Welcome to Week 02

W261 - Machine Learning at Scale

Housekeeping

Teaching Team

Name	Day Job	w261 Role	Time Zone	Email/Slack
Jimi Shanahan	deep learning | computer vision | edge-based computing | large scale data science consultant	Lecturer | Course Creator | Course Coordinator	PST (SF)	jimi@ischool.berkeley.edu
Ramki Gummadi	CEO Argonaut AI	Lecturer	PST (SV)	rkgmd1729@gmail.com
Vinicio De Sola	Senior Data Scientist Aspen Capital (W205 Lecturer)	Lecturer	PST+3 (NY)	vinicio.desola@ischool.berkeley.edu
Luis Villarreal	Cloud Enterprise Architect, Accenture	Lecturer	PST+2 (Texas)	luis.villarreal@ischool.berkeley.edu
Toby Petty	W261 Alum	TA	NYC	
Sam Gomez	W261 Alum	TA	PST	
Zachary G.	W261 Alum	TA	EST	

Google Hadoop/Spark Cluster (aka DataProc)

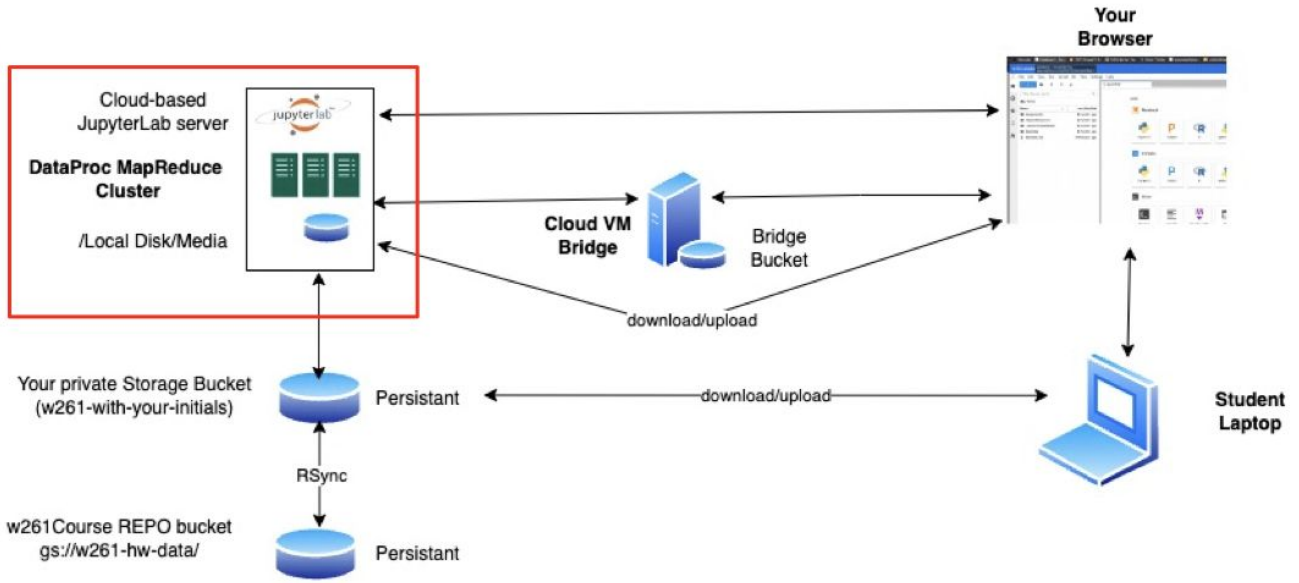
Click here: https://console.cloud.google.com/

JupterLab PaaS at $0.20 per hour

Dataproc is a fully managed and highly scalable service for running Hadoop, Apache Spark, Apache Flink, Presto, and 30+ open source tools and frameworks.

Setup a hadoop/Spark (aka DataProc) cluster on Google cloud with a **single node** (4 CPUs)
The cluster costs $0.20 per hour and has 100 Gig of local Disk

~$35 per week
~ $300 for 9 weeks

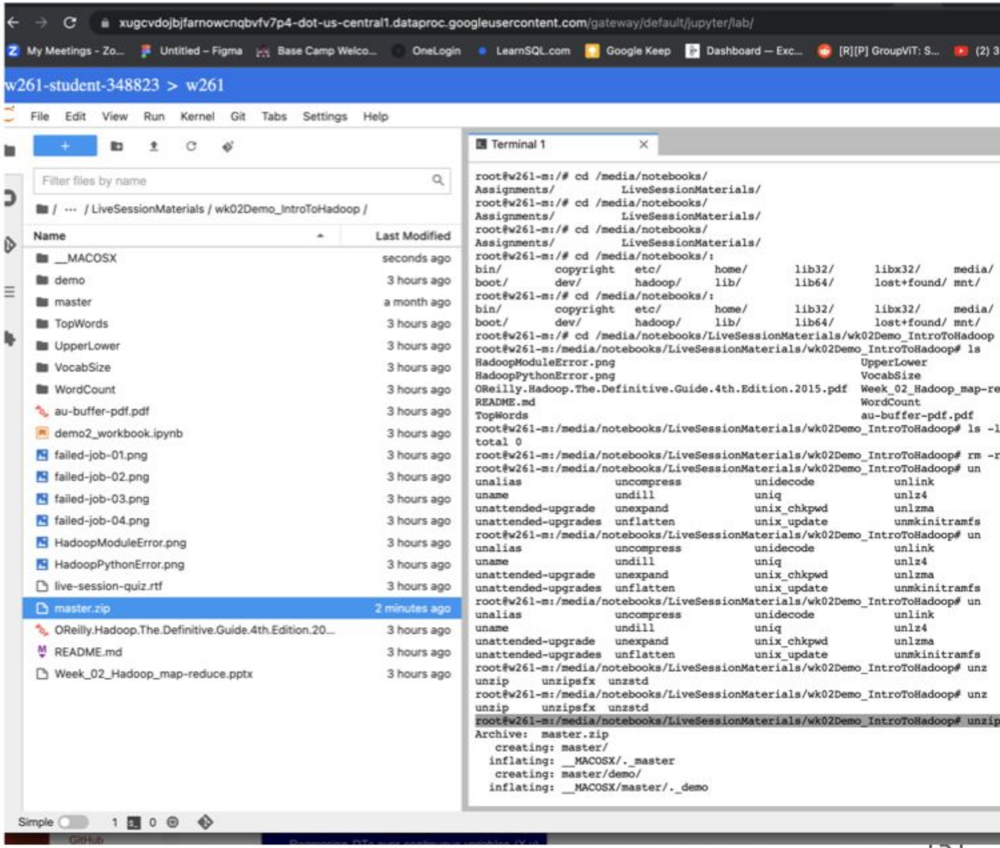


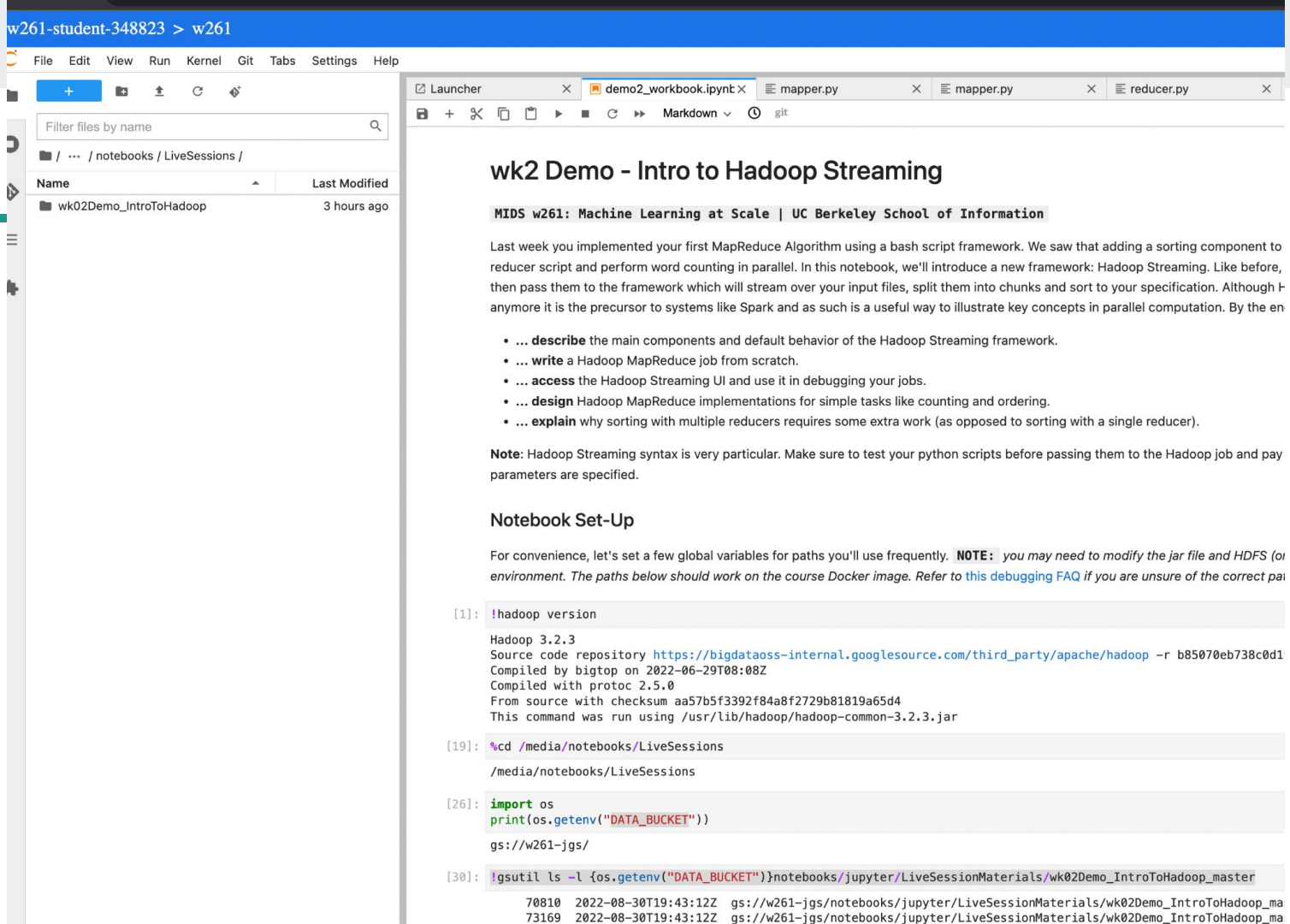
Click here to access CloudShell/Bridge: https://console.cloud.google.com/

Orchestration shell script cmd: gsutil cat gs://w261-hw-data/w261_env.sh | bash -euo pipefail

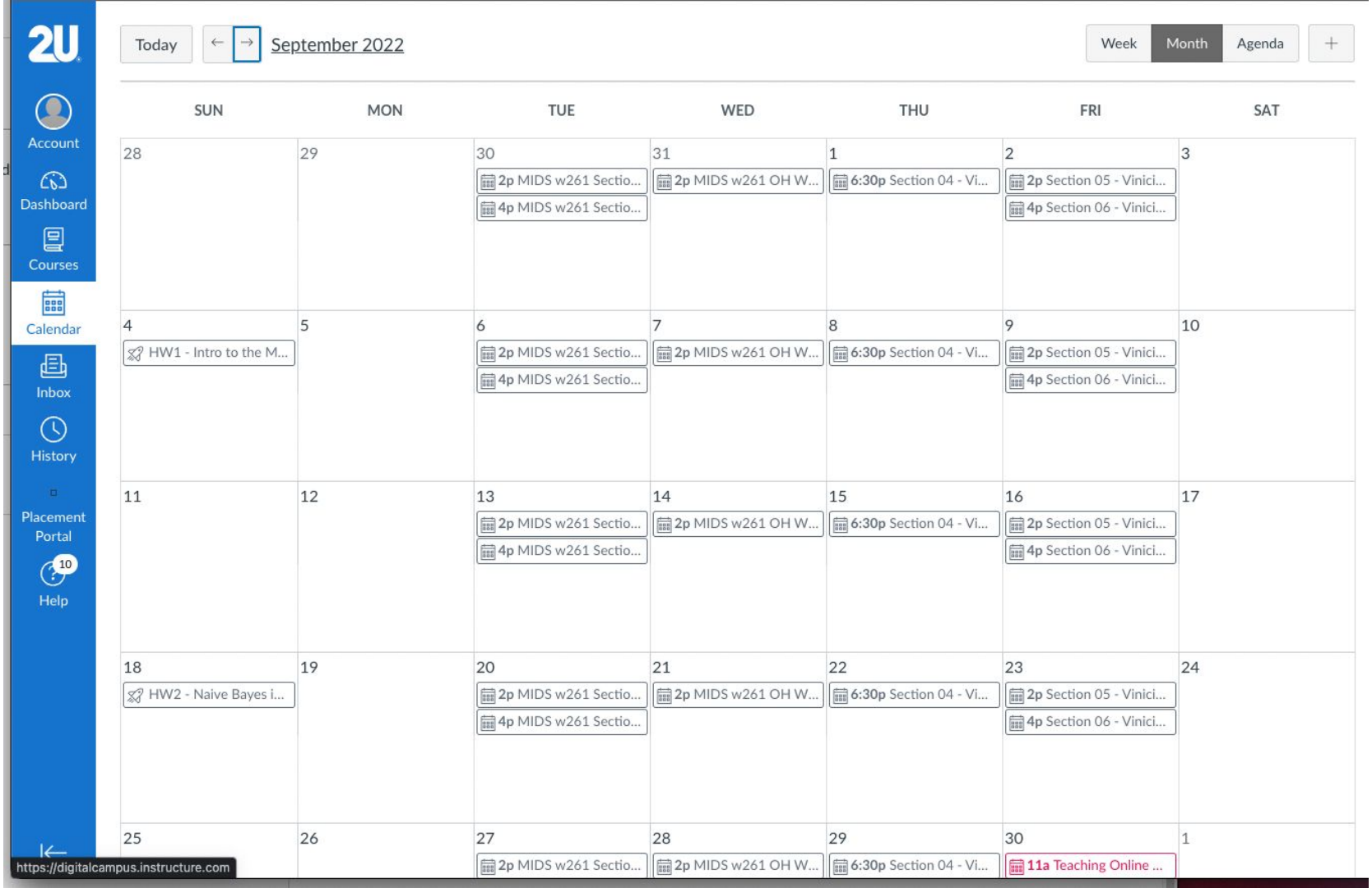
Copy a folder from your laptop to the Media disk on the master node (of Hadoop cluster)

- Zip file on laptop (e.g., zip ***master*** folder)
- Drag ***master.zip*** file to destination folder on Media
- Open terminal and run the following commands
- `cd /media/notebooks/LiveSessionMaterials/wk02Demo_IntroToHadoop`
- `unzip master.zip`



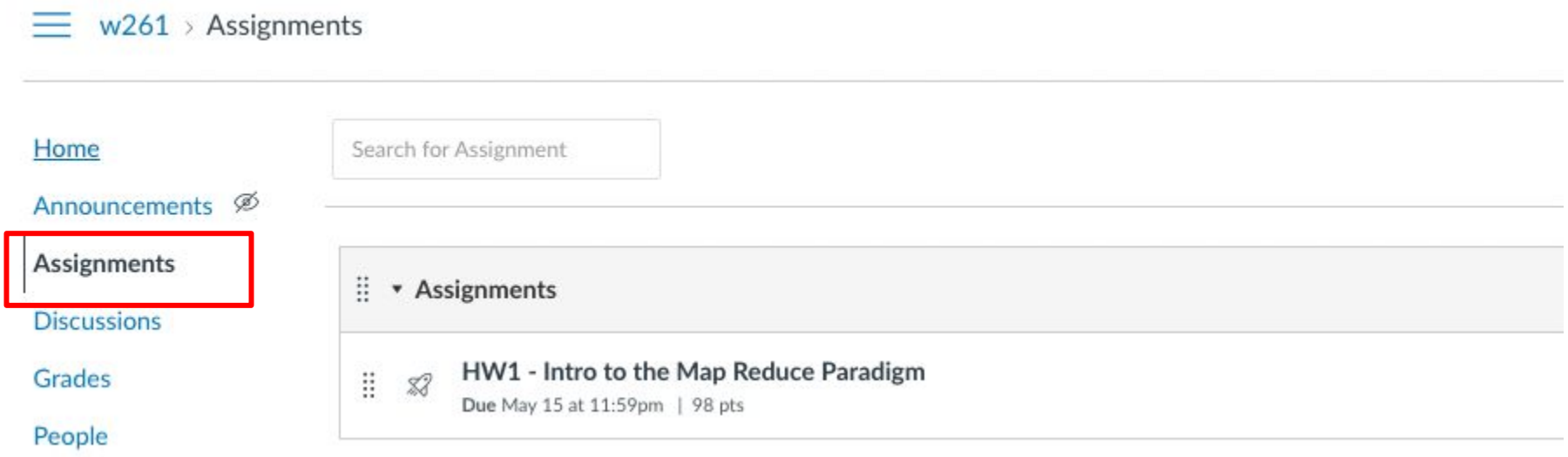

w261-student-348823 > w261
File Edit View Run Kernel Git Tabs Settings Help
Filter files by name
/ ··· / notebooks / LiveSessions /
Name
Last Modified
wk02Demo_IntroToHadoop
3 hours ago
Launcher
demo2_workbook.ipynb
mapper.py
mapper.py
reducer.py
Markdown
git
wk2 Demo - Intro to Hadoop Streaming
MIDS w261: Machine Learning at Scale | UC Berkeley School of Information
Last week you implemented your first MapReduce Algorithm using a bash script framework. We saw that adding a sorting component to
reducer script and perform word counting in parallel. In this notebook, we'll introduce a new framework: Hadoop Streaming. Like before,
then pass them to the framework which will stream over your input files, split them into chunks and sort to your specification. Although H
anymore it is the precursor to systems like Spark and as such is a useful way to illustrate key concepts in parallel computation. By the en
... describe the main components and default behavior of the Hadoop Streaming framework.
... write a Hadoop MapReduce job from scratch.
... access the Hadoop Streaming UI and use it in debugging your jobs.
... design Hadoop MapReduce implementations for simple tasks like counting and ordering.
... explain why sorting with multiple reducers requires some extra work (as opposed to sorting with a single reducer).
Note: Hadoop Streaming syntax is very particular. Make sure to test your python scripts before passing them to the Hadoop job and pay
parameters are specified.
Notebook Set-Up
For convenience, let's set a few global variables for paths you'll use frequently. NOTE: you may need to modify the jar file and HDFS (o
environment. The paths below should work on the course Docker image. Refer to this debugging FAQ if you are unsure of the correct pa
[1]: !hadoop version
Hadoop 3.2.3
Source code repository https://bigdataoss-internal.googlesource.com/third_party/apache/hadoop -r b85070eb738c0d1
Compiled by bigtop on 2022-06-29T08:08Z
Compiled with protoc 2.5.0
From source with checksum aa57b5f3392f84a8f2729b81819a65d4
This command was run using /usr/lib/hadoop/hadoop-common-3.2.3.jar
[19]: %cd /media/notebooks/LiveSessions
/media/notebooks/LiveSessions
[26]: import os
print(os.getenv("DATA_BUCKET"))
gs://w261-jgs/
[30]: !gsutil ls -l {os.getenv("DATA_BUCKET")}notebooks/jupyter/LiveSessionMaterials/wk02Demo_IntroToHadoop_master
70810 2022-08-30T19:43:12Z gs://w261-jgs/notebooks/jupyter/LiveSessionMaterials/wk02Demo_IntroToHadoop_ma
73169 2022-08-30T19:43:12Z gs://w261-jgs/notebooks/jupyter/LiveSessionMaterials/wk02Demo_IntroToHadoop_ma

Preview of HW 1 And submission instructions


Today
September 2022
Week
Month
Agenda
Account
Dashboard
Courses
Calendar
Inbox
History
Placement Portal
Help
10
SUN
MON
TUE
WED
THU
FRI
SAT
28
29
30
2p MIDS w261 Sectio...
4p MIDS w261 Sectio...
31
2p MIDS w261 OH W...
1
6:30p Section 04 - Vi...
2
2p Section 05 - Vinici...
4p Section 06 - Vinici...
3
4
HW1 - Intro to the M...
5
6
7
8
9
10
11
12
13
14
15
16
17
18
HW2 - Naive Bayes i...
19
20
21
22
23
24
25
26
27
28
29
30
11a Teaching Online ...
1
https://digitalcampus.instructure.com

HW submission via DigitalCampus (LMS)

https://digitalcampus.instructure.com/courses/4868/assignments/syllabus



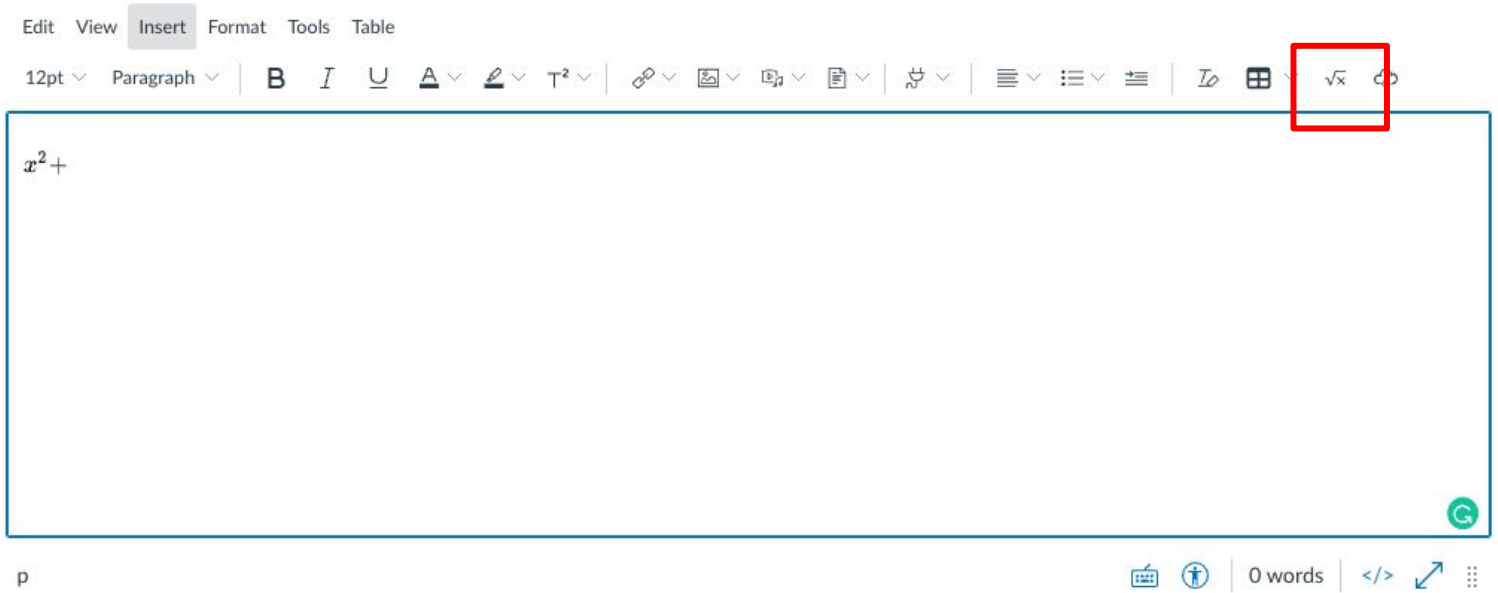
Question 1

5 pts

`The Caterpillar and Alice looked at each other for some time in silence: at last the Caterpillar took the hookah out of its mouth, and addressed her in a languid, sleepy voice. "Who are you?" said the Caterpillar.`

-- Lewis Carroll, *Alice's Adventures in Wonderland*, Chapter 4

Tell us about yourself! Briefly describe where you live, how far along you are in MIDS, what other classes you are taking and what you want to get out of w261.



Question 1

5 pts

The Caterpillar and Al

languid, sleepy voice.

-- Lewis Carroll, *Alice's*

Tell us about yourself!

w261.

addressed her in a

to get out of

Edit View Insert F

12pt Paragraph

p

0 words

Equation Editor

Basic | Greek | Operators | Relationships | Arrows | Delimiters | Misc

$x_{\square}$ $x^{\square}$ $\frac{n}{m}$ $\sqrt{}$ $\sqrt[n]{}$ $\langle$ $\rangle$ $\binom{n}{m}$ f $/$ $+$ $-$ $\pm$ $\mp$ $\bullet$ $=$ $\times$

$\div$ $*$ $\therefore$ $\because$ Σ Π $\amalg$ $/$ $\mathbb{N}$ $\mathbb{P}$ $\mathbb{Z}$ $\mathbb{Q}$ $\mathbb{R}$ $\mathbb{C}$ $\mathbb{H}$ $\overline{x}$ $\hat{x}$

x^2 +

Directly Edit LaTeX

x^2+

Cancel Done

Question 1

The Caterpillar and Alice looked at each other for some time in silence: at last the Caterpillar took the hookah out of its mouth, and addressed her in a languid, sleepy voice. "Who are you?" sai

-- Lewis Carroll, *Alice's Adventures in Wonderland*, Chapter 4

Question 1: Tell us about yourself! Briefly describe where you live,

Edit View Insert Format Tools Table

12pt Paragraph

This s the text

w_2^3

p ▸ img

Equation Editor

Basic | Greek | Operators | Relationships | Arrows | Delimiters | Misc

w_2^3

Directly Edit LaTeX

w_2^3

Cancel Done

Question 2

Question 1: Introductions

`The Caterpillar and Alice looked at each other for some time in silence: at last the Caterpillar took the hookah out of its mouth, and addressed her in a languid, sleepy voice. "Who are you?" said the Caterpillar.`

-- Lewis Carroll, _Alice's Adventures in Wonderland_, Chapter 4

Tell us about yourself! Briefly describe where you live, how far along you are in MIDS, what other classes you are taking and what you want to get out of w261.

> Type your response here!

Question 2: Bias - Variance

In 1-2 paragraphs (~200 and *absolutely no more* than 300 words!), explain the bias-variance trade off. Describe what it means to "decompose" sources of error. How is this used in machine learning? Please use mathematical equation(s) to support your explanation. (Use `$` signs to take advantage of $LaTeX$ formatting, eg. `$f(x)$` will look like: $f(x)$). Please also cite any sources that informed your answer.

> Type your response here!

Question 3: Tokenizing

A number of our assignments this term will involve extracting information from text. A common preprocessing step when working with raw files is to 'tokenize' (i.e. extract words from) the text. Within the field of Natural Language Processing a lot of thought goes into what specific tokenizing makes most sense for a given task. For example, you might choose to remove punctuation or to consider punctuation symbols 'tokens' in their own right. **In this question you'll use the Python `re` module to create a tokenizer to use when you perform WordCount on the *Alice In Wonderland* text.**

Q3 Tasks:

- **a) short response:** In the Naive Bayes algorithm (which we'll implement next week), we'll estimate the *likelihood* of a word by counting the number of times it appears and dividing by the size of the vocabulary (total number of unique words). Using the text: *"Alice had an adventure that took alice to wonderland"*, give a concrete example of how two different tokenizers could cause us to get two different results on this calculation. [`HINT` : *you should not need to read up on Naive Bayes to answer this question*]
- **b) short response:** When tokenizing in this assignment we'll remove punctuation and discard numerical digits by making everything lowercase and then capturing only consecutive letters a to z. Suppose **`tokenize(x)`** is a Python function that performs the desired tokenization. What would **`tokenize("By-the-bye, what became of Alice's 12 hats?!")`** output? Type the answer in the space provided below.
- **c) code:** Fill in the regular expression pattern in the cell labeled `part c` so that the subsequent call to `re.findall(RE_PATTERN, ...)` returns the

Question 1: Introductions

`The Caterpillar and Alice looked at each other for some time in silence: at last the Caterpillar took the hookah out of its mouth, and addressed her in a languid, sleepy voice. "Who are you?" said the Caterpillar.`

-- Lewis Carroll, *Alice's Adventures in Wonderland*, Chapter 4

Tell us about yourself! Briefly describe where you live, how far along you are in MIDS, what other classes you are taking and what you want to get out of w261.

Question 2: Bias - Variance

In 1-2 paragraphs (~200 and *absolutely no more* than 300 words!), explain the bias-variance trade off. Describe what it means to "decompose" sources of error. How is this used in machine learning? Please use mathematical equation(s) to support your explanation. (Use `$` signs to take advantage of $LaTeX$ formatting, eg. `$f(x)$` will look like: $f(x)$. Please also cite any sources that informed your answer.

Question 3: Tokenizing Pick 3 questions, 5 pts per question

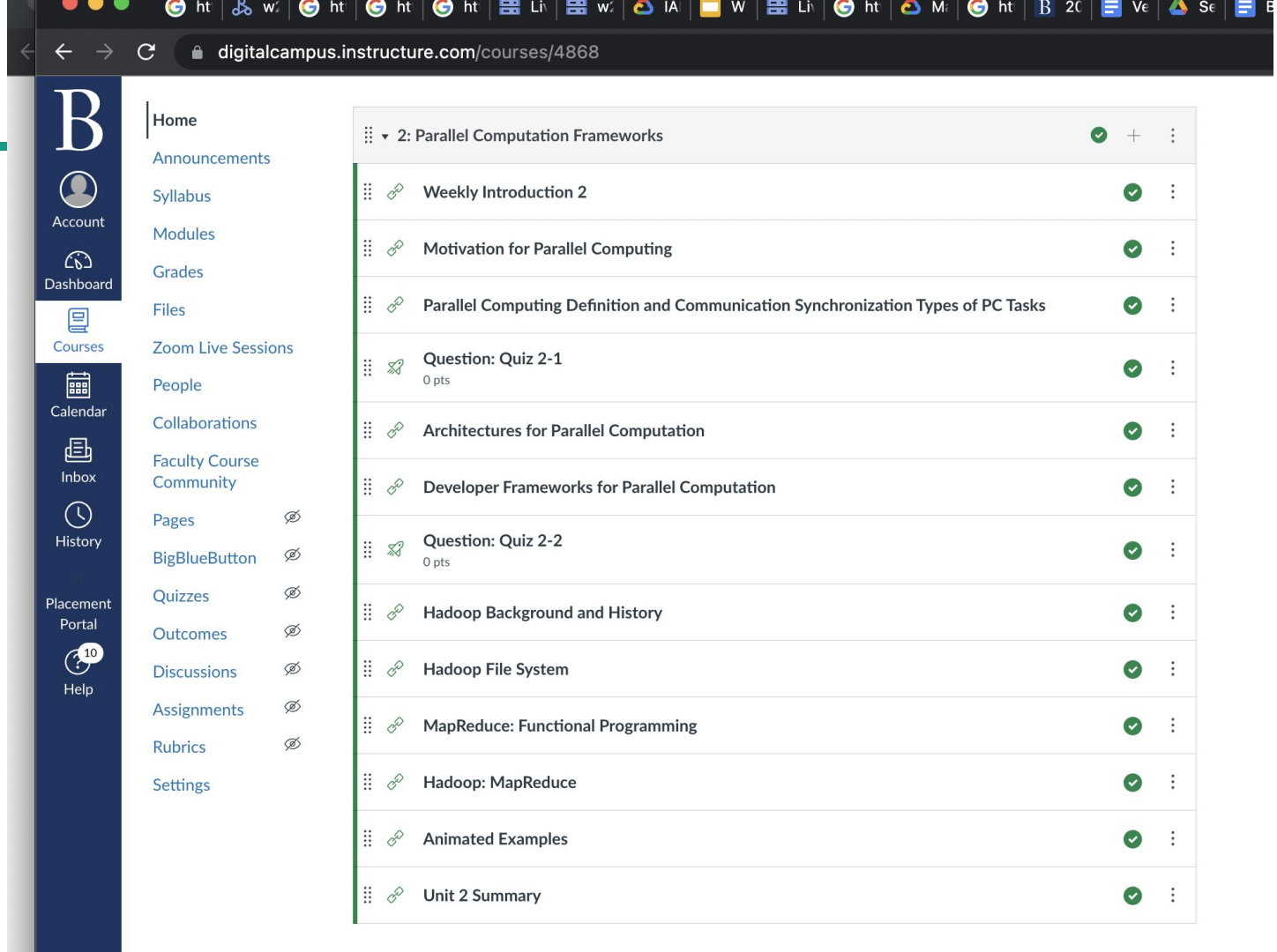
3.a

Welcome to Week 02!

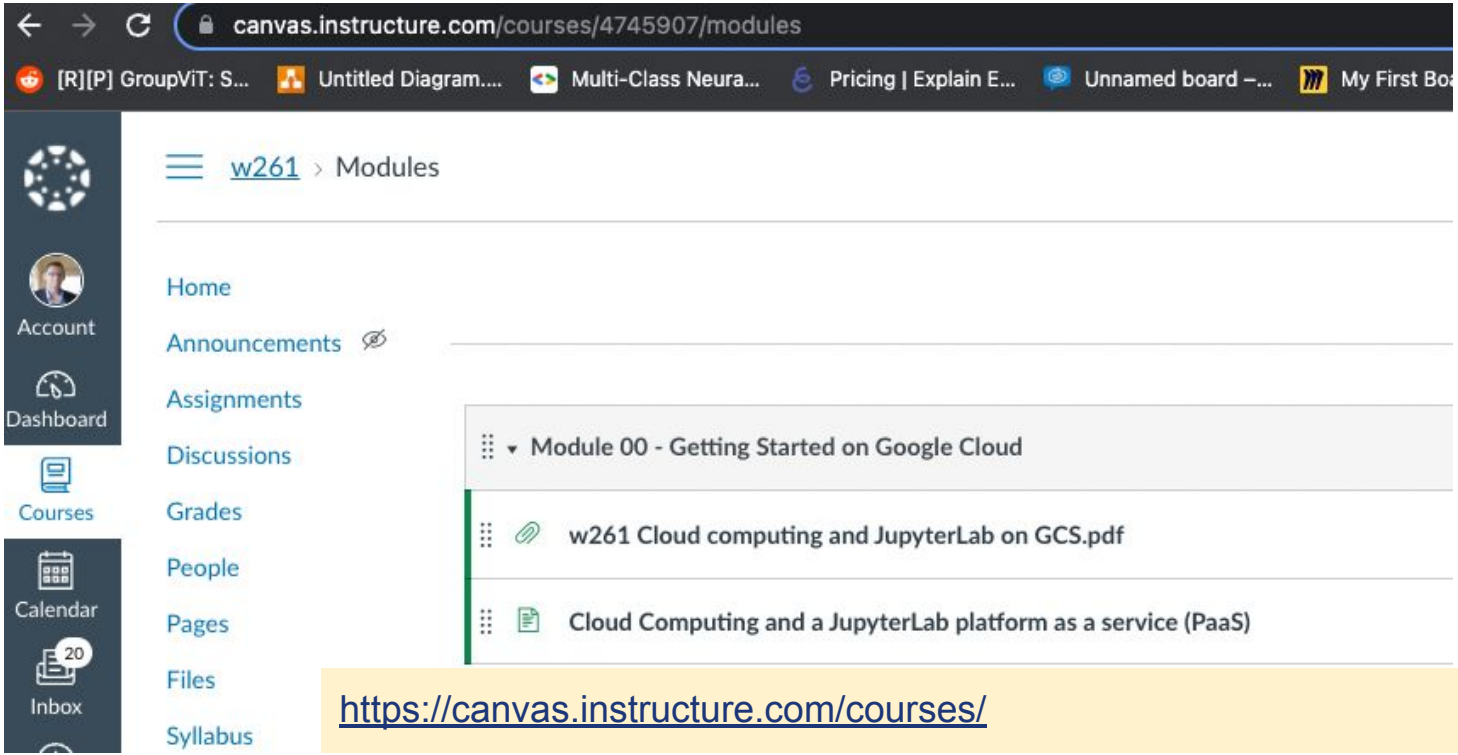
W261 - Machine Learning at Scale

JupyterLab PaaS Map-Reduce cluster on GC setup

Complete online/offline


digitalcampus.instructure.com/courses/4868
Account
Dashboard
Courses
Calendar
Inbox
History
Placement Portal
Help
10
Home
Announcements
Syllabus
Modules
Grades
Files
Zoom Live Sessions
People
Collaborations
Faculty Course Community
Pages
BigBlueButton
Quizzes
Outcomes
Discussions
Assignments
Rubrics
Settings
2: Parallel Computation Frameworks
Weekly Introduction 2
Motivation for Parallel Computing
Parallel Computing Definition and Communication Synchronization Types of PC Tasks
Question: Quiz 2-1
0 pts
Architectures for Parallel Computation
Developer Frameworks for Parallel Computation
Question: Quiz 2-2
0 pts
Hadoop Background and History
Hadoop File System
MapReduce: Functional Programming
Hadoop: MapReduce
Animated Examples
Unit 2 Summary

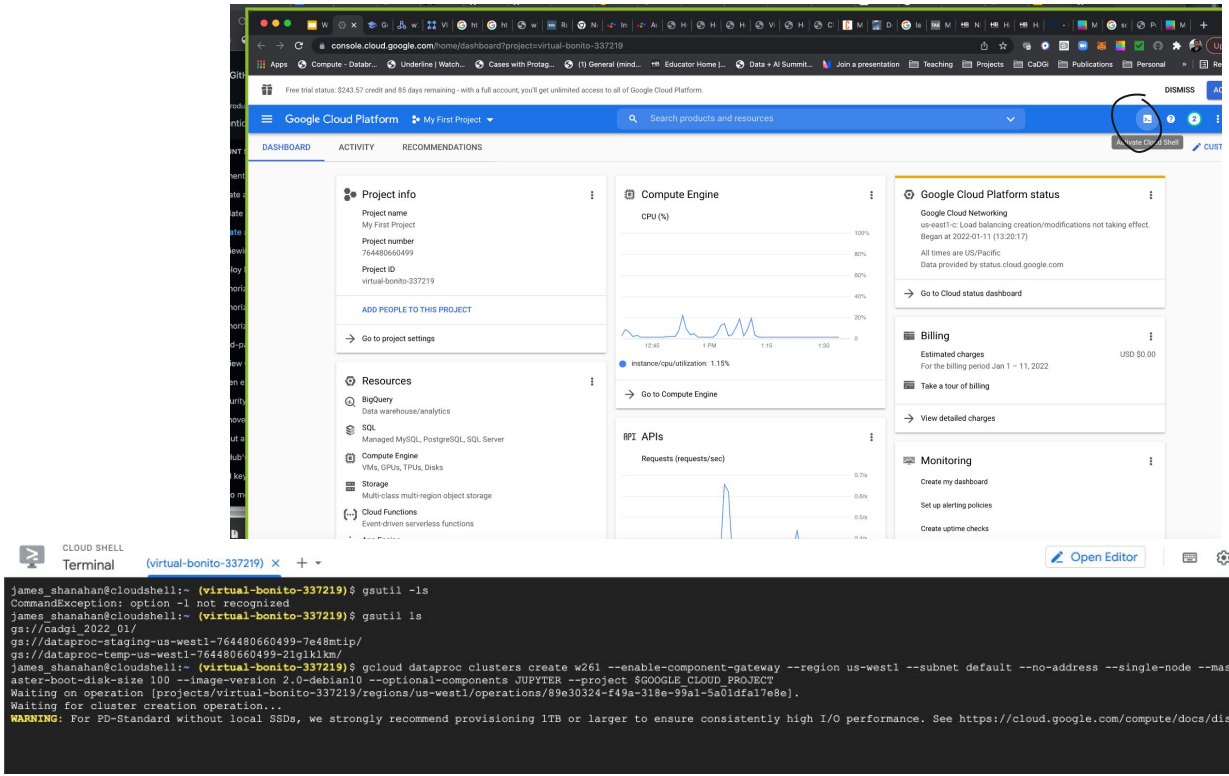
For JupyterLab PaaS setup: Check Module 00 on DigitalCampus



https://canvas.instructure.com/courses/

startup a hadoop cluster (aka dataproc) on google cloud with a single node (4cpus)

Click here: https://console.cloud.google.com/



Live Session #2

Live Session #2

Lab Notebook is located here (contains 4-5 tasks to complete):

-

The master solution for each weekly lab (from Week 2 onwards) will be published on each Saturday Midday.

Breakout 1: WordCount in Hadoop MapReduce

- Breakout 1 Tasks:
- Reducer: stateful or stateless?

Breakout 2: Uppercase and Lowercase Counts

- Breakout 2 Tasks:
- Solution Mapper
- Stateless and stateful mapper

Breakout 3: Number of Unique Words

- Breakout 3 Tasks:
- Solution Reducer

Breakout 4: Secondary Sort

- Breakout 4 Tasks:
- Solution tasks
- solution mapper.py

Breakout 5: Tracking Down Errors in Python Code

- Follow-Up:

Lab 2

Breakout 1: WordCount in Hadoop MapReduce

Breakout 2: Uppercase and Lowercase Counts

Breakout 3: Number of Unique Words

Breakout 4: Secondary Sort

Exercise 5: debugging in Hadoop

Hadoop Streaming API

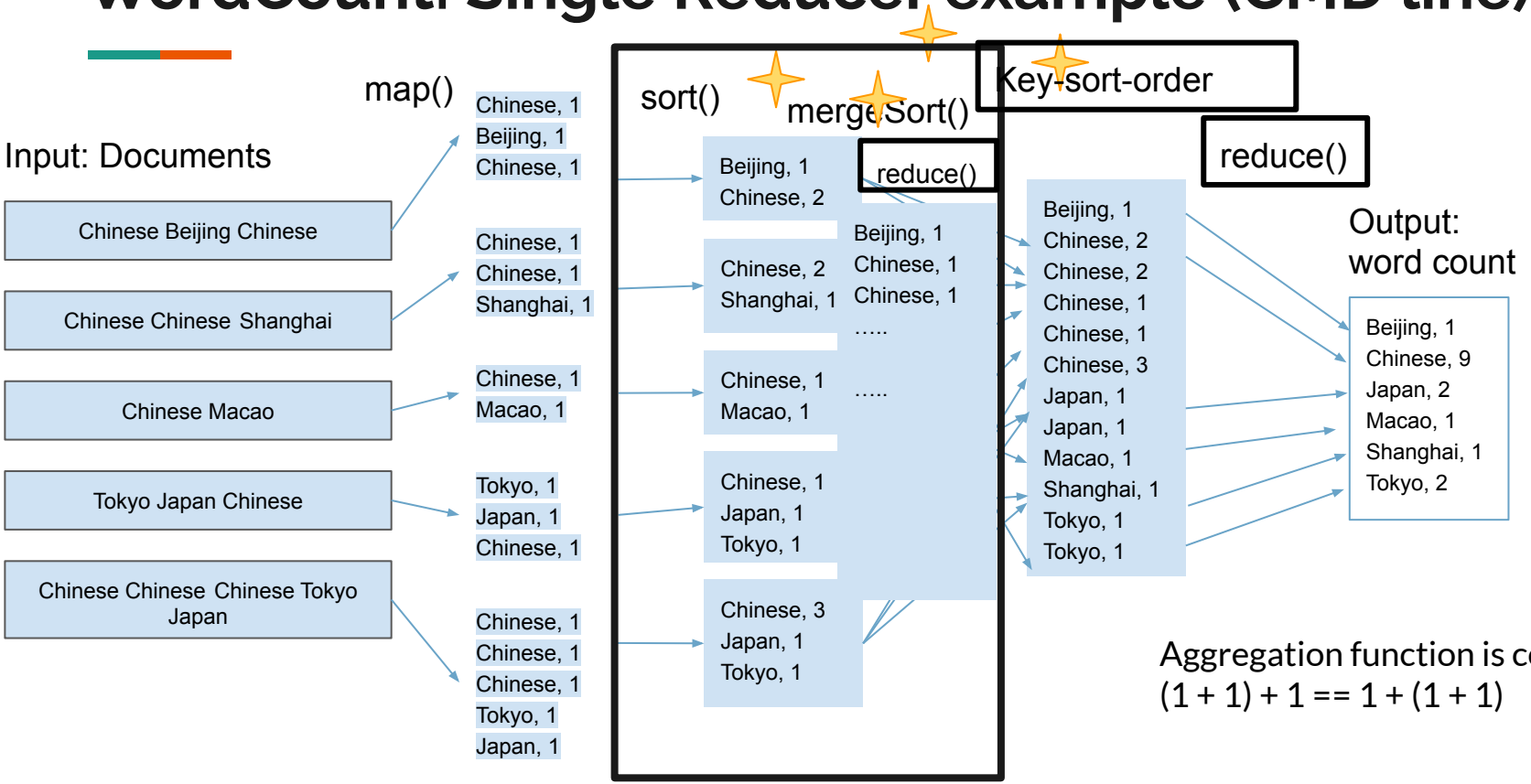
Hadoop streaming is a utility that comes with the Hadoop distribution. The utility allows you to create and run Map/Reduce jobs with any executable or script as the mapper and/or the reducer.

Have a look here to learn more about the specifics of Hadoop Streaming:

- https://hadoop.apache.org/docs/r2.6.0/hadoop-mapreduce-client/hadoop-mapreduce-client-core/HadoopStreaming.html
- https://data-flair.training/blogs/hadoop-streaming/

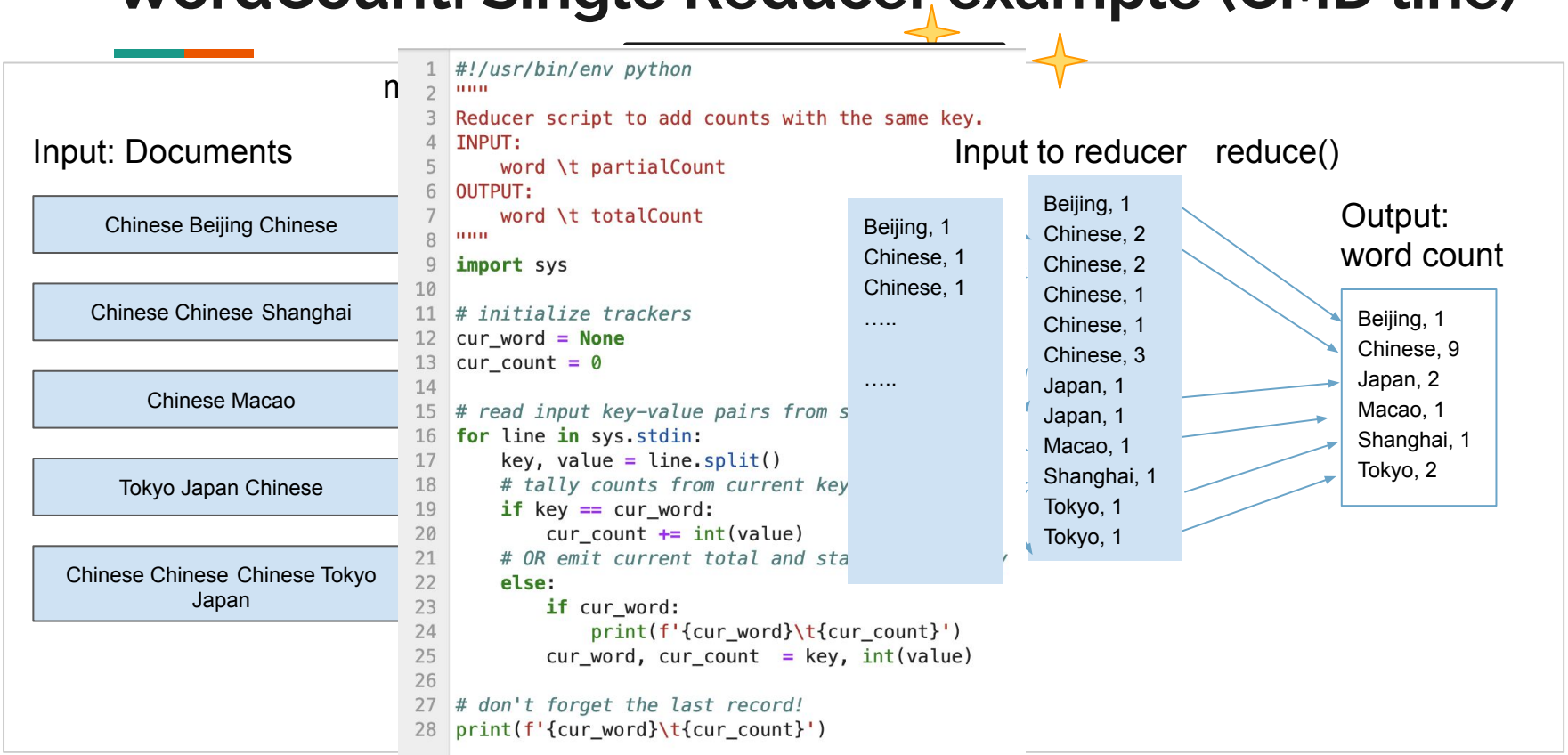
In the case of Hadoop Streaming Jobs, both the mapper and the reducer are python scripts that read the input from standard input and emit the output to standard output. The Hadoop Streaming command will create a Map/Reduce job, submit the job to an appropriate cluster, and monitor the progress of the job until it completes.

WordCount: Single Reducer example (CMD line)

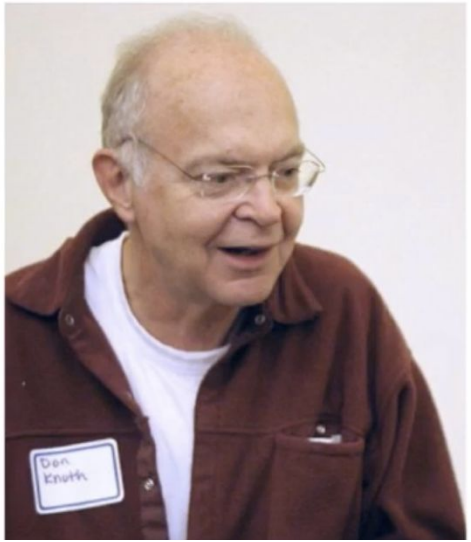


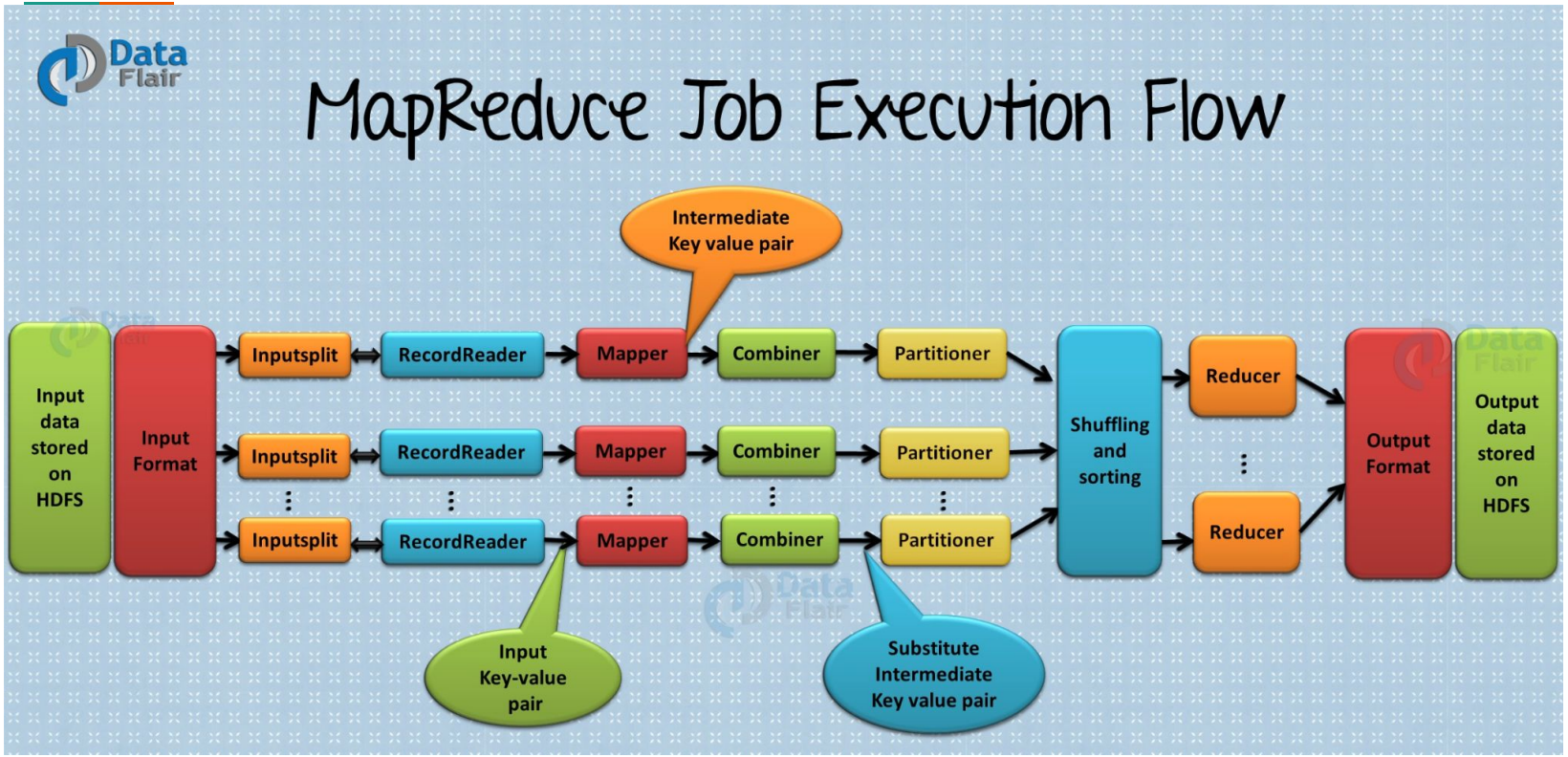
Aggregation function is comm + ass
(1 + 1) + 1 == 1 + (1 + 1)

WordCount: Single Reducer example (CMD line)



“Computer manufacturers of the 1960s estimated that more than 25% of the running time on their computers was spent on sorting… in fact, there were many installations in which the task of sorting was responsible for more than half of the computing time.” (Donald Knuth, *The Art of Computer Programming, Vol 3*)




Data Flair
MapReduce Job Execution Flow
Intermediate Key value pair
Input data stored on HDFS
Input Format
Inputsplit
RecordReader
Mapper
Combiner
Partitioner
Inputsplit
RecordReader
Mapper
Combiner
Partitioner
Inputsplit
RecordReader
Mapper
Combiner
Partitioner
Shuffling and sorting
Reducer
Reducer
Output Format
Output data stored on HDFS
Input Key-value pair
Substitute Intermediate Key value pair

Hadoop Streaming

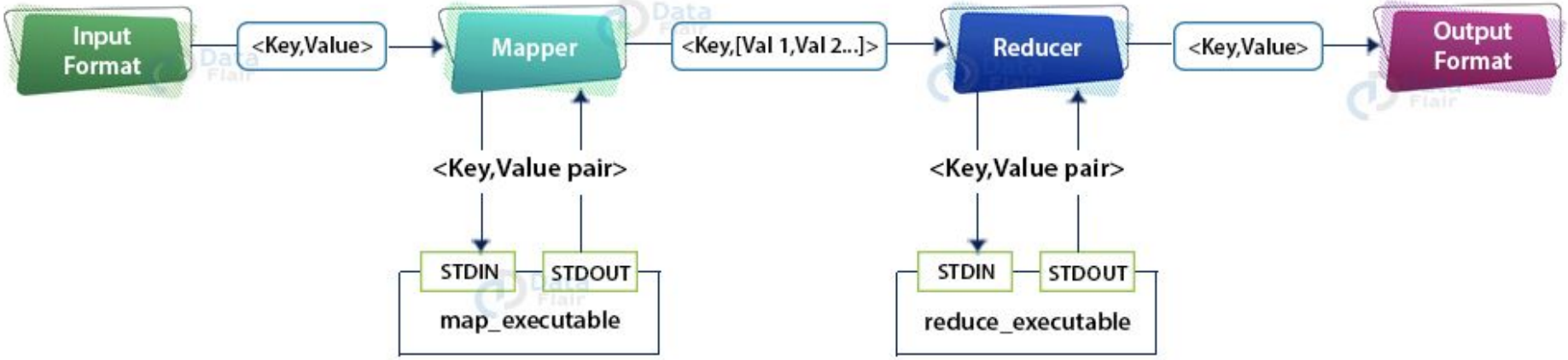
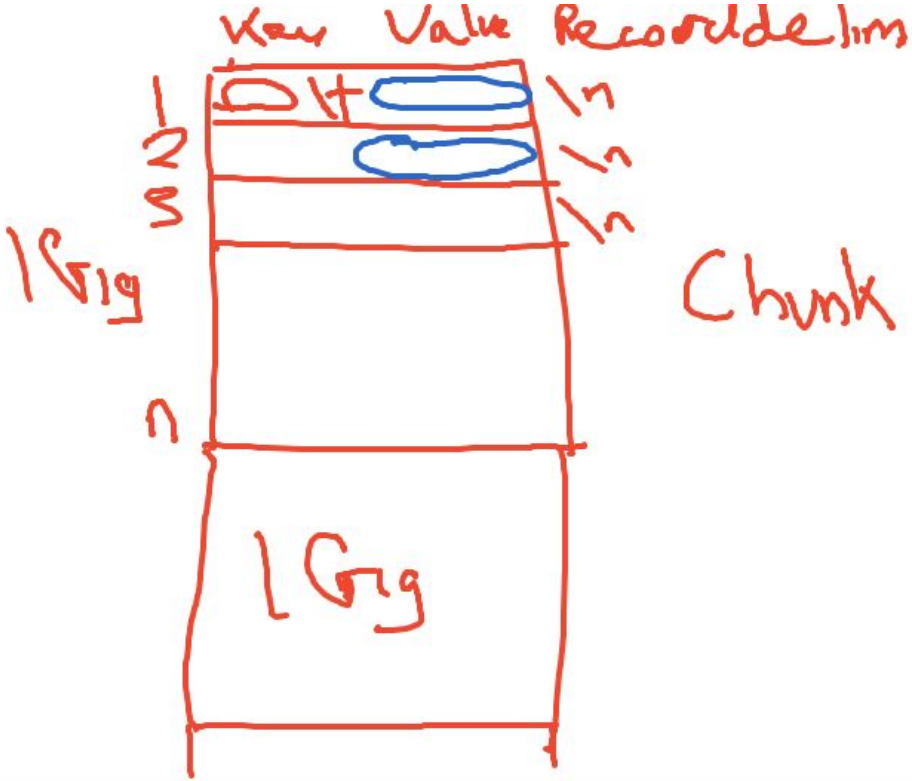


*image source: https://data-flair.training/blogs/hadoop-streaming/

You can use the below syntax to run MapReduce code written in a language other than JAVA to process data using the Hadoop MapReduce framework.

```
$HADOOP_HOME/bin/hadoop jar $HADOOP_HOME/hadoop-streaming.jar
-input myInputDirs \
-output myOutputDir \
-mapper /bin/cat \
-reducer /usr/bin/wc
```


Key Value Recorddelim
Chunk
1GiB

```
$HADOOP_HOME/bin/hadoop jar $HADOOP_HOME/hadoop-streaming.jar
-input myInputDirs \
-output myOutputDir \
-mapper /bin/cat \
-reducer /usr/bin/wc
```

Parameters Description

Parameter	Description
-input myInputDirs	Input location for mapper
-output myOutputDir	Output location for reducer
-mapper /bin/cat	Mapper executable
-reducer /usr/bin/wc	Reducer executable

The mapper and the reducer (in the above example) are the scripts that read the input line-by-line from stdin and emit the output to stdout.

Customizing How Lines are Split into Key/Value Pairs

By default for both the mapper and the reducer the key and value are extracted from the input record as follows:

- the prefix of a line until the first tab character is the key,
- and the rest of the line is the value except the tab character.

In the case of no tab character in the line, the entire line is considered as key, and the value is considered null. This is customizable by setting -inputformat command option for mapper and -outputformat option for reducer that we will see later in this article.

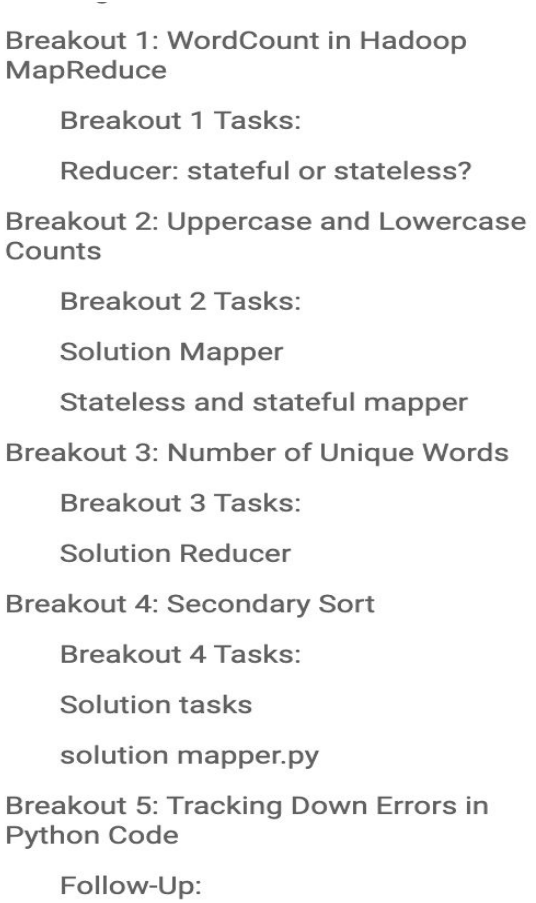
As noted earlier, when the Map/Reduce framework reads a line from the stdout of the mapper, it splits the line into a key/value pair. By default, **the prefix of the line up to the first tab character is the key** and the rest of the line (excluding the tab character) is the value.

Live Session #2

Lab Notebook is located here (contains 4-5 tasks to complete):

-

The master solution for each weekly lab (from Week 2 onwards) will be published on each Saturday Midday.



CSV; “.” as a field delimiter
101.2.34.127

Specify a field separator or just use the default TAB

However, you can customize this default. You can specify a field separator other than the tab character (the default), and you can specify the nth (n >= 1) character rather than the first character in a line (the default) as the separator between the key and value. For example:

```
hadoop jar hadoop-streaming-2.6.0.jar \
    -D stream.map.output.field.separator=. \
    -D stream.num.map.output.key.fields=4 \
    -input myInputDirs \
    -output myOutputDir \
    -mapper /bin/cat \
    -reducer /bin/cat
```

Mapper output records are sorted

In the above example, `-D stream.map.output.field.separator=.` specifies `.` as the field separator for the map outputs, and the prefix up to the fourth "." in a line will be the key and the rest of the line (excluding the fourth ".") will be the value. If a line has less than four "."s, then the whole line will be the key and the value will be an empty Text object (like the one created by new Text("")).

Similarly, you can use `-D stream.reduce.output.field.separator=SEP` and `-D stream.num.reduce.output.fields=NUM` to specify the nth field separator in a line of the reduce outputs as the separator between the key and the value.

Similarly, you can specify `stream.map.input.field.separator` and `stream.reduce.input.field.separator` as the input separator for Map/Reduce inputs. By default the separator is the tab character.

File Edit View Run Kernel Tabs Settings Help

/ ··· / master / WordCount /

Name	Last Modified
mapper.py	a year ago
reducer.py	a year ago
results.txt	2 hours ago

demo2_workbook.ipynb ● | HadoopModuleError.png ×

Code | Python 3

Breakout 1 Tasks:

- **a) read scripts & docstrings:** Read through `WordCount/mapper.py` and `WordCou
(briefly) explain what the script does and the expected input/output record formats. [
self/collaborator/grader) quickly orient to a piece of code. They should describe what
for Python_] The use of docstrings is recommended in all code that is written for this c
- **b) discuss:** What are the 'keys' and what are the 'values' in this MapReduce job? What
expect Hadoop to sort the records emitted by the mapper script? Why is this order im
- **c) run provided code:** Run the cells provided to make sure that your mapper and redu
directory where the job will write its output. You will need to do these preparation step
- **d) unit test:** A good habit when writing Hadoop streaming jobs is to test your mappers
An easy way to do this is to pipe in a small line of text. We've provided the unix code to
cells to confirm that our mapper and reducer work properly. (Observe how the reducer
- **e) code:** We've provided the code to run your Hadoop streaming command on the test
that we're passing in, then run it and confirm that the output performs word counting
on the *Alice and Wonderland* text instead of the test file. Remember that the input path
not a local path. Take a look at the output and confirm you get the same count for 'alic

`part c` Prep for Hadoop Streaming Job

```
[20]:
# part c - make sure the mapper and reducer are executable (RUN THIS
!chmod a+x WordCount/mapper.py
!chmod a+x WordCount/reducer.py
```

```
[21]:
# part c - load the input files into HDFS (RUN THIS CELL AS IS)
!hdfs dfs -copyFromLocal {TEST_TXT} {HDFS_DIR}
!hdfs dfs -copyFromLocal {ALICE_TXT} {HDFS_DIR}
```

```
copyFromLocal: `/user/root/demo2/alice_test.txt': File exists
```

```
[22]:
# part c - clear the output directory (RUN THIS CELL AS IS)
!hdfs dfs -rm -r {HDFS_DIR}/wordcount-output
# NOTE: this directory won't exist unless you are re-running a job, that's fine.
```

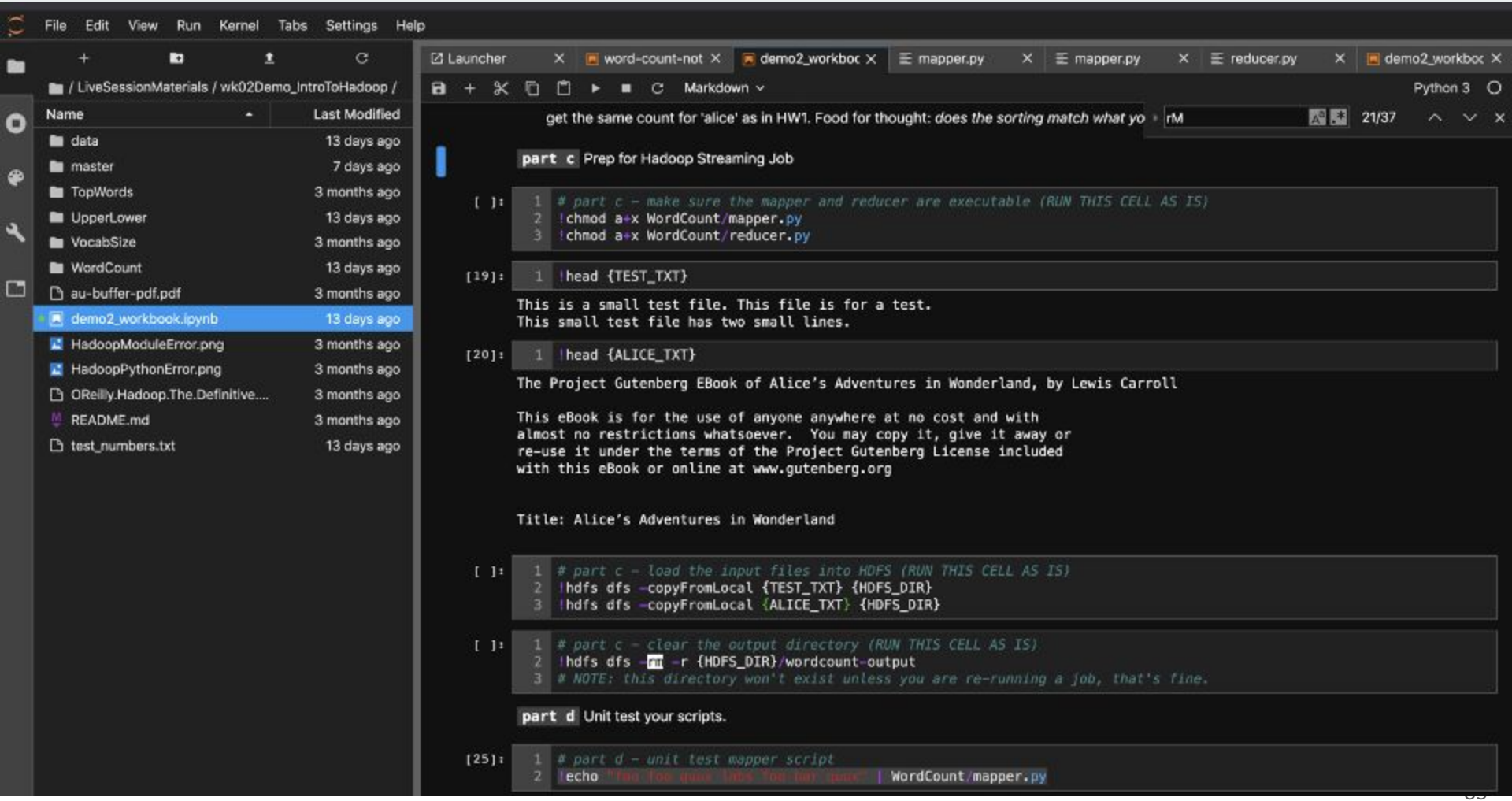
```
rm: `/user/root/demo2/wordcount-output': No such file or directory
```

```
#!/usr/bin/env python
"""
Reducer script to add counts with the same key.
INPUT:
    word \t partialCount
OUTPUT:
    word \t totalCount
"""
import sys

# initialize trackers
cur_word = None
cur_count = 0

# read input key-value pairs from standard input
for line in sys.stdin:
    key, value = line.split()
    # tally counts from current key
    if key == cur_word:
        cur_count += int(value)
    # OR emit current total and start a new tally
    else:
        if cur_word:
            print(f'{cur_word}\t{cur_count}')
        cur_word, cur_count  = key, int(value)

# don't forget the last record!
print(f'{cur_word}\t{cur_count}')
```

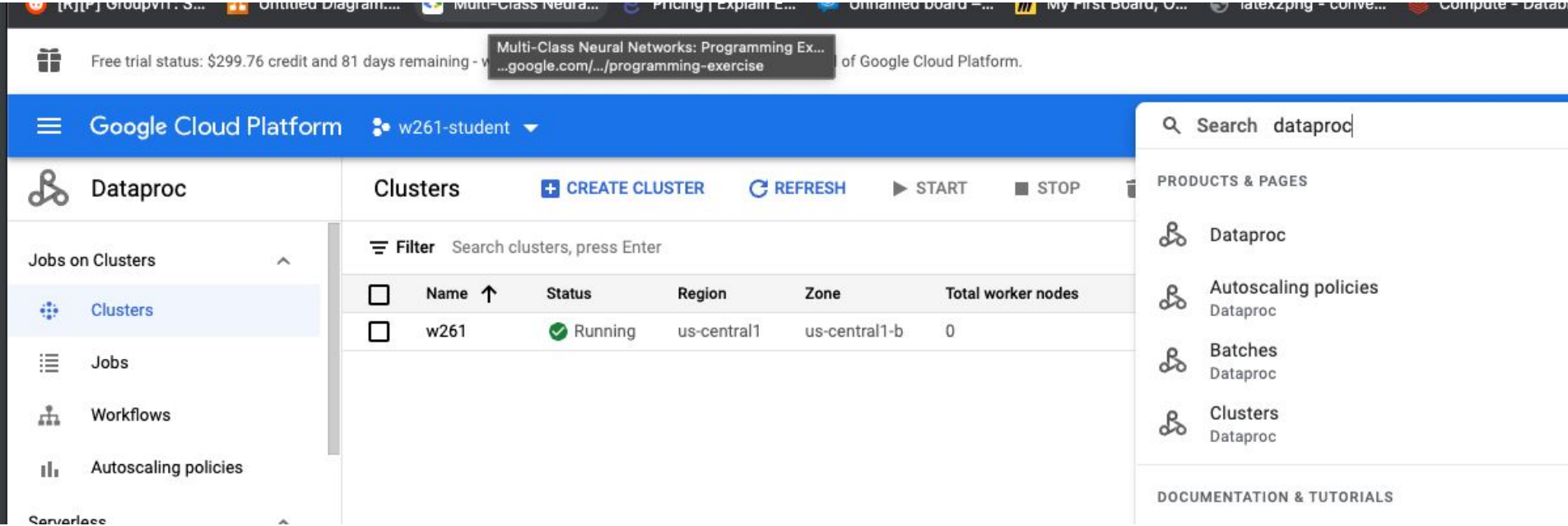

File Edit View Run Kernel Tabs Settings Help
/ LiveSessionMaterials / wk02Demo_IntroToHadoop /
Name
Last Modified
data 13 days ago
master 7 days ago
TopWords 3 months ago
UpperLower 13 days ago
VocabSize 3 months ago
WordCount 13 days ago
au-buffer-pdf.pdf 3 months ago
demo2_workbook.ipynb 13 days ago
HadoopModuleError.png 3 months ago
HadoopPythonError.png 3 months ago
OReilly.Hadoop.The.Definitive.... 3 months ago
README.md 3 months ago
test_numbers.txt 13 days ago
Launcher
word-count-not
demo2_workboc
mapper.py
mapper.py
reducer.py
demo2_workboc
Markdown
Python 3
get the same count for 'alice' as in HW1. Food for thought: does the sorting match what yo
rM
21/37
part c Prep for Hadoop Streaming Job
[ ]:
part c - make sure the mapper and reducer are executable (RUN THIS CELL AS IS)
!chmod a+x WordCount/mapper.py
!chmod a+x WordCount/reducer.py
[19]:
!head {TEST_TXT}
This is a small test file. This file is for a test.
This small test file has two small lines.
[20]:
!head {ALICE_TXT}
The Project Gutenberg EBook of Alice's Adventures in Wonderland, by Lewis Carroll
This eBook is for the use of anyone anywhere at no cost and with
almost no restrictions whatsoever. You may copy it, give it away or
re-use it under the terms of the Project Gutenberg License included
with this eBook or online at www.gutenberg.org
Title: Alice's Adventures in Wonderland
[ ]:
part c - load the input files into HDFS (RUN THIS CELL AS IS)
!hdfs dfs -copyFromLocal {TEST_TXT} {HDFS_DIR}
!hdfs dfs -copyFromLocal {ALICE_TXT} {HDFS_DIR}
[ ]:
part c - clear the output directory (RUN THIS CELL AS IS)
!hdfs dfs -rm -r {HDFS_DIR}/wordcount-output
NOTE: this directory won't exist unless you are re-running a job, that's fine.
part d Unit test your scripts.
[25]:
part d - unit test mapper script
!echo "foo foo quux labs foo bar quux" | WordCount/mapper.py

How to Access Hadoop logs

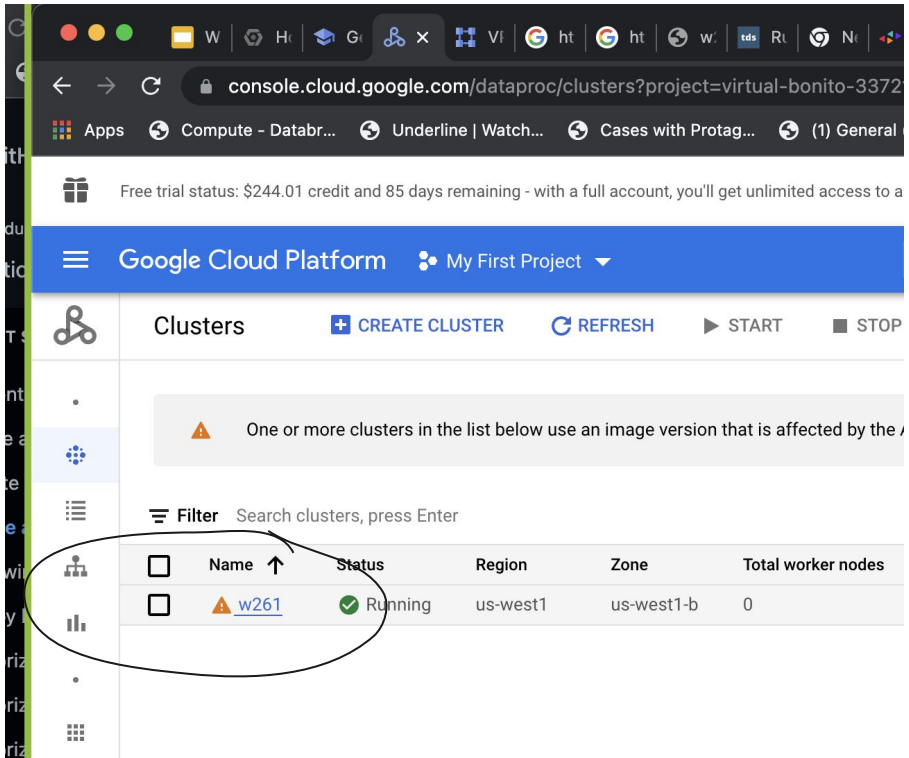
STEP 1: open GC Console

(do NOT click the link in the Hadoop output as it is local link only)

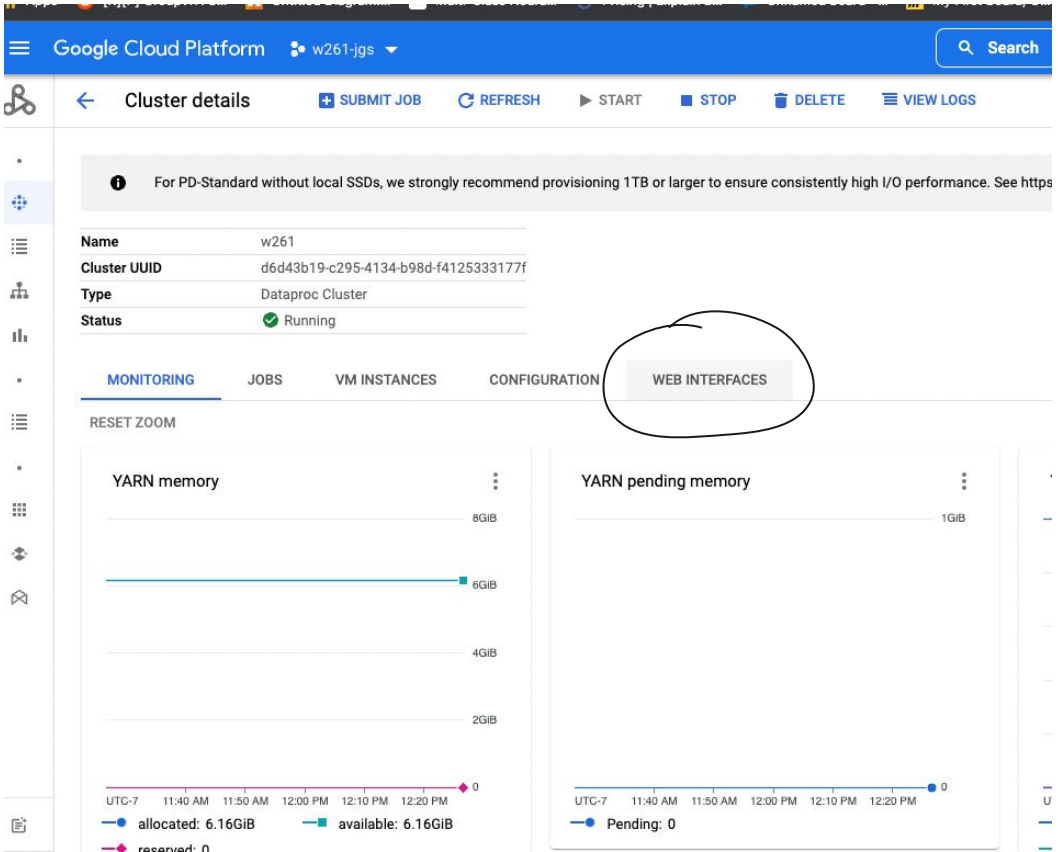
STEP 2: Search for dataproc GC console



Click on w261 cluster

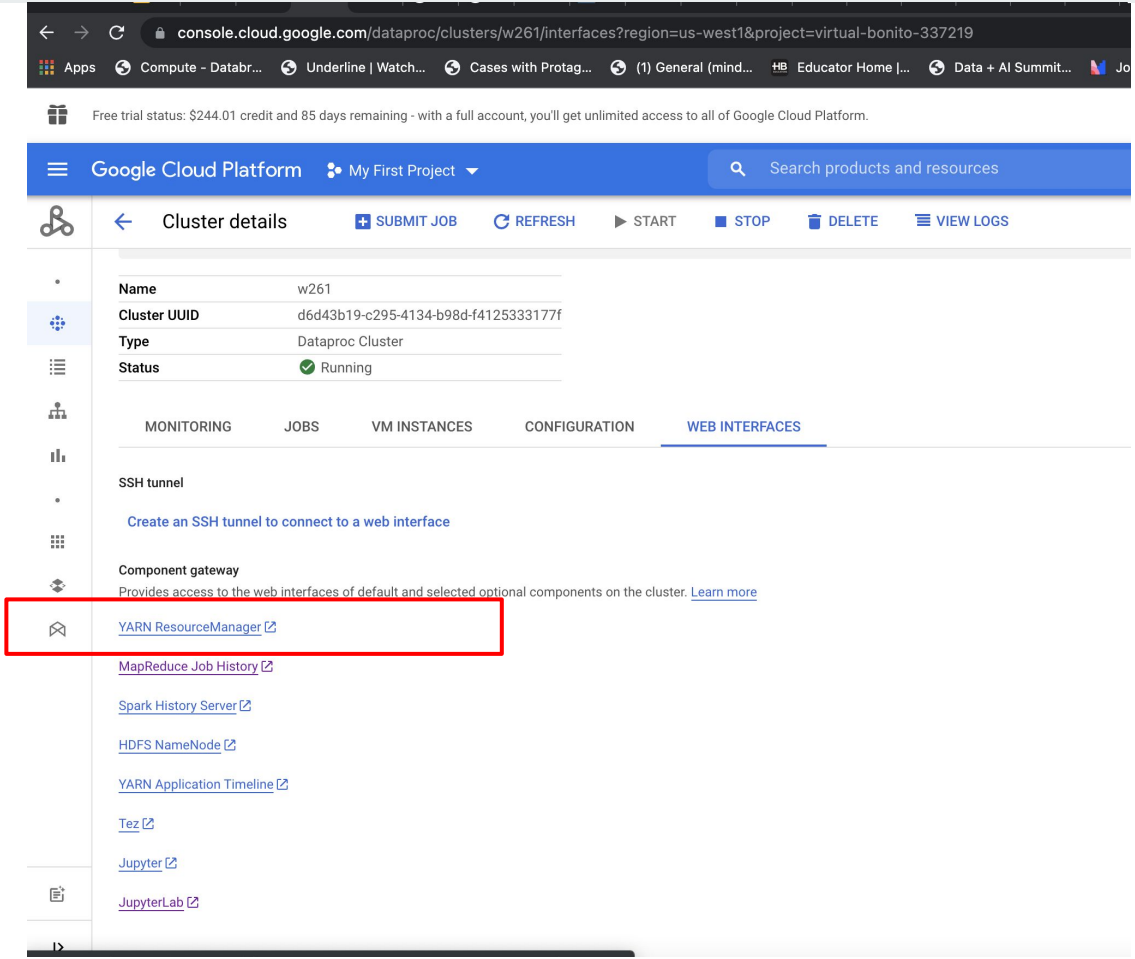


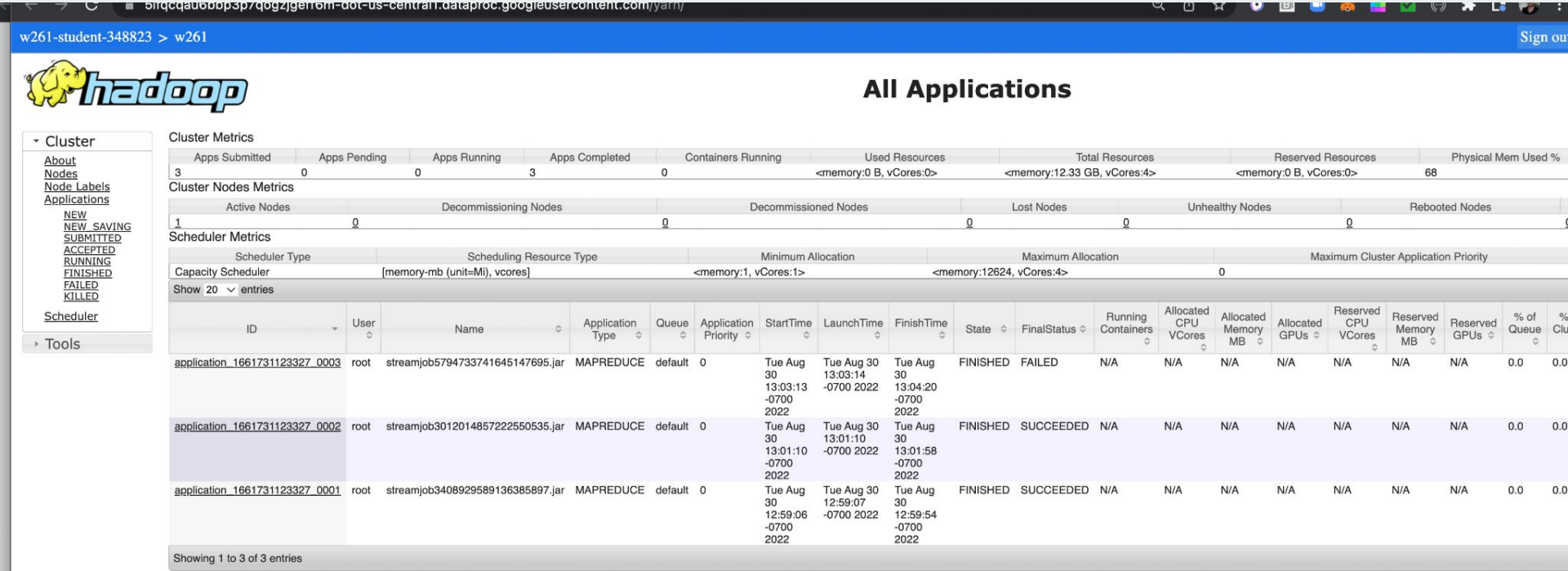
Click on Web Interfaces



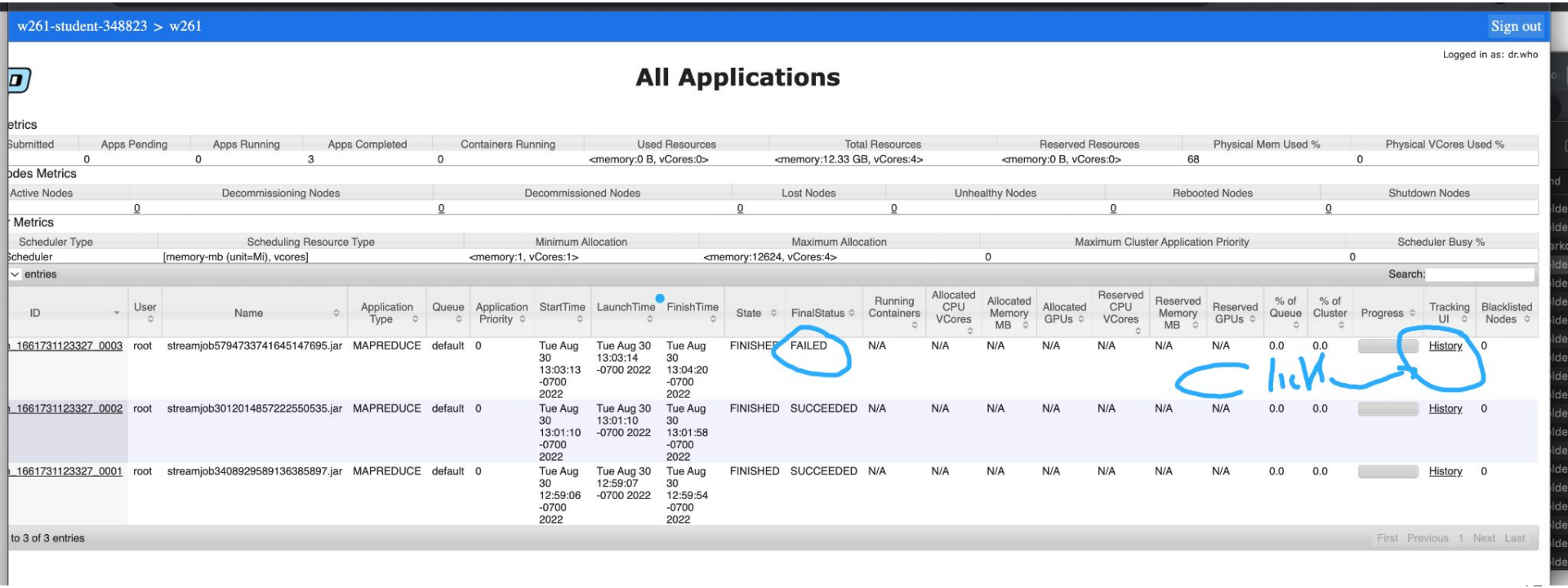
Hadoop Logs

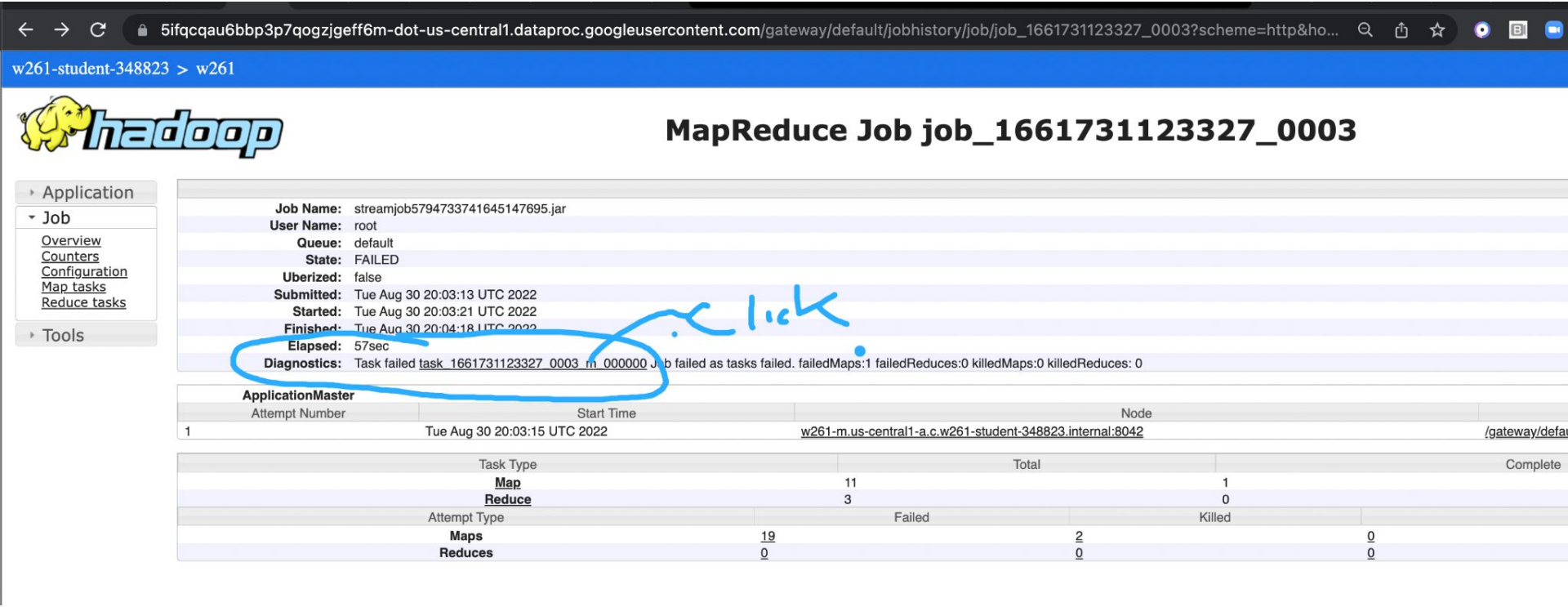
Hadoop Logs: click YARN Link

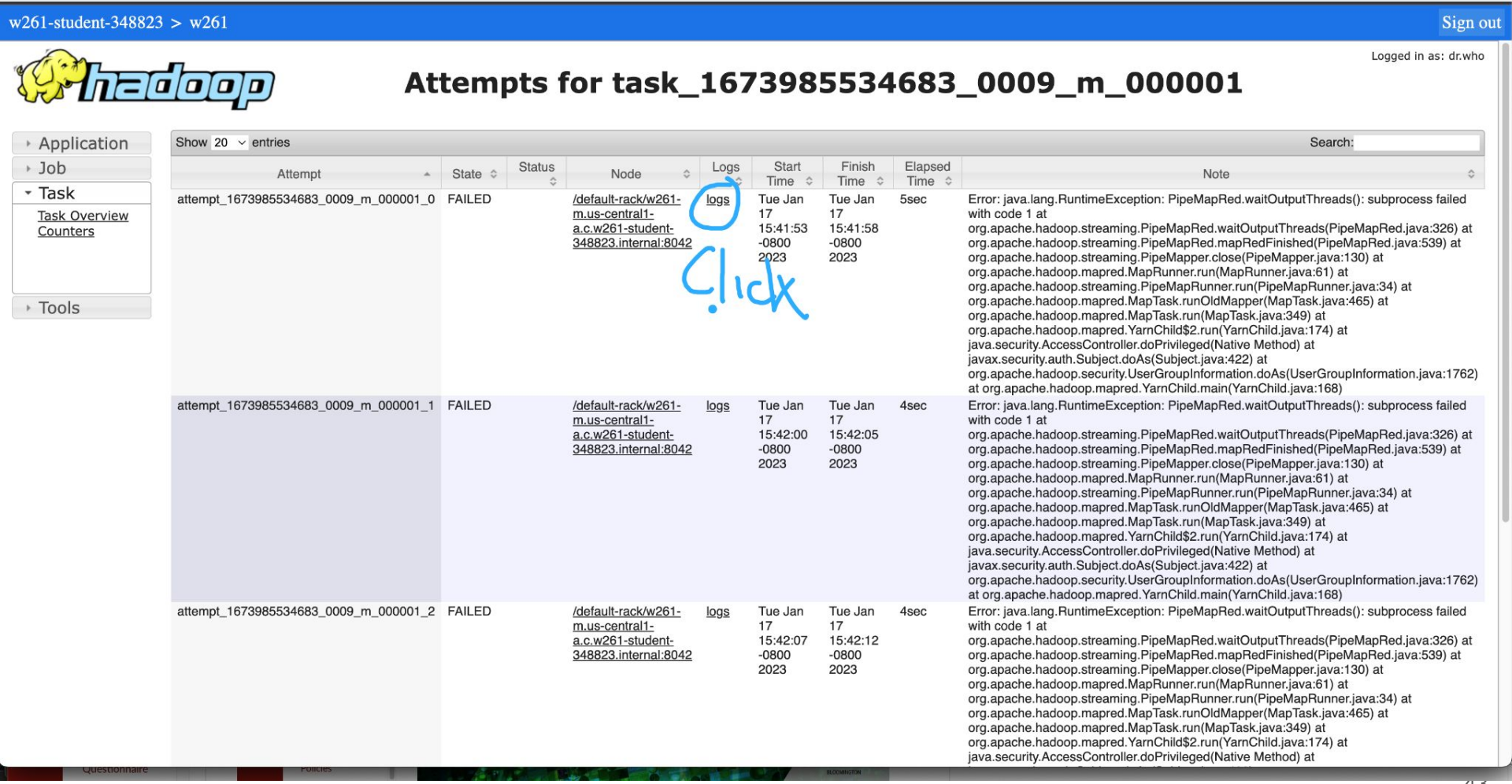



w261-student-348823 > w261
Sign out
hadoop
All Applications
Cluster
About
Nodes
Node Labels
Applications
NEW
NEW_SAVING
SUBMITTED
ACCEPTED
RUNNING
FINISHED
FAILED
KILLED
Scheduler
Tools
Cluster Metrics
Apps Submitted | Apps Pending | Apps Running | Apps Completed | Containers Running | Used Resources | Total Resources | Reserved Resources | Physical Mem Used %
3 | 0 | 0 | 3 | 0 | <memory:0 B, vCores:0> | <memory:12.33 GB, vCores:4> | <memory:0 B, vCores:0> | 68
Cluster Nodes Metrics
Active Nodes | Decommissioning Nodes | Decommissioned Nodes | Lost Nodes | Unhealthy Nodes | Rebooted Nodes
1 | 0 | 0 | 0 | 0 | 0
Scheduler Metrics
Scheduler Type | Scheduling Resource Type | Minimum Allocation | Maximum Allocation | Maximum Cluster Application Priority
Capacity Scheduler | [memory-mb (unit=Mi), vcores] | <memory:1, vCores:1> | <memory:12624, vCores:4> | 0
Show 20 entries
ID | User | Name | Application Type | Queue | Application Priority | StartTime | LaunchTime | FinishTime | State | FinalStatus | Running Containers | Allocated CPU VCores | Allocated Memory MB | Allocated GPUs | Reserved CPU VCores | Reserved Memory MB | Reserved GPUs | % of Queue
application_1661731123327_0003 | root | streamjob5794733741645147695.jar | MAPREDUCE | default | 0 | Tue Aug 30 13:03:13 -0700 2022 | Tue Aug 30 13:03:14 -0700 2022 | Tue Aug 30 13:04:20 -0700 2022 | FINISHED | FAILED | N/A | N/A | N/A | N/A | N/A | N/A | N/A | 0.0
application_1661731123327_0002 | root | streamjob3012014857222550535.jar | MAPREDUCE | default | 0 | Tue Aug 30 13:01:10 -0700 2022 | Tue Aug 30 13:01:10 -0700 2022 | Tue Aug 30 13:01:58 -0700 2022 | FINISHED | SUCCEEDED | N/A | N/A | N/A | N/A | N/A | N/A | N/A | 0.0
application_1661731123327_0001 | root | streamjob3408929589136385897.jar | MAPREDUCE | default | 0 | Tue Aug 30 12:59:06 -0700 2022 | Tue Aug 30 12:59:07 -0700 2022 | Tue Aug 30 12:59:54 -0700 2022 | FINISHED | SUCCEEDED | N/A | N/A | N/A | N/A | N/A | N/A | N/A | 0.0
Showing 1 to 3 of 3 entries
Scroll Right

After scrolling right click on the history of the failed job




5ifqcqau6bbp3p7qogzjgeff6m-dot-us-central1.dataproc.googleusercontent.com/gateway/default/jobhistory/job/job_1661731123327_0003?scheme=http&ho...
w261-student-348823 > w261
hadoop
MapReduce Job job_1661731123327_0003
Application
Job
Overview
Counters
Configuration
Map tasks
Reduce tasks
Tools
Job Name: streamjob5794733741645147695.jar
User Name: root
Queue: default
State: FAILED
Uberized: false
Submitted: Tue Aug 30 20:03:13 UTC 2022
Started: Tue Aug 30 20:03:21 UTC 2022
Finished: Tue Aug 30 20:04:18 UTC 2022
Elapsed: 57sec
Diagnostics: Task failed task_1661731123327_0003_m_000000 Job failed as tasks failed. failedMaps:1 failedReduces:0 killedMaps:0 killedReduces: 0
click
ApplicationMaster
Attempt Number
Start Time
Node
1
Tue Aug 30 20:03:15 UTC 2022
w261-m.us-central1-a.c.w261-student-348823.internal:8042
/gateway/defau
Task Type
Total
Complete
Map
11
1
Reduce
3
0
Attempt Type
Failed
Killed
Maps
19
2
0
Reduces
0
0
0


w261-student-348823 > w261
Sign out
Logged in as: dr.who
hadoop
Attempts for task_1673985534683_0009_m_000001
Application
Job
Task
Task Overview
Counters
Tools
Show 20 entries
Search:
Attempt
State
Status
Node
Logs
Start Time
Finish Time
Elapsed Time
Note
attempt_1673985534683_0009_m_000001_0
FAILED
/default-rack/w261-m.us-central1-a.c.w261-student-348823.internal:8042
logs
Tue Jan 17 15:41:53 -0800 2023
Tue Jan 17 15:41:58 -0800 2023
5sec
Error: java.lang.RuntimeException: PipeMapRed.waitOutputThreads(): subprocess failed with code 1 at org.apache.hadoop.streaming.PipeMapRed.waitOutputThreads(PipeMapRed.java:326) at org.apache.hadoop.streaming.PipeMapRed.mapRedFinished(PipeMapRed.java:539) at org.apache.hadoop.streaming.PipeMapper.close(PipeMapper.java:130) at org.apache.hadoop.mapred.MapRunner.run(MapRunner.java:61) at org.apache.hadoop.streaming.PipeMapRunner.run(PipeMapRunner.java:34) at org.apache.hadoop.mapred.MapTask.runOldMapper(MapTask.java:465) at org.apache.hadoop.mapred.MapTask.run(MapTask.java:349) at org.apache.hadoop.mapred.YarnChild$2.run(YarnChild.java:174) at java.security.AccessController.doPrivileged(Native Method) at javax.security.auth.Subject.doAs(Subject.java:422) at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1762) at org.apache.hadoop.mapred.YarnChild.main(YarnChild.java:168)
attempt_1673985534683_0009_m_000001_1
FAILED
/default-rack/w261-m.us-central1-a.c.w261-student-348823.internal:8042
logs
Tue Jan 17 15:42:00 -0800 2023
Tue Jan 17 15:42:05 -0800 2023
4sec
Error: java.lang.RuntimeException: PipeMapRed.waitOutputThreads(): subprocess failed with code 1 at org.apache.hadoop.streaming.PipeMapRed.waitOutputThreads(PipeMapRed.java:326) at org.apache.hadoop.streaming.PipeMapRed.mapRedFinished(PipeMapRed.java:539) at org.apache.hadoop.streaming.PipeMapper.close(PipeMapper.java:130) at org.apache.hadoop.mapred.MapRunner.run(MapRunner.java:61) at org.apache.hadoop.streaming.PipeMapRunner.run(PipeMapRunner.java:34) at org.apache.hadoop.mapred.MapTask.runOldMapper(MapTask.java:465) at org.apache.hadoop.mapred.MapTask.run(MapTask.java:349) at org.apache.hadoop.mapred.YarnChild$2.run(YarnChild.java:174) at java.security.AccessController.doPrivileged(Native Method) at javax.security.auth.Subject.doAs(Subject.java:422) at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1762) at org.apache.hadoop.mapred.YarnChild.main(YarnChild.java:168)
attempt_1673985534683_0009_m_000001_2
FAILED
/default-rack/w261-m.us-central1-a.c.w261-student-348823.internal:8042
logs
Tue Jan 17 15:42:07 -0800 2023
Tue Jan 17 15:42:12 -0800 2023
4sec
Error: java.lang.RuntimeException: PipeMapRed.waitOutputThreads(): subprocess failed with code 1 at org.apache.hadoop.streaming.PipeMapRed.waitOutputThreads(PipeMapRed.java:326) at org.apache.hadoop.streaming.PipeMapRed.mapRedFinished(PipeMapRed.java:539) at org.apache.hadoop.streaming.PipeMapper.close(PipeMapper.java:130) at org.apache.hadoop.mapred.MapRunner.run(MapRunner.java:61) at org.apache.hadoop.streaming.PipeMapRunner.run(PipeMapRunner.java:34) at org.apache.hadoop.mapred.MapTask.runOldMapper(MapTask.java:465) at org.apache.hadoop.mapred.MapTask.run(MapTask.java:349) at org.apache.hadoop.mapred.YarnChild$2.run(YarnChild.java:174) at java.security.AccessController.doPrivileged(Native Method) at
Click

- Application
- Job
- Task
  - Task Overview
  - Counters
- Tools

Log Type: stdout
Log Upload Time: Tue Aug 30 20:03:23 +0000 2022
Log Length: 0

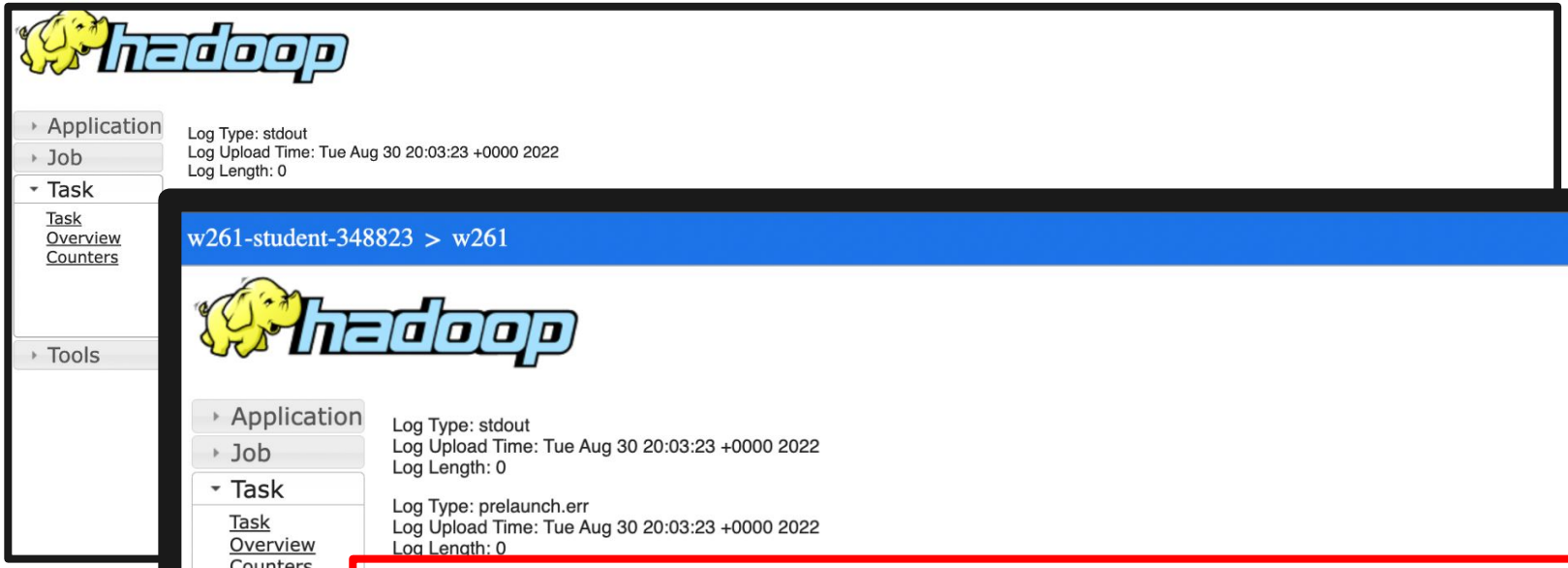
Log Type: prelaunch.err
Log Upload Time: Tue Aug 30 20:03:23 +0000 2022
Log Length: 0

Log Type: stderr
Log Upload Time: Tue Aug 30 20:03:30 +0000 2022
Log Length: 240

```
  File "/hadoop/yarn/nm-local-dir/usercache/root/appcache/application_1661731123327_0003/container_1661731123327_0003_01_000002/./mapper.py", line 10
    print(1/0)            # dividing by zero is a no-go
    ^
SyntaxError: invalid syntax
```

Log Type: launch_container.sh
Log Upload Time: Tue Aug 30 20:03:23 +0000 2022
Log Length: 5172
Showing 4096 bytes of 5172 total. Click here for the full log.

```
ontainer_1661731123327_0003_01_000002"
export NM_PORT="8026"
export NM_HOST="w261-m.us-central1-a.c.w261-student-348823.internal"
export NM_HTTP_PORT="8042"
export LOCAL_DIRS="/hadoop/yarn/nm-local-dir/usercache/root/appcache/application_1661731123327_0003"
export LOCAL_USER_DIRS="/hadoop/yarn/nm-local-dir/usercache/root/"
export LOG_DIRS="/var/log/hadoop-yarn/userlogs/application_1661731123327_0003/container_1661731123327_0003_01_000002"
export USER="root"
export LOGNAME="root"
export HOME="/home/"
export PWD="/hadoop/yarn/nm-local-dir/usercache/root/appcache/application_1661731123327_0003/container_1661731123327_0003_01_000002"
export LOCALIZATION_COUNTERS="391980,317,2,1,102"
export JVM_PID="$$"
export NM_AUX_SERVICE_spark_shuffle=""
export NM_AUX_SERVICE_mapreduce_shuffle="AAA0+gAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA="
export STDOUT_LOGFILE_ENV="/var/log/hadoop-yarn/userlogs/application_1661731123327_0003/container_1661731123327_0003_01_000002/stdout"
export SHELL="/bin/bash"
export HADOOP_ROOT_LOGGER="INFO,console"
export CLASSPATH="$PWD:$HADOOP_CONF_DIR:$HADOOP_COMMON_HOME/*:$HADOOP_COMMON_HOME/lib/*:$HADOOP_HDFS_HOME/*:$HADOOP_HDFS_HOME/lib/*:$HADOOP_MAPRED_HOME/*:$HADOOP_MAPRED_HOME/lib/*:$HADOOP_YARN_HOME/*:$HADOOP_YA
export LD_LIBRARY_PATH="$PWD:$HADOOP_COMMON_HOME/lib/native"
export STDERR_LOGFILE_ENV="/var/log/hadoop-yarn/userlogs/application_1661731123327_0003/container_1661731123327_0003_01_000002/stderr"
export HADOOP_CLIENT_OPTS=""
export MALLOC_ARENA_MAX="4"
echo "Setting up job resources"
ln -sf -- "/hadoop/yarn/nm-local-dir/usercache/root/appcache/application_1661731123327_0003/filecache/14/job.jar" "job.jar"
ln -sf -- "/hadoop/yarn/nm-local-dir/usercache/root/appcache/application_1661731123327_0003/filecache/15/job.xml" "job.xml"
ln -sf -- "/hadoop/yarn/nm-local-dir/usercache/root/filecache/10/mapper.py" "mapper.py"
echo "Copying debugging information"
# Creating copy of launch script
cp "launch_container.sh" "/var/log/hadoop-yarn/userlogs/application_1661731123327_0003/container_1661731123327_0003_01_000002/launch_container.sh"
```



Application
Job
Task
Task Overview
Counters
Tools

Log Type: stdout
Log Upload Time: Tue Aug 30 20:03:23 +0000 2022
Log Length: 0

w261-student-348823 > w261

Application
Job
Task
Task Overview
Counters
Tools

Log Type: stdout
Log Upload Time: Tue Aug 30 20:03:23 +0000 2022
Log Length: 0

Log Type: prelaunch.err
Log Upload Time: Tue Aug 30 20:03:23 +0000 2022
Log Length: 0

Log Type: stderr
Log Upload Time: Tue Aug 30 20:03:30 +0000 2022
Log Length: 240

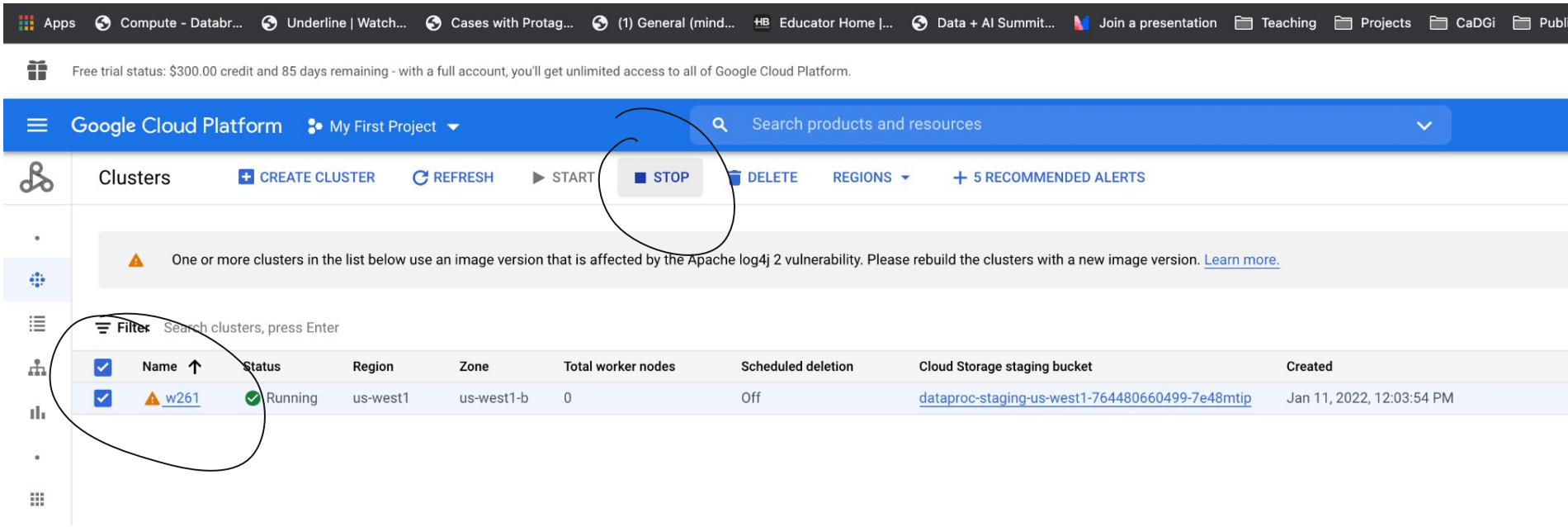
```
  File "/hadoop/yarn/nm-local-dir/usercache/root/appcache/application_1661731123327_0003/container_1661731123327_0003_01_000002/./mapper.py", line 10
    print(1/0)            # dividing by zero is a no-go
    ^
SyntaxError: invalid syntax
```

Log Type: launch_container.sh
Log Upload Time: Tue Aug 30 20:03:23 +0000 2022

```
export STDERR_LOGFILE_ENV="/var/log/hadoop-yarn/userlogs/application_1661731123327_0003/container_1661731123327_0003_01_000002/stderr"
export HADOOP_CLIENT_OPTS=""
export MALLOC_ARENA_MAX="4"
echo "Setting up job resources"
ln -sf -- "/hadoop/yarn/nm-local-dir/usercache/root/appcache/application_1661731123327_0003/filecache/14/job.jar" "job.jar"
ln -sf -- "/hadoop/yarn/nm-local-dir/usercache/root/appcache/application_1661731123327_0003/filecache/15/job.xml" "job.xml"
ln -sf -- "/hadoop/yarn/nm-local-dir/usercache/root/filecache/10/mapper.py" "mapper.py"
echo "Copying debugging information"
# Creating copy of launch script
cp "launch_container.sh" "/var/log/hadoop-yarn/userlogs/application_1661731123327_0003/container_1661731123327_0003_01_000002/launch_container.sh"
```

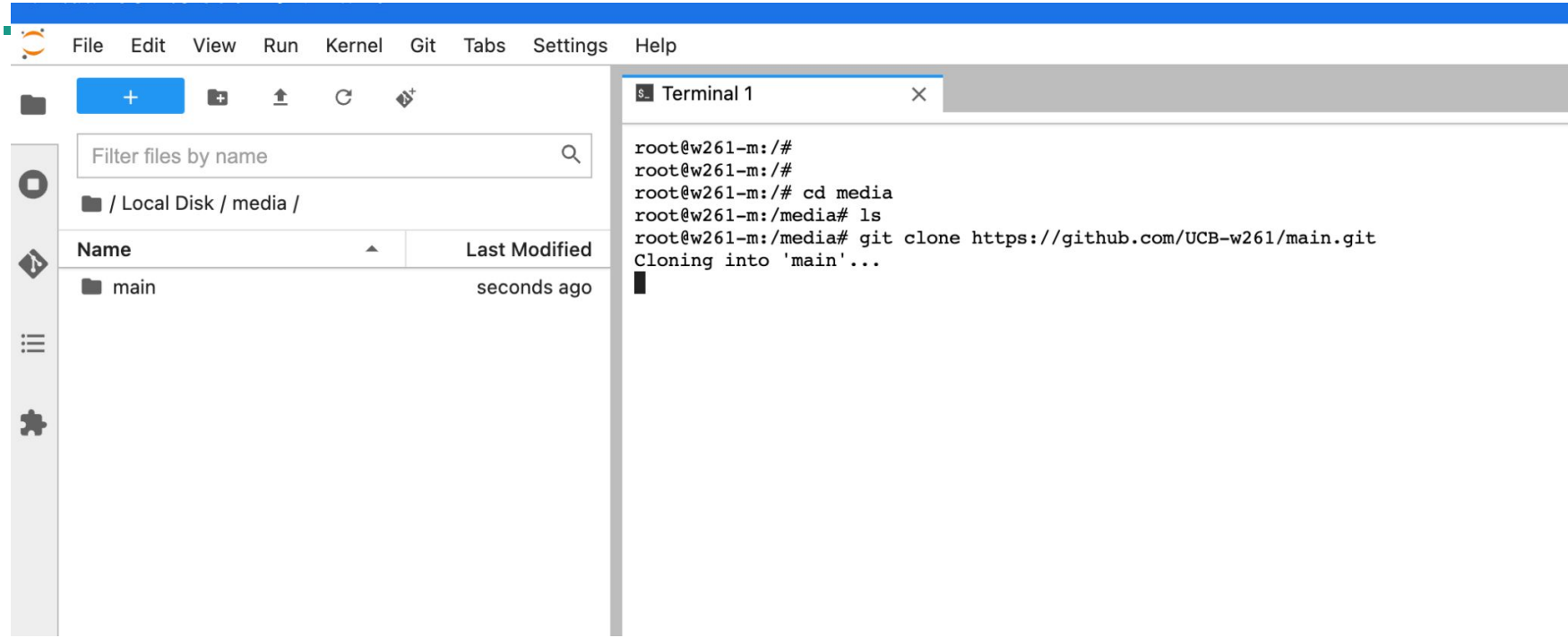
Stop your cluster when you are not using it!!

Select cluster and click the DELETE button



Please ignore slides following this one

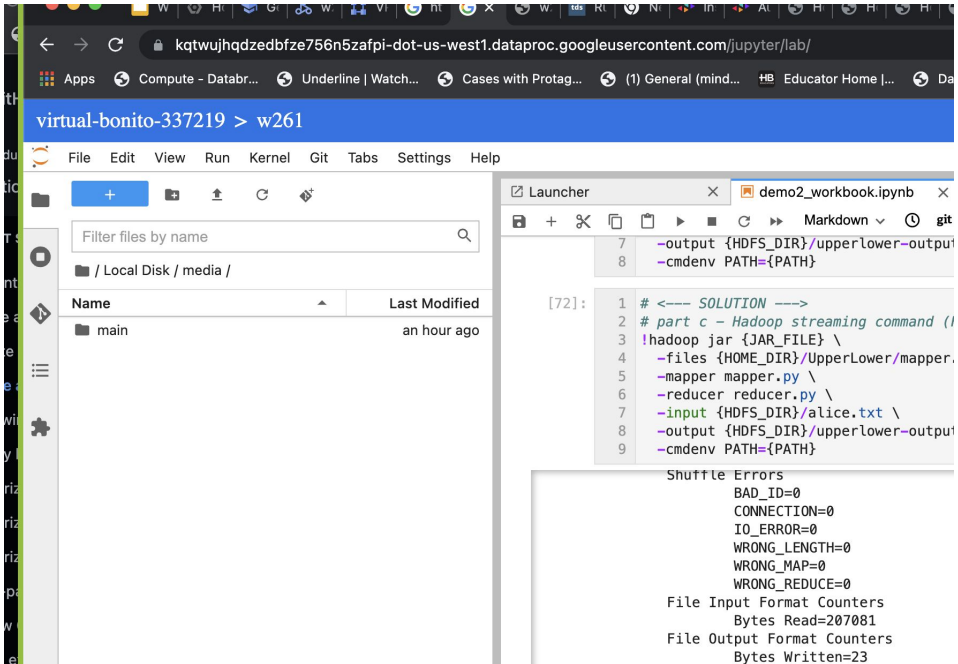
CLi


File Edit View Run Kernel Git Tabs Settings Help
Filter files by name
/ Local Disk / media /
Name
Last Modified
main
seconds ago
Terminal 1
root@w261-m:/#
root@w261-m:/#
root@w261-m:/# cd media
root@w261-m:/media# ls
root@w261-m:/media# git clone https://github.com/UCB-w261/main.git
Cloning into 'main'...

Jimi's command history

```
james_shanahan@cloudshell:~ (virtual-bonito-337219)$ history
  1 ls
  2 pwd
  3 docker run hello-world
  4 echo $GOOGLE_CLOUD_PROJECT  #pre-defined locally  virtual-bonito-337219
  5 gcloud compute instances create w261  --project=$GOOGLE_CLOUD_PROJECT  --zone=us-central1-a  --machine-type=n2-standard-4  --network-interface=subnet=default,no-address  --maintenance-policy=MIGRATE
--create-disk=auto-delete=yes,boot=yes,device-name=w261,image=projects/w261-trusted-images/global/images/w261-image,mode=rw,size=10  --no-shielded-secure-boot  --shielded-vtpm  --shielded-integrity-monitoring  --reservation-affinity=any
  6 gcloud compute instances create w261  --project=$GOOGLE_CLOUD_PROJECT  --zone=us-central1-a  --machine-type=n2-standard-4  --network-interface=subnet=default,no-address  --maintenance-policy=MIGRATE
--create-disk=auto-delete=yes,boot=yes,device-name=w261,image=projects/w261-trusted-images/global/images/w261-image,mode=rw,size=10  --no-shielded-secure-boot  --shielded-vtpm  --shielded-integrity-monitoring \
  7 gcloud compute instances create w261  --project=$GOOGLE_CLOUD_PROJECT  --zone=us-central1-a  --machine-type=n2-standard-4  --network-interface=subnet=default,no-address  --maintenance-policy=MIGRATE \
  8 --no-shielded-secure-boot  --shielded-vtpm  --shielded-integrity-monitoring  --reservation-affinity=any
  9 gsutil -ls
 10 gsutil ls
 11 gcloud dataproc clusters create w261 --enable-component-gateway --region us-west1 --subnet default --no-address --single-node --master-machine-type n1-standard-4 --master-boot-disk-size 100 --image-version 2.0-debian10 --optional-components JUPYTER
--project $GOOGLE_CLOUD_PROJECT
 12 gsutil cp -r gs://w261-hw-data/* gs://dataproc-staging-us-west1-764480660499-7e48mtip/
 13 gsutil ls gs://dataproc-staging-us-west1-764480660499-7e48mtip/
 14 gsutil ls gs://dataproc-staging-us-west1-764480660499-7e48mtip/main
 15 gsutil ls gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments
 16 gsutil ls gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3
 17 gsutil ls gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3/docker
 18 gsutil ls gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3/docker/student
 19 gsutil ls -l gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3/docker/student
 20 git clone
 21 git clone https://github.com/UCB-w261/main.git
 22 pwd
 23 ls
 24 gsutil cp main/Assignments/HW3/docker/student/hw3_Workbook.ipynb      gs://dataproc-staging-us-west1-764480660499-7e48mtip/notebooks/jupyter/main/Assignments/HW3/docker/student      #copy NOTEBOOKS
 25 gsutil cp gs://w261-hw-data/main/Assignments/HW3/docker/student/ngrams.zip .
 26 unzip ngrams.zip
 27 ls
 28 gsutil -m cp -r data/* gs://cadgi_2022_01/main/Assignments/HW3/docker/student/
 29 gsutil mv gs://cadgi_2022_01/main/Assignments/HW3/docker/student/* gs://cadgi_2022_01/main/Assignments/HW3/docker/student/data/
 30 gsutil cp main/Assignments/HW3/docker/student/hw3_Workbook.ipynb gs://dataproc-staging-us-west1-764480660499-7e48mtip/notebooks/jupyter/main/Assignments/HW3/docker/student/
 31 REGION=us-central1
 32 # CREATE DATAPROC CLUSTER
 33 gcloud dataproc clusters create w261  --enable-component-gateway  --region ${REGION}  --subnet default  --no-address  --single-node  --master-machine-type n1-standard-4  --master-boot-disk-size 100  --image-version 2.0-debian10  --optional-components
JUPYTER  --project $GOOGLE_CLOUD_PROJECT  --properties spark:spark.jars="gs://spark-lib/bigquery/spark-bigquery-latest_2.12.jar"  --properties spark:spark.jars.packages="org.apache.spark:spark-avro_2.12:3.1.2"  --max-idle 3h  --async
 34 gsutil ls -l gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3/docker/student
 35 history
james_shanahan@cloudshell:~ (virtual-bonito-337219)$ echo $GOOGLE_CLOUD_PROJECT
virtual-bonito-337219
(failed reverse-i-search)`PROEJECT=': echo $GOOGLE_CLOUD_PROJECT
```

Goto /media and clone github repo



HW3 data sets (600Meg zipped)

gsutil cp -r gs://w261-hw-data/* gs://dataproc-staging-us-west1-764480660499-7e48mtip/

gsutil ls
gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3/docker/student

```
gs://dataproc-staging-us-west1-764480[redacted]n/Assignments/HW4/
james_shanahan@cloudshell:~ (virtual-bonito-337219)$ gsutil ls gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3
gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3/docker/
james_shanahan@cloudshell:~ (virtual-bonito-337219)$ gsutil ls gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3/docker
^[[Ags://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3/docker/student/
james_shanahan@cloudshell:~ (virtual-bonito-337219)$ gsutil ls gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3/docker/student
gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3/docker/student/ngrams.zip
james_shanahan@cloudshell:~ (virtual-bonito-337219)$ gsutil ls -l gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3/docker/student
 723310158  2022-01-11T20:28:33Z  gs://dataproc-staging-us-west1-764480660499-7e48mtip/main/Assignments/HW3/docker/student/ngrams.zip
TOTAL: 1 objects, 723310158 bytes (689.8 MiB)
james_shanahan@cloudshell:~ (virtual-bonito-337219)$
```

Open a new tab with project...