal:SIGKILL while converting dta files to parquet format**

Got this error because the docker container memory was exhausted. The dta file was upto 800MB but my docker container does not have enough memory to handle that.

Solution was to load the file in chunks with Pandas, then create multiple parquet files for each dat file I was processing. This worked smoothly and the issue was resolved.

# Project

## Does anyone know nice and relatively large datasets?

See a list of datasets here: <https://github.com/DataTalksClub/data-engineering-zoomcamp/blob/main/week_7_project/datasets.md>

## Spark fails when reading from BigQuery and using `.show()` on `SELECT` queries

I got it working using `gcs-connector-hadoop3-2.2.5-shaded.jar` and Spark 3.1

I also added the google\_credentials.json and .p12 to auth with gcs. These files are downloadable from GCP Service account.

To create the SparkSession:

spark = SparkSession.builder.master('local[\*]') \

.appName('spark-read-from-bigquery') \

.config('BigQueryProjectId','razor-project-xxxxxxx) \

.config('BigQueryDatasetLocation','de\_final\_data') \

.config('parentProject','razor-project-xxxxxxx) \

.config("google.cloud.auth.service.account.enable", "true") \

.config("credentialsFile", "google\_credentials.json") \

.config("GcpJsonKeyFile", "google\_credentials.json") \

.config("spark.driver.memory", "4g") \

.config("spark.executor.memory", "2g") \

.config("spark.memory.offHeap.enabled",True) \

.config("spark.memory.offHeap.size","5g") \

.config('google.cloud.auth.service.account.json.keyfile', "google\_credentials.json") \

.config("fs.gs.project.id", "razor-project-xxxxxxx") \

.config("fs.gs.impl", "com.google.cloud.hadoop.fs.gcs.GoogleHadoopFileSystem") \

.config("fs.AbstractFileSystem.gs.impl", "com.google.cloud.hadoop.fs.gcs.GoogleHadoopFS") \

.getOrCreate()

## Spark BigQuery connector Automatic configuration

While creating a SparkSession using the config **spark.jars.packages** as *com.google.cloud.spark:spark-bigquery-with-dependencies\_2.12:0.23.2*

spark = SparkSession.builder.master('local').appName('bq').config("spark.jars.packages", "com.google.cloud.spark:spark-bigquery-with-dependencies\_2.12:0.23.2").getOrCreate()

automatically downloads the required dependency jars and configures the connector, removing the need to manage this dependency. More details available [here](https://github.com/GoogleCloudDataproc/spark-bigquery-connector)

## Spark Cloud Storage connector

Link to Slack Thread <https://datatalks-club.slack.com/archives/C01FABYF2RG/p1646013709648279?thread_ts=1646008578.136059&cid=C01FABYF2RG>

**has anyone figured out how to read from GCP data lake instead of downloading all the taxi data again?**

There’s a few extra steps to go into reading from GCS with PySpark

1.) IMPORTANT: Download the Cloud Storage connector for Hadoop here:<https://cloud.google.com/dataproc/docs/concepts/connectors/cloud-storage#clusters>

As the name implies, this .jar file is what essentially connects PySpark with your GCS

2.) Move the .jar file to your Spark file directory. I installed Spark using homebrew on my MacOS machine and I had to create a /jars directory under "/opt/homebrew/Cellar/apache-spark/3.2.1/ (where my spark dir is located)

3.) In your Python script, there are a few extra classes you’ll have to import:

import pyspark

from pyspark.sql import SparkSession

from pyspark.conf import SparkConf

from pyspark.context import SparkContext

4.) You must set up your configurations before building your SparkSession. Here’s my code snippet:

conf = SparkConf() \

.setMaster('local[\*]') \

.setAppName('test') \

.set("spark.jars", "/opt/homebrew/Cellar/apache-spark/3.2.1/jars/gcs-connector-hadoop3-latest.jar") \

.set("spark.hadoop.google.cloud.auth.service.account.enable", "true") \

.set("spark.hadoop.google.cloud.auth.service.account.json.keyfile", "path/to/google\_credentials.json")

sc = SparkContext(conf=conf)

sc.\_jsc.hadoopConfiguration().set("fs.AbstractFileSystem.gs.impl", "com.google.cloud.hadoop.fs.gcs.GoogleHadoopFS")

sc.\_jsc.hadoopConfiguration().set("fs.gs.impl", "com.google.cloud.hadoop.fs.gcs.GoogleHadoopFileSystem")

sc.\_jsc.hadoopConfiguration().set("fs.gs.auth.service.account.json.keyfile", "path/to/google\_credentials.json")

sc.\_jsc.hadoopConfiguration().set("fs.gs.auth.service.account.enable", "true")

5.) Once you run that, build your SparkSession with the new parameters we’d just instantiated in the previous step:

spark = SparkSession.builder \

.config(conf=sc.getConf()) \

.getOrCreate()

6.) Finally, you’re able to read your files straight from GCS!

df\_green = spark.read.parquet("gs://{BUCKET}/green/202\*/")

**Spark Streaming - How do I read from multiple topics in the same Spark Session**

Initiate a Spark Session

spark = (SparkSession

.builder

.appName(app\_name)

.master(master=master)

.getOrCreate())

spark.streams.resetTerminated()

.

.

.

query1 = spark

.readStream

…

…

.load()

query2 = spark

.readStream

…

…

.load()

query3 = spark

.readStream

…

…

.load()

query1.start()

query2.start()

query3.start()

spark.streams.awaitAnyTermination() #waits for any one of the query to receive kill signal or error failure. This is asynchronous

# On the contrary query3.start().awaitTermination() is a blocking call. Works well when we are reading only from one topic.

## Orchestrating dbt with Airflow

The trial dbt account provides access to dbt API. Job will still be needed to be added manually. Airflow will run the job using a python operator calling the API. You will need to provide api key, job id, etc. (be careful not committing it to highub).

Detailed explanation here: <https://docs.getdbt.com/blog/dbt-airflow-spiritual-alignment>

Source code example here: <https://github.com/sungchun12/airflow-toolkit/blob/95d40ac76122de337e1b1cdc8eed35ba1c3051ed/dags/examples/dbt_cloud_example.py>

## Orchestrating DataProc with Airflow

<https://airflow.apache.org/docs/apache-airflow-providers-google/stable/_api/airflow/providers/google/cloud/operators/dataproc/index.html>

<https://airflow.apache.org/docs/apache-airflow-providers-google/stable/_modules/airflow/providers/google/cloud/operators/dataproc.html>

Give the following roles to you service account:

* DataProc Administrator
* Service Account User (explanation [here](https://stackoverflow.com/questions/63941429/user-not-authorized-to-act-as-service-account-when-using-workload-identity))

Use DataprocSubmitPySparkJobOperator, DataprocDeleteClusterOperator and DataprocCreateClusterOperator.

When using DataprocSubmitPySparkJobOperator, do not forget to add:

dataproc\_jars = ["gs://spark-lib/bigquery/spark-bigquery-with-dependencies\_2.12-0.24.0.jar"]

Because DataProc does not have already the BigQuery Connector.