Big Data Coursework - Questions

Data Processing and Machine Learning in the Cloud

This is the **INM432 Big Data coursework 2025**. This coursework contains extended elements of **theory** and **practice**, mainly around parallelisation of tasks withSpark and a bit about parallel training using TensorFlow.

Code and Report

Your tasks parallelization of tasks in PySpark, extension, evaluation, and theoretical reflection. Please complete and submit the **coding tasks** in a copy of **this notebook**. Write your code in the **indicated cells** and **include** the **output** in the submitted notebook.
Make sure that **your code contains comments** on its **stucture** and explanations of its **purpose**.

Provide also a **report** with the **textual answers in a separate document**.
Include **screenshots** from the Google Cloud web interface (don't use the SCREENSHOT function that Google provides, but take a picture of the graphs you see for the VMs) and result tables, as well as written text about the analysis.

Submission

Download and submit **your version of this notebook** as an **.ipynb** file and also submit a **shareable link** to your notebook on Colab in your report (created with the Colab 'Share' function) (**and don't change the online version after submission**).

Further, provide your **report as a PDF document**. **State the number of words** in the document at the end. The report should **not have more than 2000 words**.

Please also submit **a PDF of your Jupyter notebook**.

Introduction and Description

This coursework focuses on parallelisation and scalability in the cloud with Spark and TesorFlow/Keras. We start with code based on **lessons 3 and 4** of the *Fast and Lean Data Science* course by Martin Gorner. The course is based on Tensorflow for data processing and MachineLearning. Tensorflow's data processing approach is somewhat similar to that of Spark, but you don't need to study Tensorflow, just make sure you understand the high-level structure.
What we will do here is **parallelising pre-processing**, and **measuring** performance, and we will perform **evaluation** and **analysis** on the cloud performance, as well as **theoretical discussion**.

This coursework contains **3 sections**.

Section 0

This section just contains some necessary code for setting up the environment. It has no tasks for you (but do read the code and comments).

Section 1

Section 1 is about preprocessing a set of image files. We will work with a public dataset "Flowers" (3600 images, 5 classes). This is not a vast dataset, but it keeps the tasks more manageable for development and you can scale up later, if you like.

In **'Getting Started'** we will work through the data preprocessing code from *Fast and Lean Data Science* which uses TensorFlow's `tf.data` package. There is no task for you here, but you will need to re-use some of this code later.

In **Task 1** you will **parallelise the data preprocessing in Spark**, using Google Cloud (GC) Dataproc. This involves adapting the code from 'Getting Started' to use Spark and running it in the cloud.

Section 2

In **Section 2** we are going to **measure the speed of reading data** in the cloud. In **Task 2** we will **paralellize the measuring** of different configurations **using Spark**.

Section 3

This section is about the theoretical discussion, based on one paper, in **Task 3**. The answers should be given in the PDF report.

General points

For **all coding tasks**, take the **time of the operations** and for the cloud operations, get performance **information from the web interfaces** for your reporting and analysis.

The **tasks** are **mostly independent** of each other. The later tasks can mostly be addressed without needing the solution to the earlier ones.

Section 0: Set-up

As usual, you need to run the **imports and authentication every time you work with this notebook**. Use the **local Spark** installation for development before you send jobs to the cloud.

Read through this section once and **fill in the project ID the first time**, then you can just step straight throught this at the beginning of each session - except for the two authentication cells.

Imports

We import some **packages that will be needed throughout**. For the **code that runs in the cloud**, we will need **separate import sections** that will need to be partly different from the one below.

In [1]:
```
import os, sys, math
import numpy as np
import scipy as sp
import scipy.stats
import time
import datetime
import string
import random
from matplotlib import pyplot as plt
import tensorflow as tf
print("Tensorflow version " + tf.__version__)
import pickle
```

```
Tensorflow version 2.18.0
```

Cloud and Drive authentication

This is for **authenticating with with GCS Google Drive**, so that we can create and use our own buckets and access Dataproc and AI-Platform.

This section **starts with the two interactive authentications**.

First, we mount Google Drive for persistent local storage and create a directory `DB-CW` thay you can use for this work. Then we'll set up the cloud environment, including a storage bucket.

In [2]:
```
print('Mounting google drive...')
from google.colab import drive
drive.mount('/content/drive')
%cd "/content/drive/MyDrive"
!mkdir BD-CW
%cd "/content/drive/MyDrive/BD-CW"
```

```
Mounting google drive...
Mounted at /content/drive
/content/drive/MyDrive
mkdir: cannot create directory ‘BD-CW’: File exists
/content/drive/MyDrive/BD-CW
```

Next, we authenticate with the GCS to enable access to Dataproc and AI-Platform.

In [3]:
```
import sys
if 'google.colab' in sys.modules:
    from google.colab import auth
    auth.authenticate_user()
```

It is useful to **create a new Google Cloud project** for this coursework. You can do this on the GC Console page by clicking on the entry at the top, right of the *Google Cloud Platform* and choosing *New Project*. **Copy** the **generated project ID** to the next cell. Also **enable billing** and the **Compute, Storage and Dataproc** APIs like we did during the labs.

We also specify the **default project and region**. The REGION should be `europe-west2` as it is closest to us geographically. This way we don't have to specify this information every time we access the cloud.

In [4]:
```
PROJECT = 'eng-throne-453420-e1'  ### USE YOUR GOOGLE CLOUD PROJECT ID HERE. ###
!gcloud config set project $PROJECT
REGION = 'europe-west2'
```

```
CLUSTER = '{}-cluster'.format(PROJECT)
!gcloud config set compute/region $REGION
!gcloud config set dataproc/region $REGION

!gcloud config list # show some information
```

```
Updated property [core/project].
WARNING: Property validation for compute/region was skipped.
Updated property [compute/region].
Updated property [dataproc/region].
[component_manager]
disable_update_check = True
[compute]
region = europe-west2
[core]
account = elias.michael@city.ac.uk
project = eng-throne-453420-e1
[dataproc]
region = europe-west2

Your active configuration is: [default]
```

With the cell below, we **create a storage bucket** that we will use later for **global storage**. If the bucket exists you will see a "ServiceException: 409 ...", which does not cause any problems. **You must create your own bucket to have write access.**

In [5]:
```
BUCKET = 'gs://{}-storage'.format(PROJECT)
!gsutil mb $BUCKET
```

```
Creating gs://eng-throne-453420-e1-storage/...
ServiceException: 409 A Cloud Storage bucket named 'eng-throne-453420-e1-storage' already exists. Try another na
me. Bucket names must be globally unique across all Google Cloud projects, including those outside of your organ
ization.
```

The cell below just **defines some routines for displaying images** that will be **used later**. You can see the code by double-clicking, but you don't need to study this.

In [6]:
```
#@title Utility functions for image display **[RUN THIS TO ACTIVATE]** { display-mode: "form" }
def display_9_images_from_dataset(dataset):
  plt.figure(figsize=(13,13))
  subplot=331
  for i, (image, label) in enumerate(dataset):
    plt.subplot(subplot)
    plt.axis('off')
    plt.imshow(image.numpy().astype(np.uint8))
    plt.title(str(label.numpy()), fontsize=16)
    # plt.title(label.numpy().decode(), fontsize=16)
    subplot += 1
    if i==8:
      break
  plt.tight_layout()
  plt.subplots_adjust(wspace=0.1, hspace=0.1)
  plt.show()

def display_training_curves(training, validation, title, subplot):
  if subplot%10==1: # set up the subplots on the first call
    plt.subplots(figsize=(10,10), facecolor='#F0F0F0')
    plt.tight_layout()
  ax = plt.subplot(subplot)
  ax.set_facecolor('#F8F8F8')
  ax.plot(training)
  ax.plot(validation)
  ax.set_title('model '+ title)
  ax.set_ylabel(title)
  ax.set_xlabel('epoch')
  ax.legend(['train', 'valid.'])

def dataset_to_numpy_util(dataset, N):
    dataset = dataset.batch(N)
    for images, labels in dataset:
        numpy_images = images.numpy()
        numpy_labels = labels.numpy()
        break;
    return numpy_images, numpy_labels

def title_from_label_and_target(label, correct_label):
  correct = (label == correct_label)
  return "{} [{}{}{}]".format(CLASSES[label], str(correct), ', shoud be ' if not correct else '',
                              CLASSES[correct_label] if not correct else ''), correct

def display_one_flower(image, title, subplot, red=False):
    plt.subplot(subplot)
    plt.axis('off')
    plt.imshow(image)
```

```
    plt.title(title, fontsize=16, color='red' if red else 'black')
    return subplot+1

def display_9_images_with_predictions(images, predictions, labels):
  subplot=331
  plt.figure(figsize=(13,13))
  classes = np.argmax(predictions, axis=-1)
  for i, image in enumerate(images):
    title, correct = title_from_label_and_target(classes[i], labels[i])
    subplot = display_one_flower(image, title, subplot, not correct)
    if i >= 8:
      break;

  plt.tight_layout()
  plt.subplots_adjust(wspace=0.1, hspace=0.1)
  plt.show()
```

Install Spark locally for quick testing

You can use the cell below to **install Spark locally on this Colab VM** (like in the labs), to do quicker small-scale interactive testing. Using Spark in the cloud with **Dataproc is still required for the final version**.

In [7]:
```
%cd
!apt-get update -qq
!apt-get install openjdk-8-jdk-headless -qq >> /dev/null # send any output to null device
!tar -xzf "/content/drive/My Drive/Big_Data/data/spark/spark-3.5.0-bin-hadoop3.tgz" # unpack

!pip install -q findspark
import os
os.environ["JAVA_HOME"] = "/usr/lib/jvm/java-8-openjdk-amd64"
os.environ["SPARK_HOME"] = "/root/spark-3.5.0-bin-hadoop3"
import findspark
findspark.init()
import pyspark
print(pyspark.__version__)
sc = pyspark.SparkContext.getOrCreate()
print(sc)
```

```
/root
W: Skipping acquire of configured file 'main/source/Sources' as repository 'https://r2u.stat.illinois.edu/ubuntu
jammy InRelease' does not seem to provide it (sources.list entry misspelt?)
3.5.0
<SparkContext master=local[*] appName=pyspark-shell>
```

Section 1: Data pre-processing

This section is about the **pre-processing of a dataset** for deep learning. We first look at a ready-made solution using Tensorflow and then we build a implement the same process with Spark. The tasks are about **parallelisation** and **analysis** the performance of the cloud implementations.

1.1 Getting started

In this section, we get started with the data pre-processing. The code is based on lecture 3 of the 'Fast and Lean Data Science' course.

This code is using the TensorFlow `tf.data` package, which supports map functions, similar to Spark. Your **task** will be to **re-implement the same approach in Spark**.

We start by **setting some variables for the *Flowers* dataset**.

In [8]:
```
GCS_PATTERN = 'gs://cloud-samples-data/ai-platform/flowers_tfrec/*/*.jpg' # glob  pattern for input files
PARTITIONS = 16 # no of partitions we will use later
TARGET_SIZE = [192, 192] # target resolution for the images
CLASSES = [b'daisy', b'dandelion', b'roses', b'sunflowers', b'tulips']
    # labels for the data
```

We **read the image files** from the public GCS bucket that contains the *Flowers* dataset. **TensorFlow** has **functions** to execute glob patterns that we use to calculate the the number of images in total and per partition (rounded up as we cannont deal with parts of images).

In [9]:
```
nb_images = len(tf.io.gfile.glob(GCS_PATTERN)) # number of images
partition_size = math.ceil(1.0 * nb_images / PARTITIONS) # images per partition (float)
print("GCS_PATTERN matches {} images, to be divided into {} partitions with up to {} images each.".format(nb_ima
```

```
GCS_PATTERN matches 3670 images, to be divided into 16 partitions with up to 230 images each.
```

Map functions

In order to read use the images for learning, they need to be **preprocessed** (decoded, resized, cropped, and potentially recompressed). Below are **map functions** for these steps. You **don't need to study** the **internals of these functions** in detail.

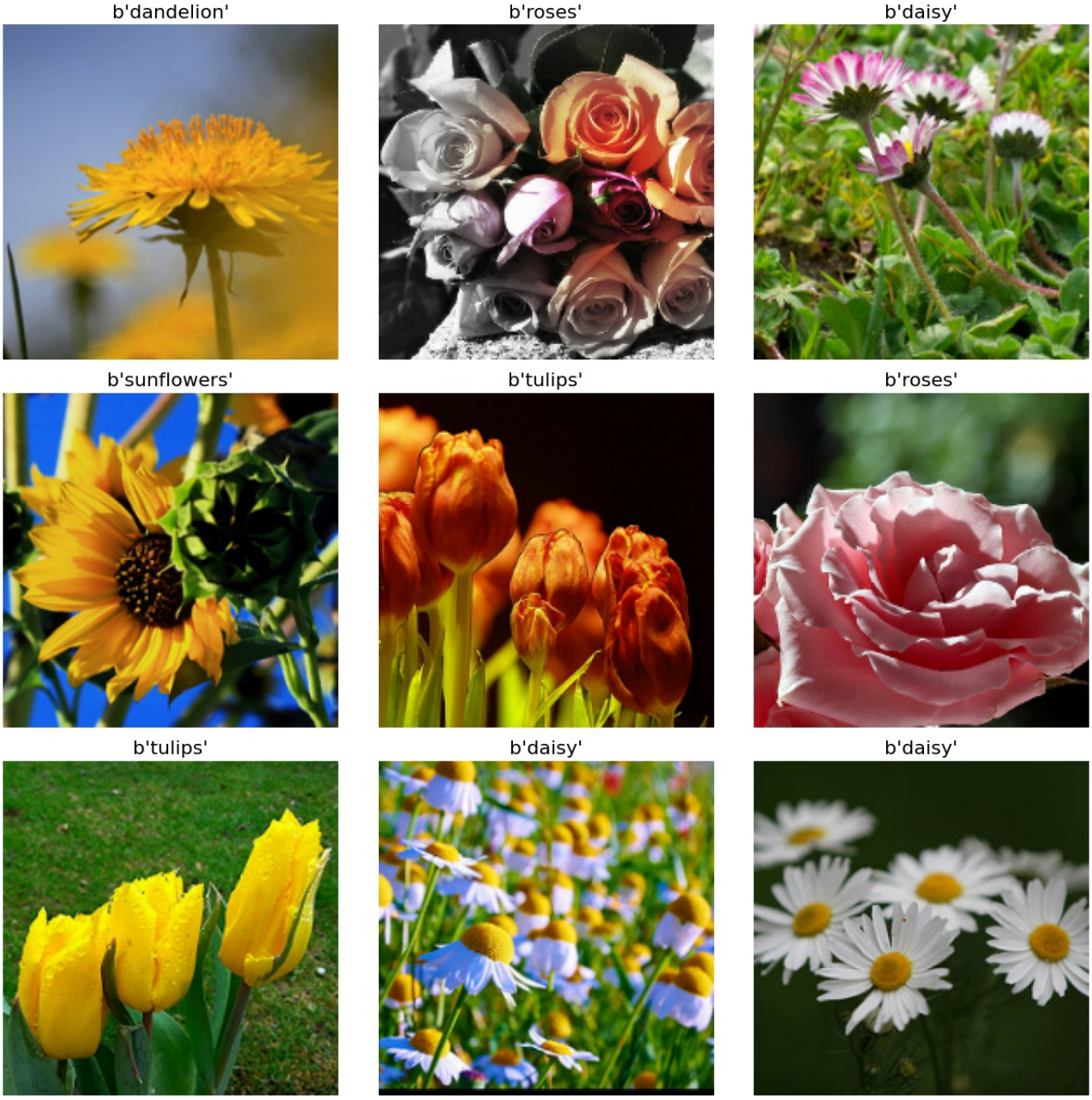
In [10]:
```
def decode_jpeg_and_label(filepath):
    # extracts the image data and creates a class label, based on the filepath
    bits = tf.io.read_file(filepath)
    image = tf.image.decode_jpeg(bits)
    # parse flower name from containing directory
    label = tf.strings.split(tf.expand_dims(filepath, axis=-1), sep='/')
    label2 = label.values[-2]
    return image, label2

def resize_and_crop_image(image, label):
    # Resizes and cropd using "fill" algorithm:
    # always make sure the resulting image is cut out from the source image
    # so that it fills the TARGET_SIZE entirely with no black bars
    # and a preserved aspect ratio.
    w = tf.shape(image)[0]
    h = tf.shape(image)[1]
    tw = TARGET_SIZE[1]
    th = TARGET_SIZE[0]
    resize_crit = (w * th) / (h * tw)
    image = tf.cond(resize_crit < 1,
                    lambda: tf.image.resize(image, [w*tw/w, h*tw/w]), # if true
                    lambda: tf.image.resize(image, [w*th/h, h*th/h])  # if false
                    )
    nw = tf.shape(image)[0]
    nh = tf.shape(image)[1]
    image = tf.image.crop_to_bounding_box(image, (nw - tw) // 2, (nh - th) // 2, tw, th)
    return image, label

def recompress_image(image, label):
    # this reduces the amount of data, but takes some time
    image = tf.cast(image, tf.uint8)
    image = tf.image.encode_jpeg(image, optimize_size=True, chroma_downsampling=False)
    return image, label
```

With `tf.data`, we can apply decoding and resizing as map functions.

In [11]:
```
dsetFiles = tf.data.Dataset.list_files(GCS_PATTERN) # This also shuffles the images
dsetDecoded = dsetFiles.map(decode_jpeg_and_label)
dsetResized = dsetDecoded.map(resize_and_crop_image)
```

We can also look at some images using the image display function defined above (the one with the hidden code).

In [12]:
```
display_9_images_from_dataset(dsetResized)
```



Now, let's test continuous reading from the dataset. We can see that reading the first 100 files already takes some time.

In [13]:
```
sample_set = dsetResized.batch(10).take(10) # take 10 batches of 10 images for testing
for image, label in sample_set:
    print("Image batch shape {}, {})".format(image.numpy().shape,
        [lbl.decode('utf8') for lbl in label.numpy()]))
```

```
Image batch shape (10, 192, 192, 3), ['roses', 'daisy', 'sunflowers', 'tulips', 'dandelion', 'roses', 'dandelion
', 'dandelion', 'dandelion', 'dandelion'])
Image batch shape (10, 192, 192, 3), ['roses', 'sunflowers', 'sunflowers', 'tulips', 'dandelion', 'daisy', 'tuli
ps', 'sunflowers', 'dandelion', 'dandelion'])
Image batch shape (10, 192, 192, 3), ['daisy', 'dandelion', 'dandelion', 'daisy', 'dandelion', 'daisy', 'dandeli
on', 'dandelion', 'dandelion', 'sunflowers'])
Image batch shape (10, 192, 192, 3), ['tulips', 'sunflowers', 'daisy', 'roses', 'sunflowers', 'daisy', 'tulips',
'roses', 'sunflowers', 'daisy'])
Image batch shape (10, 192, 192, 3), ['tulips', 'tulips', 'tulips', 'sunflowers', 'tulips', 'dandelion', 'dandel
ion', 'dandelion', 'tulips', 'dandelion'])
Image batch shape (10, 192, 192, 3), ['dandelion', 'sunflowers', 'tulips', 'sunflowers', 'tulips', 'roses', 'ros
es', 'sunflowers', 'roses', 'dandelion'])
Image batch shape (10, 192, 192, 3), ['dandelion', 'daisy', 'daisy', 'tulips', 'tulips', 'tulips', 'roses', 'sun
flowers', 'tulips', 'roses'])
Image batch shape (10, 192, 192, 3), ['daisy', 'roses', 'roses', 'sunflowers', 'tulips', 'roses', 'tulips', 'ros
es', 'daisy', 'sunflowers'])
Image batch shape (10, 192, 192, 3), ['roses', 'dandelion', 'dandelion', 'tulips', 'tulips', 'dandelion', 'dande
lion', 'tulips', 'dandelion', 'dandelion'])
Image batch shape (10, 192, 192, 3), ['daisy', 'sunflowers', 'roses', 'roses', 'roses', 'daisy', 'roses', 'sunfl
owers', 'tulips', 'sunflowers'])
```

1.2 Improving Speed

Using individual image files didn't look very fast. The 'Lean and Fast Data Science' course introduced **two techniques to improve the speed**.

Recompress the images

By **compressing** the images in the **reduced resolution** we save on the size. This **costs some CPU time** upfront, but **saves network and disk bandwith**, especially when the data are **read multiple times**.

In [14]:
```
# This is a quick test to get an idea how long recompressions takes.
dataset4 = dsetResized.map(recompress_image)
test_set = dataset4.batch(10).take(10)
for image, label in test_set:
    print("Image batch shape {}, {})".format(image.numpy().shape, [lbl.decode('utf8') for lbl in label.numpy()]
```

```
Image batch shape (10,), ['roses', 'tulips', 'dandelion', 'daisy', 'roses', 'daisy', 'tulips', 'sunflowers', 'su
nflowers', 'dandelion'])
Image batch shape (10,), ['daisy', 'sunflowers', 'roses', 'dandelion', 'sunflowers', 'daisy', 'daisy', 'tulips',
'tulips', 'tulips'])
Image batch shape (10,), ['sunflowers', 'dandelion', 'tulips', 'tulips', 'tulips', 'sunflowers', 'sunflowers', '
daisy', 'roses', 'dandelion'])
Image batch shape (10,), ['tulips', 'roses', 'sunflowers', 'dandelion', 'dandelion', 'roses', 'tulips', 'dandeli
on', 'sunflowers', 'sunflowers'])
Image batch shape (10,), ['tulips', 'tulips', 'dandelion', 'dandelion', 'tulips', 'dandelion', 'sunflowers', 'tu
lips', 'dandelion', 'tulips'])
Image batch shape (10,), ['sunflowers', 'daisy', 'roses', 'dandelion', 'tulips', 'tulips', 'sunflowers', 'daisy'
, 'daisy', 'tulips'])
Image batch shape (10,), ['tulips', 'tulips', 'daisy', 'tulips', 'sunflowers', 'daisy', 'roses', 'daisy', 'roses
', 'sunflowers'])
Image batch shape (10,), ['daisy', 'dandelion', 'tulips', 'roses', 'tulips', 'sunflowers', 'daisy', 'roses', 'ro
ses', 'sunflowers'])
Image batch shape (10,), ['tulips', 'sunflowers', 'tulips', 'dandelion', 'dandelion', 'roses', 'dandelion', 'dan
delion', 'tulips', 'tulips'])
Image batch shape (10,), ['daisy', 'tulips', 'tulips', 'tulips', 'daisy', 'tulips', 'dandelion', 'tulips', 'dand
elion', 'daisy'])
```

Write the dataset to TFRecord files

By writing **multiple preprocessed samples into a single file**, we can make further speed gains. We distribute the data over **partitions** to facilitate **parallelisation** when the data are used. First we need to **define a location** where we want to put the file.

In [15]:
```
GCS_OUTPUT = BUCKET + '/tfrecords-jpeg-192x192-2/flowers'  # prefix for output file names
```

Now we can **write the TFRecord files** to the bucket.

Running the cell takes some time and **only needs to be done once** or not at all, as you can use the publicly available data for the next few cells. For convenience I have commented out the call to `write_tfrecords` at the end of the next cell. You don't need to run it (it takes some time), but you'll need to use the code below later (but there is no need to study it in detail).

There is a **ready-made pre-processed data** versions available here: `gs://cloud-samples-data/ai-platform/flowers_tfrec/tfrecords-jpeg-192x192-2/`, that we can use for testing.

In [16]:
```
# functions for writing TFRecord entries
# Feature values are always stored as lists, a single data element will be a list of size 1
def _bytestring_feature(list_of_bytestrings):
    return tf.train.Feature(bytes_list=tf.train.BytesList(value=list_of_bytestrings))

def _int_feature(list_of_ints): # int64
    return tf.train.Feature(int64_list=tf.train.Int64List(value=list_of_ints))

def to_tfrecord(tfrec_filewriter, img_bytes, label): # Create tf data records
    class_num = np.argmax(np.array(CLASSES)==label) # 'roses' => 2 (order defined in CLASSES)
    one_hot_class = np.eye(len(CLASSES))[class_num]     # [0, 0, 1, 0, 0] for class #2, roses
    feature = {
        "image": _bytestring_feature([img_bytes]), # one image in the list
        "class": _int_feature([class_num]) #,      # one class in the list
    }
    return tf.train.Example(features=tf.train.Features(feature=feature))

def write_tfrecords(GCS_PATTERN,GCS_OUTPUT,partition_size): # write the images to files.
    print("Writing TFRecords")
    tt0 = time.time()
    filenames = tf.data.Dataset.list_files(GCS_PATTERN)
    dataset1 = filenames.map(decode_jpeg_and_label)
    dataset2 = dataset1.map(resize_and_crop_image)
    dataset3 = dataset2.map(recompress_image)
    dataset4 = dataset3.batch(partition_size) # partitioning: there will be one "batch" of images per file
    for partition, (image, label) in enumerate(dataset4):
        # batch size used as partition size here
```

```
        partition_size = image.numpy().shape[0]
        # good practice to have the number of records in the filename
        filename = GCS_OUTPUT + "{:02d}-{}.tfrec".format(partition, partition_size)
        # You need to change GCS_OUTPUT to your own bucket to actually create new files
        with tf.io.TFRecordWriter(filename) as out_file:
            for i in range(partition_size):
                example = to_tfrecord(out_file,
                                      image.numpy()[i], # re-compressed image: already a byte string
                                      label.numpy()[i] #
                                      )
                out_file.write(example.SerializeToString())
        print("Wrote file {} containing {} records".format(filename, partition_size))
    print("Total time: "+str(time.time()-tt0))

write_tfrecords(GCS_PATTERN,GCS_OUTPUT,partition_size) # uncomment to run this cell
```

```
Writing TFRecords
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers00-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers01-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers02-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers03-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers04-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers05-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers06-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers07-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers08-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers09-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers10-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers11-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers12-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers13-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers14-230.tfrec containing 230 records
Wrote file gs://eng-throne-453420-e1-storage/tfrecords-jpeg-192x192-2/flowers15-220.tfrec containing 220 records
Total time: 322.5682816505432
```

Test the TFRecord files

We can now **read from the TFRecord files**. By default, we use the files in the public bucket. Comment out the 1st line of the cell below to use the files written in the cell above.

In [17]:
```
# GCS_OUTPUT = 'gs://cloud-samples-data/ai-platform/flowers_tfrec/tfrecords-jpeg-192x192-2/'
# remove the line above to use your own files that you generated above

def read_tfrecord(example):
    features = {
        "image": tf.io.FixedLenFeature([], tf.string),  # tf.string = bytestring (not text string)
        "class": tf.io.FixedLenFeature([], tf.int64) #,   # shape [] means scalar
    }
    # decode the TFRecord
    example = tf.io.parse_single_example(example, features)
    image = tf.image.decode_jpeg(example['image'], channels=3)
    image = tf.reshape(image, [*TARGET_SIZE, 3])
    class_num = example['class']
    return image, class_num

def load_dataset(filenames):
    # read from TFRecords. For optimal performance, read from multiple
    # TFRecord files at once and set the option experimental_deterministic = False
    # to allow order-altering optimizations.
    option_no_order = tf.data.Options()
    option_no_order.experimental_deterministic = False

    dataset = tf.data.TFRecordDataset(filenames)
    dataset = dataset.with_options(option_no_order)
    dataset = dataset.map(read_tfrecord)
    return dataset


filenames = tf.io.gfile.glob(GCS_OUTPUT + "*.tfrec")
datasetTfrec = load_dataset(filenames)
```

Let's have a look **if reading from the TFRecord** files is **quicker**.

In [18]:
```
batched_dataset = datasetTfrec.batch(10)
sample_set = batched_dataset.take(10)
for image, label in sample_set:
    print("Image batch shape {}, {})".format(image.numpy().shape, \
                        [str(lbl) for lbl in label.numpy()]))
```

```
Image batch shape (10, 192, 192, 3), ['4', '4', '1', '0', '3', '4', '1', '1', '4', '1'])
Image batch shape (10, 192, 192, 3), ['2', '0', '1', '1', '1', '4', '4', '0', '0', '0'])
Image batch shape (10, 192, 192, 3), ['3', '0', '4', '3', '1', '1', '4', '0', '0', '1'])
Image batch shape (10, 192, 192, 3), ['1', '1', '4', '0', '1', '0', '4', '4', '3', '2'])
Image batch shape (10, 192, 192, 3), ['1', '4', '2', '4', '1', '2', '2', '4', '3', '1'])
Image batch shape (10, 192, 192, 3), ['0', '3', '0', '4', '0', '4', '0', '1', '1', '0'])
Image batch shape (10, 192, 192, 3), ['1', '3', '1', '1', '0', '0', '2', '1', '3', '0'])
Image batch shape (10, 192, 192, 3), ['4', '3', '2', '0', '0', '1', '1', '1', '1', '4'])
Image batch shape (10, 192, 192, 3), ['4', '2', '3', '3', '1', '2', '1', '4', '2', '3'])
Image batch shape (10, 192, 192, 3), ['3', '0', '2', '3', '4', '1', '3', '4', '4', '2'])
```

Wow, we have a **massive speed-up**! The repackageing is worthwhile :-)

Task 1: Write TFRecord files to the cloud with Spark (40%)

Since recompressing and repackaging is very effective, we would like to be able to do it inparallel for large datasets. This is a relatively straightforward case of **parallelisation**. We will **use Spark to implement** the same process as above, but in parallel.

1a) Create the script (14%)

Re-implement the pre-processing in Spark, using Spark mechanisms for **distributing** the workload **over multiple machines**.

You need to:

i) **Copy** over the **mapping functions** (see section 1.1) and **adapt** the resizing and recompression functions **to Spark** (only one argument). (3%)

ii) **Replace** the TensorFlow **Dataset objects with RDDs**, starting with an RDD that contains the list of image filenames. (3%)

iii) **Sample** the the RDD to a smaller number at an appropriate position in the code. Specify a sampling factor of 0.02 for short tests. (1%)

iv) Then **use the functions from above** to write the TFRecord files. (3%)

v) The code for **writing to the TFRecord files** needs to be put into a function, that can be applied to every partition with the 'RDD.mapPartitionsWithIndex' function. The return value of that function is not used here, but you should return the filename, so that you have a list of the created TFRecord files. (4%)

In [19]:
```python
# Import necessary libraries
import tensorflow as tf
import numpy as np
import io
from PIL import Image

# ---------- Step 1: Adapt functions for Spark ----------

# Function to decode a JPEG image file and extract its label from the directory structure
def decode_image_and_label(filepath):
    bits = tf.io.read_file(filepath)  # Read image file
    image = tf.image.decode_jpeg(bits)  # Decode JPEG
    label = tf.strings.split(tf.expand_dims(filepath, axis=-1), sep='/').values[-2]  # Extract class label
    return image, label.numpy()

# Function to resize and crop the image to target size
def resize_image(image):
    image = tf.image.resize_with_crop_or_pad(image, TARGET_SIZE[0], TARGET_SIZE[1])
    return image

# Function to recompress the image as JPEG to reduce size
def recompress_image(image):
    image = tf.cast(image, tf.uint8).numpy()  # Convert tensor to numpy array
    pil_img = Image.fromarray(image)  # Create PIL image
    buf = io.BytesIO()  # Create in-memory buffer
    pil_img.save(buf, format='JPEG', optimize=True)  # Save optimized JPEG to buffer
    return buf.getvalue()

# Function to create a TensorFlow Example record from image bytes and label
def encode_example(img_bytes, label):
    class_num = int(np.argmax(np.array(CLASSES) == label))  # Convert label string to integer
    feature = {
        "image": tf.train.Feature(bytes_list=tf.train.BytesList(value=[img_bytes])),  # Image feature
        "class": tf.train.Feature(int64_list=tf.train.Int64List(value=[class_num]))   # Class label feature
    }
    return tf.train.Example(features=tf.train.Features(feature=feature))

# ---------- Step 2: Create RDD of image file paths ----------

# Define path to the images stored in GCS
GCS_IMAGE_PATH = 'gs://cloud-samples-data/ai-platform/flowers_tfrec/*/*.jpg'
```

```
# Get list of filenames matching the pattern
filenames = tf.io.gfile.glob(GCS_IMAGE_PATH)
# Create an RDD from the filenames for parallel processing
rdd = sc.parallelize(filenames)

# ---------- Step 3: Sample 2% for fast test ----------

# Randomly sample 2% of the files to make testing faster
sampled_rdd = rdd.sample(False, 0.02)

# ---------- Step 4 + 5: Write TFRecords per partition ----------

# Define output path for TFRecord files
OUTPUT_PATH = BUCKET + "/spark-output/flowers-partition-"

# Function to write a partition of image files into a single TFRecord file
def write_partition(partition_idx, iterator):
    filenames = list(iterator)
    output_filename = OUTPUT_PATH + f"{partition_idx:02d}.tfrec"  # Partition-specific output filename
    with tf.io.TFRecordWriter(output_filename) as writer:
        for filepath in filenames:
            try:
                # Decode, preprocess, recompress and encode each image
                image, label = decode_image_and_label(filepath)
                image = resize_image(image)
                img_bytes = recompress_image(image)
                example = encode_example(img_bytes, label)
                writer.write(example.SerializeToString())  # Write serialized example to TFRecord
            except Exception as e:
                print(f"Error processing file {filepath}: {e}")  # Handle possible errors
    yield output_filename  # Return the output filename for tracking

# Apply the partition writing function across all partitions and collect filenames
written_files = sampled_rdd.mapPartitionsWithIndex(write_partition).collect()

# Show which TFRecord files were written
print("TFRecord files written:")
for f in written_files:
    print(f)
```

```
TFRecord files written:
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-00.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-01.tfrec
```

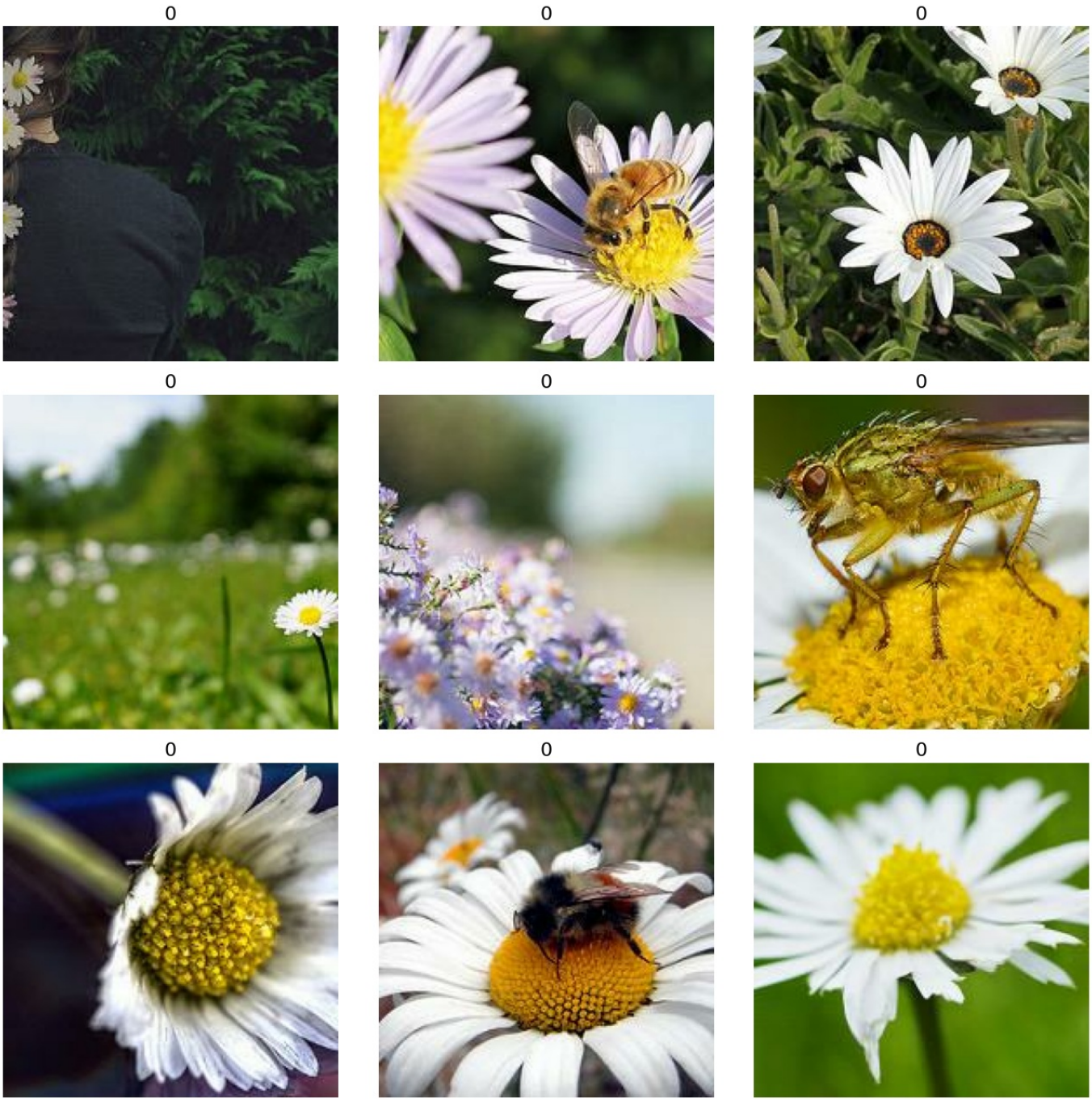
1b) Testing (3%)

i) Read from the TFRecord Dataset, using `load_dataset` and `display_9_images_from_dataset` to test.

In [20]:
```
GCS_OUTPUT = BUCKET + "/spark-output/"
```

In [21]:
```
filenames = tf.io.gfile.glob(GCS_OUTPUT + "*.tfrec")
datasetTfrec = load_dataset(filenames)
```

In [22]:
```
display_9_images_from_dataset(datasetTfrec)
```



ii) Write your code above into a file using the *cell magic* `%%writefile spark_write_tfrec.py` at the beginning of the file. Then, run the file locally in Spark.

In [42]:
```
%%writefile spark_write_tfrec.py
# Write the script into a .py file

# Import necessary libraries
import tensorflow as tf
import numpy as np
import io
from PIL import Image
from pyspark import SparkContext

# ---- Config ----

# Set the target image size after resizing
TARGET_SIZE = [192, 192]

# Define the list of classes (flower types)
CLASSES = [b'daisy', b'dandelion', b'roses', b'sunflowers', b'tulips']

# Define the GCS bucket and input/output paths
BUCKET = 'gs://eng-throne-453420-e1-storage'
GCS_IMAGE_PATH = 'gs://cloud-samples-data/ai-platform/flowers_tfrec/*/*.jpg'
OUTPUT_PATH = BUCKET + "/spark-output/flowers-partition-"

# ---- Image functions ----

# Decode image file and extract label
```

```
def decode_image_and_label(filepath):
    bits = tf.io.read_file(filepath)  # Read raw image bytes
    image = tf.image.decode_jpeg(bits)  # Decode JPEG format
    label = tf.strings.split(tf.expand_dims(filepath, axis=-1), sep='/').values[-2]  # Extract label
    return image, label.numpy()

# Resize the image to the target size by cropping or padding
def resize_image(image):
    image = tf.image.resize_with_crop_or_pad(image, TARGET_SIZE[0], TARGET_SIZE[1])
    return image

# Recompress the resized image into JPEG to reduce size
def recompress_image(image):
    image = tf.cast(image, tf.uint8).numpy()  # Convert to uint8 numpy array
    pil_img = Image.fromarray(image)  # Convert numpy array to PIL image
    buf = io.BytesIO()  # Create in-memory buffer
    pil_img.save(buf, format='JPEG', optimize=True)  # Save optimized JPEG
    return buf.getvalue()

# Encode the image bytes and label into a TensorFlow Example format
def encode_example(img_bytes, label):
    class_num = int(np.argmax(np.array(CLASSES) == label))  # Find the integer class index
    feature = {
        "image": tf.train.Feature(bytes_list=tf.train.BytesList(value=[img_bytes])),  # Image bytes
        "class": tf.train.Feature(int64_list=tf.train.Int64List(value=[class_num]))    # Class label
    }
    return tf.train.Example(features=tf.train.Features(feature=feature))

# Write images in each Spark partition to a TFRecord file
def write_partition(partition_idx, iterator):
    filenames = list(iterator)
    output_filename = OUTPUT_PATH + f"{partition_idx:02d}.tfrec"  # Unique filename per partition
    with tf.io.TFRecordWriter(output_filename) as writer:
        for filepath in filenames:
            try:
                image, label = decode_image_and_label(filepath)  # Decode and label
                image = resize_image(image)  # Resize
                img_bytes = recompress_image(image)  # Recompress
                example = encode_example(img_bytes, label)  # Encode example
                writer.write(example.SerializeToString())  # Write serialized example
            except Exception as e:
                print(f"Error processing file {filepath}: {e}")  # Handle exceptions gracefully
    yield output_filename  # Yield the output filename

# ---- Spark Job ----

if __name__ == "__main__":
    sc = SparkContext.getOrCreate()  # Get or create the SparkContext

    # List all image file paths matching the GCS pattern
    filenames = tf.io.gfile.glob(GCS_IMAGE_PATH)

    # Parallelize filenames into an RDD, explicitly setting 128 partitions for better parallelism ### TASK 1d #
    rdd = sc.parallelize(filenames, 128)

    # Randomly sample 2% of the images for quick test runs
    sampled_rdd = rdd.sample(False, 0.02)

    # Process each partition: decode, preprocess, recompress, and save to TFRecord files
    written_files = sampled_rdd.mapPartitionsWithIndex(write_partition).collect()

    # Print the list of TFRecord files that were successfully written
    print("TFRecord files written:")
    for f in written_files:
        print(f)
```

```
Overwriting spark_write_tfrec.py
```

In [43]:
```
# Run the spark_write_tfrec.py script locally using spark-submit
!spark-submit spark_write_tfrec.py
```

```
2025-04-26 08:42:18.860057: E external/local_xla/xla/stream_executor/cuda/cuda_fft.cc:477] Unable to register cu
FFT factory: Attempting to register factory for plugin cuFFT when one has already been registered
WARNING: All log messages before absl::InitializeLog() is called are written to STDERR
E0000 00:00:1745656938.910446   20646 cuda_dnn.cc:8310] Unable to register cuDNN factory: Attempting to register
factory for plugin cuDNN when one has already been registered
E0000 00:00:1745656938.927646   20646 cuda_blas.cc:1418] Unable to register cuBLAS factory: Attempting to regist
er factory for plugin cuBLAS when one has already been registered
25/04/26 08:42:25 INFO SparkContext: Running Spark version 3.5.0
25/04/26 08:42:25 INFO SparkContext: OS info Linux, 6.1.123+, amd64
25/04/26 08:42:25 INFO SparkContext: Java version 1.8.0_442
25/04/26 08:42:26 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin
-java classes where applicable
25/04/26 08:42:26 INFO ResourceUtils: ==============================================================
```

```
25/04/26 08:42:26 INFO ResourceUtils: No custom resources configured for spark.driver.
25/04/26 08:42:26 INFO ResourceUtils: ==============================================================
25/04/26 08:42:26 INFO SparkContext: Submitted application: spark_write_tfrec.py
25/04/26 08:42:26 INFO ResourceProfile: Default ResourceProfile created, executor resources: Map(cores -> name:
cores, amount: 1, script: , vendor: , memory -> name: memory, amount: 1024, script: , vendor: , offHeap -> name:
offHeap, amount: 0, script: , vendor: ), task resources: Map(cpus -> name: cpus, amount: 1.0)
25/04/26 08:42:26 INFO ResourceProfile: Limiting resource is cpu
25/04/26 08:42:26 INFO ResourceProfileManager: Added ResourceProfile id: 0
25/04/26 08:42:26 INFO SecurityManager: Changing view acls to: root
25/04/26 08:42:26 INFO SecurityManager: Changing modify acls to: root
25/04/26 08:42:26 INFO SecurityManager: Changing view acls groups to:
25/04/26 08:42:26 INFO SecurityManager: Changing modify acls groups to:
25/04/26 08:42:26 INFO SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users with v
iew permissions: root; groups with view permissions: EMPTY; users with modify permissions: root; groups with mod
ify permissions: EMPTY
25/04/26 08:42:27 INFO Utils: Successfully started service 'sparkDriver' on port 46159.
25/04/26 08:42:27 INFO SparkEnv: Registering MapOutputTracker
25/04/26 08:42:27 INFO SparkEnv: Registering BlockManagerMaster
25/04/26 08:42:27 INFO BlockManagerMasterEndpoint: Using org.apache.spark.storage.DefaultTopologyMapper for gett
ing topology information
25/04/26 08:42:27 INFO BlockManagerMasterEndpoint: BlockManagerMasterEndpoint up
25/04/26 08:42:27 INFO SparkEnv: Registering BlockManagerMasterHeartbeat
25/04/26 08:42:27 INFO DiskBlockManager: Created local directory at /tmp/blockmgr-48b0e502-0b6b-4d1d-ad38-5112c9
b10c5d
25/04/26 08:42:27 INFO MemoryStore: MemoryStore started with capacity 366.3 MiB
25/04/26 08:42:27 INFO SparkEnv: Registering OutputCommitCoordinator
25/04/26 08:42:27 INFO JettyUtils: Start Jetty 0.0.0.0:4040 for SparkUI
25/04/26 08:42:27 WARN Utils: Service 'SparkUI' could not bind on port 4040. Attempting port 4041.
25/04/26 08:42:27 INFO Utils: Successfully started service 'SparkUI' on port 4041.
25/04/26 08:42:27 INFO Executor: Starting executor ID driver on host 8bd937941c6d
25/04/26 08:42:27 INFO Executor: OS info Linux, 6.1.123+, amd64
25/04/26 08:42:27 INFO Executor: Java version 1.8.0_442
25/04/26 08:42:27 INFO Executor: Starting executor with user classpath (userClassPathFirst = false): ''
25/04/26 08:42:27 INFO Executor: Created or updated repl class loader org.apache.spark.util.MutableURLClassLoade
r@2226256f for default.
25/04/26 08:42:27 INFO Utils: Successfully started service 'org.apache.spark.network.netty.NettyBlockTransferSer
vice' on port 45059.
25/04/26 08:42:27 INFO NettyBlockTransferService: Server created on 8bd937941c6d:45059
25/04/26 08:42:27 INFO BlockManager: Using org.apache.spark.storage.RandomBlockReplicationPolicy for block repli
cation policy
25/04/26 08:42:27 INFO BlockManagerMaster: Registering BlockManager BlockManagerId(driver, 8bd937941c6d, 45059,
None)
25/04/26 08:42:27 INFO BlockManagerMasterEndpoint: Registering block manager 8bd937941c6d:45059 with 366.3 MiB R
AM, BlockManagerId(driver, 8bd937941c6d, 45059, None)
25/04/26 08:42:27 INFO BlockManagerMaster: Registered BlockManager BlockManagerId(driver, 8bd937941c6d, 45059, N
one)
25/04/26 08:42:27 INFO BlockManager: Initialized BlockManager: BlockManagerId(driver, 8bd937941c6d, 45059, None)
25/04/26 08:42:29 INFO SparkContext: Starting job: collect at /root/spark_write_tfrec.py:61
25/04/26 08:42:29 INFO DAGScheduler: Got job 0 (collect at /root/spark_write_tfrec.py:61) with 128 output partit
ions
25/04/26 08:42:29 INFO DAGScheduler: Final stage: ResultStage 0 (collect at /root/spark_write_tfrec.py:61)
25/04/26 08:42:29 INFO DAGScheduler: Parents of final stage: List()
25/04/26 08:42:29 INFO DAGScheduler: Missing parents: List()
25/04/26 08:42:29 INFO DAGScheduler: Submitting ResultStage 0 (PythonRDD[1] at collect at /root/spark_write_tfre
c.py:61), which has no missing parents
25/04/26 08:42:30 INFO MemoryStore: Block broadcast_0 stored as values in memory (estimated size 9.8 KiB, free 3
66.3 MiB)
25/04/26 08:42:30 INFO MemoryStore: Block broadcast_0_piece0 stored as bytes in memory (estimated size 6.3 KiB,
free 366.3 MiB)
25/04/26 08:42:30 INFO BlockManagerInfo: Added broadcast_0_piece0 in memory on 8bd937941c6d:45059 (size: 6.3 KiB
, free: 366.3 MiB)
25/04/26 08:42:30 INFO SparkContext: Created broadcast 0 from broadcast at DAGScheduler.scala:1580
25/04/26 08:42:30 INFO DAGScheduler: Submitting 128 missing tasks from ResultStage 0 (PythonRDD[1] at collect at
/root/spark_write_tfrec.py:61) (first 15 tasks are for partitions Vector(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 1
2, 13, 14))
25/04/26 08:42:30 INFO TaskSchedulerImpl: Adding task set 0.0 with 128 tasks resource profile 0
25/04/26 08:42:30 INFO TaskSetManager: Starting task 0.0 in stage 0.0 (TID 0) (8bd937941c6d, executor driver, pa
rtition 0, PROCESS_LOCAL, 10016 bytes)
25/04/26 08:42:30 INFO TaskSetManager: Starting task 1.0 in stage 0.0 (TID 1) (8bd937941c6d, executor driver, pa
rtition 1, PROCESS_LOCAL, 10020 bytes)
25/04/26 08:42:30 INFO Executor: Running task 0.0 in stage 0.0 (TID 0)
25/04/26 08:42:30 INFO Executor: Running task 1.0 in stage 0.0 (TID 1)
2025-04-26 08:42:32.660272: E external/local_xla/xla/stream_executor/cuda/cuda_fft.cc:477] Unable to register cu
FFT factory: Attempting to register factory for plugin cuFFT when one has already been registered
2025-04-26 08:42:32.667952: E external/local_xla/xla/stream_executor/cuda/cuda_fft.cc:477] Unable to register cu
FFT factory: Attempting to register factory for plugin cuFFT when one has already been registered
WARNING: All log messages before absl::InitializeLog() is called are written to STDERR
E0000 00:00:1745656952.719052   20756 cuda_dnn.cc:8310] Unable to register cuDNN factory: Attempting to register
factory for plugin cuDNN when one has already been registered
WARNING: All log messages before absl::InitializeLog() is called are written to STDERR
E0000 00:00:1745656952.724552   20755 cuda_dnn.cc:8310] Unable to register cuDNN factory: Attempting to register
factory for plugin cuDNN when one has already been registered
E0000 00:00:1745656952.737056   20756 cuda_blas.cc:1418] Unable to register cuBLAS factory: Attempting to regist
```

```
er factory for plugin cuBLAS when one has already been registered
E0000 00:00:1745656952.742277   20755 cuda_blas.cc:1418] Unable to register cuBLAS factory: Attempting to regist
er factory for plugin cuBLAS when one has already been registered
25/04/26 08:42:40 INFO PythonRunner: Times: total = 9652, boot = 761, init = 8469, finish = 422
25/04/26 08:42:40 INFO PythonRunner: Times: total = 9684, boot = 750, init = 8549, finish = 385
25/04/26 08:42:40 INFO Executor: Finished task 0.0 in stage 0.0 (TID 0). 1483 bytes result sent to driver
25/04/26 08:42:40 INFO Executor: Finished task 1.0 in stage 0.0 (TID 1). 1440 bytes result sent to driver
25/04/26 08:42:40 INFO TaskSetManager: Starting task 2.0 in stage 0.0 (TID 2) (8bd937941c6d, executor driver, pa
rtition 2, PROCESS_LOCAL, 10029 bytes)
25/04/26 08:42:40 INFO Executor: Running task 2.0 in stage 0.0 (TID 2)
25/04/26 08:42:40 INFO TaskSetManager: Starting task 3.0 in stage 0.0 (TID 3) (8bd937941c6d, executor driver, pa
rtition 3, PROCESS_LOCAL, 10024 bytes)
25/04/26 08:42:40 INFO Executor: Running task 3.0 in stage 0.0 (TID 3)
25/04/26 08:42:40 INFO TaskSetManager: Finished task 1.0 in stage 0.0 (TID 1) in 10535 ms on 8bd937941c6d (execu
tor driver) (1/128)
25/04/26 08:42:40 INFO PythonAccumulatorV2: Connected to AccumulatorServer at host: 127.0.0.1 port: 59331
25/04/26 08:42:40 INFO TaskSetManager: Finished task 0.0 in stage 0.0 (TID 0) in 10594 ms on 8bd937941c6d (execu
tor driver) (2/128)
25/04/26 08:42:41 INFO PythonRunner: Times: total = 609, boot = 237, init = 119, finish = 253
25/04/26 08:42:41 INFO Executor: Finished task 3.0 in stage 0.0 (TID 3). 1397 bytes result sent to driver
25/04/26 08:42:41 INFO TaskSetManager: Starting task 4.0 in stage 0.0 (TID 4) (8bd937941c6d, executor driver, pa
rtition 4, PROCESS_LOCAL, 10025 bytes)
25/04/26 08:42:41 INFO Executor: Running task 4.0 in stage 0.0 (TID 4)
25/04/26 08:42:41 INFO TaskSetManager: Finished task 3.0 in stage 0.0 (TID 3) in 648 ms on 8bd937941c6d (executo
r driver) (3/128)
25/04/26 08:42:41 INFO PythonRunner: Times: total = 780, boot = 292, init = 110, finish = 378
25/04/26 08:42:41 INFO Executor: Finished task 2.0 in stage 0.0 (TID 2). 1397 bytes result sent to driver
25/04/26 08:42:41 INFO TaskSetManager: Starting task 5.0 in stage 0.0 (TID 5) (8bd937941c6d, executor driver, pa
rtition 5, PROCESS_LOCAL, 10011 bytes)
25/04/26 08:42:41 INFO Executor: Running task 5.0 in stage 0.0 (TID 5)
25/04/26 08:42:41 INFO TaskSetManager: Finished task 2.0 in stage 0.0 (TID 2) in 835 ms on 8bd937941c6d (executo
r driver) (4/128)
25/04/26 08:42:42 INFO PythonRunner: Times: total = 590, boot = 257, init = 89, finish = 244
25/04/26 08:42:42 INFO Executor: Finished task 4.0 in stage 0.0 (TID 4). 1397 bytes result sent to driver
25/04/26 08:42:42 INFO TaskSetManager: Starting task 6.0 in stage 0.0 (TID 6) (8bd937941c6d, executor driver, pa
rtition 6, PROCESS_LOCAL, 10029 bytes)
25/04/26 08:42:42 INFO TaskSetManager: Finished task 4.0 in stage 0.0 (TID 4) in 624 ms on 8bd937941c6d (executo
r driver) (5/128)
25/04/26 08:42:42 INFO Executor: Running task 6.0 in stage 0.0 (TID 6)
25/04/26 08:42:42 INFO PythonRunner: Times: total = 559, boot = 262, init = 61, finish = 236
25/04/26 08:42:42 INFO Executor: Finished task 5.0 in stage 0.0 (TID 5). 1397 bytes result sent to driver
25/04/26 08:42:42 INFO TaskSetManager: Starting task 7.0 in stage 0.0 (TID 7) (8bd937941c6d, executor driver, pa
rtition 7, PROCESS_LOCAL, 10020 bytes)
25/04/26 08:42:42 INFO TaskSetManager: Finished task 5.0 in stage 0.0 (TID 5) in 578 ms on 8bd937941c6d (executo
r driver) (6/128)
25/04/26 08:42:42 INFO Executor: Running task 7.0 in stage 0.0 (TID 7)
2025-04-26 08:42:42.335567: E external/local_xla/xla/stream_executor/cuda/cuda_driver.cc:152] failed call to cuI
nit: INTERNAL: CUDA error: Failed call to cuInit: UNKNOWN ERROR (303)
2025-04-26 08:42:42.484087: E external/local_xla/xla/stream_executor/cuda/cuda_driver.cc:152] failed call to cuI
nit: INTERNAL: CUDA error: Failed call to cuInit: UNKNOWN ERROR (303)
25/04/26 08:42:42 INFO PythonRunner: Times: total = 800, boot = 203, init = 87, finish = 510
25/04/26 08:42:42 INFO Executor: Finished task 6.0 in stage 0.0 (TID 6). 1397 bytes result sent to driver
25/04/26 08:42:42 INFO TaskSetManager: Starting task 8.0 in stage 0.0 (TID 8) (8bd937941c6d, executor driver, pa
rtition 8, PROCESS_LOCAL, 10006 bytes)
25/04/26 08:42:42 INFO Executor: Running task 8.0 in stage 0.0 (TID 8)
25/04/26 08:42:42 INFO TaskSetManager: Finished task 6.0 in stage 0.0 (TID 6) in 836 ms on 8bd937941c6d (executo
r driver) (7/128)
25/04/26 08:42:43 INFO PythonRunner: Times: total = 919, boot = 259, init = 58, finish = 602
25/04/26 08:42:43 INFO Executor: Finished task 7.0 in stage 0.0 (TID 7). 1397 bytes result sent to driver
25/04/26 08:42:43 INFO TaskSetManager: Starting task 9.0 in stage 0.0 (TID 9) (8bd937941c6d, executor driver, pa
rtition 9, PROCESS_LOCAL, 10004 bytes)
25/04/26 08:42:43 INFO TaskSetManager: Finished task 7.0 in stage 0.0 (TID 7) in 948 ms on 8bd937941c6d (executo
r driver) (8/128)
25/04/26 08:42:43 INFO Executor: Running task 9.0 in stage 0.0 (TID 9)
25/04/26 08:42:43 INFO PythonRunner: Times: total = 634, boot = 202, init = 91, finish = 341
25/04/26 08:42:43 INFO Executor: Finished task 8.0 in stage 0.0 (TID 8). 1397 bytes result sent to driver
25/04/26 08:42:43 INFO TaskSetManager: Starting task 10.0 in stage 0.0 (TID 10) (8bd937941c6d, executor driver,
partition 10, PROCESS_LOCAL, 10003 bytes)
25/04/26 08:42:43 INFO Executor: Running task 10.0 in stage 0.0 (TID 10)
25/04/26 08:42:43 INFO TaskSetManager: Finished task 8.0 in stage 0.0 (TID 8) in 651 ms on 8bd937941c6d (executo
r driver) (9/128)
25/04/26 08:42:43 INFO PythonRunner: Times: total = 791, boot = 219, init = 59, finish = 513
25/04/26 08:42:43 INFO Executor: Finished task 9.0 in stage 0.0 (TID 9). 1397 bytes result sent to driver
25/04/26 08:42:43 INFO TaskSetManager: Starting task 11.0 in stage 0.0 (TID 11) (8bd937941c6d, executor driver,
partition 11, PROCESS_LOCAL, 10010 bytes)
25/04/26 08:42:43 INFO Executor: Running task 11.0 in stage 0.0 (TID 11)
25/04/26 08:42:43 INFO TaskSetManager: Finished task 9.0 in stage 0.0 (TID 9) in 813 ms on 8bd937941c6d (executo
r driver) (10/128)
25/04/26 08:42:44 INFO PythonRunner: Times: total = 625, boot = 217, init = 63, finish = 345
25/04/26 08:42:44 INFO Executor: Finished task 10.0 in stage 0.0 (TID 10). 1397 bytes result sent to driver
25/04/26 08:42:44 INFO TaskSetManager: Starting task 12.0 in stage 0.0 (TID 12) (8bd937941c6d, executor driver,
partition 12, PROCESS_LOCAL, 10011 bytes)
25/04/26 08:42:44 INFO Executor: Running task 12.0 in stage 0.0 (TID 12)
```

```
25/04/26 08:42:44 INFO TaskSetManager: Finished task 10.0 in stage 0.0 (TID 10) in 651 ms on 8bd937941c6d (execu
tor driver) (11/128)
25/04/26 08:42:44 INFO PythonRunner: Times: total = 539, boot = 204, init = 92, finish = 243
25/04/26 08:42:44 INFO Executor: Finished task 11.0 in stage 0.0 (TID 11). 1397 bytes result sent to driver
25/04/26 08:42:44 INFO TaskSetManager: Starting task 13.0 in stage 0.0 (TID 13) (8bd937941c6d, executor driver,
partition 13, PROCESS_LOCAL, 10004 bytes)
25/04/26 08:42:44 INFO Executor: Running task 13.0 in stage 0.0 (TID 13)
25/04/26 08:42:44 INFO TaskSetManager: Finished task 11.0 in stage 0.0 (TID 11) in 553 ms on 8bd937941c6d (execu
tor driver) (12/128)
25/04/26 08:42:44 INFO PythonRunner: Times: total = 733, boot = 214, init = 62, finish = 457
25/04/26 08:42:44 INFO Executor: Finished task 12.0 in stage 0.0 (TID 12). 1397 bytes result sent to driver
25/04/26 08:42:44 INFO TaskSetManager: Starting task 14.0 in stage 0.0 (TID 14) (8bd937941c6d, executor driver,
partition 14, PROCESS_LOCAL, 10009 bytes)
25/04/26 08:42:44 INFO Executor: Running task 14.0 in stage 0.0 (TID 14)
25/04/26 08:42:44 INFO TaskSetManager: Finished task 12.0 in stage 0.0 (TID 12) in 747 ms on 8bd937941c6d (execu
tor driver) (13/128)
25/04/26 08:42:44 INFO PythonRunner: Times: total = 494, boot = 182, init = 78, finish = 234
25/04/26 08:42:44 INFO Executor: Finished task 13.0 in stage 0.0 (TID 13). 1397 bytes result sent to driver
25/04/26 08:42:44 INFO TaskSetManager: Starting task 15.0 in stage 0.0 (TID 15) (8bd937941c6d, executor driver,
partition 15, PROCESS_LOCAL, 9995 bytes)
25/04/26 08:42:44 INFO TaskSetManager: Finished task 13.0 in stage 0.0 (TID 13) in 520 ms on 8bd937941c6d (execu
tor driver) (14/128)
25/04/26 08:42:44 INFO Executor: Running task 15.0 in stage 0.0 (TID 15)
25/04/26 08:42:45 INFO PythonRunner: Times: total = 677, boot = 337, init = 88, finish = 252
25/04/26 08:42:45 INFO Executor: Finished task 14.0 in stage 0.0 (TID 14). 1397 bytes result sent to driver
25/04/26 08:42:45 INFO TaskSetManager: Starting task 16.0 in stage 0.0 (TID 16) (8bd937941c6d, executor driver,
partition 16, PROCESS_LOCAL, 10000 bytes)
25/04/26 08:42:45 INFO Executor: Running task 16.0 in stage 0.0 (TID 16)
25/04/26 08:42:45 INFO TaskSetManager: Finished task 14.0 in stage 0.0 (TID 14) in 696 ms on 8bd937941c6d (execu
tor driver) (15/128)
25/04/26 08:42:45 INFO PythonRunner: Times: total = 629, boot = 311, init = 78, finish = 240
25/04/26 08:42:45 INFO Executor: Finished task 15.0 in stage 0.0 (TID 15). 1397 bytes result sent to driver
25/04/26 08:42:45 INFO TaskSetManager: Starting task 17.0 in stage 0.0 (TID 17) (8bd937941c6d, executor driver,
partition 17, PROCESS_LOCAL, 9997 bytes)
25/04/26 08:42:45 INFO TaskSetManager: Finished task 15.0 in stage 0.0 (TID 15) in 646 ms on 8bd937941c6d (execu
tor driver) (16/128)
25/04/26 08:42:45 INFO Executor: Running task 17.0 in stage 0.0 (TID 17)
25/04/26 08:42:46 INFO PythonRunner: Times: total = 591, boot = 268, init = 90, finish = 233
25/04/26 08:42:46 INFO Executor: Finished task 17.0 in stage 0.0 (TID 17). 1397 bytes result sent to driver
25/04/26 08:42:46 INFO TaskSetManager: Starting task 18.0 in stage 0.0 (TID 18) (8bd937941c6d, executor driver,
partition 18, PROCESS_LOCAL, 10010 bytes)
25/04/26 08:42:46 INFO Executor: Running task 18.0 in stage 0.0 (TID 18)
25/04/26 08:42:46 INFO TaskSetManager: Finished task 17.0 in stage 0.0 (TID 17) in 605 ms on 8bd937941c6d (execu
tor driver) (17/128)
25/04/26 08:42:46 INFO PythonRunner: Times: total = 921, boot = 267, init = 90, finish = 564
25/04/26 08:42:46 INFO Executor: Finished task 16.0 in stage 0.0 (TID 16). 1397 bytes result sent to driver
25/04/26 08:42:46 INFO TaskSetManager: Starting task 19.0 in stage 0.0 (TID 19) (8bd937941c6d, executor driver,
partition 19, PROCESS_LOCAL, 10000 bytes)
25/04/26 08:42:46 INFO Executor: Running task 19.0 in stage 0.0 (TID 19)
25/04/26 08:42:46 INFO TaskSetManager: Finished task 16.0 in stage 0.0 (TID 16) in 935 ms on 8bd937941c6d (execu
tor driver) (18/128)
25/04/26 08:42:46 INFO PythonRunner: Times: total = 538, boot = 213, init = 62, finish = 263
25/04/26 08:42:46 INFO Executor: Finished task 18.0 in stage 0.0 (TID 18). 1397 bytes result sent to driver
25/04/26 08:42:46 INFO TaskSetManager: Starting task 20.0 in stage 0.0 (TID 20) (8bd937941c6d, executor driver,
partition 20, PROCESS_LOCAL, 10011 bytes)
25/04/26 08:42:46 INFO Executor: Running task 20.0 in stage 0.0 (TID 20)
25/04/26 08:42:46 INFO TaskSetManager: Finished task 18.0 in stage 0.0 (TID 18) in 553 ms on 8bd937941c6d (execu
tor driver) (19/128)
25/04/26 08:42:47 INFO PythonRunner: Times: total = 689, boot = 184, init = 103, finish = 402
25/04/26 08:42:47 INFO Executor: Finished task 19.0 in stage 0.0 (TID 19). 1397 bytes result sent to driver
25/04/26 08:42:47 INFO TaskSetManager: Starting task 21.0 in stage 0.0 (TID 21) (8bd937941c6d, executor driver,
partition 21, PROCESS_LOCAL, 10002 bytes)
25/04/26 08:42:47 INFO TaskSetManager: Finished task 19.0 in stage 0.0 (TID 19) in 712 ms on 8bd937941c6d (execu
tor driver) (20/128)
25/04/26 08:42:47 INFO Executor: Running task 21.0 in stage 0.0 (TID 21)
25/04/26 08:42:47 INFO PythonRunner: Times: total = 646, boot = 212, init = 71, finish = 363
25/04/26 08:42:47 INFO Executor: Finished task 20.0 in stage 0.0 (TID 20). 1397 bytes result sent to driver
25/04/26 08:42:47 INFO TaskSetManager: Starting task 22.0 in stage 0.0 (TID 22) (8bd937941c6d, executor driver,
partition 22, PROCESS_LOCAL, 10059 bytes)
25/04/26 08:42:47 INFO TaskSetManager: Finished task 20.0 in stage 0.0 (TID 20) in 661 ms on 8bd937941c6d (execu
tor driver) (21/128)
25/04/26 08:42:47 INFO Executor: Running task 22.0 in stage 0.0 (TID 22)
25/04/26 08:42:47 INFO PythonRunner: Times: total = 550, boot = 217, init = 91, finish = 242
25/04/26 08:42:47 INFO Executor: Finished task 21.0 in stage 0.0 (TID 21). 1397 bytes result sent to driver
25/04/26 08:42:47 INFO TaskSetManager: Starting task 23.0 in stage 0.0 (TID 23) (8bd937941c6d, executor driver,
partition 23, PROCESS_LOCAL, 10118 bytes)
25/04/26 08:42:47 INFO TaskSetManager: Finished task 21.0 in stage 0.0 (TID 21) in 574 ms on 8bd937941c6d (execu
tor driver) (22/128)
25/04/26 08:42:47 INFO Executor: Running task 23.0 in stage 0.0 (TID 23)
25/04/26 08:42:48 INFO PythonRunner: Times: total = 870, boot = 232, init = 64, finish = 574
25/04/26 08:42:48 INFO Executor: Finished task 22.0 in stage 0.0 (TID 22). 1397 bytes result sent to driver
25/04/26 08:42:48 INFO TaskSetManager: Starting task 24.0 in stage 0.0 (TID 24) (8bd937941c6d, executor driver,
partition 24, PROCESS_LOCAL, 10128 bytes)
```

```
25/04/26 08:42:48 INFO TaskSetManager: Finished task 22.0 in stage 0.0 (TID 22) in 883 ms on 8bd937941c6d (execu
tor driver) (23/128)
25/04/26 08:42:48 INFO Executor: Running task 24.0 in stage 0.0 (TID 24)
25/04/26 08:42:48 INFO PythonRunner: Times: total = 669, boot = 236, init = 69, finish = 364
25/04/26 08:42:48 INFO Executor: Finished task 23.0 in stage 0.0 (TID 23). 1397 bytes result sent to driver
25/04/26 08:42:48 INFO TaskSetManager: Starting task 25.0 in stage 0.0 (TID 25) (8bd937941c6d, executor driver,
partition 25, PROCESS_LOCAL, 10130 bytes)
25/04/26 08:42:48 INFO Executor: Running task 25.0 in stage 0.0 (TID 25)
25/04/26 08:42:48 INFO TaskSetManager: Finished task 23.0 in stage 0.0 (TID 23) in 694 ms on 8bd937941c6d (execu
tor driver) (24/128)
25/04/26 08:42:49 INFO PythonRunner: Times: total = 781, boot = 215, init = 95, finish = 471
25/04/26 08:42:49 INFO Executor: Finished task 24.0 in stage 0.0 (TID 24). 1397 bytes result sent to driver
25/04/26 08:42:49 INFO TaskSetManager: Starting task 26.0 in stage 0.0 (TID 26) (8bd937941c6d, executor driver,
partition 26, PROCESS_LOCAL, 10132 bytes)
25/04/26 08:42:49 INFO TaskSetManager: Finished task 24.0 in stage 0.0 (TID 24) in 801 ms on 8bd937941c6d (execu
tor driver) (25/128)
25/04/26 08:42:49 INFO Executor: Running task 26.0 in stage 0.0 (TID 26)
25/04/26 08:42:49 INFO PythonRunner: Times: total = 651, boot = 235, init = 78, finish = 338
25/04/26 08:42:49 INFO Executor: Finished task 25.0 in stage 0.0 (TID 25). 1397 bytes result sent to driver
25/04/26 08:42:49 INFO TaskSetManager: Starting task 27.0 in stage 0.0 (TID 27) (8bd937941c6d, executor driver,
partition 27, PROCESS_LOCAL, 10121 bytes)
25/04/26 08:42:49 INFO TaskSetManager: Finished task 25.0 in stage 0.0 (TID 25) in 664 ms on 8bd937941c6d (execu
tor driver) (26/128)
25/04/26 08:42:49 INFO Executor: Running task 27.0 in stage 0.0 (TID 27)
25/04/26 08:42:49 INFO PythonRunner: Times: total = 589, boot = 254, init = 91, finish = 244
25/04/26 08:42:49 INFO Executor: Finished task 27.0 in stage 0.0 (TID 27). 1397 bytes result sent to driver
25/04/26 08:42:49 INFO TaskSetManager: Starting task 28.0 in stage 0.0 (TID 28) (8bd937941c6d, executor driver,
partition 28, PROCESS_LOCAL, 10128 bytes)
25/04/26 08:42:49 INFO TaskSetManager: Finished task 27.0 in stage 0.0 (TID 27) in 600 ms on 8bd937941c6d (execu
tor driver) (27/128)
25/04/26 08:42:49 INFO Executor: Running task 28.0 in stage 0.0 (TID 28)
25/04/26 08:42:49 INFO PythonRunner: Times: total = 753, boot = 252, init = 94, finish = 407
25/04/26 08:42:49 INFO Executor: Finished task 26.0 in stage 0.0 (TID 26). 1397 bytes result sent to driver
25/04/26 08:42:49 INFO TaskSetManager: Starting task 29.0 in stage 0.0 (TID 29) (8bd937941c6d, executor driver,
partition 29, PROCESS_LOCAL, 10149 bytes)
25/04/26 08:42:49 INFO TaskSetManager: Finished task 26.0 in stage 0.0 (TID 26) in 783 ms on 8bd937941c6d (execu
tor driver) (28/128)
25/04/26 08:42:49 INFO Executor: Running task 29.0 in stage 0.0 (TID 29)
25/04/26 08:42:50 INFO PythonRunner: Times: total = 930, boot = 388, init = 265, finish = 277
25/04/26 08:42:50 INFO Executor: Finished task 28.0 in stage 0.0 (TID 28). 1397 bytes result sent to driver
25/04/26 08:42:50 INFO TaskSetManager: Starting task 30.0 in stage 0.0 (TID 30) (8bd937941c6d, executor driver,
partition 30, PROCESS_LOCAL, 10135 bytes)
25/04/26 08:42:50 INFO Executor: Running task 30.0 in stage 0.0 (TID 30)
25/04/26 08:42:50 INFO TaskSetManager: Finished task 28.0 in stage 0.0 (TID 28) in 947 ms on 8bd937941c6d (execu
tor driver) (29/128)
25/04/26 08:42:50 INFO PythonRunner: Times: total = 1036, boot = 642, init = 98, finish = 296
25/04/26 08:42:50 INFO Executor: Finished task 29.0 in stage 0.0 (TID 29). 1397 bytes result sent to driver
25/04/26 08:42:50 INFO TaskSetManager: Starting task 31.0 in stage 0.0 (TID 31) (8bd937941c6d, executor driver,
partition 31, PROCESS_LOCAL, 12679 bytes)
25/04/26 08:42:50 INFO TaskSetManager: Finished task 29.0 in stage 0.0 (TID 29) in 1065 ms on 8bd937941c6d (exec
utor driver) (30/128)
25/04/26 08:42:50 INFO Executor: Running task 31.0 in stage 0.0 (TID 31)
25/04/26 08:42:51 INFO PythonRunner: Times: total = 813, boot = 382, init = 175, finish = 256
25/04/26 08:42:51 INFO Executor: Finished task 30.0 in stage 0.0 (TID 30). 1397 bytes result sent to driver
25/04/26 08:42:51 INFO TaskSetManager: Starting task 32.0 in stage 0.0 (TID 32) (8bd937941c6d, executor driver,
partition 32, PROCESS_LOCAL, 10139 bytes)
25/04/26 08:42:51 INFO Executor: Running task 32.0 in stage 0.0 (TID 32)
25/04/26 08:42:51 INFO TaskSetManager: Finished task 30.0 in stage 0.0 (TID 30) in 848 ms on 8bd937941c6d (execu
tor driver) (31/128)
25/04/26 08:42:51 INFO PythonRunner: Times: total = 789, boot = 336, init = 109, finish = 344
25/04/26 08:42:51 INFO Executor: Finished task 31.0 in stage 0.0 (TID 31). 1397 bytes result sent to driver
25/04/26 08:42:51 INFO TaskSetManager: Starting task 33.0 in stage 0.0 (TID 33) (8bd937941c6d, executor driver,
partition 33, PROCESS_LOCAL, 10119 bytes)
25/04/26 08:42:51 INFO Executor: Running task 33.0 in stage 0.0 (TID 33)
25/04/26 08:42:51 INFO TaskSetManager: Finished task 31.0 in stage 0.0 (TID 31) in 808 ms on 8bd937941c6d (execu
tor driver) (32/128)
25/04/26 08:42:52 INFO PythonRunner: Times: total = 703, boot = 307, init = 126, finish = 270
25/04/26 08:42:52 INFO Executor: Finished task 32.0 in stage 0.0 (TID 32). 1397 bytes result sent to driver
25/04/26 08:42:52 INFO TaskSetManager: Starting task 34.0 in stage 0.0 (TID 34) (8bd937941c6d, executor driver,
partition 34, PROCESS_LOCAL, 10117 bytes)
25/04/26 08:42:52 INFO Executor: Running task 34.0 in stage 0.0 (TID 34)
25/04/26 08:42:52 INFO TaskSetManager: Finished task 32.0 in stage 0.0 (TID 32) in 727 ms on 8bd937941c6d (execu
tor driver) (33/128)
25/04/26 08:42:52 INFO PythonRunner: Times: total = 819, boot = 388, init = 169, finish = 262
25/04/26 08:42:52 INFO Executor: Finished task 33.0 in stage 0.0 (TID 33). 1440 bytes result sent to driver
25/04/26 08:42:52 INFO TaskSetManager: Starting task 35.0 in stage 0.0 (TID 35) (8bd937941c6d, executor driver,
partition 35, PROCESS_LOCAL, 10116 bytes)
25/04/26 08:42:52 INFO Executor: Running task 35.0 in stage 0.0 (TID 35)
25/04/26 08:42:52 INFO TaskSetManager: Finished task 33.0 in stage 0.0 (TID 33) in 842 ms on 8bd937941c6d (execu
tor driver) (34/128)
25/04/26 08:42:53 INFO PythonRunner: Times: total = 1162, boot = 571, init = 91, finish = 500
25/04/26 08:42:53 INFO Executor: Finished task 34.0 in stage 0.0 (TID 34). 1397 bytes result sent to driver
25/04/26 08:42:53 INFO TaskSetManager: Starting task 36.0 in stage 0.0 (TID 36) (8bd937941c6d, executor driver,
```

```
partition 36, PROCESS_LOCAL, 10112 bytes)
25/04/26 08:42:53 INFO Executor: Running task 36.0 in stage 0.0 (TID 36)
25/04/26 08:42:53 INFO TaskSetManager: Finished task 34.0 in stage 0.0 (TID 34) in 1180 ms on 8bd937941c6d (exec
utor driver) (35/128)
25/04/26 08:42:53 INFO PythonRunner: Times: total = 928, boot = 380, init = 64, finish = 484
25/04/26 08:42:53 INFO Executor: Finished task 35.0 in stage 0.0 (TID 35). 1397 bytes result sent to driver
25/04/26 08:42:53 INFO TaskSetManager: Starting task 37.0 in stage 0.0 (TID 37) (8bd937941c6d, executor driver,
partition 37, PROCESS_LOCAL, 10123 bytes)
25/04/26 08:42:53 INFO Executor: Running task 37.0 in stage 0.0 (TID 37)
25/04/26 08:42:53 INFO TaskSetManager: Finished task 35.0 in stage 0.0 (TID 35) in 937 ms on 8bd937941c6d (execu
tor driver) (36/128)
25/04/26 08:42:53 INFO PythonRunner: Times: total = 549, boot = 224, init = 92, finish = 233
25/04/26 08:42:53 INFO Executor: Finished task 36.0 in stage 0.0 (TID 36). 1397 bytes result sent to driver
25/04/26 08:42:53 INFO TaskSetManager: Starting task 38.0 in stage 0.0 (TID 38) (8bd937941c6d, executor driver,
partition 38, PROCESS_LOCAL, 10123 bytes)
25/04/26 08:42:53 INFO TaskSetManager: Finished task 36.0 in stage 0.0 (TID 36) in 560 ms on 8bd937941c6d (execu
tor driver) (37/128)
25/04/26 08:42:53 INFO Executor: Running task 38.0 in stage 0.0 (TID 38)
25/04/26 08:42:54 INFO PythonRunner: Times: total = 548, boot = 256, init = 64, finish = 228
25/04/26 08:42:54 INFO Executor: Finished task 37.0 in stage 0.0 (TID 37). 1397 bytes result sent to driver
25/04/26 08:42:54 INFO TaskSetManager: Starting task 39.0 in stage 0.0 (TID 39) (8bd937941c6d, executor driver,
partition 39, PROCESS_LOCAL, 10111 bytes)
25/04/26 08:42:54 INFO Executor: Running task 39.0 in stage 0.0 (TID 39)
25/04/26 08:42:54 INFO TaskSetManager: Finished task 37.0 in stage 0.0 (TID 37) in 567 ms on 8bd937941c6d (execu
tor driver) (38/128)
25/04/26 08:42:54 INFO PythonRunner: Times: total = 789, boot = 341, init = 95, finish = 353
25/04/26 08:42:54 INFO Executor: Finished task 38.0 in stage 0.0 (TID 38). 1397 bytes result sent to driver
25/04/26 08:42:54 INFO TaskSetManager: Starting task 40.0 in stage 0.0 (TID 40) (8bd937941c6d, executor driver,
partition 40, PROCESS_LOCAL, 10119 bytes)
25/04/26 08:42:54 INFO Executor: Running task 40.0 in stage 0.0 (TID 40)
25/04/26 08:42:54 INFO TaskSetManager: Finished task 38.0 in stage 0.0 (TID 38) in 801 ms on 8bd937941c6d (execu
tor driver) (39/128)
25/04/26 08:42:54 INFO PythonRunner: Times: total = 685, boot = 294, init = 132, finish = 259
25/04/26 08:42:54 INFO Executor: Finished task 39.0 in stage 0.0 (TID 39). 1397 bytes result sent to driver
25/04/26 08:42:54 INFO TaskSetManager: Starting task 41.0 in stage 0.0 (TID 41) (8bd937941c6d, executor driver,
partition 41, PROCESS_LOCAL, 10113 bytes)
25/04/26 08:42:54 INFO TaskSetManager: Finished task 39.0 in stage 0.0 (TID 39) in 719 ms on 8bd937941c6d (execu
tor driver) (40/128)
25/04/26 08:42:54 INFO Executor: Running task 41.0 in stage 0.0 (TID 41)
25/04/26 08:42:55 INFO PythonRunner: Times: total = 620, boot = 240, init = 108, finish = 272
25/04/26 08:42:55 INFO Executor: Finished task 41.0 in stage 0.0 (TID 41). 1397 bytes result sent to driver
25/04/26 08:42:55 INFO TaskSetManager: Starting task 42.0 in stage 0.0 (TID 42) (8bd937941c6d, executor driver,
partition 42, PROCESS_LOCAL, 10110 bytes)
25/04/26 08:42:55 INFO Executor: Running task 42.0 in stage 0.0 (TID 42)
25/04/26 08:42:55 INFO TaskSetManager: Finished task 41.0 in stage 0.0 (TID 41) in 637 ms on 8bd937941c6d (execu
tor driver) (41/128)
25/04/26 08:42:55 INFO PythonRunner: Times: total = 737, boot = 262, init = 116, finish = 359
25/04/26 08:42:55 INFO Executor: Finished task 40.0 in stage 0.0 (TID 40). 1397 bytes result sent to driver
25/04/26 08:42:55 INFO TaskSetManager: Starting task 43.0 in stage 0.0 (TID 43) (8bd937941c6d, executor driver,
partition 43, PROCESS_LOCAL, 10117 bytes)
25/04/26 08:42:55 INFO TaskSetManager: Finished task 40.0 in stage 0.0 (TID 40) in 748 ms on 8bd937941c6d (execu
tor driver) (42/128)
25/04/26 08:42:55 INFO Executor: Running task 43.0 in stage 0.0 (TID 43)
25/04/26 08:42:56 INFO PythonRunner: Times: total = 576, boot = 214, init = 94, finish = 268
25/04/26 08:42:56 INFO Executor: Finished task 42.0 in stage 0.0 (TID 42). 1397 bytes result sent to driver
25/04/26 08:42:56 INFO TaskSetManager: Starting task 44.0 in stage 0.0 (TID 44) (8bd937941c6d, executor driver,
partition 44, PROCESS_LOCAL, 10116 bytes)
25/04/26 08:42:56 INFO TaskSetManager: Finished task 42.0 in stage 0.0 (TID 42) in 591 ms on 8bd937941c6d (execu
tor driver) (43/128)
25/04/26 08:42:56 INFO Executor: Running task 44.0 in stage 0.0 (TID 44)
25/04/26 08:42:56 INFO PythonRunner: Times: total = 758, boot = 270, init = 65, finish = 423
25/04/26 08:42:56 INFO Executor: Finished task 43.0 in stage 0.0 (TID 43). 1397 bytes result sent to driver
25/04/26 08:42:56 INFO TaskSetManager: Starting task 45.0 in stage 0.0 (TID 45) (8bd937941c6d, executor driver,
partition 45, PROCESS_LOCAL, 10122 bytes)
25/04/26 08:42:56 INFO TaskSetManager: Finished task 43.0 in stage 0.0 (TID 43) in 769 ms on 8bd937941c6d (execu
tor driver) (44/128)
25/04/26 08:42:56 INFO Executor: Running task 45.0 in stage 0.0 (TID 45)
25/04/26 08:42:56 INFO PythonRunner: Times: total = 812, boot = 212, init = 125, finish = 475
25/04/26 08:42:56 INFO Executor: Finished task 44.0 in stage 0.0 (TID 44). 1397 bytes result sent to driver
25/04/26 08:42:56 INFO TaskSetManager: Starting task 46.0 in stage 0.0 (TID 46) (8bd937941c6d, executor driver,
partition 46, PROCESS_LOCAL, 10121 bytes)
25/04/26 08:42:56 INFO TaskSetManager: Finished task 44.0 in stage 0.0 (TID 44) in 824 ms on 8bd937941c6d (execu
tor driver) (45/128)
25/04/26 08:42:56 INFO Executor: Running task 46.0 in stage 0.0 (TID 46)
25/04/26 08:42:57 INFO PythonRunner: Times: total = 856, boot = 241, init = 58, finish = 557
25/04/26 08:42:57 INFO Executor: Finished task 45.0 in stage 0.0 (TID 45). 1397 bytes result sent to driver
25/04/26 08:42:57 INFO TaskSetManager: Starting task 47.0 in stage 0.0 (TID 47) (8bd937941c6d, executor driver,
partition 47, PROCESS_LOCAL, 10123 bytes)
25/04/26 08:42:57 INFO TaskSetManager: Finished task 45.0 in stage 0.0 (TID 45) in 866 ms on 8bd937941c6d (execu
tor driver) (46/128)
25/04/26 08:42:57 INFO Executor: Running task 47.0 in stage 0.0 (TID 47)
25/04/26 08:42:57 INFO PythonRunner: Times: total = 505, boot = 199, init = 69, finish = 237
25/04/26 08:42:57 INFO Executor: Finished task 46.0 in stage 0.0 (TID 46). 1397 bytes result sent to driver
```

```
25/04/26 08:42:57 INFO TaskSetManager: Starting task 48.0 in stage 0.0 (TID 48) (8bd937941c6d, executor driver,
partition 48, PROCESS_LOCAL, 10124 bytes)
25/04/26 08:42:57 INFO Executor: Running task 48.0 in stage 0.0 (TID 48)
25/04/26 08:42:57 INFO TaskSetManager: Finished task 46.0 in stage 0.0 (TID 46) in 517 ms on 8bd937941c6d (execu
tor driver) (47/128)
25/04/26 08:42:57 INFO PythonRunner: Times: total = 554, boot = 210, init = 106, finish = 238
25/04/26 08:42:57 INFO Executor: Finished task 47.0 in stage 0.0 (TID 47). 1397 bytes result sent to driver
25/04/26 08:42:57 INFO TaskSetManager: Starting task 49.0 in stage 0.0 (TID 49) (8bd937941c6d, executor driver,
partition 49, PROCESS_LOCAL, 10108 bytes)
25/04/26 08:42:57 INFO Executor: Running task 49.0 in stage 0.0 (TID 49)
25/04/26 08:42:57 INFO TaskSetManager: Finished task 47.0 in stage 0.0 (TID 47) in 576 ms on 8bd937941c6d (execu
tor driver) (48/128)
25/04/26 08:42:58 INFO PythonRunner: Times: total = 510, boot = 206, init = 61, finish = 243
25/04/26 08:42:58 INFO Executor: Finished task 49.0 in stage 0.0 (TID 49). 1397 bytes result sent to driver
25/04/26 08:42:58 INFO TaskSetManager: Starting task 50.0 in stage 0.0 (TID 50) (8bd937941c6d, executor driver,
partition 50, PROCESS_LOCAL, 10113 bytes)
25/04/26 08:42:58 INFO Executor: Running task 50.0 in stage 0.0 (TID 50)
25/04/26 08:42:58 INFO TaskSetManager: Finished task 49.0 in stage 0.0 (TID 49) in 520 ms on 8bd937941c6d (execu
tor driver) (49/128)
25/04/26 08:42:58 INFO PythonRunner: Times: total = 975, boot = 238, init = 62, finish = 675
25/04/26 08:42:58 INFO Executor: Finished task 48.0 in stage 0.0 (TID 48). 1397 bytes result sent to driver
25/04/26 08:42:58 INFO TaskSetManager: Starting task 51.0 in stage 0.0 (TID 51) (8bd937941c6d, executor driver,
partition 51, PROCESS_LOCAL, 10121 bytes)
25/04/26 08:42:58 INFO TaskSetManager: Finished task 48.0 in stage 0.0 (TID 48) in 986 ms on 8bd937941c6d (execu
tor driver) (50/128)
25/04/26 08:42:58 INFO Executor: Running task 51.0 in stage 0.0 (TID 51)
25/04/26 08:42:58 INFO PythonRunner: Times: total = 555, boot = 211, init = 95, finish = 249
25/04/26 08:42:58 INFO Executor: Finished task 50.0 in stage 0.0 (TID 50). 1397 bytes result sent to driver
25/04/26 08:42:58 INFO TaskSetManager: Starting task 52.0 in stage 0.0 (TID 52) (8bd937941c6d, executor driver,
partition 52, PROCESS_LOCAL, 10118 bytes)
25/04/26 08:42:58 INFO TaskSetManager: Finished task 50.0 in stage 0.0 (TID 50) in 570 ms on 8bd937941c6d (execu
tor driver) (51/128)
25/04/26 08:42:58 INFO Executor: Running task 52.0 in stage 0.0 (TID 52)
25/04/26 08:42:58 INFO PythonRunner: Times: total = 547, boot = 248, init = 65, finish = 234
25/04/26 08:42:58 INFO Executor: Finished task 51.0 in stage 0.0 (TID 51). 1440 bytes result sent to driver
25/04/26 08:42:58 INFO TaskSetManager: Starting task 53.0 in stage 0.0 (TID 53) (8bd937941c6d, executor driver,
partition 53, PROCESS_LOCAL, 10079 bytes)
25/04/26 08:42:58 INFO TaskSetManager: Finished task 51.0 in stage 0.0 (TID 51) in 560 ms on 8bd937941c6d (execu
tor driver) (52/128)
25/04/26 08:42:58 INFO Executor: Running task 53.0 in stage 0.0 (TID 53)
25/04/26 08:42:59 INFO PythonRunner: Times: total = 640, boot = 212, init = 97, finish = 331
25/04/26 08:42:59 INFO Executor: Finished task 52.0 in stage 0.0 (TID 52). 1397 bytes result sent to driver
25/04/26 08:42:59 INFO TaskSetManager: Starting task 54.0 in stage 0.0 (TID 54) (8bd937941c6d, executor driver,
partition 54, PROCESS_LOCAL, 10031 bytes)
25/04/26 08:42:59 INFO TaskSetManager: Finished task 52.0 in stage 0.0 (TID 52) in 651 ms on 8bd937941c6d (execu
tor driver) (53/128)
25/04/26 08:42:59 INFO Executor: Running task 54.0 in stage 0.0 (TID 54)
25/04/26 08:42:59 INFO PythonRunner: Times: total = 610, boot = 278, init = 70, finish = 262
25/04/26 08:42:59 INFO Executor: Finished task 53.0 in stage 0.0 (TID 53). 1397 bytes result sent to driver
25/04/26 08:42:59 INFO TaskSetManager: Starting task 55.0 in stage 0.0 (TID 55) (8bd937941c6d, executor driver,
partition 55, PROCESS_LOCAL, 10020 bytes)
25/04/26 08:42:59 INFO TaskSetManager: Finished task 53.0 in stage 0.0 (TID 53) in 620 ms on 8bd937941c6d (execu
tor driver) (54/128)
25/04/26 08:42:59 INFO Executor: Running task 55.0 in stage 0.0 (TID 55)
25/04/26 08:43:00 INFO PythonRunner: Times: total = 548, boot = 250, init = 73, finish = 225
25/04/26 08:43:00 INFO Executor: Finished task 55.0 in stage 0.0 (TID 55). 1397 bytes result sent to driver
25/04/26 08:43:00 INFO TaskSetManager: Starting task 56.0 in stage 0.0 (TID 56) (8bd937941c6d, executor driver,
partition 56, PROCESS_LOCAL, 10027 bytes)
25/04/26 08:43:00 INFO TaskSetManager: Finished task 55.0 in stage 0.0 (TID 55) in 561 ms on 8bd937941c6d (execu
tor driver) (55/128)
25/04/26 08:43:00 INFO Executor: Running task 56.0 in stage 0.0 (TID 56)
25/04/26 08:43:00 INFO PythonRunner: Times: total = 967, boot = 235, init = 93, finish = 639
25/04/26 08:43:00 INFO Executor: Finished task 54.0 in stage 0.0 (TID 54). 1397 bytes result sent to driver
25/04/26 08:43:00 INFO TaskSetManager: Starting task 57.0 in stage 0.0 (TID 57) (8bd937941c6d, executor driver,
partition 57, PROCESS_LOCAL, 10039 bytes)
25/04/26 08:43:00 INFO TaskSetManager: Finished task 54.0 in stage 0.0 (TID 54) in 978 ms on 8bd937941c6d (execu
tor driver) (56/128)
25/04/26 08:43:00 INFO Executor: Running task 57.0 in stage 0.0 (TID 57)
25/04/26 08:43:00 INFO PythonRunner: Times: total = 512, boot = 196, init = 73, finish = 243
25/04/26 08:43:00 INFO Executor: Finished task 56.0 in stage 0.0 (TID 56). 1440 bytes result sent to driver
25/04/26 08:43:00 INFO TaskSetManager: Starting task 58.0 in stage 0.0 (TID 58) (8bd937941c6d, executor driver,
partition 58, PROCESS_LOCAL, 10032 bytes)
25/04/26 08:43:00 INFO Executor: Running task 58.0 in stage 0.0 (TID 58)
25/04/26 08:43:00 INFO TaskSetManager: Finished task 56.0 in stage 0.0 (TID 56) in 523 ms on 8bd937941c6d (execu
tor driver) (57/128)
25/04/26 08:43:00 INFO PythonRunner: Times: total = 536, boot = 189, init = 100, finish = 247
25/04/26 08:43:01 INFO Executor: Finished task 57.0 in stage 0.0 (TID 57). 1397 bytes result sent to driver
25/04/26 08:43:01 INFO TaskSetManager: Starting task 59.0 in stage 0.0 (TID 59) (8bd937941c6d, executor driver,
partition 59, PROCESS_LOCAL, 10025 bytes)
25/04/26 08:43:01 INFO TaskSetManager: Finished task 57.0 in stage 0.0 (TID 57) in 547 ms on 8bd937941c6d (execu
tor driver) (58/128)
25/04/26 08:43:01 INFO Executor: Running task 59.0 in stage 0.0 (TID 59)
25/04/26 08:43:01 INFO PythonRunner: Times: total = 661, boot = 248, init = 64, finish = 349
```

```
25/04/26 08:43:01 INFO Executor: Finished task 58.0 in stage 0.0 (TID 58). 1397 bytes result sent to driver
25/04/26 08:43:01 INFO TaskSetManager: Starting task 60.0 in stage 0.0 (TID 60) (8bd937941c6d, executor driver,
partition 60, PROCESS_LOCAL, 10032 bytes)
25/04/26 08:43:01 INFO TaskSetManager: Finished task 58.0 in stage 0.0 (TID 58) in 679 ms on 8bd937941c6d (execu
tor driver) (59/128)
25/04/26 08:43:01 INFO Executor: Running task 60.0 in stage 0.0 (TID 60)
25/04/26 08:43:01 INFO PythonRunner: Times: total = 510, boot = 204, init = 70, finish = 236
25/04/26 08:43:01 INFO Executor: Finished task 59.0 in stage 0.0 (TID 59). 1397 bytes result sent to driver
25/04/26 08:43:01 INFO TaskSetManager: Starting task 61.0 in stage 0.0 (TID 61) (8bd937941c6d, executor driver,
partition 61, PROCESS_LOCAL, 10014 bytes)
25/04/26 08:43:01 INFO Executor: Running task 61.0 in stage 0.0 (TID 61)
25/04/26 08:43:01 INFO TaskSetManager: Finished task 59.0 in stage 0.0 (TID 59) in 523 ms on 8bd937941c6d (execu
tor driver) (60/128)
25/04/26 08:43:01 INFO PythonRunner: Times: total = 570, boot = 209, init = 109, finish = 252
25/04/26 08:43:01 INFO Executor: Finished task 60.0 in stage 0.0 (TID 60). 1397 bytes result sent to driver
25/04/26 08:43:01 INFO TaskSetManager: Starting task 62.0 in stage 0.0 (TID 62) (8bd937941c6d, executor driver,
partition 62, PROCESS_LOCAL, 10002 bytes)
25/04/26 08:43:01 INFO TaskSetManager: Finished task 60.0 in stage 0.0 (TID 60) in 583 ms on 8bd937941c6d (execu
tor driver) (61/128)
25/04/26 08:43:01 INFO Executor: Running task 62.0 in stage 0.0 (TID 62)
25/04/26 08:43:02 INFO PythonRunner: Times: total = 541, boot = 235, init = 63, finish = 243
25/04/26 08:43:02 INFO Executor: Finished task 61.0 in stage 0.0 (TID 61). 1397 bytes result sent to driver
25/04/26 08:43:02 INFO TaskSetManager: Starting task 63.0 in stage 0.0 (TID 63) (8bd937941c6d, executor driver,
partition 63, PROCESS_LOCAL, 12416 bytes)
25/04/26 08:43:02 INFO Executor: Running task 63.0 in stage 0.0 (TID 63)
25/04/26 08:43:02 INFO TaskSetManager: Finished task 61.0 in stage 0.0 (TID 61) in 550 ms on 8bd937941c6d (execu
tor driver) (62/128)
25/04/26 08:43:02 INFO PythonRunner: Times: total = 694, boot = 195, init = 100, finish = 399
25/04/26 08:43:02 INFO Executor: Finished task 62.0 in stage 0.0 (TID 62). 1397 bytes result sent to driver
25/04/26 08:43:02 INFO TaskSetManager: Starting task 64.0 in stage 0.0 (TID 64) (8bd937941c6d, executor driver,
partition 64, PROCESS_LOCAL, 10012 bytes)
25/04/26 08:43:02 INFO Executor: Running task 64.0 in stage 0.0 (TID 64)
25/04/26 08:43:02 INFO TaskSetManager: Finished task 62.0 in stage 0.0 (TID 62) in 705 ms on 8bd937941c6d (execu
tor driver) (63/128)
25/04/26 08:43:02 INFO PythonRunner: Times: total = 754, boot = 227, init = 86, finish = 441
25/04/26 08:43:02 INFO Executor: Finished task 63.0 in stage 0.0 (TID 63). 1397 bytes result sent to driver
25/04/26 08:43:02 INFO TaskSetManager: Starting task 65.0 in stage 0.0 (TID 65) (8bd937941c6d, executor driver,
partition 65, PROCESS_LOCAL, 10003 bytes)
25/04/26 08:43:02 INFO Executor: Running task 65.0 in stage 0.0 (TID 65)
25/04/26 08:43:02 INFO TaskSetManager: Finished task 63.0 in stage 0.0 (TID 63) in 767 ms on 8bd937941c6d (execu
tor driver) (64/128)
25/04/26 08:43:03 INFO PythonRunner: Times: total = 663, boot = 209, init = 202, finish = 252
25/04/26 08:43:03 INFO Executor: Finished task 64.0 in stage 0.0 (TID 64). 1397 bytes result sent to driver
25/04/26 08:43:03 INFO TaskSetManager: Starting task 66.0 in stage 0.0 (TID 66) (8bd937941c6d, executor driver,
partition 66, PROCESS_LOCAL, 9999 bytes)
25/04/26 08:43:03 INFO Executor: Running task 66.0 in stage 0.0 (TID 66)
25/04/26 08:43:03 INFO TaskSetManager: Finished task 64.0 in stage 0.0 (TID 64) in 678 ms on 8bd937941c6d (execu
tor driver) (65/128)
25/04/26 08:43:03 INFO PythonRunner: Times: total = 917, boot = 290, init = 106, finish = 521
25/04/26 08:43:03 INFO Executor: Finished task 65.0 in stage 0.0 (TID 65). 1397 bytes result sent to driver
25/04/26 08:43:03 INFO TaskSetManager: Starting task 67.0 in stage 0.0 (TID 67) (8bd937941c6d, executor driver,
partition 67, PROCESS_LOCAL, 10008 bytes)
25/04/26 08:43:03 INFO TaskSetManager: Finished task 65.0 in stage 0.0 (TID 65) in 939 ms on 8bd937941c6d (execu
tor driver) (66/128)
25/04/26 08:43:03 INFO Executor: Running task 67.0 in stage 0.0 (TID 67)
25/04/26 08:43:04 INFO PythonRunner: Times: total = 817, boot = 299, init = 129, finish = 389
25/04/26 08:43:04 INFO Executor: Finished task 66.0 in stage 0.0 (TID 66). 1397 bytes result sent to driver
25/04/26 08:43:04 INFO TaskSetManager: Starting task 68.0 in stage 0.0 (TID 68) (8bd937941c6d, executor driver,
partition 68, PROCESS_LOCAL, 10004 bytes)
25/04/26 08:43:04 INFO TaskSetManager: Finished task 66.0 in stage 0.0 (TID 66) in 827 ms on 8bd937941c6d (execu
tor driver) (67/128)
25/04/26 08:43:04 INFO Executor: Running task 68.0 in stage 0.0 (TID 68)
25/04/26 08:43:04 INFO PythonRunner: Times: total = 720, boot = 273, init = 199, finish = 248
25/04/26 08:43:04 INFO Executor: Finished task 67.0 in stage 0.0 (TID 67). 1397 bytes result sent to driver
25/04/26 08:43:04 INFO TaskSetManager: Starting task 69.0 in stage 0.0 (TID 69) (8bd937941c6d, executor driver,
partition 69, PROCESS_LOCAL, 10001 bytes)
25/04/26 08:43:04 INFO TaskSetManager: Finished task 67.0 in stage 0.0 (TID 67) in 736 ms on 8bd937941c6d (execu
tor driver) (68/128)
25/04/26 08:43:04 INFO Executor: Running task 69.0 in stage 0.0 (TID 69)
25/04/26 08:43:04 INFO PythonRunner: Times: total = 741, boot = 363, init = 118, finish = 260
25/04/26 08:43:04 INFO Executor: Finished task 68.0 in stage 0.0 (TID 68). 1397 bytes result sent to driver
25/04/26 08:43:04 INFO TaskSetManager: Starting task 70.0 in stage 0.0 (TID 70) (8bd937941c6d, executor driver,
partition 70, PROCESS_LOCAL, 10009 bytes)
25/04/26 08:43:04 INFO TaskSetManager: Finished task 68.0 in stage 0.0 (TID 68) in 765 ms on 8bd937941c6d (execu
tor driver) (69/128)
25/04/26 08:43:04 INFO Executor: Running task 70.0 in stage 0.0 (TID 70)
25/04/26 08:43:05 INFO PythonRunner: Times: total = 780, boot = 314, init = 202, finish = 264
25/04/26 08:43:05 INFO Executor: Finished task 69.0 in stage 0.0 (TID 69). 1397 bytes result sent to driver
25/04/26 08:43:05 INFO TaskSetManager: Starting task 71.0 in stage 0.0 (TID 71) (8bd937941c6d, executor driver,
partition 71, PROCESS_LOCAL, 10010 bytes)
25/04/26 08:43:05 INFO TaskSetManager: Finished task 69.0 in stage 0.0 (TID 69) in 801 ms on 8bd937941c6d (execu
tor driver) (70/128)
25/04/26 08:43:05 INFO Executor: Running task 71.0 in stage 0.0 (TID 71)
```

```
25/04/26 08:43:05 INFO PythonRunner: Times: total = 707, boot = 321, init = 110, finish = 276
25/04/26 08:43:05 INFO Executor: Finished task 70.0 in stage 0.0 (TID 70). 1397 bytes result sent to driver
25/04/26 08:43:05 INFO TaskSetManager: Starting task 72.0 in stage 0.0 (TID 72) (8bd937941c6d, executor driver,
partition 72, PROCESS_LOCAL, 10012 bytes)
25/04/26 08:43:05 INFO Executor: Running task 72.0 in stage 0.0 (TID 72)
25/04/26 08:43:05 INFO TaskSetManager: Finished task 70.0 in stage 0.0 (TID 70) in 737 ms on 8bd937941c6d (execu
tor driver) (71/128)
25/04/26 08:43:05 INFO PythonRunner: Times: total = 596, boot = 249, init = 101, finish = 246
25/04/26 08:43:05 INFO Executor: Finished task 71.0 in stage 0.0 (TID 71). 1397 bytes result sent to driver
25/04/26 08:43:05 INFO TaskSetManager: Starting task 73.0 in stage 0.0 (TID 73) (8bd937941c6d, executor driver,
partition 73, PROCESS_LOCAL, 10000 bytes)
25/04/26 08:43:05 INFO Executor: Running task 73.0 in stage 0.0 (TID 73)
25/04/26 08:43:05 INFO TaskSetManager: Finished task 71.0 in stage 0.0 (TID 71) in 621 ms on 8bd937941c6d (execu
tor driver) (72/128)
25/04/26 08:43:06 INFO PythonRunner: Times: total = 761, boot = 255, init = 64, finish = 442
25/04/26 08:43:06 INFO Executor: Finished task 72.0 in stage 0.0 (TID 72). 1397 bytes result sent to driver
25/04/26 08:43:06 INFO TaskSetManager: Starting task 74.0 in stage 0.0 (TID 74) (8bd937941c6d, executor driver,
partition 74, PROCESS_LOCAL, 10002 bytes)
25/04/26 08:43:06 INFO Executor: Running task 74.0 in stage 0.0 (TID 74)
25/04/26 08:43:06 INFO TaskSetManager: Finished task 72.0 in stage 0.0 (TID 72) in 773 ms on 8bd937941c6d (execu
tor driver) (73/128)
25/04/26 08:43:06 INFO PythonRunner: Times: total = 637, boot = 338, init = 63, finish = 236
25/04/26 08:43:06 INFO Executor: Finished task 73.0 in stage 0.0 (TID 73). 1397 bytes result sent to driver
25/04/26 08:43:06 INFO TaskSetManager: Starting task 75.0 in stage 0.0 (TID 75) (8bd937941c6d, executor driver,
partition 75, PROCESS_LOCAL, 10078 bytes)
25/04/26 08:43:06 INFO TaskSetManager: Finished task 73.0 in stage 0.0 (TID 73) in 646 ms on 8bd937941c6d (execu
tor driver) (74/128)
25/04/26 08:43:06 INFO Executor: Running task 75.0 in stage 0.0 (TID 75)
25/04/26 08:43:07 INFO PythonRunner: Times: total = 703, boot = 201, init = 107, finish = 395
25/04/26 08:43:07 INFO Executor: Finished task 74.0 in stage 0.0 (TID 74). 1397 bytes result sent to driver
25/04/26 08:43:07 INFO TaskSetManager: Starting task 76.0 in stage 0.0 (TID 76) (8bd937941c6d, executor driver,
partition 76, PROCESS_LOCAL, 10166 bytes)
25/04/26 08:43:07 INFO Executor: Running task 76.0 in stage 0.0 (TID 76)
25/04/26 08:43:07 INFO TaskSetManager: Finished task 74.0 in stage 0.0 (TID 74) in 709 ms on 8bd937941c6d (execu
tor driver) (75/128)
25/04/26 08:43:07 INFO PythonRunner: Times: total = 762, boot = 369, init = 65, finish = 328
25/04/26 08:43:07 INFO Executor: Finished task 75.0 in stage 0.0 (TID 75). 1397 bytes result sent to driver
25/04/26 08:43:07 INFO TaskSetManager: Starting task 77.0 in stage 0.0 (TID 77) (8bd937941c6d, executor driver,
partition 77, PROCESS_LOCAL, 10173 bytes)
25/04/26 08:43:07 INFO TaskSetManager: Finished task 75.0 in stage 0.0 (TID 75) in 769 ms on 8bd937941c6d (execu
tor driver) (76/128)
25/04/26 08:43:07 INFO Executor: Running task 77.0 in stage 0.0 (TID 77)
25/04/26 08:43:07 INFO PythonRunner: Times: total = 841, boot = 205, init = 75, finish = 561
25/04/26 08:43:07 INFO Executor: Finished task 76.0 in stage 0.0 (TID 76). 1397 bytes result sent to driver
25/04/26 08:43:07 INFO TaskSetManager: Starting task 78.0 in stage 0.0 (TID 78) (8bd937941c6d, executor driver,
partition 78, PROCESS_LOCAL, 10191 bytes)
25/04/26 08:43:07 INFO TaskSetManager: Finished task 76.0 in stage 0.0 (TID 76) in 858 ms on 8bd937941c6d (execu
tor driver) (77/128)
25/04/26 08:43:07 INFO Executor: Running task 78.0 in stage 0.0 (TID 78)
25/04/26 08:43:08 INFO PythonRunner: Times: total = 876, boot = 350, init = 74, finish = 452
25/04/26 08:43:08 INFO Executor: Finished task 77.0 in stage 0.0 (TID 77). 1397 bytes result sent to driver
25/04/26 08:43:08 INFO TaskSetManager: Starting task 79.0 in stage 0.0 (TID 79) (8bd937941c6d, executor driver,
partition 79, PROCESS_LOCAL, 10183 bytes)
25/04/26 08:43:08 INFO TaskSetManager: Finished task 77.0 in stage 0.0 (TID 77) in 887 ms on 8bd937941c6d (execu
tor driver) (78/128)
25/04/26 08:43:08 INFO Executor: Running task 79.0 in stage 0.0 (TID 79)
25/04/26 08:43:08 INFO PythonRunner: Times: total = 665, boot = 206, init = 67, finish = 392
25/04/26 08:43:08 INFO Executor: Finished task 78.0 in stage 0.0 (TID 78). 1397 bytes result sent to driver
25/04/26 08:43:08 INFO TaskSetManager: Starting task 80.0 in stage 0.0 (TID 80) (8bd937941c6d, executor driver,
partition 80, PROCESS_LOCAL, 10154 bytes)
25/04/26 08:43:08 INFO TaskSetManager: Finished task 78.0 in stage 0.0 (TID 78) in 677 ms on 8bd937941c6d (execu
tor driver) (79/128)
25/04/26 08:43:08 INFO Executor: Running task 80.0 in stage 0.0 (TID 80)
25/04/26 08:43:08 INFO PythonRunner: Times: total = 714, boot = 198, init = 70, finish = 446
25/04/26 08:43:08 INFO Executor: Finished task 79.0 in stage 0.0 (TID 79). 1397 bytes result sent to driver
25/04/26 08:43:08 INFO TaskSetManager: Starting task 81.0 in stage 0.0 (TID 81) (8bd937941c6d, executor driver,
partition 81, PROCESS_LOCAL, 10162 bytes)
25/04/26 08:43:08 INFO TaskSetManager: Finished task 79.0 in stage 0.0 (TID 79) in 726 ms on 8bd937941c6d (execu
tor driver) (80/128)
25/04/26 08:43:08 INFO Executor: Running task 81.0 in stage 0.0 (TID 81)
25/04/26 08:43:09 INFO PythonRunner: Times: total = 497, boot = 202, init = 60, finish = 235
25/04/26 08:43:09 INFO Executor: Finished task 80.0 in stage 0.0 (TID 80). 1397 bytes result sent to driver
25/04/26 08:43:09 INFO TaskSetManager: Starting task 82.0 in stage 0.0 (TID 82) (8bd937941c6d, executor driver,
partition 82, PROCESS_LOCAL, 10154 bytes)
25/04/26 08:43:09 INFO Executor: Running task 82.0 in stage 0.0 (TID 82)
25/04/26 08:43:09 INFO TaskSetManager: Finished task 80.0 in stage 0.0 (TID 80) in 513 ms on 8bd937941c6d (execu
tor driver) (81/128)
25/04/26 08:43:09 INFO PythonRunner: Times: total = 524, boot = 196, init = 97, finish = 231
25/04/26 08:43:09 INFO Executor: Finished task 81.0 in stage 0.0 (TID 81). 1397 bytes result sent to driver
25/04/26 08:43:09 INFO TaskSetManager: Starting task 83.0 in stage 0.0 (TID 83) (8bd937941c6d, executor driver,
partition 83, PROCESS_LOCAL, 10173 bytes)
25/04/26 08:43:09 INFO TaskSetManager: Finished task 81.0 in stage 0.0 (TID 81) in 539 ms on 8bd937941c6d (execu
tor driver) (82/128)
```

```
25/04/26 08:43:09 INFO Executor: Running task 83.0 in stage 0.0 (TID 83)
25/04/26 08:43:09 INFO PythonRunner: Times: total = 673, boot = 258, init = 63, finish = 352
25/04/26 08:43:09 INFO Executor: Finished task 82.0 in stage 0.0 (TID 82). 1397 bytes result sent to driver
25/04/26 08:43:09 INFO TaskSetManager: Starting task 84.0 in stage 0.0 (TID 84) (8bd937941c6d, executor driver,
partition 84, PROCESS_LOCAL, 10143 bytes)
25/04/26 08:43:09 INFO TaskSetManager: Finished task 82.0 in stage 0.0 (TID 82) in 682 ms on 8bd937941c6d (execu
tor driver) (83/128)
25/04/26 08:43:09 INFO Executor: Running task 84.0 in stage 0.0 (TID 84)
25/04/26 08:43:09 INFO PythonRunner: Times: total = 507, boot = 197, init = 66, finish = 244
25/04/26 08:43:09 INFO Executor: Finished task 83.0 in stage 0.0 (TID 83). 1397 bytes result sent to driver
25/04/26 08:43:09 INFO TaskSetManager: Starting task 85.0 in stage 0.0 (TID 85) (8bd937941c6d, executor driver,
partition 85, PROCESS_LOCAL, 10135 bytes)
25/04/26 08:43:09 INFO TaskSetManager: Finished task 83.0 in stage 0.0 (TID 83) in 516 ms on 8bd937941c6d (execu
tor driver) (84/128)
25/04/26 08:43:09 INFO Executor: Running task 85.0 in stage 0.0 (TID 85)
25/04/26 08:43:10 INFO PythonRunner: Times: total = 513, boot = 195, init = 96, finish = 222
25/04/26 08:43:10 INFO Executor: Finished task 84.0 in stage 0.0 (TID 84). 1440 bytes result sent to driver
25/04/26 08:43:10 INFO TaskSetManager: Starting task 86.0 in stage 0.0 (TID 86) (8bd937941c6d, executor driver,
partition 86, PROCESS_LOCAL, 10150 bytes)
25/04/26 08:43:10 INFO TaskSetManager: Finished task 84.0 in stage 0.0 (TID 84) in 534 ms on 8bd937941c6d (execu
tor driver) (85/128)
25/04/26 08:43:10 INFO Executor: Running task 86.0 in stage 0.0 (TID 86)
25/04/26 08:43:10 INFO PythonRunner: Times: total = 513, boot = 222, init = 64, finish = 227
25/04/26 08:43:10 INFO Executor: Finished task 85.0 in stage 0.0 (TID 85). 1440 bytes result sent to driver
25/04/26 08:43:10 INFO TaskSetManager: Starting task 87.0 in stage 0.0 (TID 87) (8bd937941c6d, executor driver,
partition 87, PROCESS_LOCAL, 10147 bytes)
25/04/26 08:43:10 INFO TaskSetManager: Finished task 85.0 in stage 0.0 (TID 85) in 529 ms on 8bd937941c6d (execu
tor driver) (86/128)
25/04/26 08:43:10 INFO Executor: Running task 87.0 in stage 0.0 (TID 87)
25/04/26 08:43:10 INFO PythonRunner: Times: total = 558, boot = 200, init = 110, finish = 248
25/04/26 08:43:10 INFO Executor: Finished task 86.0 in stage 0.0 (TID 86). 1397 bytes result sent to driver
25/04/26 08:43:10 INFO TaskSetManager: Starting task 88.0 in stage 0.0 (TID 88) (8bd937941c6d, executor driver,
partition 88, PROCESS_LOCAL, 10135 bytes)
25/04/26 08:43:10 INFO Executor: Running task 88.0 in stage 0.0 (TID 88)
25/04/26 08:43:10 INFO TaskSetManager: Finished task 86.0 in stage 0.0 (TID 86) in 568 ms on 8bd937941c6d (execu
tor driver) (87/128)
25/04/26 08:43:11 INFO PythonRunner: Times: total = 713, boot = 244, init = 65, finish = 404
25/04/26 08:43:11 INFO Executor: Finished task 87.0 in stage 0.0 (TID 87). 1483 bytes result sent to driver
25/04/26 08:43:11 INFO TaskSetManager: Starting task 89.0 in stage 0.0 (TID 89) (8bd937941c6d, executor driver,
partition 89, PROCESS_LOCAL, 10155 bytes)
25/04/26 08:43:11 INFO TaskSetManager: Finished task 87.0 in stage 0.0 (TID 87) in 742 ms on 8bd937941c6d (execu
tor driver) (88/128)
25/04/26 08:43:11 INFO Executor: Running task 89.0 in stage 0.0 (TID 89)
25/04/26 08:43:11 INFO PythonRunner: Times: total = 741, boot = 217, init = 66, finish = 458
25/04/26 08:43:11 INFO Executor: Finished task 88.0 in stage 0.0 (TID 88). 1440 bytes result sent to driver
25/04/26 08:43:11 INFO TaskSetManager: Starting task 90.0 in stage 0.0 (TID 90) (8bd937941c6d, executor driver,
partition 90, PROCESS_LOCAL, 10143 bytes)
25/04/26 08:43:11 INFO Executor: Running task 90.0 in stage 0.0 (TID 90)
25/04/26 08:43:11 INFO TaskSetManager: Finished task 88.0 in stage 0.0 (TID 88) in 750 ms on 8bd937941c6d (execu
tor driver) (89/128)
25/04/26 08:43:11 INFO PythonRunner: Times: total = 661, boot = 189, init = 70, finish = 402
25/04/26 08:43:11 INFO Executor: Finished task 89.0 in stage 0.0 (TID 89). 1397 bytes result sent to driver
25/04/26 08:43:11 INFO TaskSetManager: Starting task 91.0 in stage 0.0 (TID 91) (8bd937941c6d, executor driver,
partition 91, PROCESS_LOCAL, 10133 bytes)
25/04/26 08:43:11 INFO TaskSetManager: Finished task 89.0 in stage 0.0 (TID 89) in 674 ms on 8bd937941c6d (execu
tor driver) (90/128)
25/04/26 08:43:11 INFO Executor: Running task 91.0 in stage 0.0 (TID 91)
25/04/26 08:43:12 INFO PythonRunner: Times: total = 703, boot = 218, init = 67, finish = 418
25/04/26 08:43:12 INFO Executor: Finished task 90.0 in stage 0.0 (TID 90). 1397 bytes result sent to driver
25/04/26 08:43:12 INFO TaskSetManager: Starting task 92.0 in stage 0.0 (TID 92) (8bd937941c6d, executor driver,
partition 92, PROCESS_LOCAL, 10153 bytes)
25/04/26 08:43:12 INFO TaskSetManager: Finished task 90.0 in stage 0.0 (TID 90) in 709 ms on 8bd937941c6d (execu
tor driver) (91/128)
25/04/26 08:43:12 INFO Executor: Running task 92.0 in stage 0.0 (TID 92)
25/04/26 08:43:12 INFO PythonRunner: Times: total = 519, boot = 208, init = 73, finish = 238
25/04/26 08:43:12 INFO Executor: Finished task 91.0 in stage 0.0 (TID 91). 1397 bytes result sent to driver
25/04/26 08:43:12 INFO TaskSetManager: Starting task 93.0 in stage 0.0 (TID 93) (8bd937941c6d, executor driver,
partition 93, PROCESS_LOCAL, 10147 bytes)
25/04/26 08:43:12 INFO TaskSetManager: Finished task 91.0 in stage 0.0 (TID 91) in 527 ms on 8bd937941c6d (execu
tor driver) (92/128)
25/04/26 08:43:12 INFO Executor: Running task 93.0 in stage 0.0 (TID 93)
25/04/26 08:43:12 INFO PythonRunner: Times: total = 545, boot = 215, init = 92, finish = 238
25/04/26 08:43:12 INFO Executor: Finished task 92.0 in stage 0.0 (TID 92). 1397 bytes result sent to driver
25/04/26 08:43:12 INFO TaskSetManager: Starting task 94.0 in stage 0.0 (TID 94) (8bd937941c6d, executor driver,
partition 94, PROCESS_LOCAL, 10149 bytes)
25/04/26 08:43:12 INFO TaskSetManager: Finished task 92.0 in stage 0.0 (TID 92) in 555 ms on 8bd937941c6d (execu
tor driver) (93/128)
25/04/26 08:43:12 INFO Executor: Running task 94.0 in stage 0.0 (TID 94)
25/04/26 08:43:13 INFO PythonRunner: Times: total = 551, boot = 244, init = 69, finish = 238
25/04/26 08:43:13 INFO Executor: Finished task 93.0 in stage 0.0 (TID 93). 1397 bytes result sent to driver
25/04/26 08:43:13 INFO TaskSetManager: Starting task 95.0 in stage 0.0 (TID 95) (8bd937941c6d, executor driver,
partition 95, PROCESS_LOCAL, 12693 bytes)
25/04/26 08:43:13 INFO Executor: Running task 95.0 in stage 0.0 (TID 95)
```

```
25/04/26 08:43:13 INFO TaskSetManager: Finished task 93.0 in stage 0.0 (TID 93) in 559 ms on 8bd937941c6d (execu
tor driver) (94/128)
25/04/26 08:43:13 INFO PythonRunner: Times: total = 665, boot = 210, init = 99, finish = 356
25/04/26 08:43:13 INFO Executor: Finished task 94.0 in stage 0.0 (TID 94). 1397 bytes result sent to driver
25/04/26 08:43:13 INFO TaskSetManager: Starting task 96.0 in stage 0.0 (TID 96) (8bd937941c6d, executor driver,
partition 96, PROCESS_LOCAL, 10150 bytes)
25/04/26 08:43:13 INFO Executor: Running task 96.0 in stage 0.0 (TID 96)
25/04/26 08:43:13 INFO TaskSetManager: Finished task 94.0 in stage 0.0 (TID 94) in 674 ms on 8bd937941c6d (execu
tor driver) (95/128)
25/04/26 08:43:13 INFO PythonRunner: Times: total = 785, boot = 255, init = 76, finish = 454
25/04/26 08:43:13 INFO Executor: Finished task 95.0 in stage 0.0 (TID 95). 1397 bytes result sent to driver
25/04/26 08:43:13 INFO TaskSetManager: Starting task 97.0 in stage 0.0 (TID 97) (8bd937941c6d, executor driver,
partition 97, PROCESS_LOCAL, 10140 bytes)
25/04/26 08:43:13 INFO Executor: Running task 97.0 in stage 0.0 (TID 97)
25/04/26 08:43:13 INFO TaskSetManager: Finished task 95.0 in stage 0.0 (TID 95) in 800 ms on 8bd937941c6d (execu
tor driver) (96/128)
25/04/26 08:43:14 INFO PythonRunner: Times: total = 503, boot = 197, init = 76, finish = 230
25/04/26 08:43:14 INFO Executor: Finished task 96.0 in stage 0.0 (TID 96). 1397 bytes result sent to driver
25/04/26 08:43:14 INFO TaskSetManager: Starting task 98.0 in stage 0.0 (TID 98) (8bd937941c6d, executor driver,
partition 98, PROCESS_LOCAL, 10149 bytes)
25/04/26 08:43:14 INFO TaskSetManager: Finished task 96.0 in stage 0.0 (TID 96) in 512 ms on 8bd937941c6d (execu
tor driver) (97/128)
25/04/26 08:43:14 INFO Executor: Running task 98.0 in stage 0.0 (TID 98)
25/04/26 08:43:14 INFO PythonRunner: Times: total = 709, boot = 205, init = 82, finish = 422
25/04/26 08:43:14 INFO Executor: Finished task 97.0 in stage 0.0 (TID 97). 1397 bytes result sent to driver
25/04/26 08:43:14 INFO TaskSetManager: Starting task 99.0 in stage 0.0 (TID 99) (8bd937941c6d, executor driver,
partition 99, PROCESS_LOCAL, 10084 bytes)
25/04/26 08:43:14 INFO Executor: Running task 99.0 in stage 0.0 (TID 99)
25/04/26 08:43:14 INFO TaskSetManager: Finished task 97.0 in stage 0.0 (TID 97) in 718 ms on 8bd937941c6d (execu
tor driver) (98/128)
25/04/26 08:43:14 INFO PythonRunner: Times: total = 623, boot = 201, init = 73, finish = 349
25/04/26 08:43:14 INFO Executor: Finished task 98.0 in stage 0.0 (TID 98). 1397 bytes result sent to driver
25/04/26 08:43:14 INFO TaskSetManager: Starting task 100.0 in stage 0.0 (TID 100) (8bd937941c6d, executor driver
, partition 100, PROCESS_LOCAL, 10030 bytes)
25/04/26 08:43:14 INFO TaskSetManager: Finished task 98.0 in stage 0.0 (TID 98) in 633 ms on 8bd937941c6d (execu
tor driver) (99/128)
25/04/26 08:43:14 INFO Executor: Running task 100.0 in stage 0.0 (TID 100)
25/04/26 08:43:15 INFO PythonRunner: Times: total = 537, boot = 193, init = 109, finish = 235
25/04/26 08:43:15 INFO Executor: Finished task 99.0 in stage 0.0 (TID 99). 1397 bytes result sent to driver
25/04/26 08:43:15 INFO TaskSetManager: Starting task 101.0 in stage 0.0 (TID 101) (8bd937941c6d, executor driver
, partition 101, PROCESS_LOCAL, 10049 bytes)
25/04/26 08:43:15 INFO TaskSetManager: Finished task 99.0 in stage 0.0 (TID 99) in 545 ms on 8bd937941c6d (execu
tor driver) (100/128)
25/04/26 08:43:15 INFO Executor: Running task 101.0 in stage 0.0 (TID 101)
25/04/26 08:43:15 INFO PythonRunner: Times: total = 714, boot = 254, init = 63, finish = 397
25/04/26 08:43:15 INFO Executor: Finished task 100.0 in stage 0.0 (TID 100). 1398 bytes result sent to driver
25/04/26 08:43:15 INFO TaskSetManager: Starting task 102.0 in stage 0.0 (TID 102) (8bd937941c6d, executor driver
, partition 102, PROCESS_LOCAL, 10052 bytes)
25/04/26 08:43:15 INFO TaskSetManager: Finished task 100.0 in stage 0.0 (TID 100) in 722 ms on 8bd937941c6d (exe
cutor driver) (101/128)
25/04/26 08:43:15 INFO Executor: Running task 102.0 in stage 0.0 (TID 102)
25/04/26 08:43:15 INFO PythonRunner: Times: total = 500, boot = 199, init = 62, finish = 239
25/04/26 08:43:15 INFO Executor: Finished task 101.0 in stage 0.0 (TID 101). 1398 bytes result sent to driver
25/04/26 08:43:15 INFO TaskSetManager: Starting task 103.0 in stage 0.0 (TID 103) (8bd937941c6d, executor driver
, partition 103, PROCESS_LOCAL, 10053 bytes)
25/04/26 08:43:15 INFO Executor: Running task 103.0 in stage 0.0 (TID 103)
25/04/26 08:43:15 INFO TaskSetManager: Finished task 101.0 in stage 0.0 (TID 101) in 527 ms on 8bd937941c6d (exe
cutor driver) (102/128)
25/04/26 08:43:16 INFO PythonRunner: Times: total = 958, boot = 267, init = 262, finish = 429
25/04/26 08:43:16 INFO Executor: Finished task 102.0 in stage 0.0 (TID 102). 1398 bytes result sent to driver
25/04/26 08:43:16 INFO TaskSetManager: Starting task 104.0 in stage 0.0 (TID 104) (8bd937941c6d, executor driver
, partition 104, PROCESS_LOCAL, 10071 bytes)
25/04/26 08:43:16 INFO Executor: Running task 104.0 in stage 0.0 (TID 104)
25/04/26 08:43:16 INFO TaskSetManager: Finished task 102.0 in stage 0.0 (TID 102) in 968 ms on 8bd937941c6d (exe
cutor driver) (103/128)
25/04/26 08:43:16 INFO PythonRunner: Times: total = 997, boot = 592, init = 150, finish = 255
25/04/26 08:43:16 INFO Executor: Finished task 103.0 in stage 0.0 (TID 103). 1398 bytes result sent to driver
25/04/26 08:43:16 INFO TaskSetManager: Starting task 105.0 in stage 0.0 (TID 105) (8bd937941c6d, executor driver
, partition 105, PROCESS_LOCAL, 10059 bytes)
25/04/26 08:43:16 INFO TaskSetManager: Finished task 103.0 in stage 0.0 (TID 103) in 1017 ms on 8bd937941c6d (ex
ecutor driver) (104/128)
25/04/26 08:43:16 INFO Executor: Running task 105.0 in stage 0.0 (TID 105)
25/04/26 08:43:17 INFO PythonRunner: Times: total = 706, boot = 277, init = 138, finish = 291
25/04/26 08:43:17 INFO Executor: Finished task 104.0 in stage 0.0 (TID 104). 1398 bytes result sent to driver
25/04/26 08:43:17 INFO TaskSetManager: Starting task 106.0 in stage 0.0 (TID 106) (8bd937941c6d, executor driver
, partition 106, PROCESS_LOCAL, 10045 bytes)
25/04/26 08:43:17 INFO TaskSetManager: Finished task 104.0 in stage 0.0 (TID 104) in 715 ms on 8bd937941c6d (exe
cutor driver) (105/128)
25/04/26 08:43:17 INFO Executor: Running task 106.0 in stage 0.0 (TID 106)
25/04/26 08:43:17 INFO PythonRunner: Times: total = 771, boot = 380, init = 138, finish = 253
25/04/26 08:43:17 INFO Executor: Finished task 105.0 in stage 0.0 (TID 105). 1398 bytes result sent to driver
25/04/26 08:43:17 INFO TaskSetManager: Starting task 107.0 in stage 0.0 (TID 107) (8bd937941c6d, executor driver
, partition 107, PROCESS_LOCAL, 10066 bytes)
```

```
25/04/26 08:43:17 INFO TaskSetManager: Finished task 105.0 in stage 0.0 (TID 105) in 789 ms on 8bd937941c6d (exe
cutor driver) (106/128)
25/04/26 08:43:17 INFO Executor: Running task 107.0 in stage 0.0 (TID 107)
25/04/26 08:43:18 INFO PythonRunner: Times: total = 896, boot = 325, init = 154, finish = 417
25/04/26 08:43:18 INFO Executor: Finished task 106.0 in stage 0.0 (TID 106). 1398 bytes result sent to driver
25/04/26 08:43:18 INFO TaskSetManager: Starting task 108.0 in stage 0.0 (TID 108) (8bd937941c6d, executor driver
, partition 108, PROCESS_LOCAL, 10055 bytes)
25/04/26 08:43:18 INFO TaskSetManager: Finished task 106.0 in stage 0.0 (TID 106) in 912 ms on 8bd937941c6d (exe
cutor driver) (107/128)
25/04/26 08:43:18 INFO Executor: Running task 108.0 in stage 0.0 (TID 108)
25/04/26 08:43:18 INFO PythonRunner: Times: total = 738, boot = 375, init = 126, finish = 237
25/04/26 08:43:18 INFO Executor: Finished task 107.0 in stage 0.0 (TID 107). 1398 bytes result sent to driver
25/04/26 08:43:18 INFO TaskSetManager: Starting task 109.0 in stage 0.0 (TID 109) (8bd937941c6d, executor driver
, partition 109, PROCESS_LOCAL, 10069 bytes)
25/04/26 08:43:18 INFO Executor: Running task 109.0 in stage 0.0 (TID 109)
25/04/26 08:43:18 INFO TaskSetManager: Finished task 107.0 in stage 0.0 (TID 107) in 767 ms on 8bd937941c6d (exe
cutor driver) (108/128)
25/04/26 08:43:18 INFO PythonRunner: Times: total = 796, boot = 336, init = 138, finish = 322
25/04/26 08:43:18 INFO Executor: Finished task 108.0 in stage 0.0 (TID 108). 1398 bytes result sent to driver
25/04/26 08:43:18 INFO TaskSetManager: Starting task 110.0 in stage 0.0 (TID 110) (8bd937941c6d, executor driver
, partition 110, PROCESS_LOCAL, 10065 bytes)
25/04/26 08:43:18 INFO TaskSetManager: Finished task 108.0 in stage 0.0 (TID 108) in 804 ms on 8bd937941c6d (exe
cutor driver) (109/128)
25/04/26 08:43:18 INFO Executor: Running task 110.0 in stage 0.0 (TID 110)
25/04/26 08:43:18 INFO PythonRunner: Times: total = 661, boot = 341, init = 69, finish = 251
25/04/26 08:43:18 INFO Executor: Finished task 109.0 in stage 0.0 (TID 109). 1398 bytes result sent to driver
25/04/26 08:43:18 INFO TaskSetManager: Starting task 111.0 in stage 0.0 (TID 111) (8bd937941c6d, executor driver
, partition 111, PROCESS_LOCAL, 10054 bytes)
25/04/26 08:43:18 INFO TaskSetManager: Finished task 109.0 in stage 0.0 (TID 109) in 685 ms on 8bd937941c6d (exe
cutor driver) (110/128)
25/04/26 08:43:18 INFO Executor: Running task 111.0 in stage 0.0 (TID 111)
25/04/26 08:43:19 INFO PythonRunner: Times: total = 618, boot = 266, init = 110, finish = 242
25/04/26 08:43:19 INFO Executor: Finished task 110.0 in stage 0.0 (TID 110). 1398 bytes result sent to driver
25/04/26 08:43:19 INFO TaskSetManager: Starting task 112.0 in stage 0.0 (TID 112) (8bd937941c6d, executor driver
, partition 112, PROCESS_LOCAL, 10037 bytes)
25/04/26 08:43:19 INFO TaskSetManager: Finished task 110.0 in stage 0.0 (TID 110) in 625 ms on 8bd937941c6d (exe
cutor driver) (111/128)
25/04/26 08:43:19 INFO Executor: Running task 112.0 in stage 0.0 (TID 112)
25/04/26 08:43:19 INFO PythonRunner: Times: total = 607, boot = 249, init = 110, finish = 248
25/04/26 08:43:19 INFO Executor: Finished task 111.0 in stage 0.0 (TID 111). 1398 bytes result sent to driver
25/04/26 08:43:19 INFO TaskSetManager: Starting task 113.0 in stage 0.0 (TID 113) (8bd937941c6d, executor driver
, partition 113, PROCESS_LOCAL, 10029 bytes)
25/04/26 08:43:19 INFO TaskSetManager: Finished task 111.0 in stage 0.0 (TID 111) in 617 ms on 8bd937941c6d (exe
cutor driver) (112/128)
25/04/26 08:43:19 INFO Executor: Running task 113.0 in stage 0.0 (TID 113)
25/04/26 08:43:20 INFO PythonRunner: Times: total = 592, boot = 238, init = 103, finish = 251
25/04/26 08:43:20 INFO Executor: Finished task 112.0 in stage 0.0 (TID 112). 1398 bytes result sent to driver
25/04/26 08:43:20 INFO TaskSetManager: Starting task 114.0 in stage 0.0 (TID 114) (8bd937941c6d, executor driver
, partition 114, PROCESS_LOCAL, 10043 bytes)
25/04/26 08:43:20 INFO TaskSetManager: Finished task 112.0 in stage 0.0 (TID 112) in 617 ms on 8bd937941c6d (exe
cutor driver) (113/128)
25/04/26 08:43:20 INFO Executor: Running task 114.0 in stage 0.0 (TID 114)
25/04/26 08:43:20 INFO PythonRunner: Times: total = 603, boot = 239, init = 101, finish = 263
25/04/26 08:43:20 INFO Executor: Finished task 113.0 in stage 0.0 (TID 113). 1398 bytes result sent to driver
25/04/26 08:43:20 INFO TaskSetManager: Starting task 115.0 in stage 0.0 (TID 115) (8bd937941c6d, executor driver
, partition 115, PROCESS_LOCAL, 10035 bytes)
25/04/26 08:43:20 INFO TaskSetManager: Finished task 113.0 in stage 0.0 (TID 113) in 631 ms on 8bd937941c6d (exe
cutor driver) (114/128)
25/04/26 08:43:20 INFO Executor: Running task 115.0 in stage 0.0 (TID 115)
25/04/26 08:43:20 INFO PythonRunner: Times: total = 567, boot = 229, init = 103, finish = 235
25/04/26 08:43:20 INFO Executor: Finished task 115.0 in stage 0.0 (TID 115). 1441 bytes result sent to driver
25/04/26 08:43:20 INFO TaskSetManager: Starting task 116.0 in stage 0.0 (TID 116) (8bd937941c6d, executor driver
, partition 116, PROCESS_LOCAL, 10029 bytes)
25/04/26 08:43:20 INFO TaskSetManager: Finished task 115.0 in stage 0.0 (TID 115) in 582 ms on 8bd937941c6d (exe
cutor driver) (115/128)
25/04/26 08:43:20 INFO Executor: Running task 116.0 in stage 0.0 (TID 116)
25/04/26 08:43:20 INFO PythonRunner: Times: total = 889, boot = 237, init = 104, finish = 548
25/04/26 08:43:20 INFO Executor: Finished task 114.0 in stage 0.0 (TID 114). 1398 bytes result sent to driver
25/04/26 08:43:20 INFO TaskSetManager: Starting task 117.0 in stage 0.0 (TID 117) (8bd937941c6d, executor driver
, partition 117, PROCESS_LOCAL, 10031 bytes)
25/04/26 08:43:20 INFO TaskSetManager: Finished task 114.0 in stage 0.0 (TID 114) in 896 ms on 8bd937941c6d (exe
cutor driver) (116/128)
25/04/26 08:43:20 INFO Executor: Running task 117.0 in stage 0.0 (TID 117)
25/04/26 08:43:21 INFO PythonRunner: Times: total = 664, boot = 199, init = 65, finish = 400
25/04/26 08:43:21 INFO Executor: Finished task 116.0 in stage 0.0 (TID 116). 1355 bytes result sent to driver
25/04/26 08:43:21 INFO TaskSetManager: Starting task 118.0 in stage 0.0 (TID 118) (8bd937941c6d, executor driver
, partition 118, PROCESS_LOCAL, 10012 bytes)
25/04/26 08:43:21 INFO TaskSetManager: Finished task 116.0 in stage 0.0 (TID 116) in 674 ms on 8bd937941c6d (exe
cutor driver) (117/128)
25/04/26 08:43:21 INFO Executor: Running task 118.0 in stage 0.0 (TID 118)
25/04/26 08:43:21 INFO PythonRunner: Times: total = 509, boot = 206, init = 67, finish = 236
25/04/26 08:43:21 INFO Executor: Finished task 117.0 in stage 0.0 (TID 117). 1398 bytes result sent to driver
25/04/26 08:43:21 INFO TaskSetManager: Starting task 119.0 in stage 0.0 (TID 119) (8bd937941c6d, executor driver
```

```
, partition 119, PROCESS_LOCAL, 10033 bytes)
25/04/26 08:43:21 INFO Executor: Running task 119.0 in stage 0.0 (TID 119)
25/04/26 08:43:21 INFO TaskSetManager: Finished task 117.0 in stage 0.0 (TID 117) in 529 ms on 8bd937941c6d (exe
cutor driver) (118/128)
25/04/26 08:43:21 INFO PythonRunner: Times: total = 549, boot = 216, init = 98, finish = 235
25/04/26 08:43:21 INFO Executor: Finished task 118.0 in stage 0.0 (TID 118). 1398 bytes result sent to driver
25/04/26 08:43:21 INFO TaskSetManager: Starting task 120.0 in stage 0.0 (TID 120) (8bd937941c6d, executor driver
, partition 120, PROCESS_LOCAL, 10025 bytes)
25/04/26 08:43:21 INFO Executor: Running task 120.0 in stage 0.0 (TID 120)
25/04/26 08:43:21 INFO TaskSetManager: Finished task 118.0 in stage 0.0 (TID 118) in 562 ms on 8bd937941c6d (exe
cutor driver) (119/128)
25/04/26 08:43:22 INFO PythonRunner: Times: total = 617, boot = 224, init = 72, finish = 321
25/04/26 08:43:22 INFO Executor: Finished task 119.0 in stage 0.0 (TID 119). 1398 bytes result sent to driver
25/04/26 08:43:22 INFO TaskSetManager: Starting task 121.0 in stage 0.0 (TID 121) (8bd937941c6d, executor driver
, partition 121, PROCESS_LOCAL, 10032 bytes)
25/04/26 08:43:22 INFO TaskSetManager: Finished task 119.0 in stage 0.0 (TID 119) in 627 ms on 8bd937941c6d (exe
cutor driver) (120/128)
25/04/26 08:43:22 INFO Executor: Running task 121.0 in stage 0.0 (TID 121)
25/04/26 08:43:22 INFO PythonRunner: Times: total = 709, boot = 198, init = 106, finish = 405
25/04/26 08:43:22 INFO Executor: Finished task 120.0 in stage 0.0 (TID 120). 1398 bytes result sent to driver
25/04/26 08:43:22 INFO TaskSetManager: Starting task 122.0 in stage 0.0 (TID 122) (8bd937941c6d, executor driver
, partition 122, PROCESS_LOCAL, 10037 bytes)
25/04/26 08:43:22 INFO TaskSetManager: Finished task 120.0 in stage 0.0 (TID 120) in 720 ms on 8bd937941c6d (exe
cutor driver) (121/128)
25/04/26 08:43:22 INFO Executor: Running task 122.0 in stage 0.0 (TID 122)
25/04/26 08:43:22 INFO PythonRunner: Times: total = 530, boot = 221, init = 82, finish = 227
25/04/26 08:43:22 INFO Executor: Finished task 121.0 in stage 0.0 (TID 121). 1398 bytes result sent to driver
25/04/26 08:43:22 INFO TaskSetManager: Starting task 123.0 in stage 0.0 (TID 123) (8bd937941c6d, executor driver
, partition 123, PROCESS_LOCAL, 10030 bytes)
25/04/26 08:43:22 INFO Executor: Running task 123.0 in stage 0.0 (TID 123)
25/04/26 08:43:22 INFO TaskSetManager: Finished task 121.0 in stage 0.0 (TID 121) in 544 ms on 8bd937941c6d (exe
cutor driver) (122/128)
25/04/26 08:43:23 INFO PythonRunner: Times: total = 597, boot = 251, init = 97, finish = 249
25/04/26 08:43:23 INFO Executor: Finished task 123.0 in stage 0.0 (TID 123). 1398 bytes result sent to driver
25/04/26 08:43:23 INFO TaskSetManager: Starting task 124.0 in stage 0.0 (TID 124) (8bd937941c6d, executor driver
, partition 124, PROCESS_LOCAL, 10036 bytes)
25/04/26 08:43:23 INFO TaskSetManager: Finished task 123.0 in stage 0.0 (TID 123) in 603 ms on 8bd937941c6d (exe
cutor driver) (123/128)
25/04/26 08:43:23 INFO Executor: Running task 124.0 in stage 0.0 (TID 124)
25/04/26 08:43:23 INFO PythonRunner: Times: total = 812, boot = 241, init = 108, finish = 463
25/04/26 08:43:23 INFO Executor: Finished task 122.0 in stage 0.0 (TID 122). 1398 bytes result sent to driver
25/04/26 08:43:23 INFO TaskSetManager: Starting task 125.0 in stage 0.0 (TID 125) (8bd937941c6d, executor driver
, partition 125, PROCESS_LOCAL, 10039 bytes)
25/04/26 08:43:23 INFO TaskSetManager: Finished task 122.0 in stage 0.0 (TID 122) in 823 ms on 8bd937941c6d (exe
cutor driver) (124/128)
25/04/26 08:43:23 INFO Executor: Running task 125.0 in stage 0.0 (TID 125)
25/04/26 08:43:23 INFO PythonRunner: Times: total = 720, boot = 210, init = 100, finish = 410
25/04/26 08:43:23 INFO Executor: Finished task 124.0 in stage 0.0 (TID 124). 1398 bytes result sent to driver
25/04/26 08:43:23 INFO TaskSetManager: Starting task 126.0 in stage 0.0 (TID 126) (8bd937941c6d, executor driver
, partition 126, PROCESS_LOCAL, 10039 bytes)
25/04/26 08:43:23 INFO TaskSetManager: Finished task 124.0 in stage 0.0 (TID 124) in 729 ms on 8bd937941c6d (exe
cutor driver) (125/128)
25/04/26 08:43:23 INFO Executor: Running task 126.0 in stage 0.0 (TID 126)
25/04/26 08:43:23 INFO PythonRunner: Times: total = 552, boot = 242, init = 73, finish = 237
25/04/26 08:43:23 INFO Executor: Finished task 125.0 in stage 0.0 (TID 125). 1398 bytes result sent to driver
25/04/26 08:43:23 INFO TaskSetManager: Starting task 127.0 in stage 0.0 (TID 127) (8bd937941c6d, executor driver
, partition 127, PROCESS_LOCAL, 10242 bytes)
25/04/26 08:43:23 INFO TaskSetManager: Finished task 125.0 in stage 0.0 (TID 125) in 563 ms on 8bd937941c6d (exe
cutor driver) (126/128)
25/04/26 08:43:23 INFO Executor: Running task 127.0 in stage 0.0 (TID 127)
25/04/26 08:43:24 INFO PythonRunner: Times: total = 595, boot = 274, init = 103, finish = 218
25/04/26 08:43:24 INFO Executor: Finished task 126.0 in stage 0.0 (TID 126). 1398 bytes result sent to driver
25/04/26 08:43:24 INFO TaskSetManager: Finished task 126.0 in stage 0.0 (TID 126) in 612 ms on 8bd937941c6d (exe
cutor driver) (127/128)
25/04/26 08:43:24 INFO PythonRunner: Times: total = 598, boot = 274, init = 104, finish = 220
25/04/26 08:43:24 INFO Executor: Finished task 127.0 in stage 0.0 (TID 127). 1398 bytes result sent to driver
25/04/26 08:43:24 INFO TaskSetManager: Finished task 127.0 in stage 0.0 (TID 127) in 605 ms on 8bd937941c6d (exe
cutor driver) (128/128)
25/04/26 08:43:24 INFO TaskSchedulerImpl: Removed TaskSet 0.0, whose tasks have all completed, from pool
25/04/26 08:43:24 INFO DAGScheduler: ResultStage 0 (collect at /root/spark_write_tfrec.py:61) finished in 54.742
s
25/04/26 08:43:24 INFO DAGScheduler: Job 0 is finished. Cancelling potential speculative or zombie tasks for thi
s job
25/04/26 08:43:24 INFO TaskSchedulerImpl: Killing all running tasks in stage 0: Stage finished
25/04/26 08:43:24 INFO DAGScheduler: Job 0 finished: collect at /root/spark_write_tfrec.py:61, took 54.899036 s
TFRecord files written:
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-00.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-01.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-02.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-03.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-04.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-05.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-06.tfrec
```

gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-07.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-08.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-09.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-10.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-11.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-12.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-13.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-14.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-15.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-16.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-17.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-18.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-19.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-20.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-21.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-22.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-23.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-24.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-25.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-26.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-27.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-28.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-29.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-30.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-31.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-32.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-33.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-34.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-35.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-36.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-37.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-38.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-39.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-40.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-41.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-42.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-43.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-44.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-45.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-46.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-47.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-48.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-49.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-50.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-51.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-52.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-53.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-54.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-55.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-56.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-57.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-58.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-59.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-60.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-61.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-62.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-63.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-64.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-65.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-66.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-67.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-68.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-69.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-70.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-71.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-72.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-73.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-74.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-75.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-76.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-77.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-78.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-79.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-80.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-81.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-82.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-83.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-84.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-85.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-86.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-87.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-88.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-89.tfrec

```
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-90.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-91.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-92.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-93.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-94.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-95.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-96.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-97.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-98.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-99.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-100.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-101.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-102.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-103.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-104.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-105.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-106.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-107.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-108.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-109.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-110.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-111.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-112.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-113.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-114.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-115.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-116.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-117.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-118.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-119.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-120.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-121.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-122.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-123.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-124.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-125.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-126.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-127.tfrec
25/04/26 08:43:26 INFO SparkContext: Invoking stop() from shutdown hook
25/04/26 08:43:26 INFO SparkContext: SparkContext is stopping with exitCode 0.
25/04/26 08:43:26 INFO SparkUI: Stopped Spark web UI at http://8bd937941c6d:4041
25/04/26 08:43:26 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
25/04/26 08:43:26 INFO MemoryStore: MemoryStore cleared
25/04/26 08:43:26 INFO BlockManager: BlockManager stopped
25/04/26 08:43:26 INFO BlockManagerMaster: BlockManagerMaster stopped
25/04/26 08:43:26 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
25/04/26 08:43:26 INFO SparkContext: Successfully stopped SparkContext
25/04/26 08:43:26 INFO ShutdownHookManager: Shutdown hook called
25/04/26 08:43:26 INFO ShutdownHookManager: Deleting directory /tmp/spark-53de0773-1eb7-4572-a882-f17029f690e5/p
yspark-db97fcfb-95e2-4ebf-8158-cd4c569f6e3b
25/04/26 08:43:26 INFO ShutdownHookManager: Deleting directory /tmp/spark-ecf1fcf2-9dc1-4cb9-937a-60997d5cbe43
25/04/26 08:43:26 INFO ShutdownHookManager: Deleting directory /tmp/spark-53de0773-1eb7-4572-a882-f17029f690e5
```

1c) Set up a cluster and run the script. (6%)

Following the example from the labs, set up a cluster to run PySpark jobs in the cloud. You need to set up so that TensorFlow is installed on all nodes in the cluster.

i) Single machine cluster

Set up a cluster with a single machine using the maximal SSD size (100) and 8 vCPUs.

Enable **package installation** by passing a flag `--initialization-actions` with argument `gs://goog-dataproc-initialization-actions-$REGION/python/pip-install.sh` (this is a public script that will read metadata to determine which packages to install). Then, the **packages are specified** by providing a `--metadata` flag with the argument `PIP_PACKAGES=tensorflow==2.4.0`.

Note: consider using `PIP_PACKAGES="tensorflow numpy"` or `PIP_PACKAGES=tensorflow` in case an older version of tensorflow is causing issues.

When the cluster is running, run your script to check that it works and keep the output cell output. (3%)

Run the script in the cloud and test the output.

In [31]:
```
# List all Compute Engine VM instances in the current Google Cloud project
!gcloud compute instances list

# List all Dataproc clusters in the current Google Cloud project
!gcloud dataproc clusters list
```

```
NAME                           ZONE           MACHINE_TYPE   PREEMPTIBLE  INTERNAL_IP  EXTERNAL_IP   STATUS
eng-throne-453420-e1-cluster-m  europe-west2-c  n1-standard-8               10.154.0.9   34.89.35.121  RUNNING
NAME                         PLATFORM  PRIMARY_WORKER_COUNT  SECONDARY_WORKER_COUNT  STATUS   ZONE            S
CHEDULED_DELETE  SCHEDULED_STOP
eng-throne-453420-e1-cluster  GCE                                                     RUNNING  europe-west2-c
```

In [26]:
```
# Delete any active clusters to create a new one
!gcloud dataproc clusters delete eng-throne-453420-e1-cluster --region=europe-west2 --quiet
```

```
Waiting on operation [projects/eng-throne-453420-e1/regions/europe-west2/operations/bd73e71a-adb9-3a3e-bbf2-afaa
b8039106].
Deleted [https://dataproc.googleapis.com/v1/projects/eng-throne-453420-e1/regions/europe-west2/clusters/eng-thro
ne-453420-e1-cluster].
Waiting on operation [projects/eng-throne-453420-e1/regions/europe-west2/operations/0f4a28d3-56c9-3b75-9b0d-9ba6
acbf6f31].
Deleted [https://dataproc.googleapis.com/v1/projects/eng-throne-453420-e1/regions/europe-west2/clusters/eng-thro
ne-453420-e1-maxcluster].
```

In [27]:
```
# Create a Dataproc single-node cluster with specified configuration
!gcloud dataproc clusters create eng-throne-453420-e1-cluster \
  --region=europe-west2 \
  --single-node \
  --master-machine-type=n1-standard-8 \
  --master-boot-disk-size=100GB \
  --image-version=2.0-debian10 \
  --initialization-actions=gs://goog-dataproc-initialization-actions-europe-west2/python/pip-install.sh \
  --metadata=PIP_PACKAGES="tensorflow numpy"
```

```
Waiting on operation [projects/eng-throne-453420-e1/regions/europe-west2/operations/057dad88-2e34-39fa-985d-dfc6
c9808eba].

WARNING: Don't create production clusters that reference initialization actions located in the gs://goog-datapro
c-initialization-actions-REGION public buckets. These scripts are provided as reference implementations, and the
y are synchronized with ongoing GitHub repository changes—a new version of a initialization action in public buc
kets may break your cluster creation. Instead, copy the following initialization actions from public buckets int
o your bucket : gs://goog-dataproc-initialization-actions-europe-west2/python/pip-install.sh
WARNING: For PD-Standard without local SSDs, we strongly recommend provisioning 1TB or larger to ensure consiste
ntly high I/O performance. See https://cloud.google.com/compute/docs/disks/performance for information on disk I
/O performance.
WARNING: The firewall rules for specified network or subnetwork would allow ingress traffic from 0.0.0.0/0, whic
h could be a security risk.
WARNING: The specified custom staging bucket 'dataproc-staging-europe-west2-410514942591-juafyoqs' is not using
uniform bucket level access IAM configuration. It is recommended to update bucket to enable the same. See https:
//cloud.google.com/storage/docs/uniform-bucket-level-access.
Created [https://dataproc.googleapis.com/v1/projects/eng-throne-453420-e1/regions/europe-west2/clusters/eng-thro
ne-453420-e1-cluster] Cluster placed in zone [europe-west2-c].
```

In [28]:
```
# Submit the spark_write_tfrec.py PySpark job to the Dataproc cluster
!gcloud dataproc jobs submit pyspark /content/drive/MyDrive/BD-CW/spark_write_tfrec.py \
  --cluster=eng-throne-453420-e1-cluster \
  --region=europe-west2
```

```
Job [d89cb6cadbc34229be496cd40925711c] submitted.
Waiting for job output...
2025-04-26 07:45:16.259235: I tensorflow/tsl/cuda/cudart_stub.cc:28] Could not find cuda drivers on your machine
, GPU will not be used.
2025-04-26 07:45:16.317219: I tensorflow/tsl/cuda/cudart_stub.cc:28] Could not find cuda drivers on your machine
, GPU will not be used.
2025-04-26 07:45:16.317854: I tensorflow/core/platform/cpu_feature_guard.cc:182] This TensorFlow binary is optim
ized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate com
piler flags.
2025-04-26 07:45:17.469079: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Could not fi
nd TensorRT
/opt/conda/default/lib/python3.8/site-packages/scipy/__init__.py:138: UserWarning: A NumPy version >=1.16.5 and
<1.23.0 is required for this version of SciPy (detected version 1.24.3)
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion} is required for this version of "
25/04/26 07:45:20 INFO org.apache.spark.SparkEnv: Registering MapOutputTracker
25/04/26 07:45:20 INFO org.apache.spark.SparkEnv: Registering BlockManagerMaster
25/04/26 07:45:21 INFO org.apache.spark.SparkEnv: Registering BlockManagerMasterHeartbeat
25/04/26 07:45:21 INFO org.apache.spark.SparkEnv: Registering OutputCommitCoordinator
25/04/26 07:45:21 INFO org.sparkproject.jetty.util.log: Logging initialized @7209ms to org.sparkproject.jetty.ut
il.log.Slf4jLog
25/04/26 07:45:21 INFO org.sparkproject.jetty.server.Server: jetty-9.4.40.v20210413; built: 2021-04-13T20:42:42.
668Z; git: b881a572662e1943a14ae12e7e1207989f218b74; jvm 1.8.0_442-b06
25/04/26 07:45:21 INFO org.sparkproject.jetty.server.Server: Started @7319ms
25/04/26 07:45:21 INFO org.sparkproject.jetty.server.AbstractConnector: Started ServerConnector@5fbc1c32{HTTP/1.
1, (http/1.1)}{0.0.0.0:39667}
25/04/26 07:45:22 INFO com.google.cloud.dataproc.DataprocSparkPlugin: Registered 128 driver metrics
25/04/26 07:45:23 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at eng-throne-453420
-e1-cluster-m/10.154.0.9:8032
25/04/26 07:45:23 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to Application History server at eng-t
hrone-453420-e1-cluster-m/10.154.0.9:10200
25/04/26 07:45:24 INFO org.apache.hadoop.conf.Configuration: resource-types.xml not found
25/04/26 07:45:24 INFO org.apache.hadoop.yarn.util.resource.ResourceUtils: Unable to find 'resource-types.xml'.
```

```
25/04/26 07:45:25 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitted application application_
1745653377215_0001
25/04/26 07:45:27 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at eng-throne-453420
-e1-cluster-m/10.154.0.9:8030
25/04/26 07:45:28 INFO com.google.cloud.hadoop.repackaged.gcs.com.google.cloud.hadoop.gcsio.GoogleCloudStorageIm
pl: Ignoring exception of type GoogleJsonResponseException; verified object already exists with desired state.
TFRecord files written:
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-00.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-01.tfrec
25/04/26 07:45:58 INFO org.sparkproject.jetty.server.AbstractConnector: Stopped Spark@5fbc1c32{HTTP/1.1, (http/1
.1)}{0.0.0.0:0}
25/04/26 07:45:58 INFO com.google.cloud.dataproc.DataprocSparkPlugin: Shutting down driver plugin. metrics=[file
s_created=1, gcs_api_server_not_implemented_error_count=0, gcs_api_server_timeout_count=0, action_http_post_requ
est_failures=0, op_get_list_status_result_size=0, op_open=0, gcs_api_client_unauthorized_response_count=0, actio
n_http_head_request_failures=0, stream_read_close_operations=0, stream_read_bytes_backwards_on_seek=0, exception
_count=13, gcs_api_total_request_count=11, op_create=1, gcs_api_client_bad_request_count=0, op_create_non_recurs
ive=0, gcs_api_client_gone_response_count=0, stream_write_operations=0, stream_read_operations=0, gcs_api_client
_request_timeout_count=0, op_rename=0, op_get_file_status=1, stream_read_total_bytes=0, op_glob_status=0, stream
_read_exceptions=0, action_http_get_request_failures=0, op_exists=0, stream_write_bytes=104815, op_xattr_list=0,
stream_write_exceptions=0, gcs_api_server_unavailable_count=0, directories_created=0, files_delete_rejected=0, o
p_xattr_get_named=0, op_hsync=0, stream_read_operations_incomplete=0, op_delete=0, stream_read_bytes=0, gcs_api_
client_non_found_response_count=6, gcs_api_client_requested_range_not_statisfiable_count=0, op_hflush=0, op_list
_status=0, op_xattr_get_named_map=0, gcs_api_client_side_error_count=20, op_get_file_checksum=0, action_http_del
ete_request_failures=0, gcs_api_server_internal_error_count=0, stream_read_seek_bytes_skipped=0, stream_write_cl
ose_operations=0, op_list_files=0, files_deleted=0, op_mkdirs=1, gcs_api_client_rate_limit_error_count=0, action
_http_put_request_failures=0, gcs_api_server_bad_gateway_count=0, stream_read_seek_backward_operations=0, gcs_ap
i_server_side_error_count=0, action_http_patch_request_failures=0, stream_read_seek_operations=0, stream_read_se
ek_forward_operations=0, gcs_api_client_precondition_failed_response_count=1, directories_deleted=0, op_xattr_ge
t_map=0, delegation_tokens_issued=0, op_create_min=72, op_delete_min=0, op_mkdirs_min=177, op_create_non_recursi
ve_min=0, op_glob_status_min=0, op_hsync_min=0, op_xattr_get_named_min=0, op_list_status_min=0, op_xattr_get_nam
ed_map_min=0, stream_read_close_operations_min=0, stream_read_operations_min=0, stream_read_seek_operations_min=
0, op_hflush_min=0, op_xattr_get_map_min=0, op_xattr_list_min=0, stream_write_operations_min=0, op_get_file_stat
us_min=187, op_open_min=0, op_rename_min=0, delegation_tokens_issued_min=0, stream_write_close_operations_min=0,
stream_read_close_operations_max=0, stream_read_operations_max=0, stream_read_seek_operations_max=0, op_hflush_m
ax=0, op_xattr_list_max=0, op_xattr_get_map_max=0, op_xattr_get_named_max=0, op_create_non_recursive_max=0, op_g
lob_status_max=0, op_get_file_status_max=187, stream_write_close_operations_max=0, op_open_max=0, delegation_tok
ens_issued_max=0, op_mkdirs_max=177, op_rename_max=0, op_create_max=72, op_delete_max=0, op_list_status_max=0, o
p_xattr_get_named_map_max=0, stream_write_operations_max=0, op_hsync_max=0, op_list_status_mean=0, stream_read_c
lose_operations_mean=0, op_open_mean=0, op_xattr_get_named_map_mean=0, op_xattr_list_mean=0, op_mkdirs_mean=177,
stream_write_close_operations_mean=0, op_rename_mean=0, op_hsync_mean=0, delegation_tokens_issued_mean=0, stream
_read_operations_mean=0, op_xattr_get_map_mean=0, op_create_mean=72, op_glob_status_mean=0, op_delete_mean=0, st
ream_read_seek_operations_mean=0, stream_write_operations_mean=0, op_create_non_recursive_mean=0, op_hflush_mean
=0, op_xattr_get_named_mean=0, op_get_file_status_mean=187, stream_write_operations_duration=0, stream_read_oper
ations_duration=0]
Job [d89cb6cadbc34229be496cd40925711c] finished successfully.
done: true
driverControlFilesUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo/5
7cd7b1e-235a-4440-b50b-0884f16de0a2/jobs/d89cb6cadbc34229be496cd40925711c/
driverOutputResourceUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo
/57cd7b1e-235a-4440-b50b-0884f16de0a2/jobs/d89cb6cadbc34229be496cd40925711c/driveroutput
jobUuid: 05eaf7f5-cc37-31be-8385-3e3b63ccebc7
placement:
  clusterName: eng-throne-453420-e1-cluster
  clusterUuid: 57cd7b1e-235a-4440-b50b-0884f16de0a2
pysparkJob:
  mainPythonFileUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo/57c
d7b1e-235a-4440-b50b-0884f16de0a2/jobs/d89cb6cadbc34229be496cd40925711c/staging/spark_write_tfrec.py
reference:
  jobId: d89cb6cadbc34229be496cd40925711c
  projectId: eng-throne-453420-e1
status:
  state: DONE
  stateStartTime: '2025-04-26T07:45:59.166279Z'
statusHistory:
- state: PENDING
  stateStartTime: '2025-04-26T07:45:12.729485Z'
- state: SETUP_DONE
  stateStartTime: '2025-04-26T07:45:12.789707Z'
- details: Agent reported job success
  state: RUNNING
  stateStartTime: '2025-04-26T07:45:13.127313Z'
yarnApplications:
- name: spark_write_tfrec.py
  progress: 1.0
  state: FINISHED
  trackingUrl: http://eng-throne-453420-e1-cluster-m:8088/proxy/application_1745653377215_0001/
```

In [29]:

```
GCS_OUTPUT = 'gs://eng-throne-453420-e1-storage/spark-output/'

filenames = tf.io.gfile.glob(GCS_OUTPUT + "*.tfrec")
datasetTfrec = load_dataset(filenames)
display_9_images_from_dataset(datasetTfrec)
```

In the free credit tier on Google Cloud, there are normally the following **restrictions** on compute machines:

- max 100GB of *SSD persistent disk*
- max 2000GB of *standard persistent disk*
- max 8 *vCPU*s
- no GPUs

See here for details The **disks are virtual** disks, where **I/O speed is limited in proportion to the size**, so we should allocate them evenly. This has mainly an effect on the **time the cluster needs to start**, as we are reading the data mainly from the bucket and we are not writing much to disk at all.

ii) Maximal cluster

Use the **largest possible cluster** within these constraints, i.e. **1 master and 7 worker nodes**. Each of them with 1 (virtual) CPU. The master should get the full *SSD* capacity and the 7 worker nodes should get equal shares of the *standard* disk capacity to maximise throughput.

Once the cluster is running, test your script. (3%)

In [37]:
```
# List all Compute Engine VM instances in the current Google Cloud project
!gcloud compute instances list

# List all Dataproc clusters in the current Google Cloud project
!gcloud dataproc clusters list
```

```
NAME                                ZONE            MACHINE_TYPE   PREEMPTIBLE  INTERNAL_IP  EXTERNAL_IP     ST
ATUS
eng-throne-453420-e1-maxcluster-m   europe-west2-c  e2-standard-2               10.154.0.11  34.105.246.17   RU
NNING
eng-throne-453420-e1-maxcluster-w-0  europe-west2-c  e2-standard-2               10.154.0.16  34.147.247.135  RU
NNING
eng-throne-453420-e1-maxcluster-w-1  europe-west2-c  e2-standard-2               10.154.0.17  34.142.33.221   RU
NNING
eng-throne-453420-e1-maxcluster-w-2  europe-west2-c  e2-standard-2               10.154.0.14  35.197.215.124  RU
NNING
eng-throne-453420-e1-maxcluster-w-3  europe-west2-c  e2-standard-2               10.154.0.13  35.197.226.248  RU
NNING
eng-throne-453420-e1-maxcluster-w-4  europe-west2-c  e2-standard-2               10.154.0.15  34.89.61.229    RU
NNING
eng-throne-453420-e1-maxcluster-w-5  europe-west2-c  e2-standard-2               10.154.0.12  34.89.35.121    RU
NNING
eng-throne-453420-e1-maxcluster-w-6  europe-west2-c  e2-standard-2               10.154.0.10  34.147.240.249  RU
NNING
NAME                          PLATFORM  PRIMARY_WORKER_COUNT  SECONDARY_WORKER_COUNT  STATUS   ZONE
SCHEDULED_DELETE  SCHEDULED_STOP
eng-throne-453420-e1-maxcluster  GCE       7                                       RUNNING  europe-west2-c
```

In [38]:
```
# Delete previous clusters
!gcloud dataproc clusters delete eng-throne-453420-e1-maxcluster --region=europe-west2 --quiet
```

```
Waiting on operation [projects/eng-throne-453420-e1/regions/europe-west2/operations/80c5a820-1059-3c73-9793-2eaa
3020c7cd].
Deleted [https://dataproc.googleapis.com/v1/projects/eng-throne-453420-e1/regions/europe-west2/clusters/eng-thro
ne-453420-e1-maxcluster].
```

In [39]:
```
# Create a Dataproc cluster with 1 master and 7 worker nodes for maximum parallelism
!gcloud dataproc clusters create eng-throne-453420-e1-maxcluster \
  --region=europe-west2 \
  --num-workers=7 \
  --worker-machine-type=e2-standard-2 \
  --worker-boot-disk-size=50GB \
  --master-machine-type=e2-standard-2 \
  --master-boot-disk-size=100GB \
  --image-version=2.0-debian10 \
  --initialization-actions=gs://goog-dataproc-initialization-actions-europe-west2/python/pip-install.sh \
  --metadata=PIP_PACKAGES="tensorflow numpy" \
  --enable-component-gateway
```

```
Waiting on operation [projects/eng-throne-453420-e1/regions/europe-west2/operations/58bb5517-709a-319a-bdcc-9665
a928cc61].

WARNING: Don't create production clusters that reference initialization actions located in the gs://goog-datapro
c-initialization-actions-REGION public buckets. These scripts are provided as reference implementations, and the
y are synchronized with ongoing GitHub repository changes—a new version of a initialization action in public buc
kets may break your cluster creation. Instead, copy the following initialization actions from public buckets int
o your bucket : gs://goog-dataproc-initialization-actions-europe-west2/python/pip-install.sh
WARNING: For PD-Standard without local SSDs, we strongly recommend provisioning 1TB or larger to ensure consiste
ntly high I/O performance. See https://cloud.google.com/compute/docs/disks/performance for information on disk I
/O performance.
WARNING: The firewall rules for specified network or subnetwork would allow ingress traffic from 0.0.0.0/0, whic
h could be a security risk.
WARNING: The specified custom staging bucket 'dataproc-staging-europe-west2-410514942591-juafyoqs' is not using
uniform bucket level access IAM configuration. It is recommended to update bucket to enable the same. See https:
//cloud.google.com/storage/docs/uniform-bucket-level-access.
Created [https://dataproc.googleapis.com/v1/projects/eng-throne-453420-e1/regions/europe-west2/clusters/eng-thro
ne-453420-e1-maxcluster] Cluster placed in zone [europe-west2-c].
```

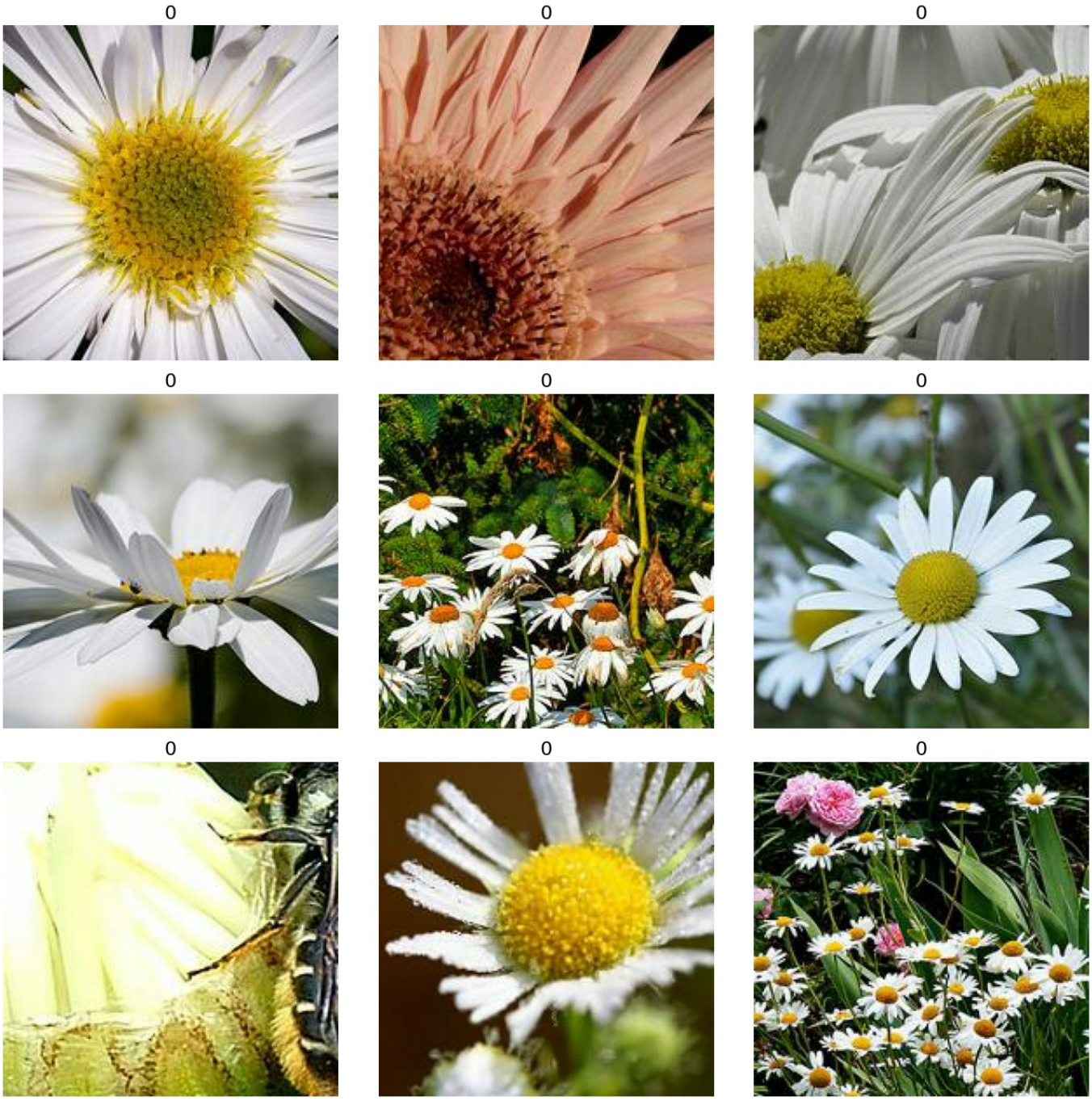
In [44]:
```
# Submit the spark_write_tfrec.py PySpark job to the maxcluster with 7 workers
!gcloud dataproc jobs submit pyspark /content/drive/MyDrive/BD-CW/spark_write_tfrec.py \
  --cluster=eng-throne-453420-e1-maxcluster \
  --region=europe-west2
```

```
Job [e22816e9a14045b3ba3975949a5829e3] submitted.
Waiting for job output...
2025-04-26 08:46:09.611126: I tensorflow/tsl/cuda/cudart_stub.cc:28] Could not find cuda drivers on your machine
, GPU will not be used.
2025-04-26 08:46:10.025720: I tensorflow/tsl/cuda/cudart_stub.cc:28] Could not find cuda drivers on your machine
, GPU will not be used.
2025-04-26 08:46:10.027138: I tensorflow/core/platform/cpu_feature_guard.cc:182] This TensorFlow binary is optim
ized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate com
piler flags.
2025-04-26 08:46:13.314171: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Could not fi
nd TensorRT
/opt/conda/default/lib/python3.8/site-packages/scipy/__init__.py:138: UserWarning: A NumPy version >=1.16.5 and
<1.23.0 is required for this version of SciPy (detected version 1.24.3)
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion} is required for this version of "
25/04/26 08:46:19 INFO org.apache.spark.SparkEnv: Registering MapOutputTracker
25/04/26 08:46:19 INFO org.apache.spark.SparkEnv: Registering BlockManagerMaster
25/04/26 08:46:19 INFO org.apache.spark.SparkEnv: Registering BlockManagerMasterHeartbeat
```

```
25/04/26 08:46:19 INFO org.apache.spark.SparkEnv: Registering OutputCommitCoordinator
25/04/26 08:46:20 INFO org.sparkproject.jetty.util.log: Logging initialized @18076ms to org.sparkproject.jetty.u
til.log.Slf4jLog
25/04/26 08:46:20 INFO org.sparkproject.jetty.server.Server: jetty-9.4.40.v20210413; built: 2021-04-13T20:42:42.
668Z; git: b881a572662e1943a14ae12e7e1207989f218b74; jvm 1.8.0_442-b06
25/04/26 08:46:20 INFO org.sparkproject.jetty.server.Server: Started @18332ms
25/04/26 08:46:20 INFO org.sparkproject.jetty.server.AbstractConnector: Started ServerConnector@2e903cc0{HTTP/1.
1, (http/1.1)}{0.0.0.0:45075}
25/04/26 08:46:23 INFO com.google.cloud.dataproc.DataprocSparkPlugin: Registered 128 driver metrics
25/04/26 08:46:24 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at eng-throne-453420
-e1-maxcluster-m/10.154.0.18:8032
25/04/26 08:46:25 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to Application History server at eng-t
hrone-453420-e1-maxcluster-m/10.154.0.18:10200
25/04/26 08:46:26 INFO org.apache.hadoop.conf.Configuration: resource-types.xml not found
25/04/26 08:46:26 INFO org.apache.hadoop.yarn.util.resource.ResourceUtils: Unable to find 'resource-types.xml'.
25/04/26 08:46:28 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitted application application_
1745656330861_0002
25/04/26 08:46:29 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at eng-throne-453420
-e1-maxcluster-m/10.154.0.18:8030
25/04/26 08:46:31 INFO com.google.cloud.hadoop.repackaged.gcs.com.google.cloud.hadoop.gcsio.GoogleCloudStorageIm
pl: Ignoring exception of type GoogleJsonResponseException; verified object already exists with desired state.
TFRecord files written:
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-00.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-01.tfrec
25/04/26 08:47:05 INFO org.sparkproject.jetty.server.AbstractConnector: Stopped Spark@2e903cc0{HTTP/1.1, (http/1
.1)}{0.0.0.0:0}
25/04/26 08:47:05 INFO com.google.cloud.dataproc.DataprocSparkPlugin: Shutting down driver plugin. metrics=[file
s_created=1, gcs_api_server_not_implemented_error_count=0, gcs_api_server_timeout_count=0, action_http_post_requ
est_failures=0, op_get_list_status_result_size=0, op_open=0, gcs_api_client_unauthorized_response_count=0, actio
n_http_head_request_failures=0, stream_read_close_operations=0, stream_read_bytes_backwards_on_seek=0, exception
_count=13, gcs_api_total_request_count=11, op_create=1, gcs_api_client_bad_request_count=0, op_create_non_recurs
ive=0, gcs_api_client_gone_response_count=0, stream_write_operations=0, stream_read_operations=0, gcs_api_client
_request_timeout_count=0, op_rename=0, op_get_file_status=1, stream_read_total_bytes=0, op_glob_status=0, stream
_read_exceptions=0, action_http_get_request_failures=0, op_exists=0, stream_write_bytes=104663, op_xattr_list=0,
stream_write_exceptions=0, gcs_api_server_unavailable_count=0, directories_created=0, files_delete_rejected=0, o
p_xattr_get_named=0, op_hsync=0, stream_read_operations_incomplete=0, op_delete=0, stream_read_bytes=0, gcs_api_
client_non_found_response_count=6, gcs_api_client_requested_range_not_statisfiable_count=0, op_hflush=0, op_list
_status=0, op_xattr_get_named_map=0, gcs_api_client_side_error_count=20, op_get_file_checksum=0, action_http_del
ete_request_failures=0, gcs_api_server_internal_error_count=0, stream_read_seek_bytes_skipped=0, stream_write_cl
ose_operations=0, op_list_files=0, files_deleted=0, op_mkdirs=1, gcs_api_client_rate_limit_error_count=0, action
_http_put_request_failures=0, gcs_api_server_bad_gateway_count=0, stream_read_seek_backward_operations=0, gcs_ap
i_server_side_error_count=0, action_http_patch_request_failures=0, stream_read_seek_operations=0, stream_read_se
ek_forward_operations=0, gcs_api_client_precondition_failed_response_count=1, directories_deleted=0, op_xattr_ge
t_map=0, delegation_tokens_issued=0, op_create_min=108, op_delete_min=0, op_mkdirs_min=238, op_create_non_recurs
ive_min=0, op_glob_status_min=0, op_hsync_min=0, op_xattr_get_named_min=0, op_list_status_min=0, op_xattr_get_na
med_map_min=0, stream_read_close_operations_min=0, stream_read_operations_min=0, stream_read_seek_operations_min
=0, op_hflush_min=0, op_xattr_get_map_min=0, op_xattr_list_min=0, stream_write_operations_min=0, op_get_file_sta
tus_min=473, op_open_min=0, op_rename_min=0, delegation_tokens_issued_min=0, stream_write_close_operations_min=0
, stream_read_close_operations_max=0, stream_read_operations_max=0, stream_read_seek_operations_max=0, op_hflush
_max=0, op_xattr_list_max=0, op_xattr_get_map_max=0, op_xattr_get_named_max=0, op_create_non_recursive_max=0, op
_glob_status_max=0, op_get_file_status_max=473, stream_write_close_operations_max=0, op_open_max=0, delegation_t
okens_issued_max=0, op_mkdirs_max=238, op_rename_max=0, op_create_max=108, op_delete_max=0, op_list_status_max=0
, op_xattr_get_named_map_max=0, stream_write_operations_max=0, op_hsync_max=0, op_list_status_mean=0, stream_rea
d_close_operations_mean=0, op_open_mean=0, op_xattr_get_named_map_mean=0, op_xattr_list_mean=0, op_mkdirs_mean=2
38, stream_write_close_operations_mean=0, op_rename_mean=0, op_hsync_mean=0, delegation_tokens_issued_mean=0, st
ream_read_operations_mean=0, op_xattr_get_map_mean=0, op_create_mean=108, op_glob_status_mean=0, op_delete_mean=
0, stream_read_seek_operations_mean=0, stream_write_operations_mean=0, op_create_non_recursive_mean=0, op_hflush
_mean=0, op_xattr_get_named_mean=0, op_get_file_status_mean=473, stream_write_operations_duration=0, stream_read
_operations_duration=0]
Job [e22816e9a14045b3ba3975949a5829e3] finished successfully.
done: true
driverControlFilesUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo/c
670966b-8fb0-43b7-9317-992aba95a6b3/jobs/e22816e9a14045b3ba3975949a5829e3/
driverOutputResourceUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo
/c670966b-8fb0-43b7-9317-992aba95a6b3/jobs/e22816e9a14045b3ba3975949a5829e3/driveroutput
jobUuid: a2d8db6f-d07d-3058-adf2-8c644274ef87
placement:
  clusterName: eng-throne-453420-e1-maxcluster
  clusterUuid: c670966b-8fb0-43b7-9317-992aba95a6b3
pysparkJob:
  mainPythonFileUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo/c67
0966b-8fb0-43b7-9317-992aba95a6b3/jobs/e22816e9a14045b3ba3975949a5829e3/staging/spark_write_tfrec.py
reference:
  jobId: e22816e9a14045b3ba3975949a5829e3
  projectId: eng-throne-453420-e1
status:
  state: DONE
  stateStartTime: '2025-04-26T08:47:09.157880Z'
statusHistory:
- state: PENDING
  stateStartTime: '2025-04-26T08:46:01.169600Z'
- state: SETUP_DONE
  stateStartTime: '2025-04-26T08:46:01.194778Z'
```

```
- details: Agent reported job success
  state: RUNNING
  stateStartTime: '2025-04-26T08:46:01.403413Z'
yarnApplications:
- name: spark_write_tfrec.py
  progress: 1.0
  state: FINISHED
  trackingUrl: http://eng-throne-453420-e1-maxcluster-m:8088/proxy/application_1745656330861_0002/
```

In [45]:
```
GCS_OUTPUT = 'gs://eng-throne-453420-e1-storage/spark-output/'

filenames = tf.io.gfile.glob(GCS_OUTPUT + "*.tfrec")
datasetTfrec = load_dataset(filenames)
display_9_images_from_dataset(datasetTfrec)
```



1d) Optimisation, experiments, and discussion (17%)

i) Improve parallelisation

If you implemented a straightfoward version, you will **probably** observe that **all the computation** is done on only **two nodes**. This can be adressed by using the **second parameter** in the initial call to **parallelize**. Make the **suitable change** in the code you have written above and mark it up in comments as `### TASK 1d ###`.

Demonstrate the difference in cluster utilisation before and after the change based on different parameter values with **screenshots from Google Cloud** and measure the **difference in the processing time**. (6%)

ii) Experiment with cluster configurations.

In addition to the experiments above (using 8 VMs),test your program with 4 machines with double the resources each (2 vCPUs, memory, disk) and 1 machine with eightfold resources. Discuss the results in terms of disk I/O and network bandwidth allocation in the cloud. (7%)

iii) Explain the difference between this use of Spark and most standard applications like e.g. in our labs in terms of where the data is stored. What kind of parallelisation approach is used here? (4%)

Write the code below and your answers in the report.

In [48]:
```
# Again delete previous clusters
!gcloud dataproc clusters delete eng-throne-453420-e1-maxcluster --region=europe-west2 --quiet
```

```
Waiting on operation [projects/eng-throne-453420-e1/regions/europe-west2/operations/79213fce-d22f-3833-88da-72a2
7ba6a5ab].
Deleted [https://dataproc.googleapis.com/v1/projects/eng-throne-453420-e1/regions/europe-west2/clusters/eng-thro
ne-453420-e1-maxcluster].
```

In [49]:
```
# Task ii
# Create a Dataproc cluster with 1 master and 3 workers, each with double the resources
!gcloud dataproc clusters create eng-throne-453420-e1-cluster-4big \
  --region=europe-west2 \
  --num-workers=3 \
  --worker-machine-type=e2-standard-4 \
  --worker-boot-disk-size=100GB \
  --master-machine-type=e2-standard-4 \
  --master-boot-disk-size=100GB \
  --image-version=2.0-debian10 \
  --initialization-actions=gs://goog-dataproc-initialization-actions-europe-west2/python/pip-install.sh \
  --metadata=PIP_PACKAGES="tensorflow numpy"
```

```
Waiting on operation [projects/eng-throne-453420-e1/regions/europe-west2/operations/038939df-2099-331d-a646-5c2e
e727d6ec].

WARNING: Don't create production clusters that reference initialization actions located in the gs://goog-datapro
c-initialization-actions-REGION public buckets. These scripts are provided as reference implementations, and the
y are synchronized with ongoing GitHub repository changes—a new version of a initialization action in public buc
kets may break your cluster creation. Instead, copy the following initialization actions from public buckets int
o your bucket : gs://goog-dataproc-initialization-actions-europe-west2/python/pip-install.sh
WARNING: For PD-Standard without local SSDs, we strongly recommend provisioning 1TB or larger to ensure consiste
ntly high I/O performance. See https://cloud.google.com/compute/docs/disks/performance for information on disk I
/O performance.
WARNING: The firewall rules for specified network or subnetwork would allow ingress traffic from 0.0.0.0/0, whic
h could be a security risk.
WARNING: The specified custom staging bucket 'dataproc-staging-europe-west2-410514942591-juafyoqs' is not using
uniform bucket level access IAM configuration. It is recommended to update bucket to enable the same. See https:
//cloud.google.com/storage/docs/uniform-bucket-level-access.
Created [https://dataproc.googleapis.com/v1/projects/eng-throne-453420-e1/regions/europe-west2/clusters/eng-thro
ne-453420-e1-cluster-4big] Cluster placed in zone [europe-west2-c].
```

In [52]:
```
# Submit the spark_write_tfrec.py PySpark job to the 4big cluster
!gcloud dataproc jobs submit pyspark /content/drive/MyDrive/BD-CW/spark_write_tfrec.py \
  --cluster=eng-throne-453420-e1-cluster-4big \
  --region=europe-west2
```

```
Job [d33b1fb903b54379b2115821f319f929] submitted.
Waiting for job output...
2025-04-26 09:23:24.365215: I tensorflow/tsl/cuda/cudart_stub.cc:28] Could not find cuda drivers on your machine
, GPU will not be used.
2025-04-26 09:23:24.425128: I tensorflow/tsl/cuda/cudart_stub.cc:28] Could not find cuda drivers on your machine
, GPU will not be used.
2025-04-26 09:23:24.425760: I tensorflow/core/platform/cpu_feature_guard.cc:182] This TensorFlow binary is optim
ized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate com
piler flags.
2025-04-26 09:23:25.710055: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Could not fi
nd TensorRT
/opt/conda/default/lib/python3.8/site-packages/scipy/__init__.py:138: UserWarning: A NumPy version >=1.16.5 and
<1.23.0 is required for this version of SciPy (detected version 1.24.3)
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion} is required for this version of "
25/04/26 09:23:29 INFO org.apache.spark.SparkEnv: Registering MapOutputTracker
25/04/26 09:23:29 INFO org.apache.spark.SparkEnv: Registering BlockManagerMaster
25/04/26 09:23:29 INFO org.apache.spark.SparkEnv: Registering BlockManagerMasterHeartbeat
25/04/26 09:23:29 INFO org.apache.spark.SparkEnv: Registering OutputCommitCoordinator
25/04/26 09:23:29 INFO org.sparkproject.jetty.util.log: Logging initialized @8081ms to org.sparkproject.jetty.ut
il.log.Slf4jLog
25/04/26 09:23:29 INFO org.sparkproject.jetty.server.Server: jetty-9.4.40.v20210413; built: 2021-04-13T20:42:42.
668Z; git: b881a572662e1943a14ae12e7e1207989f218b74; jvm 1.8.0_442-b06
25/04/26 09:23:29 INFO org.sparkproject.jetty.server.Server: Started @8204ms
25/04/26 09:23:29 INFO org.sparkproject.jetty.server.AbstractConnector: Started ServerConnector@394b7dcf{HTTP/1.
1, (http/1.1)}{0.0.0.0:33779}
25/04/26 09:23:31 INFO com.google.cloud.dataproc.DataprocSparkPlugin: Registered 128 driver metrics
25/04/26 09:23:32 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at eng-throne-453420
-e1-cluster-4big-m/10.154.0.29:8032
```

```
25/04/26 09:23:32 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to Application History server at eng-t
hrone-453420-e1-cluster-4big-m/10.154.0.29:10200
25/04/26 09:23:33 INFO org.apache.hadoop.conf.Configuration: resource-types.xml not found
25/04/26 09:23:33 INFO org.apache.hadoop.yarn.util.resource.ResourceUtils: Unable to find 'resource-types.xml'.
25/04/26 09:23:35 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitted application application_
1745659160732_0001
25/04/26 09:23:36 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at eng-throne-453420
-e1-cluster-4big-m/10.154.0.29:8030
25/04/26 09:23:38 INFO com.google.cloud.hadoop.repackaged.gcs.com.google.cloud.hadoop.gcsio.GoogleCloudStorageIm
pl: Ignoring exception of type GoogleJsonResponseException; verified object already exists with desired state.
TFRecord files written:
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-00.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-01.tfrec
25/04/26 09:24:08 INFO org.sparkproject.jetty.server.AbstractConnector: Stopped Spark@394b7dcf{HTTP/1.1, (http/1
.1)}{0.0.0.0:0}
25/04/26 09:24:08 INFO com.google.cloud.dataproc.DataprocSparkPlugin: Shutting down driver plugin. metrics=[file
s_created=1, gcs_api_server_not_implemented_error_count=0, gcs_api_server_timeout_count=0, action_http_post_requ
est_failures=0, op_get_list_status_result_size=0, op_open=0, gcs_api_client_unauthorized_response_count=0, actio
n_http_head_request_failures=0, stream_read_close_operations=0, stream_read_bytes_backwards_on_seek=0, exception
_count=13, gcs_api_total_request_count=11, op_create=1, gcs_api_client_bad_request_count=0, op_create_non_recurs
ive=0, gcs_api_client_gone_response_count=0, stream_write_operations=0, stream_read_operations=0, gcs_api_client
_request_timeout_count=0, op_rename=0, op_get_file_status=1, stream_read_total_bytes=0, op_glob_status=0, stream
_read_exceptions=0, action_http_get_request_failures=0, op_exists=0, stream_write_bytes=105028, op_xattr_list=0,
stream_write_exceptions=0, gcs_api_server_unavailable_count=0, directories_created=0, files_delete_rejected=0, o
p_xattr_get_named=0, op_hsync=0, stream_read_operations_incomplete=0, op_delete=0, stream_read_bytes=0, gcs_api_
client_non_found_response_count=6, gcs_api_client_requested_range_not_statisfiable_count=0, op_hflush=0, op_list
_status=0, op_xattr_get_named_map=0, gcs_api_client_side_error_count=20, op_get_file_checksum=0, action_http_del
ete_request_failures=0, gcs_api_server_internal_error_count=0, stream_read_seek_bytes_skipped=0, stream_write_cl
ose_operations=0, op_list_files=0, files_deleted=0, op_mkdirs=1, gcs_api_client_rate_limit_error_count=0, action
_http_put_request_failures=0, gcs_api_server_bad_gateway_count=0, stream_read_seek_backward_operations=0, gcs_ap
i_server_side_error_count=0, action_http_patch_request_failures=0, stream_read_seek_operations=0, stream_read_se
ek_forward_operations=0, gcs_api_client_precondition_failed_response_count=1, directories_deleted=0, op_xattr_ge
t_map=0, delegation_tokens_issued=0, op_create_min=79, op_delete_min=0, op_mkdirs_min=207, op_create_non_recursi
ve_min=0, op_glob_status_min=0, op_hsync_min=0, op_xattr_get_named_min=0, op_list_status_min=0, op_xattr_get_nam
ed_map_min=0, stream_read_close_operations_min=0, stream_read_operations_min=0, stream_read_seek_operations_min=
0, op_hflush_min=0, op_xattr_get_map_min=0, op_xattr_list_min=0, stream_write_operations_min=0, op_get_file_stat
us_min=339, op_open_min=0, op_rename_min=0, delegation_tokens_issued_min=0, stream_write_close_operations_min=0,
stream_read_close_operations_max=0, stream_read_operations_max=0, stream_read_seek_operations_max=0, op_hflush_m
ax=0, op_xattr_list_max=0, op_xattr_get_map_max=0, op_xattr_get_named_max=0, op_create_non_recursive_max=0, op_g
lob_status_max=0, op_get_file_status_max=339, stream_write_close_operations_max=0, op_open_max=0, delegation_tok
ens_issued_max=0, op_mkdirs_max=207, op_rename_max=0, op_create_max=79, op_delete_max=0, op_list_status_max=0, o
p_xattr_get_named_map_max=0, stream_write_operations_max=0, op_hsync_max=0, op_list_status_mean=0, stream_read_c
lose_operations_mean=0, op_open_mean=0, op_xattr_get_named_map_mean=0, op_xattr_list_mean=0, op_mkdirs_mean=207,
stream_write_close_operations_mean=0, op_rename_mean=0, op_hsync_mean=0, delegation_tokens_issued_mean=0, stream
_read_operations_mean=0, op_xattr_get_map_mean=0, op_create_mean=79, op_glob_status_mean=0, op_delete_mean=0, st
ream_read_seek_operations_mean=0, stream_write_operations_mean=0, op_create_non_recursive_mean=0, op_hflush_mean
=0, op_xattr_get_named_mean=0, op_get_file_status_mean=339, stream_write_operations_duration=0, stream_read_oper
ations_duration=0]
Job [d33b1fb903b54379b2115821f319f929] finished successfully.
done: true
driverControlFilesUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo/c
54f5fd0-985a-484e-b3b7-fb97f8ffe82f/jobs/d33b1fb903b54379b2115821f319f929/
driverOutputResourceUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo
/c54f5fd0-985a-484e-b3b7-fb97f8ffe82f/jobs/d33b1fb903b54379b2115821f319f929/driveroutput
jobUuid: 288aca80-a7d3-39e2-a0b3-796c4e37668d
placement:
  clusterName: eng-throne-453420-e1-cluster-4big
  clusterUuid: c54f5fd0-985a-484e-b3b7-fb97f8ffe82f
pysparkJob:
  mainPythonFileUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo/c54
f5fd0-985a-484e-b3b7-fb97f8ffe82f/jobs/d33b1fb903b54379b2115821f319f929/staging/spark_write_tfrec.py
reference:
  jobId: d33b1fb903b54379b2115821f319f929
  projectId: eng-throne-453420-e1
status:
  state: DONE
  stateStartTime: '2025-04-26T09:24:12.818798Z'
statusHistory:
- state: PENDING
  stateStartTime: '2025-04-26T09:23:19.865456Z'
- state: SETUP_DONE
  stateStartTime: '2025-04-26T09:23:19.888244Z'
- details: Agent reported job success
  state: RUNNING
  stateStartTime: '2025-04-26T09:23:20.243102Z'
yarnApplications:
- name: spark_write_tfrec.py
  progress: 1.0
  state: FINISHED
  trackingUrl: http://eng-throne-453420-e1-cluster-4big-m:8088/proxy/application_1745659160732_0001/
```

In [54]:
```
# Delete running clusters
!gcloud dataproc clusters delete eng-throne-453420-e1-cluster-4big --region=europe-west2 --quiet
```

```
Waiting on operation [projects/eng-throne-453420-e1/regions/europe-west2/operations/025be622-d8a8-381a-8872-fc17c1dcb443].
Deleted [https://dataproc.googleapis.com/v1/projects/eng-throne-453420-e1/regions/europe-west2/clusters/eng-throne-453420-e1-cluster-4big].
```

In [55]:
```
# Create a Dataproc single-node cluster with a large machine (16 vCPUs) for comparison
!gcloud dataproc clusters create eng-throne-453420-e1-cluster-1huge \
  --region=europe-west2 \
  --single-node \
  --master-machine-type=e2-standard-16 \
  --master-boot-disk-size=100GB \
  --image-version=2.0-debian10 \
  --initialization-actions=gs://goog-dataproc-initialization-actions-europe-west2/python/pip-install.sh \
  --metadata=PIP_PACKAGES="tensorflow numpy"
```

```
Waiting on operation [projects/eng-throne-453420-e1/regions/europe-west2/operations/4b584b43-7db3-3340-9ca9-a00ee6838ccb].

WARNING: Don't create production clusters that reference initialization actions located in the gs://goog-dataproc-initialization-actions-REGION public buckets. These scripts are provided as reference implementations, and they are synchronized with ongoing GitHub repository changes—a new version of a initialization action in public buckets may break your cluster creation. Instead, copy the following initialization actions from public buckets into your bucket : gs://goog-dataproc-initialization-actions-europe-west2/python/pip-install.sh
WARNING: For PD-Standard without local SSDs, we strongly recommend provisioning 1TB or larger to ensure consistently high I/O performance. See https://cloud.google.com/compute/docs/disks/performance for information on disk I/O performance.
WARNING: The firewall rules for specified network or subnetwork would allow ingress traffic from 0.0.0.0/0, which could be a security risk.
WARNING: The specified custom staging bucket 'dataproc-staging-europe-west2-410514942591-juafyoqs' is not using uniform bucket level access IAM configuration. It is recommended to update bucket to enable the same. See https://cloud.google.com/storage/docs/uniform-bucket-level-access.
Created [https://dataproc.googleapis.com/v1/projects/eng-throne-453420-e1/regions/europe-west2/clusters/eng-throne-453420-e1-cluster-1huge] Cluster placed in zone [europe-west2-c].
```

In [56]:
```
# Submit the spark_write_tfrec.py PySpark job to the 1huge cluster (single powerful machine)
!gcloud dataproc jobs submit pyspark /content/drive/MyDrive/BD-CW/spark_write_tfrec.py \
  --cluster=eng-throne-453420-e1-cluster-1huge \
  --region=europe-west2
```

```
Job [499b808844334770935e66d24a80affa] submitted.
Waiting for job output...
2025-04-26 09:29:38.291705: I tensorflow/tsl/cuda/cudart_stub.cc:28] Could not find cuda drivers on your machine, GPU will not be used.
2025-04-26 09:29:38.324203: I tensorflow/tsl/cuda/cudart_stub.cc:28] Could not find cuda drivers on your machine, GPU will not be used.
2025-04-26 09:29:38.324561: I tensorflow/core/platform/cpu_feature_guard.cc:182] This TensorFlow binary is optimized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
2025-04-26 09:29:43.975431: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Could not find TensorRT
/opt/conda/default/lib/python3.8/site-packages/scipy/__init__.py:138: UserWarning: A NumPy version >=1.16.5 and <1.23.0 is required for this version of SciPy (detected version 1.24.3)
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion} is required for this version of "
25/04/26 09:29:47 INFO org.apache.spark.SparkEnv: Registering MapOutputTracker
25/04/26 09:29:47 INFO org.apache.spark.SparkEnv: Registering BlockManagerMaster
25/04/26 09:29:47 INFO org.apache.spark.SparkEnv: Registering BlockManagerMasterHeartbeat
25/04/26 09:29:47 INFO org.apache.spark.SparkEnv: Registering OutputCommitCoordinator
25/04/26 09:29:47 INFO org.sparkproject.jetty.util.log: Logging initialized @14325ms to org.sparkproject.jetty.util.log.Slf4jLog
25/04/26 09:29:48 INFO org.sparkproject.jetty.server.Server: jetty-9.4.40.v20210413; built: 2021-04-13T20:42:42.668Z; git: b881a572662e1943a14ae12e7e1207989f218b74; jvm 1.8.0_442-b06
25/04/26 09:29:48 INFO org.sparkproject.jetty.server.Server: Started @14412ms
25/04/26 09:29:48 INFO org.sparkproject.jetty.server.AbstractConnector: Started ServerConnector@12ecf764{HTTP/1.1, (http/1.1)}{0.0.0.0:41283}
25/04/26 09:29:49 INFO com.google.cloud.dataproc.DataprocSparkPlugin: Registered 128 driver metrics
25/04/26 09:29:49 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at eng-throne-453420-e1-cluster-1huge-m/10.154.0.30:8032
25/04/26 09:29:50 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to Application History server at eng-throne-453420-e1-cluster-1huge-m/10.154.0.30:10200
25/04/26 09:29:50 INFO org.apache.hadoop.conf.Configuration: resource-types.xml not found
25/04/26 09:29:50 INFO org.apache.hadoop.yarn.util.resource.ResourceUtils: Unable to find 'resource-types.xml'.
25/04/26 09:29:52 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitted application application_1745659685036_0001
25/04/26 09:29:53 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at eng-throne-453420-e1-cluster-1huge-m/10.154.0.30:8030
25/04/26 09:29:54 INFO com.google.cloud.hadoop.repackaged.gcs.com.google.cloud.hadoop.gcsio.GoogleCloudStorageImpl: Ignoring exception of type GoogleJsonResponseException; verified object already exists with desired state.
TFRecord files written:
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-00.tfrec
gs://eng-throne-453420-e1-storage/spark-output/flowers-partition-01.tfrec
25/04/26 09:30:17 INFO org.sparkproject.jetty.server.AbstractConnector: Stopped Spark@12ecf764{HTTP/1.1, (http/1.1)}{0.0.0.0:0}
25/04/26 09:30:18 INFO com.google.cloud.dataproc.DataprocSparkPlugin: Shutting down driver plugin. metrics=[file
```

s_created=1, gcs_api_server_not_implemented_error_count=0, gcs_api_server_timeout_count=0, action_http_post_request_failures=0, op_get_list_status_result_size=0, op_open=0, gcs_api_client_unauthorized_response_count=0, action_http_head_request_failures=0, stream_read_close_operations=0, stream_read_bytes_backwards_on_seek=0, exception_count=13, gcs_api_total_request_count=11, op_create=1, gcs_api_client_bad_request_count=0, op_create_non_recursive=0, gcs_api_client_gone_response_count=0, stream_write_operations=0, stream_read_operations=0, gcs_api_client_request_timeout_count=0, op_rename=0, op_get_file_status=1, stream_read_total_bytes=0, op_glob_status=0, stream_read_exceptions=0, action_http_get_request_failures=0, op_exists=0, stream_write_bytes=105033, op_xattr_list=0, stream_write_exceptions=0, gcs_api_server_unavailable_count=0, directories_created=0, files_delete_rejected=0, op_xattr_get_named=0, op_hsync=0, stream_read_operations_incomplete=0, op_delete=0, stream_read_bytes=0, gcs_api_client_non_found_response_count=6, gcs_api_client_requested_range_not_satisfiable_count=0, op_hflush=0, op_list_status=0, op_xattr_get_named_map=0, gcs_api_client_side_error_count=20, op_get_file_checksum=0, action_http_delete_request_failures=0, gcs_api_server_internal_error_count=0, stream_read_seek_bytes_skipped=0, stream_write_close_operations=0, op_list_files=0, files_deleted=0, op_mkdirs=1, gcs_api_client_rate_limit_error_count=0, action_http_put_request_failures=0, gcs_api_server_bad_gateway_count=0, stream_read_seek_backward_operations=0, gcs_api_server_side_error_count=0, action_http_patch_request_failures=0, stream_read_seek_operations=0, stream_read_seek_forward_operations=0, gcs_api_client_precondition_failed_response_count=1, directories_deleted=0, op_xattr_get_map=0, delegation_tokens_issued=0, op_create_min=75, op_delete_min=0, op_mkdirs_min=169, op_create_non_recursive_min=0, op_glob_status_min=0, op_hsync_min=0, op_xattr_get_named_min=0, op_list_status_min=0, op_xattr_get_named_map_min=0, stream_read_close_operations_min=0, stream_read_operations_min=0, stream_read_seek_operations_min=0, op_hflush_min=0, op_xattr_get_map_min=0, op_xattr_list_min=0, stream_write_operations_min=0, op_get_file_status_min=192, op_open_min=0, op_rename_min=0, delegation_tokens_issued_min=0, stream_write_close_operations_min=0, stream_read_close_operations_max=0, stream_read_operations_max=0, stream_read_seek_operations_max=0, op_hflush_max=0, op_xattr_list_max=0, op_xattr_get_map_max=0, op_xattr_get_named_max=0, op_create_non_recursive_max=0, op_glob_status_max=0, op_get_file_status_max=192, stream_write_close_operations_max=0, op_open_max=0, delegation_tokens_issued_max=0, op_mkdirs_max=169, op_rename_max=0, op_create_max=75, op_delete_max=0, op_list_status_max=0, op_xattr_get_named_map_max=0, stream_write_operations_max=0, op_hsync_max=0, op_list_status_mean=0, stream_read_close_operations_mean=0, op_open_mean=0, op_xattr_get_named_map_mean=0, op_xattr_list_mean=0, op_mkdirs_mean=169, stream_write_close_operations_mean=0, op_rename_mean=0, op_hsync_mean=0, delegation_tokens_issued_mean=0, stream_read_operations_mean=0, op_xattr_get_map_mean=0, op_create_mean=75, op_glob_status_mean=0, op_delete_mean=0, stream_read_seek_operations_mean=0, stream_write_operations_mean=0, op_create_non_recursive_mean=0, op_hflush_mean=0, op_xattr_get_named_mean=0, op_get_file_status_mean=192, stream_write_operations_duration=0, stream_read_operations_duration=0]
Job [499b808844334770935e66d24a80affa] finished successfully.
done: true
driverControlFilesUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo/40573cef-12d0-4245-837c-09d1c70fa8df/jobs/499b808844334770935e66d24a80affa/
driverOutputResourceUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo/40573cef-12d0-4245-837c-09d1c70fa8df/jobs/499b808844334770935e66d24a80affa/driveroutput
jobUuid: 9f829699-a47e-39fb-ab6a-cdf77727a525
placement:
  clusterName: eng-throne-453420-e1-cluster-1huge
  clusterUuid: 40573cef-12d0-4245-837c-09d1c70fa8df
pysparkJob:
  mainPythonFileUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo/40573cef-12d0-4245-837c-09d1c70fa8df/jobs/499b808844334770935e66d24a80affa/staging/spark_write_tfrec.py
reference:
  jobId: 499b808844334770935e66d24a80affa
  projectId: eng-throne-453420-e1
status:
  state: DONE
  stateStartTime: '2025-04-26T09:30:21.754284Z'
statusHistory:
- state: PENDING
  stateStartTime: '2025-04-26T09:29:31.176336Z'
- state: SETUP_DONE
  stateStartTime: '2025-04-26T09:29:31.197320Z'
- details: Agent reported job success
  state: RUNNING
  stateStartTime: '2025-04-26T09:29:31.466836Z'
yarnApplications:
- name: spark_write_tfrec.py
  progress: 1.0
  state: FINISHED
  trackingUrl: http://eng-throne-453420-e1-cluster-1huge-m:8088/proxy/application_1745659685036_0001/

Section 2: Speed tests

We have seen that **reading from the pre-processed TFRecord files** is **faster** than reading individual image files and decoding on the fly. This task is about **measuring this effect** and **parallelizing the tests with PySpark**.

2.1 Speed test implementation

Here is **code for time measurement** to determine the **throughput in images per second**. It doesn't render the images but extracts and prints some basic information in order to make sure the image data are read. We write the information to the null device for longer measurements `null_file=open("/dev/null", mode='w')`. That way it will not clutter our cell output.

We use batches (`dset2 = dset1.batch(batch_size)`) and select a number of batches with (`dset3 = dset2.take(batch_number)`). Then we use the `time.time()` to take the **time measurement** and take it multiple times, reading

from the same dataset to see if reading speed changes with mutiple readings.

We then **vary** the size of the batch (`batch_size`) and the number of batches (`batch_number`) and **store the results for different values**. Store also the **results for each repetition** over the same dataset (repeat 2 or 3 times).

The speed test should be combined in a **function** `time_configs()` that takes a configuration, i.e. a dataset and arrays of `batch_sizes` , `batch_numbers` , and `repetitions` (an array of integers starting from 1), as **arguments** and runs the time measurement for each combination of batch_size and batch_number for the requested number of repetitions.

In [57]:
```
# Here are some useful values for testing your code, use higher values later for actually testing throughput
batch_sizes = [2,4]
batch_numbers = [3,6]
repetitions = [1]

def time_configs(dataset, batch_sizes, batch_numbers, repetitions):
    dims = [len(batch_sizes),len(batch_numbers),len(repetitions)]
    print(dims)
    results = np.zeros(dims)
    params = np.zeros(dims + [3])
    print( results.shape )
    with open("/dev/null",mode='w') as null_file: # for printing the output without showing it
        tt = time.time() # for overall time taking
        for bsi,bs in enumerate(batch_sizes):
            for dsi, ds in enumerate(batch_numbers):
                batched_dataset = dataset.batch(bs)
                timing_set = batched_dataset.take(ds)
                for ri,rep in enumerate(repetitions):
                    print("bs: {}, ds: {}, rep: {}".format(bs,ds,rep))
                    t0 = time.time()
                    for image, label in timing_set:
                        #print("Image batch shape {}".format(image.numpy().shape),
                        print("Image batch shape {}, {})".format(image.numpy().shape,
                            [str(lbl) for lbl in label.numpy()]), null_file)
                    td = time.time() - t0 # duration for reading images
                    results[bsi,dsi,ri] = ( bs * ds) / td
                    params[bsi,dsi,ri] = [ bs, ds, rep ]
    print("total time: "+str(time.time()-tt))
    return results, params
```

Let's try this function with a **small number** of configurations of batch_sizes batch_numbers and repetions, so that we get a set of parameter combinations and corresponding reading speeds. Try reading from the image files (dataset4) and the TFRecord files (datasetTfrec).

In [58]:
```
[res,par] = time_configs(dataset4, batch_sizes, batch_numbers, repetitions)
print(res)
print(par)

print("=============")

[res,par] = time_configs(datasetTfrec, batch_sizes, batch_numbers, repetitions)
print(res)
print(par)
```

```
[2, 2, 1]
(2, 2, 1)
bs: 2, ds: 3, rep: 1
Image batch shape (2,), ["b'sunflowers'", "b'roses'"]) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='ut
f-8'>
Image batch shape (2,), ["b'dandelion'", "b'tulips'"]) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='ut
f-8'>
Image batch shape (2,), ["b'dandelion'", "b'roses'"]) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='utf
-8'>
bs: 2, ds: 6, rep: 1
Image batch shape (2,), ["b'tulips'", "b'dandelion'"]) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='ut
f-8'>
Image batch shape (2,), ["b'dandelion'", "b'roses'"]) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='utf
-8'>
Image batch shape (2,), ["b'sunflowers'", "b'roses'"]) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='ut
f-8'>
Image batch shape (2,), ["b'daisy'", "b'sunflowers'"]) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='ut
f-8'>
Image batch shape (2,), ["b'tulips'", "b'sunflowers'"]) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='u
tf-8'>
Image batch shape (2,), ["b'tulips'", "b'sunflowers'"]) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='u
tf-8'>
bs: 4, ds: 3, rep: 1
Image batch shape (4,), ["b'sunflowers'", "b'dandelion'", "b'roses'", "b'tulips'"]) <_io.TextIOWrapper name='/de
v/null' mode='w' encoding='utf-8'>
Image batch shape (4,), ["b'tulips'", "b'tulips'", "b'dandelion'", "b'sunflowers'"]) <_io.TextIOWrapper name='/d
ev/null' mode='w' encoding='utf-8'>
Image batch shape (4,), ["b'tulips'", "b'tulips'", "b'sunflowers'", "b'sunflowers'"]) <_io.TextIOWrapper name='/
```

```
dev/null' mode='w' encoding='utf-8'>
bs: 4, ds: 6, rep: 1
Image batch shape (4,), ["b'tulips'", "b'daisy'", "b'roses'", "b'sunflowers'"]) <_io.TextIOWrapper name='/dev/nu
ll' mode='w' encoding='utf-8'>
Image batch shape (4,), ["b'sunflowers'", "b'dandelion'", "b'dandelion'", "b'sunflowers'"]) <_io.TextIOWrapper n
ame='/dev/null' mode='w' encoding='utf-8'>
Image batch shape (4,), ["b'roses'", "b'tulips'", "b'tulips'", "b'sunflowers'"]) <_io.TextIOWrapper name='/dev/n
ull' mode='w' encoding='utf-8'>
Image batch shape (4,), ["b'dandelion'", "b'roses'", "b'tulips'", "b'roses'"]) <_io.TextIOWrapper name='/dev/nul
l' mode='w' encoding='utf-8'>
Image batch shape (4,), ["b'tulips'", "b'sunflowers'", "b'roses'", "b'roses'"]) <_io.TextIOWrapper name='/dev/nu
ll' mode='w' encoding='utf-8'>
Image batch shape (4,), ["b'sunflowers'", "b'sunflowers'", "b'tulips'", "b'dandelion'"]) <_io.TextIOWrapper name
='/dev/null' mode='w' encoding='utf-8'>
total time: 6.986522436141968
[[[ 4.00968411]
  [ 7.52868082]]

 [[ 7.60174389]
  [10.71324231]]]
[[[[2. 3. 1.]]

  [[2. 6. 1.]]]


 [[[4. 3. 1.]]

  [[4. 6. 1.]]]]
=============
[2, 2, 1]
(2, 2, 1)
bs: 2, ds: 3, rep: 1
Image batch shape (2, 192, 192, 3), ['0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='utf-8'>
Image batch shape (2, 192, 192, 3), ['0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='utf-8'>
Image batch shape (2, 192, 192, 3), ['0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='utf-8'>
bs: 2, ds: 6, rep: 1
Image batch shape (2, 192, 192, 3), ['0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='utf-8'>
Image batch shape (2, 192, 192, 3), ['0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='utf-8'>
Image batch shape (2, 192, 192, 3), ['0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='utf-8'>
Image batch shape (2, 192, 192, 3), ['0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='utf-8'>
Image batch shape (2, 192, 192, 3), ['0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='utf-8'>
Image batch shape (2, 192, 192, 3), ['0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding='utf-8'>
bs: 4, ds: 3, rep: 1
Image batch shape (4, 192, 192, 3), ['0', '0', '0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding=
'utf-8'>
Image batch shape (4, 192, 192, 3), ['0', '0', '0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding=
'utf-8'>
Image batch shape (4, 192, 192, 3), ['0', '0', '0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding=
'utf-8'>
bs: 4, ds: 6, rep: 1
Image batch shape (4, 192, 192, 3), ['0', '0', '0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding=
'utf-8'>
Image batch shape (4, 192, 192, 3), ['0', '0', '0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding=
'utf-8'>
Image batch shape (4, 192, 192, 3), ['0', '0', '0', '0']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding=
'utf-8'>
Image batch shape (4, 192, 192, 3), ['1', '1', '1', '1']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding=
'utf-8'>
Image batch shape (4, 192, 192, 3), ['1', '1', '1', '1']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding=
'utf-8'>
Image batch shape (4, 192, 192, 3), ['1', '1', '1', '1']) <_io.TextIOWrapper name='/dev/null' mode='w' encoding=
'utf-8'>
total time: 0.8612415790557861
[[[ 15.47450035]
  [ 74.93307607]]

 [[ 86.38283178]
  [151.8448958 ]]]
[[[[2. 3. 1.]]

  [[2. 6. 1.]]]


 [[[4. 3. 1.]]

  [[4. 6. 1.]]]]
```

Task 2: Parallelising the speed test with Spark in the cloud. (36%)

As an exercise in **Spark programming and optimisation** as well as **performance analysis**, we will now implement the **speed test** with multiple parameters in parallel with Spark. Runing multiple tests in parallel would **not be a useful approach on a single machine, but it can be in the cloud** (you will be asked to reason about this later).

2a) Create the script (14%)

Your task is now to **port the speed test above to Spark** for running it in the cloud in Dataproc. **Adapt the speed testing** as a Spark program that performs the same actions as above, but **with Spark RDDs in a distributed way**. The distribution should be such that **each parameter combination (except repetition)** is processed in a separate Spark task.

More specifically:

- i) combine the previous cells to have the code to create a dataset and create a list of parameter combinations in an RDD (2%)
- ii) get a Spark context and create the dataset and run timing test for each combination in parallel (2%)
- iii) transform the resulting RDD to the structure (parameter_combination, images_per_second) and save these values in an array (2%)
- iv) create an RDD with all results for each parameter as (parameter_value,images_per_second) and collect the result for each parameter (2%)
- v) create an RDD with the average reading speeds for each parameter value and collect the results. Keep associativity in mind when implementing the average. (3%)
- vi) write the results to a pickle file in your bucket (2%)
- vii) Write your code it into a file using the *cell magic* `%%writefile spark_job.py` (1%)

Important: The task here is not to parallelize the pre-processing, but to run multiple speed tests in parallel using Spark.

In [70]:
```
%%writefile /content/drive/MyDrive/BD-CW/spark_job.py
# Save this script as spark_job.py inside the BD-CW folder in Google Drive

# ---- Import libraries ----
import tensorflow as tf
import numpy as np
import time
import pickle
import datetime
from pyspark import SparkContext

# ---- Config ----

# Define bucket path for saving outputs
BUCKET = 'gs://eng-throne-453420-e1-storage'
GCS_OUTPUT = BUCKET + '/spark-output/'

# Define testing parameters
batch_sizes = [2, 4]       # Different batch sizes to test
batch_numbers = [3, 6]     # Different number of batches to test
repetitions = [1, 2, 3]    # Multiple repetitions for robustness

# Generate all parameter combinations (batch_size, batch_number)
param_combinations = [(bs, bn) for bs in batch_sizes for bn in batch_numbers]

# ---- Functions ----

# Function to parse TFRecord entries
def read_tfrecord(example):
    features = {
        "image": tf.io.FixedLenFeature([], tf.string),    # Image feature (as bytes)
        "class": tf.io.FixedLenFeature([], tf.int64)       # Class feature (as integer)
    }
    example = tf.io.parse_single_example(example, features)
    image = tf.image.decode_jpeg(example['image'], channels=3)
    image = tf.reshape(image, [192, 192, 3])  # Reshape image to fixed dimensions
    label = example['class']
    return image, label

# Function to load dataset from TFRecord files
def load_dataset(filenames):
    option_no_order = tf.data.Options()
    option_no_order.experimental_deterministic = False  # Allow non-deterministic reads for faster performance
    dataset = tf.data.TFRecordDataset(filenames)
    dataset = dataset.with_options(option_no_order)
    dataset = dataset.map(read_tfrecord)  # Parse each TFRecord
    return dataset

# Timing function to measure images/sec for a given (batch_size, batch_number) pair
def timing_function(params):
    bs, bn = params
    filenames = tf.io.gfile.glob(GCS_OUTPUT + "*.tfrec")  # Get list of TFRecord files
    dataset = load_dataset(filenames)
    batched_dataset = dataset.batch(bs)  # Batch the dataset
    timing_set = batched_dataset.take(bn)  # Take fixed number of batches
```

```
    times = []
    for rep in repetitions:
        t0 = time.time()  # Start timing
        for image, label in timing_set:
            _ = image.numpy(), label.numpy()  # Read batch (simulate workload)
        td = time.time() - t0  # Measure elapsed time
        images_per_second = (bs * bn) / td  # Calculate throughput
        times.append(images_per_second)

    avg_images_per_second = np.mean(times)  # Average across repetitions
    return ((bs, bn), avg_images_per_second)

# ---- Main Spark Job ----
if __name__ == "__main__":
    sc = SparkContext.getOrCreate()  # Initialize SparkContext

    # Create RDD of parameter combinations
    rdd_params = sc.parallelize(param_combinations)

    # Map each parameter combination to a timing result and cache results for efficiency ### TASK 2c ###
    rdd_results = rdd_params.map(timing_function).cache()

    # Collect timing results from all nodes
    results = rdd_results.collect()

    # ---- Save results with timestamped filename ----
    timestamp = datetime.datetime.now().strftime("%y%m%d-%H%M")  # Get current time for unique filename
    output_filename = BUCKET + f'/speedtest_results_{timestamp}.pkl'

    # Write results to a Pickle file in the GCS bucket
    with tf.io.gfile.GFile(output_filename, 'wb') as f:
        pickle.dump(results, f)

    print(f"Speed test results saved to: {output_filename}")
```

```
Overwriting /content/drive/MyDrive/BD-CW/spark_job.py
```

2b) Testing the code and collecting results (4%)

i) First, test locally with `%run`.

It is useful to create a **new filename argument**, so that old results don't get overwritten.

You can for instance use `datetime.datetime.now().strftime("%y%m%d-%H%M")` to get a string with the current date and time and use that in the file name.

In [73]:
```
# Run the spark_job.py script locally inside the current Colab environment
%run spark_job.py
```

```
Speed test results saved to: gs://eng-throne-453420-e1-storage/speedtest_results_250426-1107.pkl
```

ii) Cloud

If you have a cluster running, you can run the speed test job in the cloud.

While you run this job, switch to the Dataproc web page and take **screenshots of the CPU and network load** over time. They are displayed with some delay, so you may need to wait a little. These images will be useful in the next task. Again, don't use the SCREENSHOT function that Google provides, but just take a picture of the graphs you see for the VMs.

In [71]:
```
# Submit the spark_job.py PySpark job to the 1huge cluster (single powerful machine)
!gcloud dataproc jobs submit pyspark /content/drive/MyDrive/BD-CW/spark_job.py \
  --cluster=eng-throne-453420-e1-cluster-1huge \
  --region=europe-west2
```

```
Job [bbc94844ab3f4d1694044a185174f655] submitted.
Waiting for job output...
2025-04-26 11:02:03.078023: I tensorflow/tsl/cuda/cudart_stub.cc:28] Could not find cuda drivers on your machine
, GPU will not be used.
2025-04-26 11:02:03.111092: I tensorflow/tsl/cuda/cudart_stub.cc:28] Could not find cuda drivers on your machine
, GPU will not be used.
2025-04-26 11:02:03.111475: I tensorflow/core/platform/cpu_feature_guard.cc:182] This TensorFlow binary is optim
ized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate com
piler flags.
2025-04-26 11:02:03.854322: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Could not fi
nd TensorRT
/opt/conda/default/lib/python3.8/site-packages/scipy/__init__.py:138: UserWarning: A NumPy version >=1.16.5 and
<1.23.0 is required for this version of SciPy (detected version 1.24.3)
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion} is required for this version of "
25/04/26 11:02:05 INFO org.apache.spark.SparkEnv: Registering MapOutputTracker
25/04/26 11:02:05 INFO org.apache.spark.SparkEnv: Registering BlockManagerMaster
25/04/26 11:02:05 INFO org.apache.spark.SparkEnv: Registering BlockManagerMasterHeartbeat
```

```
25/04/26 11:02:05 INFO org.apache.spark.SparkEnv: Registering OutputCommitCoordinator
25/04/26 11:02:05 INFO org.sparkproject.jetty.util.log: Logging initialized @4213ms to org.sparkproject.jetty.ut
il.log.Slf4jLog
25/04/26 11:02:05 INFO org.sparkproject.jetty.server.Server: jetty-9.4.40.v20210413; built: 2021-04-13T20:42:42.
668Z; git: b881a572662e1943a14ae12e7e1207989f218b74; jvm 1.8.0_442-b06
25/04/26 11:02:05 INFO org.sparkproject.jetty.server.Server: Started @4308ms
25/04/26 11:02:05 INFO org.sparkproject.jetty.server.AbstractConnector: Started ServerConnector@11c4597a{HTTP/1.
1, (http/1.1)}{0.0.0.0:38277}
25/04/26 11:02:06 INFO com.google.cloud.dataproc.DataprocSparkPlugin: Registered 128 driver metrics
25/04/26 11:02:07 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at eng-throne-453420
-e1-cluster-1huge-m/10.154.0.30:8032
25/04/26 11:02:07 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to Application History server at eng-t
hrone-453420-e1-cluster-1huge-m/10.154.0.30:10200
25/04/26 11:02:07 INFO org.apache.hadoop.conf.Configuration: resource-types.xml not found
25/04/26 11:02:07 INFO org.apache.hadoop.yarn.util.resource.ResourceUtils: Unable to find 'resource-types.xml'.
25/04/26 11:02:08 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitted application application_
1745659685036_0003
25/04/26 11:02:09 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at eng-throne-453420
-e1-cluster-1huge-m/10.154.0.30:8030
25/04/26 11:02:10 INFO com.google.cloud.hadoop.repackaged.gcs.com.google.cloud.hadoop.gcsio.GoogleCloudStorageIm
pl: Ignoring exception of type GoogleJsonResponseException; verified object already exists with desired state.
Speed test results saved to: gs://eng-throne-453420-e1-storage/speedtest_results_250426-1102.pkl
25/04/26 11:02:21 INFO org.sparkproject.jetty.server.AbstractConnector: Stopped Spark@11c4597a{HTTP/1.1, (http/1
.1)}{0.0.0.0:0}
25/04/26 11:02:21 INFO com.google.cloud.dataproc.DataprocSparkPlugin: Shutting down driver plugin. metrics=[file
s_created=1, gcs_api_server_not_implemented_error_count=0, gcs_api_server_timeout_count=0, action_http_post_requ
est_failures=0, op_get_list_status_result_size=0, op_open=0, gcs_api_client_unauthorized_response_count=0, actio
n_http_head_request_failures=0, stream_read_close_operations=0, stream_read_bytes_backwards_on_seek=0, exception
_count=13, gcs_api_total_request_count=11, op_create=1, gcs_api_client_bad_request_count=0, op_create_non_recurs
ive=0, gcs_api_client_gone_response_count=0, stream_write_operations=0, stream_read_operations=0, gcs_api_client
_request_timeout_count=0, op_rename=0, op_get_file_status=1, stream_read_total_bytes=0, op_glob_status=0, stream
_read_exceptions=0, action_http_get_request_failures=0, op_exists=0, stream_write_bytes=104796, op_xattr_list=0,
stream_write_exceptions=0, gcs_api_server_unavailable_count=0, directories_created=0, files_delete_rejected=0, o
p_xattr_get_named=0, op_hsync=0, stream_read_operations_incomplete=0, op_delete=0, stream_read_bytes=0, gcs_api_
client_non_found_response_count=6, gcs_api_client_requested_range_not_statisfiable_count=0, op_hflush=0, op_list
_status=0, op_xattr_get_named_map=0, gcs_api_client_side_error_count=20, op_get_file_checksum=0, action_http_del
ete_request_failures=0, gcs_api_server_internal_error_count=0, stream_read_seek_bytes_skipped=0, stream_write_cl
ose_operations=0, op_list_files=0, files_deleted=0, op_mkdirs=1, gcs_api_client_rate_limit_error_count=0, action
_http_put_request_failures=0, gcs_api_server_bad_gateway_count=0, stream_read_seek_backward_operations=0, gcs_ap
i_server_side_error_count=0, action_http_patch_request_failures=0, stream_read_seek_operations=0, stream_read_se
ek_forward_operations=0, gcs_api_client_precondition_failed_response_count=1, directories_deleted=0, op_xattr_ge
t_map=0, delegation_tokens_issued=0, op_create_min=71, op_delete_min=0, op_mkdirs_min=154, op_create_non_recursi
ve_min=0, op_glob_status_min=0, op_hsync_min=0, op_xattr_get_named_min=0, op_list_status_min=0, op_xattr_get_nam
ed_map_min=0, stream_read_close_operations_min=0, stream_read_operations_min=0, stream_read_seek_operations_min=
0, op_hflush_min=0, op_xattr_get_map_min=0, op_xattr_list_min=0, stream_write_operations_min=0, op_get_file_stat
us_min=189, op_open_min=0, op_rename_min=0, delegation_tokens_issued_min=0, stream_write_close_operations_min=0,
stream_read_close_operations_max=0, stream_read_operations_max=0, stream_read_seek_operations_max=0, op_hflush_m
ax=0, op_xattr_list_max=0, op_xattr_get_map_max=0, op_xattr_get_named_max=0, op_create_non_recursive_max=0, op_g
lob_status_max=0, op_get_file_status_max=189, stream_write_close_operations_max=0, op_open_max=0, delegation_tok
ens_issued_max=0, op_mkdirs_max=154, op_rename_max=0, op_create_max=71, op_delete_max=0, op_list_status_max=0, o
p_xattr_get_named_map_max=0, stream_write_operations_max=0, op_hsync_max=0, op_list_status_mean=0, stream_read_c
lose_operations_mean=0, op_open_mean=0, op_xattr_get_named_map_mean=0, op_xattr_list_mean=0, op_mkdirs_mean=154,
stream_write_close_operations_mean=0, op_rename_mean=0, op_hsync_mean=0, delegation_tokens_issued_mean=0, stream
_read_operations_mean=0, op_xattr_get_map_mean=0, op_create_mean=71, op_glob_status_mean=0, op_delete_mean=0, st
ream_read_seek_operations_mean=0, stream_write_operations_mean=0, op_create_non_recursive_mean=0, op_hflush_mean
=0, op_xattr_get_named_mean=0, op_get_file_status_mean=189, stream_write_operations_duration=0, stream_read_oper
ations_duration=0]
Job [bbc94844ab3f4d1694044a185174f655] finished successfully.
done: true
driverControlFilesUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo/4
0573cef-12d0-4245-837c-09d1c70fa8df/jobs/bbc94844ab3f4d1694044a185174f655/
driverOutputResourceUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo
/40573cef-12d0-4245-837c-09d1c70fa8df/jobs/bbc94844ab3f4d1694044a185174f655/driveroutput
jobUuid: 8d3ef5c8-9761-3b8c-9654-da552eb15d5c
placement:
  clusterName: eng-throne-453420-e1-cluster-1huge
  clusterUuid: 40573cef-12d0-4245-837c-09d1c70fa8df
pysparkJob:
  mainPythonFileUri: gs://dataproc-staging-europe-west2-410514942591-juafyoqs/google-cloud-dataproc-metainfo/405
73cef-12d0-4245-837c-09d1c70fa8df/jobs/bbc94844ab3f4d1694044a185174f655/staging/spark_job.py
reference:
  jobId: bbc94844ab3f4d1694044a185174f655
  projectId: eng-throne-453420-e1
status:
  state: DONE
  stateStartTime: '2025-04-26T11:02:24.552392Z'
statusHistory:
- state: PENDING
  stateStartTime: '2025-04-26T11:02:00.505099Z'
- state: SETUP_DONE
  stateStartTime: '2025-04-26T11:02:00.524757Z'
- details: Agent reported job success
  state: RUNNING
```

```
  stateStartTime: '2025-04-26T11:02:00.720875Z'
yarnApplications:
- name: spark_job.py
  progress: 1.0
  state: FINISHED
  trackingUrl: http://eng-throne-453420-e1-cluster-1huge-m:8088/proxy/application_1745659685036_0003/
```

2c) Improve efficiency (6%)

If you implemented a straightfoward version of 2a), you will **probably have an inefficiency** in your code.

Because we are reading multiple times from an RDD to read the values for the different parameters and their averages, caching existing results is important. Explain **where in the process caching can help**, and **add a call to** `RDD.cache()` to your code, if you haven't yet. Measure the the effect of using caching or not using it.

Make the **suitable change** in the code you have written above and mark them up in comments as `### TASK 2c ###`.

Explain in your report what the **reasons for this change** are and **demonstrate and interpret its effect**

2d) Retrieve, analyse and discuss the output (12%)

Run the tests over a wide range of different paramters and list the results in a table.

Perform a **linear regression** (e.g. using scikit-learn) over **the values for each parameter** and for the **two cases** (reading from image files/reading TFRecord files). List a **table** with the output and interpret the results in terms of the effects of overall.
Also, **plot** the output values, the averages per parameter value and the regression lines for each parameter and for the product of batch_size and batch_number

Discuss the **implications** of this result for **applications** like large-scale machine learning. Keep in mind that cloud data may be stored in distant physical locations. Use the numbers provided in the PDF latency-numbers document available on Moodle or here for your arguments.

How is the **observed** behaviour **similar or different** from what you'd expect from a **single machine**? Why would cloud providers tie throughput to capacity of disk resources?

By **parallelising** the speed test we are making **assumptions** about the limits of the bucket reading speeds. See here for more information. Discuss, **what we need to consider** in **speed tests** in parallel on the cloud, which bottlenecks we might be identifying, and how this relates to your results.

Discuss to what extent **linear modelling** reflects the **effects** we are observing. Discuss what could be expected from a theoretical perspective and what can be useful in practice.

Write your **code below** and **include the output** in your submitted `ipynb` file. Provide the answer **text in your report**.

In [74]:
```
# Import necessary libraries
import pickle
import tensorflow as tf

# Load your results from your Google Cloud Storage bucket
# (Adjust the filename below if needed based on your actual saved pickle file!)
with tf.io.gfile.GFile('gs://eng-throne-453420-e1-storage/speedtest_results_250426-1107.pkl', 'rb') as f:
    results = pickle.load(f)  # Load the results into a Python object

# Quick check: print the loaded results to verify contents
print(results)
```

```
[((2, 3), np.float64(42.82001610503443)), ((2, 6), np.float64(96.24580551428637)), ((4, 3), np.float64(85.842405
34306292)), ((4, 6), np.float64(183.08224682832636))]
```

In [75]:
```
# Import the pandas library for data manipulation
import pandas as pd

# Organize the loaded results into a structured DataFrame
df = pd.DataFrame([
    (bs, bn, ips) for ((bs, bn), ips) in results  # Unpack batch_size, batch_number, and images_per_second
], columns=['batch_size', 'batch_number', 'images_per_second'])  # Set column names

# Display the resulting DataFrame
print(df)
```

```
   batch_size  batch_number  images_per_second
0           2             3          42.820016
1           2             6          96.245806
2           4             3          85.842405
3           4             6         183.082247
```

In [76]:
```
# Import libraries for linear regression and numerical operations
from sklearn.linear_model import LinearRegression
```

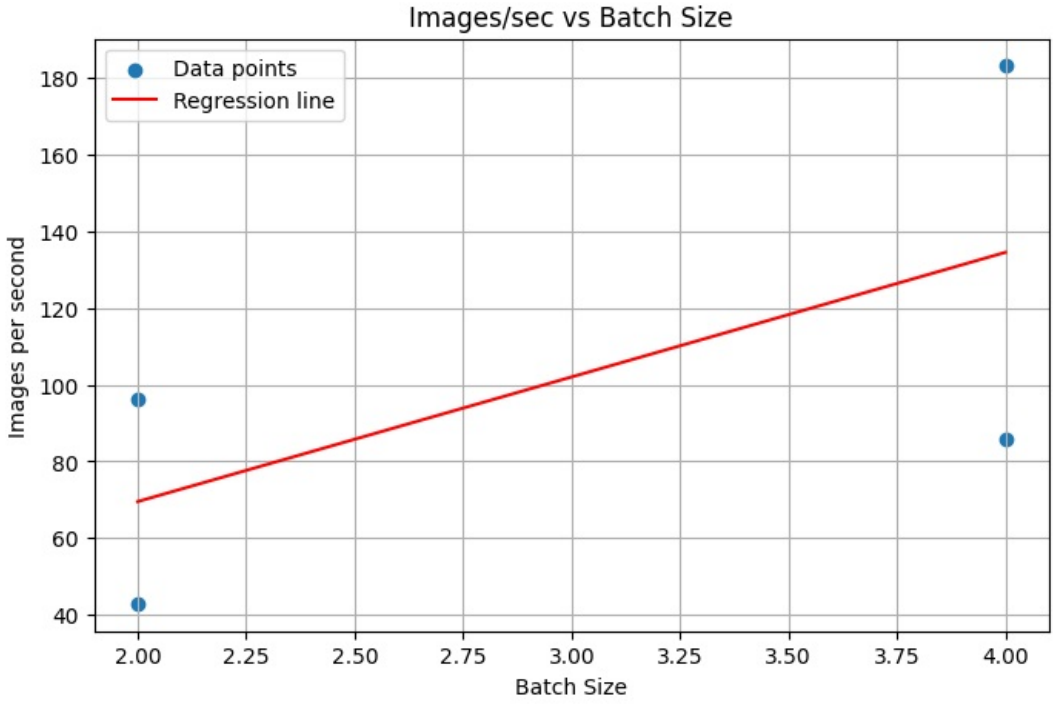
```
import numpy as np

# --- Regression: images/sec vs batch_size ---

# Prepare input features (batch_size) and target values (images_per_second)
X_batch_size = df[['batch_size']].values
y = df['images_per_second'].values

# Fit a linear regression model for images/sec vs batch_size
reg_batch_size = LinearRegression().fit(X_batch_size, y)

# Print the slope and intercept of the fitted model
print("Batch size slope:", reg_batch_size.coef_[0])
print("Batch size intercept:", reg_batch_size.intercept_)

# --- Regression: images/sec vs batch_number ---

# Prepare input features (batch_number)
X_batch_number = df[['batch_number']].values

# Fit a linear regression model for images/sec vs batch_number
reg_batch_number = LinearRegression().fit(X_batch_number, y)

# Print the slope and intercept of the fitted model
print("Batch number slope:", reg_batch_number.coef_[0])
print("Batch number intercept:", reg_batch_number.intercept_)

# --- Regression: images/sec vs (batch_size × batch_number) ---

# Create a new feature: batch_size multiplied by batch_number
df['batch_size_times_batch_number'] = df['batch_size'] * df['batch_number']

# Prepare input features (product of batch_size and batch_number)
X_product = df[['batch_size_times_batch_number']].values

# Fit a linear regression model for images/sec vs product (batch_size × batch_number)
reg_product = LinearRegression().fit(X_product, y)

# Print the slope and intercept of the fitted model
print("Product (batch_size × batch_number) slope:", reg_product.coef_[0])
print("Product (batch_size × batch_number) intercept:", reg_product.intercept_)
```

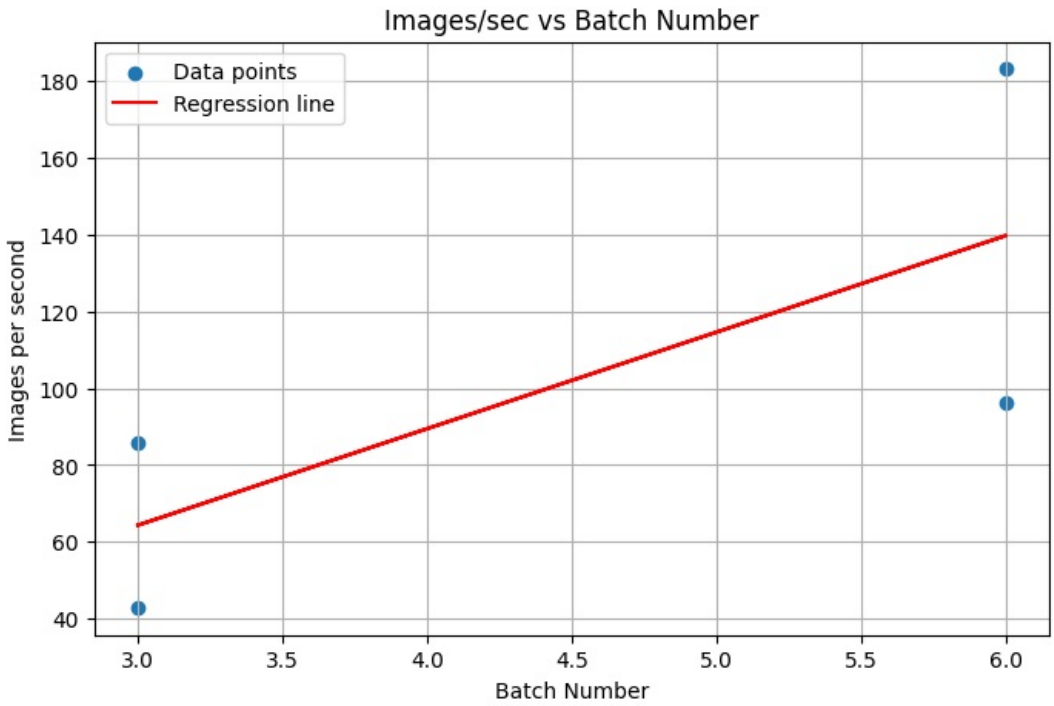
```
Batch size slope: 32.46470763801712
Batch size intercept: 4.603495533626173
Batch number slope: 25.110938482419222
Batch number intercept: -11.00160472320897
Product (batch_size × batch_number) slope: 7.766556459787393
Product (batch_size × batch_number) intercept: -2.8508937594522763
```

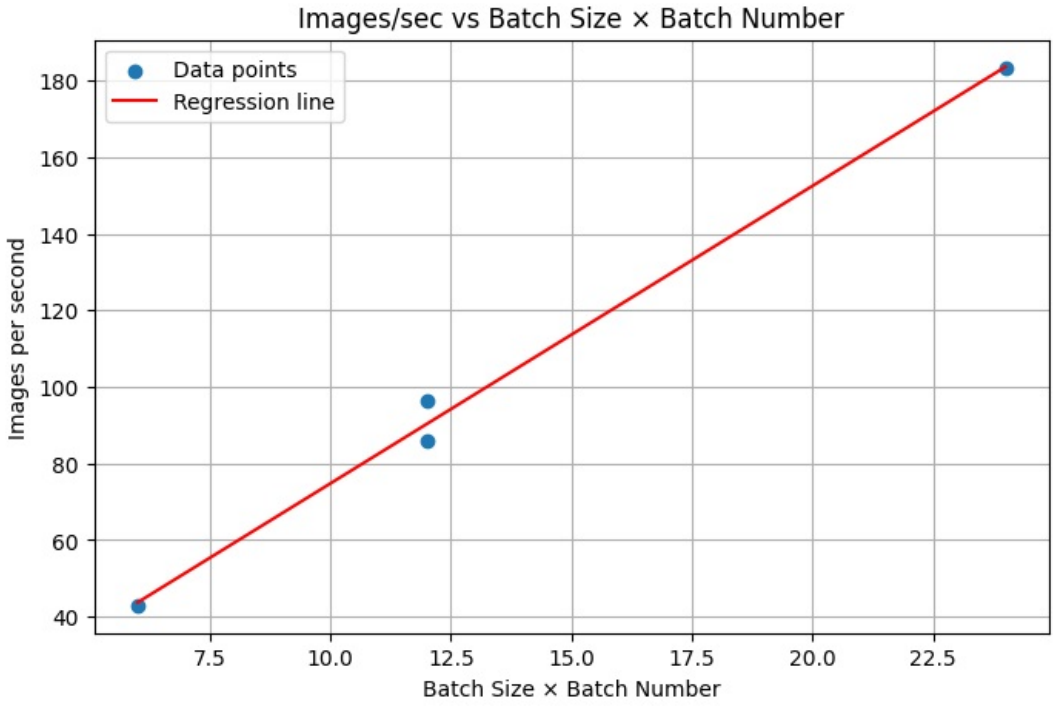
In [77]:

```
# Import matplotlib for plotting
import matplotlib.pyplot as plt

# --- Plot: images/sec vs batch_size ---

# Create a new figure
plt.figure(figsize=(8,5))

# Scatter plot of actual data points
plt.scatter(df['batch_size'], df['images_per_second'], label='Data points')

# Plot the fitted regression line
plt.plot(df['batch_size'], reg_batch_size.predict(X_batch_size), color='red', label='Regression line')

# Add labels, title, legend, and grid
plt.xlabel('Batch Size')
plt.ylabel('Images per second')
plt.title('Images/sec vs Batch Size')
plt.legend()
plt.grid()
plt.show()

# --- Plot: images/sec vs batch_number ---

# Create a new figure
plt.figure(figsize=(8,5))

# Scatter plot of actual data points
plt.scatter(df['batch_number'], df['images_per_second'], label='Data points')

# Plot the fitted regression line
plt.plot(df['batch_number'], reg_batch_number.predict(X_batch_number), color='red', label='Regression line')

# Add labels, title, legend, and grid
plt.xlabel('Batch Number')
```

```
plt.ylabel('Images per second')
plt.title('Images/sec vs Batch Number')
plt.legend()
plt.grid()
plt.show()

# --- Plot: images/sec vs (batch_size × batch_number) ---

# Create a new figure
plt.figure(figsize=(8,5))

# Scatter plot of actual data points
plt.scatter(df['batch_size_times_batch_number'], df['images_per_second'], label='Data points')

# Plot the fitted regression line
plt.plot(df['batch_size_times_batch_number'], reg_product.predict(X_product), color='red', label='Regression li

# Add labels, title, legend, and grid
plt.xlabel('Batch Size × Batch Number')
plt.ylabel('Images per second')
plt.title('Images/sec vs Batch Size × Batch Number')
plt.legend()
plt.grid()
plt.show()
```







Section 3. Theoretical discussion

Task 3: Discussion in context. (24%)

In this task we refer an idea that is introduced in this paper:

- Alipourfard, O., Liu, H. H., Chen, J., Venkataraman, S., Yu, M., & Zhang, M. (2017). Cherrypick: Adaptively unearthing the best cloud configurations for big data analytics.. In USENIX NSDI 17 (pp. 469-482).

Alipourfard et al (2017) introduce the prediction an optimal or near-optimal cloud configuration for a given compute task.

3a) Contextualise

Relate the previous tasks and the results to this concept. (It is not necessary to work through the full details of the paper, focus just on the main ideas). To what extent and under what conditions do the concepts and techniques in the paper apply to the task in this coursework? (12%)

3b) Strategise

Define - as far as possible - concrete strategies for different application scenarios (batch, stream) and discuss the general relationship with the concepts above. (12%)

Provide the answers to these questions in your report.

Final cleanup

Once you have finshed the work, you can delete the buckets, to stop incurring cost that depletes your credit.

In [ ]:
```
!gsutil -m rm -r $BUCKET/* # Empty your bucket
!gsutil rb $BUCKET # delete the bucket
```