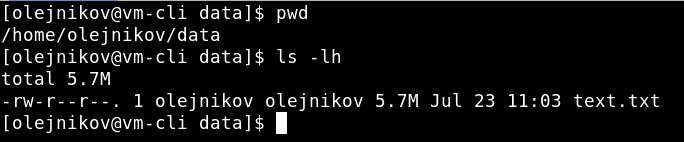
**Лабораторная работа №3**

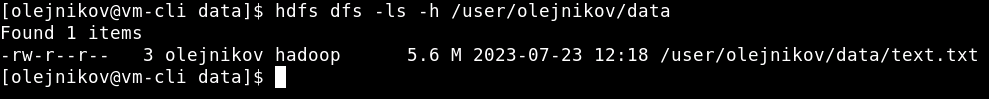
**Олейников Михаил Николаевич (olejnikov)**

**Задание №1**

1. Возьмите любой текстовый файл (расширение .txt), который может содержать текст / посты из интернета / книгу / техническое задание / документацию. Файлдолжен быть **больше 5 MB.**
2. Создайте папку **data** в вашей папке на vm-cli сервере и разместите ваш файл в этой директории.



1. Создайте папку **data** на уровне HDFS кластера под вашим пользователем и в папке, которая соответствует имени вашего пользователя
2. Скопируйте ваш файл в HDFS в папку data под вашим пользователем. Пример вывода представлен ниже.



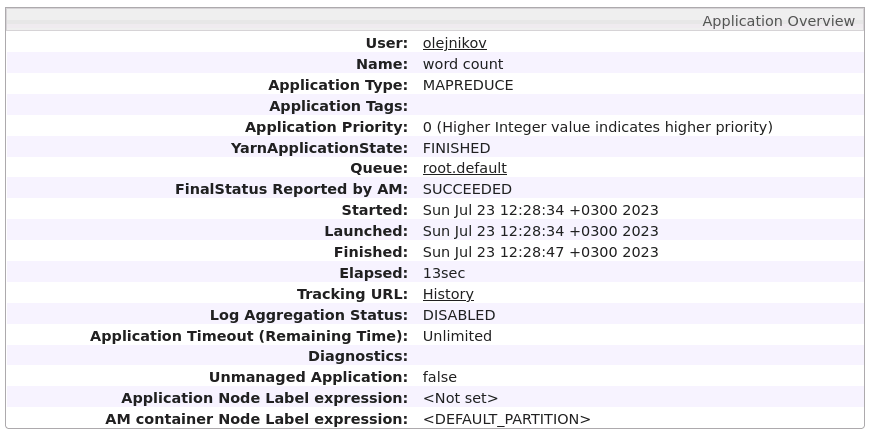
1. Запустите расчет wordcount (подсчет уникальности слов в предоставленном файле), используя bash команду, представленную ниже под вашим пользователем

hadoop jar /usr/lib/hadoop-mapreduce/hadoop-mapreduce-examples-3.2.2.jar wordcount /user/olejnikov/data/text.txt /user/olejnikov/data/results

1. Найдите цифровой след запуска MapReduce для wordcount по адресу:

<http://vm-datalake-m-1:8088/cluster>

(Предоставить преподавателю) Из списка выполненных Application Master задач, выберите свою выполненную задачу и приложите результат (в виде скриншота) к выполненной работе. ВНИМАНИЕ: значение FinalStatus должно быть SUCCEEDED

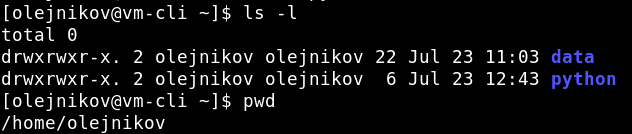


1. Посмотрите результат выполнения алгоритма wordcount , выведя на экран первых 100 строк файла из папки ../data/results (имя папки возможно может отличаться от указанного)



**Задание №2**

1. Давайте определим свой собственный алгоритм **wordcount**.
2. Создайте папку **python** в рамках вашей домашней директории на vm-cli сервере.

****

1. В папке **python** создайте файл с именем **mapper.py**. Это наша логика по работе с процессом маппинга.

Добавьте следующее содержимое файла путем копирования текста ниже. Будьте внимательны со скрытыми символами (перепроверьте содержимое файла после копирования. Не забывайте для Python очень важны отступы и правильное форматирование)

#!/usr/bin/env python3

import sys

for line in sys.stdin:

line = line.strip()

words = line.split()

for word in words:

print('%s\t%s' % (word, 1))

1. В папке **python** создайте файл с именем **reducer.py**. Это наша логика по работе с процессом редьюсинга.

#!/usr/bin/env python3

from operator import itemgetter

import sys

current\_word = None

current\_count = 0

word = None

for line in sys.stdin:

line = line.strip()

word, count = line.split('\t', 1)

try:

count = int(count)

except ValueError:

continue

if current\_word == word:

current\_count += count

else:

if current\_word:

print('%s\t%s' % (current\_word, current\_count))

current\_count = count

current\_word = word

if current\_word == word:

print('%s\t%s' % (current\_word, current\_count))

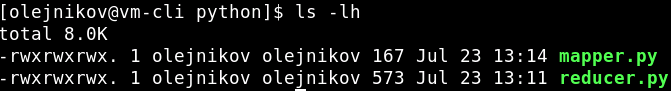
1. Установите права 777 для запуска файлов в рамках процесса MapReduce. Для этого выполните последовательно команды ниже

chmod 777 mapper.py

chmod 777 reducer.py

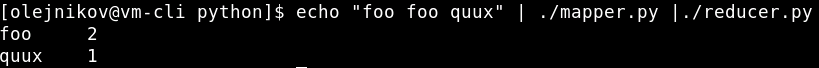
1. Проверьте что ваш алгоритм mapper.py работает, выполнив следующуюпроверку через командную строку (вы должны находиться в папке где расположен ваш файл mapper.py)

echo "foo foo quux" | ./mapper.py



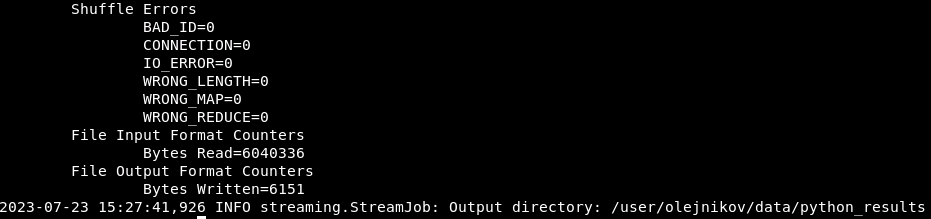
1. Проверьте что ваш алгоритм reducer.py работает, выполнив следующую проверку через командную строку (вы должны находиться в папке где расположен ваш файл reducer.py)

echo "foo foo quux" | ./mapper.py | ./reducer.py



1. Если предыдущие пункты отработали без ошибок, то выполните в командной строке следующую команду, предварительно подготовив ее под свои файлы / папки. В рамках входящего файла укажите ваш файл из **Задания #1**.

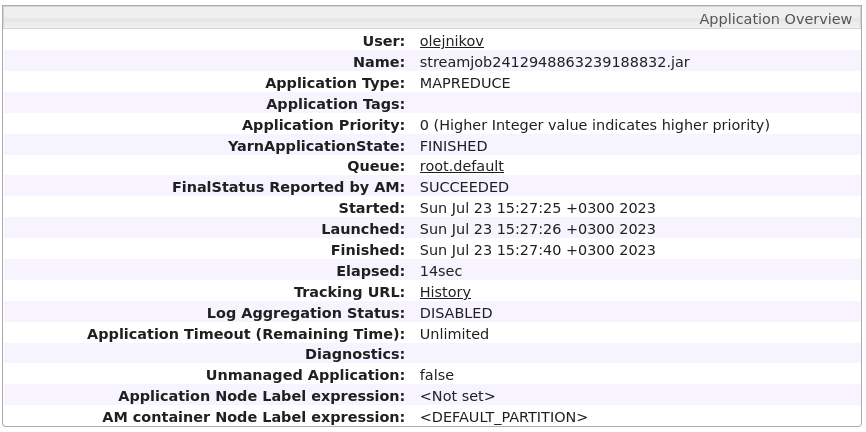
hadoop jar /usr/lib/hadoop/tools/lib/hadoop-streaming-3.2.2.jar -file /home/olejnikov/python/mapper.py -mapper /home/olejnikov/python/mapper.py -file /home/olejnikov/python/reducer.py -reducer /home/olejnikov/python/reducer.py -input /user/olejnikov/data/text.txt -output /user/olejnikov/data/python\_results



1. Найдите цифровой след запуска MapReduce для вашего алгоритма по адресу:

<http://vm-datalake-m-1:8088/cluster>

(Предоставить преподавателю) Из списка выполненных Application Master задач, выберите свою выполненную задачу и приложите результат (в виде скриншота) к выполненной работе. ВНИМАНИЕ: значение FinalStatus должно быть SUCCEEDED



1. Посмотрите результат выполнения вашего алгоритма , выведя на экран первых 100 строк файла из папки ../data/python\_results (имя папки возможно может отличаться от указанного)

(Предоставить преподавателю) приложите пожалуйста к вашей работе скриншот вывода данной команды

