

## MODUL 3 MAP REDUCE



### CAPAIAN PEMBELAJARAN

---

Mahasiswa menggunakan map reduce pada hadoop.



### KEBUTUHAN ALAT/BAHAN/SOFTWARE

---

1. Sistem Operasi Ubuntu 16
2. Hadoop 2.8.5
3. Editor teks (MCEdit, Nano, GEdit)
4. Python 3.5.2



### DASAR TEORI

---

MapReduce adalah sebuah model pemrograman yang didesain untuk dapat melakukan pemrosesan data dengan jumlah yang sangat besar dengan cara membagi pemrosesan tersebut ke beberapa tugas yang independen satu sama lain.

Pengembang aplikasi MapReduce akan membutuhkan beberapa hal berikut dalam melakukan analisis data.

1. Berkas masukan. Berkas masukan ini dapat berupa berkas-berkas teks yang tersimpan di dalam sebuah media penyimpanan terdistribusi seperti Google File System (GFS), Hadoop File System (HDFS), AWS S3, Google Cloud Storage, dan lain-lain.
2. Fungsi Map & Reduce. Untuk membuat sebuah aplikasi MapReduce yang dapat dieksekusi secara paralel (misalkan dengan Hadoop MapReduce), pengembang aplikasi menyediakan fungsi khusus yang digunakan untuk melakukan pemrosesan pada fase map dan reduce. Seluruh hal yang berkaitan dengan penjadwalan, mekanisme penanganan eror, dll. akan dilakukan oleh MapReduce framework yang digunakan.

Berikut beberapa contoh fungsi MapReduce yang dapat dibuat oleh pengembang piranti lunak:

- Word Count
- Inverted Index, dll.



### PRAKTIK

---

Jalankan sistem operasi Linux Ubuntu yang sudah terinstal hadoop. Jalankan server Hadoop dengan menuliskan perintah berikut

```
$ su - hadoop
$ hdfs namenode -format
$ start-dfs.sh
$ start-yarn.sh
$ /opt/jdk1.8.0_192/bin/jps
```

Buka texteditor seperti MCEdit, nano, GEdit atau lainnya. Tuliskan kode program python berikut simpan di alamat /home/hadoop dengan nama file mapper.py

```
mapper.py
1 #!/usr/bin/env python3
2 """mapper.py"""
3
4 import sys
5
6 # input comes from STDIN (standard input)
7 for line in sys.stdin:
8     # remove leading and trailing whitespace
9     line = line.strip()
10    # split the line into words
11    words = line.split()
12    # increase counters
13    for word in words:
14        # write the results to STDOUT (standard output);
15        # what we output here will be the input for the
16        # Reduce step, i.e. the input for reducer.py
17        #
18        # tab-delimited; the trivial word count is 1
19        print('%s\t%s' % (word, 1))
```

Ubah hak akses file menjadi executable, jalankan perintah berikut

```
$ chmod +x /home/hadoop/mapper.py
```

Buka texteditor seperti MCEdit, nano, GEdit atau lainnya. Tuliskan kode program python berikut simpan di alamat /home/hadoop dengan nama file reducer.py

```
reducer.py
1 #!/usr/bin/env python3
2 """reducer.py"""
3
4 from operator import itemgetter
5 import sys
6
7 current_word = None
8 current_count = 0
9 word = None
10
11 # input comes from STDIN
12 for line in sys.stdin:
13     # remove leading and trailing whitespace
14     line = line.strip()
15
16     # parse the input we got from mapper.py
17     word, count = line.split('\t', 1)
```

```

18
19     # convert count (currently a string) to int
20     try:
21         count = int(count)
22     except ValueError:
23         # count was not a number, so silently
24         # ignore/discard this line
25         continue
26
27     # this IF-switch only works because Hadoop sorts map output
28     # by key (here: word) before it is passed to the reducer
29     if current_word == word:
30         current_count += count
31     else:
32         if current_word:
33             # write result to STDOUT
34             print ('%s\t%s' % (current_word, current_count))
35         current_count = count
36         current_word = word
37
38 # do not forget to output the last word if needed!
39 if current_word == word:
40     print ('%s\t%s' % (current_word, current_count))

```

Ubah hak akses file menjadi executable, jalankan perintah berikut

```
$ chmod +x /home/hadoop/ reducer.py
```

Download file profil.txt dan simpan di directory /home/hadoop

Kemudian jalankan perintah seperti berikut

```
$ cat /home/hadoop/profil.txt | /home/hadoop/mapper.py | sort
-k1,1 | /home/hadoop/reducer.py
```

Amatilah hasilnya dan jelaskan di laporan!



## LATIHAN

---

Diberikan oleh dosen / instruktur pengampu



## TUGAS

---

Diberikan oleh dosen / instruktur pengampu



## REFERENSI

---

<https://medium.com/pujanggateknologi/implementasi-sederhana-framework-mapreduce-8f80f22cc54f>

<https://www.michael-noll.com/tutorials/writing-an-hadoop-mapreduce-program-in-python/>