

## HandsOn W05 – MapReduce untuk Data Tabel

Diberikan dataset yang sama, “purchases.txt”, yang digunakan di *Example 02* pada slide, buatlah program map reduce untuk masing-masing milestone di bawah ini. Untuk menjalankan map reduce di Hadoop, file “purchases.txt” tersebut harus sudah ditempatkan di suatu folder di HDFS. Sebelumnya, pastikan Hadoop sudah berjalan di VM yang digunakan (seperti yang telah dilakukan di HandsOn W05 sebelumnya).

### A. Milestone 1

1. Tampilkan total nilai penjualan untuk produk: (i) “Toys” dan (ii) “Consumer Electronics”. Sebagai catatan, nama produk dapat bermacam-macam, selama mengandung salah satu dari kedua string, (i) dan (ii), tersebut. Contoh: “Buffalo Toys”. Output dari milestone ini adalah sebagai berikut.

Consumer Electronics	<jumlah_nilai_penjualan>
Toys	<jumlah_nilai_penjualan>

2. Tampilkan hasil MapReducenya dalam terminal menggunakan perintah `hdfs dfs -cat /folder_output_kamu/file_output`, dan pastekan screenshotnya di bawah.

```

Combine output records=0
Reduce input groups=2
Reduce shuffle bytes=9956857
Reduce input records=459725
Reduce output records=2
Spilled Records=919450
Shuffled Maps=2
Failed Shuffles=0
Merged Map outputs=2
GC time elapsed (ms)=280
CPU time spent (ms)=4870
Physical memory (bytes) snapshot=581607424
Virtual memory (bytes) snapshot=7615131648
Total committed heap usage (bytes)=3986636680
Peak Map Physical memory (bytes)=229695480
Peak Map Virtual memory (bytes)=2536914944
Peak Reduce Physical memory (bytes)=125493248
Peak Reduce Virtual memory (bytes)=2541334528

Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0

File Input Format Counters
  Bytes Read=21117820
File Output Format Counters
  Bytes Written=63
2024-03-05 03:39:15,190 INFO streaming.StreamJob: Output directory: /output_milestone1
(base) bigdata@bigdata:~/project-folder$ hdfs dfs -ls /output_milestone1
Found 2 items
-rw-r--r-- 1 bigdata supergroup 0 2024-03-05 03:39 /output_milestone1/_SUCCESS
-rw-r--r-- 1 bigdata supergroup 63 2024-03-05 03:39 /output_milestone1/part-00000
(base) bigdata@bigdata:~/project-folder$ hdfs dfs -cat /output_milestone1/part-00000
Consumer_Electronics 57452374.13000056
Toys 57463477.109998636
(base) bigdata@bigdata:~/project-folder$
  
```

**Catatan:** semua file python mapper dan reducer yang digunakan selama HandsOn ini, dibutuhkan untuk disubmit. Baca detail format pengumpulannya di bagian paling bawah dari dokumen ini.

**B. Milestone 2:**

- Tampilkan nilai penjualan tertinggi beserta item produknya<sup>1</sup> untuk masing-masing toko yang berada di kota: **Miami**, **San Francisco** dan **Atlanta**. Output dari milestone ini adalah sebagai berikut.

Atlanta	<nilai_penjualan_tertinggi>	<item_produk>
Miami	<nilai_penjualan_tertinggi>	<item_produk>
San Francisco	<nilai_penjualan_tertinggi>	<item_produk>

- Tampilkan hasil MapReducenya dalam terminal menggunakan perintah `hdfs dfs -cat /folder_output_kamu/file_output`, dan pastekan screenshotnya di bawah.

```

File Edit View Search Terminal Help
bigdata@bigdata: ~/project-folder

Reduce input groups=3
Reduce shuffle bytes=3457220
Reduce input records=120086
Reduce output records=3
Spilled Records=240172
Shuffled Maps=2
Failed Shuffles=0
Merged Map outputs=2
GC time elapsed (ms)=449
CPU time spent (ms)=4510
Physical memory (bytes) snapshot=561217536
Virtual memory (bytes) snapshot=7615160320
Total committed heap usage (bytes)=398063680
Peak Map Physical memory (bytes)=219508716
Peak Map Virtual memory (bytes)=2536914944
Peak Reduce Physical memory (bytes)=124055552
Peak Reduce Virtual memory (bytes)=2541363200

Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0

File Input Format Counters
Bytes Read=21117020
File Output Format Counters
Bytes Written=89

2024-03-05 04:01:20,770 INFO streaming.StreamJob: Output directory: /output_milestone2
(base) bigdata@bigdata:~/project-folder$ hdfs dfs -ls /output_milestone2
Found 2 items
-rw-r--r-- 1 bigdata supergroup 0 2024-03-05 04:01 /output_milestone2/_SUCCESS
-rw-r--r-- 1 bigdata supergroup 89 2024-03-05 04:01 /output_milestone2/part-00000
(base) bigdata@bigdata:~/project-folder$ hdfs dfs -cat /output_milestone2/part-00000
Atlanta 499.90 Pet Supplies
Miami 499.90 Video Games
San Francisco 499.97 Men's Clothing
(base) bigdata@bigdata:~/project-folder$

```

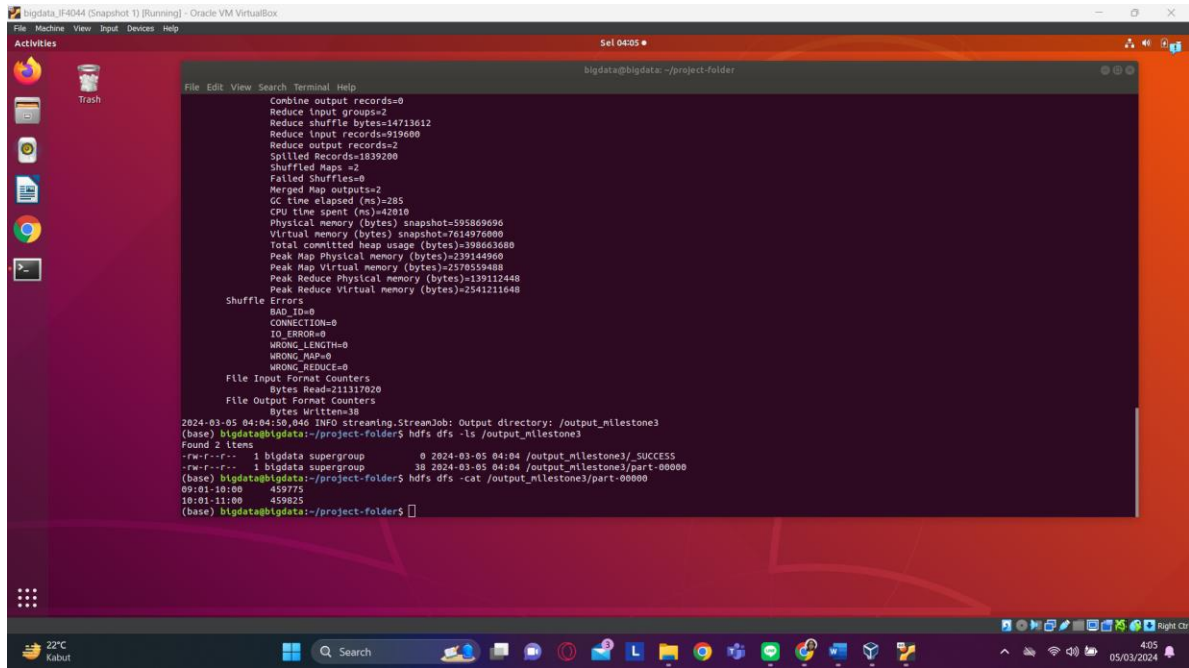
**C. Milestone 3:**

- Tampilkan banyaknya penjualan yang terjadi di rentang jam 09:01-10:00 dan jam 10:01-11:00. Output dari milestone ini adalah sebagai berikut.

09:01-10:00	<banyaknya_penjualan_yang_terjadi>
10:01-11:00	<banyaknya_penjualan_yang_terjadi>

- Tampilkan hasil MapReducenya dalam terminal menggunakan perintah `hdfs dfs -cat /folder_output_kamu/file_output`, dan pastekan screenshotnya di bawah.

<sup>1</sup> Bisa jadi penjualan tertinggi nilainya tidak unik dan terdapat pada beberapa produk. Pada kasus yang demikian, kamu hanya perlu mencantumkan salah satu produknya saja



### Pesan antara:

Implementasi MapReduce dengan membuat file kode secara *custom* untuk mapper dan reducer yang dilakukan di atas memberikan keleluasaan programmer untuk mengembangkan programnya. Akan tetapi, hal tersebut memang diperlukan usaha yang relatif besar untuk membawa permasalahan-permasalahan yang diberikan ke paradigma “map” dan “reduce”. Usaha ini sebanding dengan keuntungan yang bisa kita dapatkan, yaitu mampu mendistribusikan komputasi ke mesin-mesin dalam kluster.

Di dalam ekosistem Hadoop, tersedia sebuah *tool* yang mengubah *SQL-like query* ke komputasi MapReduce yang kemudian dapat didistribusikan ke dalam kluster, yaitu Apache Hive. Dengan menggunakan Apache Hive, seorang programmer dapat mengolah data tabel yang tersimpan (terdistribusi) di kluster layaknya melakukan *query* menggunakan SQL. Sekali lagi, query tersebut kemudian akan dikonversikan ke MapReduce dan akan memproses datanya secara terdistribusi di dalam kluster.

### D. Milestone 4

1. Masuk ke Hive dengan cara seperti yang telah dilakukan di HandsOn W04
2. Buatlah tabel “purchases” dari data “purchases.txt” yang telah disimpan di HDFS. Sebagai referensi, untuk membuat tabel “mahasiswa” dari file (dengan tiga kolom, terpisah dengan koma) yang berada di folder HDFS “/mahasiswa”, dapat dilakukan dengan kode berikut.

```
CREATE TABLE IF NOT EXISTS mahasiswa(ID int, nama string, ipk float)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
STORED AS TEXTFILE
LOCATION '/mahasiswa/';
```

3. Setelah tabel “purchases” terbuat, tes dengan query `select * from purchases limit 10;`
4. Ambil screenshot (hasil query-nya) dan pastekan di bawah ini.

```

File Edit View Search Terminal Help
bigdata@bigdata: ~/project-folder
(base) bigdata@bigdata:~/project-folder$ hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/bigdata/apache-hive-3.1.2-bin/lib/log4j-slf4j-impl-2.10.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/bigdata/hadoop-3.2.2/share/hadoop/common/lib/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
Hive Session ID = a3c26725-8363-4735-b615-85c8626cdc

Logging initialized using configuration in jar:file:/home/bigdata/apache-hive-3.1.2-bin/lib/hive-common-3.1.2.jar!/hive-log4j2.properties Async: true
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 3.X releases.
Hive Session ID = a08d3e44-ada9-45e3-b995-e7d6841f19d8
hive> select * from purchases limit 10;
OK
2012-01-01 09:00:00 San Jose Men's Clothing 214.05 Amex
2012-01-01 09:00:00 Fort Worth Women's Clothing 153.57 Visa
2012-01-01 09:00:00 San Diego Music 66.00 Cash
2012-01-01 09:00:00 Pittsburgh Pet Supplies 493.51 Discover
2012-01-01 09:00:00 Omaha Children's Clothing 235.03 MasterCard
2012-01-01 09:00:00 Stockton Men's Clothing 247.18 MasterCard
2012-01-01 09:00:00 Austin Cameras 379.6 Visa
2012-01-01 09:00:00 New York Consumer Electronics 296.8 Cash
2012-01-01 09:00:00 Corpus Christi Toys 25.38 Discover
2012-01-01 09:00:00 Fort Worth Toys 233.88 Visa
Time taken: 3.586 seconds, Fetched: 10 row(s)
hive>

```

## E. Milestone 5

1. Lakukan Milestone 1, akan tetapi menggunakan query Hive dari tabel “purchases” yang telah dibuat.<sup>2</sup>
2. Ambil screenshot (bagian ekspresi SQL dan hasil query-nya), dan pastekan di bawah ini. Hasil Milestone 1 dan 5 seharusnya memberikan keluaran yang sama<sup>3</sup>.

```

File Edit View Search Terminal Help
bigdata@bigdata: ~/project-folder
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
Hive Session ID = bb9ea2a0-a3f6-459f-a93c-2f2ae011abe7

Logging initialized using configuration in jar:file:/home/bigdata/apache-hive-3.1.2-bin/lib/hive-common-3.1.2.jar!/hive-log4j2.properties Async: true
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 3.X releases.
Hive Session ID = a1e588a8-fdd7-4711-8962-d9454b798839
hive> select product, sum(price) as total_sales from purchases where product like 'XConsumer Electronics' or product like 'XToys' group by product;
Query ID = bigdata_20240305022734_95eb3442-7566-432f-ab01-2fbeeaf314581
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.educses=number>
Starting Job = job_1709579358049_0002, Tracking URL = http://bigdata:8088/proxy/application_1709579358049_0002/
Kill Command = /home/bigdata/hadoop-3.2.2/bin/mapred job -kill job_1709579358049_0002
Hadoop job information for stage-1: number of mappers: 1; number of reducers: 1
2024-03-05 02:27:18,908 Stage-1 map = 0%, reduce = 0%
2024-03-05 02:28:12,815 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 4.97 sec
2024-03-05 02:28:12,118 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 6.68 sec
MapReduce Total cumulative CPU time: 6 seconds 680 msec
Ended Job = job_1709579358049_0002
MapReduce Jobs Launched:
Stage:Stage 1: Map: 1 Reduce: 1 Cumulative CPU: 6.68 sec HDFS Read: 223830683 HDFS Write: 177 SUCCESS
Total MapReduce CPU Time Spent: 6 seconds 680 msec
OK
Consumer Electronics 5.745237412163785E7
Toys 5.746347711329821E7
Time taken: 49.29 seconds, Fetched: 2 row(s)
hive>

```

<sup>2</sup> Bagi VM dengan size RAM kecil, kemungkinan proses akan terhenti di tengah. Jika tidak memungkinkan untuk menambahkan size RAM di VM, ambil screenshot “ekspresi SQL yang kamu buat” dan “pesan errornya”.

<sup>3</sup> Hiraukan perbedaan minor string “lowercase” dan “Capital Each Word”.

## F. Milestone 6

1. Lakukan Milestone 2, akan tetapi menggunakan query Hive dari tabel “purchases” yang telah dibuat.
2. Ambil screenshot (bagian ekspresi SQL dan hasil query-nya), dan pastekan di bawah ini. Hasil Milestone 2 dan 6 seharusnya memberikan keluaran yang sama.

```

File Edit View Search Terminal Help
bigdata@bigdata: ~/project-folder

Hive-on-HR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (
i.e., spark, tez) or using Hive 3.X releases.
Hive Session ID = 43aa192a-5c47-48a5-9751-305426199fa3
hive> select location, price, product
> from (
>   select location, price, product, row_number() over (partition by location order by price desc) as rank
>   from purchases
>   where location in ('Atlanta', 'Miami', 'San Francisco')
> ) ranked
> where rank = 1;
Query ID = bigdata_20240305024904_44386678-e574-4c9b-bd5-cf40998833fd
Total Jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1709579358049_0004, Tracking URL = http://bigdata:8080/proxy/application_1709579358049_0004/
Kill Command = /home/bigdata/hadoop-3.2.2/bin/mapred job -kill job_1709579358049_0004
Hadoop Job Information for Stage-1: number of mappers: 1; number of reducers: 1
2024-03-05 02:49:19,735 Stage-1 map = 0%, reduce = 0%
2024-03-05 02:49:31,424 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 5.09 sec
2024-03-05 02:49:42,074 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 8.19 sec
MapReduce Total cumulative CPU time: 8 seconds 190 msec
Ended Job = job_1709579358049_0004
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 8.19 sec HDFS Read: 223030008 HDFS Write: 215 SUCCESS
Total MapReduce CPU Time Spent: 8 seconds 190 msec
OK
Atlanta 499.96 Pet Supplies
Miami 499.98 Men's Clothing
San Francisco 499.97 Men's Clothing
Time taken: 40.45 seconds, Fetched: 3 row(s)
hive>

```

## G. Milestone 7

1. Lakukan Milestone 3, akan tetapi menggunakan query Hive dari tabel “purchases” yang telah dibuat.
2. Ambil screenshot (bagian ekspresi SQL dan hasil query-nya), dan pastekan di bawah ini. Hasil Milestone 3 dan 7 seharusnya memberikan keluaran yang sama.



```

File Edit View Search Terminal Help
bigdata@bigdata: ~/project-folder

MapReduce Total cumulative CPU time: 12 seconds 290 msec
Ended Job = job_1709579358049_0031
Launching Job 2 out of 3
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reducers=<number>
Starting Job = job_1709579358049_0032, Tracking URL = http://bigdata:8088/proxy/application_1709579358049_0032/
Kill Command = /home/bigdata/hadoop-3.2.2/bin/mapred job -kill job_1709579358049_0032
Hadoop job information for Stage-3: number of mappers: 1; number of reducers: 1
2024-03-05 04:34:14,308 Stage-3 map = 0%, reduce = 0%
2024-03-05 04:34:37,424 Stage-3 map = 100%, reduce = 100%, Cumulative CPU 9.85 sec
MapReduce Total cumulative CPU time: 11 seconds 450 msec
Ended Job = job_1709579358049_0032
Launching Job 3 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1709579358049_0033, Tracking URL = http://bigdata:8088/proxy/application_1709579358049_0033/
Kill Command = /home/bigdata/hadoop-3.2.2/bin/mapred job -kill job_1709579358049_0033
Hadoop job information for Stage-2: number of mappers: 2; number of reducers: 0
2024-03-05 04:34:54,582 Stage-2 map = 0%, reduce = 0%
2024-03-05 04:35:04,001 Stage-2 map = 50%, reduce = 0%, Cumulative CPU 1.14 sec
2024-03-05 04:35:05,493 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 2.32 sec
MapReduce Total cumulative CPU time: 2 seconds 320 msec
Ended Job = job_1709579358049_0033
MapReduce Job Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 12.29 sec HDFS Read: 223833781 HDFS Write: 129 SUCCESS
Stage-Stage-3: Map: 1 Reduce: 1 Cumulative CPU: 11.45 sec HDFS Read: 223833785 HDFS Write: 129 SUCCESS
Stage-Stage-2: Map: 2 Cumulative CPU: 2.32 sec HDFS Read: 7692 HDFS Write: 236 SUCCESS
Total MapReduce CPU Time Spent: 20 seconds 60 msec
OK
09:01-10:00 459775
10:01-11:00 459825
Time taken: 101.662 seconds, Fetched: 2 row(s)
hive>

```

## H. Milestone 8

1. Jika pada Milestone 5, 6 dan 7 hasil yang didapatkan tidak dapat menyamai<sup>4</sup> dari Milestone 1, 2 dan 3, berikan analisa kamu. Jika alasannya adalah terkait keterbatasan SQL-like query, berikan ide/solusinya agar hasil dari Milestone 5, 6 dan 7 secara berturut-turut sama dengan Milestone 1, 2 dan 3.

Berdasarkan handson yang telah dilakukan, terdapat perbedaan hasil antara milestone 1 dengan milestone 5 dan milestone 2 dengan milestone 6.

Perbedaan hasil antara milestone 1 dengan milestone 5 terletak pada angka yang terdapat dibelakang koma pada hasil penjumlahan data nilai penjualan produk yang ada. Hal yang dapat dilakukan adalah dengan lebih mendefenisikan jumlah angka dibelakang koma, dengan cara seperti itu maka hasil penjumlahan nilai penjualan akan lebih serupa.

Perbedaan hasil antara milestone 2 dengan milestone 6 terletak pada data dengan nilai location Miami, pada milestone 2 nilai product nya adalah Video Games sedangkan pada milestone 6 nilai product nya adalah Men's Clothing, meskipun nilai penjualan pada dua product tersebut sama. Hal tersebut dapat terjadi karena pada reducer yang telah dibuat pencarian yang dilakukan hanya mencari nilai penjualan maksimum, tetapi tidak mengupdate value dari product apabila terdapat product yang memiliki nilai penjualan serupa. Sehingga ketika record dengan product Video Games muncul lebih awal, product yang akan tercatat adalah Video Games. Namun, ketika implementasi menggunakan hive yang merupakan SQL-like query record yang ada diurutkan berdasarkan nilai penjualan kemudian nama product sehingga product yang tercatat adalah Men's Clothing.

<sup>4</sup> Hiraukan perbedaan minor string "lowercase" dan "Capital Each Word".

Setelah semua screenshot dipastekan di masing-masing milestone, upload file zip dengan nama: “W05\_NIM\_NamaLengkap.zip” ke form submission/assignment di **edunex** yang telah disediakan. Adapun isi dari file zipnya adalah:

1. File pdf dari dokumen ini, dengan nama: “W05\_NIM\_NamaLengkap.pdf”
2. File mapper dan reducer dari dari Milestone 1-3, dengan format nama “mapper\_milestone1.py” dan “reducer\_milestone1.py” untuk Milestone 1, begitu seterusnya hingga Milestone 3.

--- done ---