

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	57452374.130001
Toys	57463477.10999886

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

```

bigdata@bigdata: ~/project-folder
File Edit View Search Terminal Help
(base) bigdata@bigdata:~/project-folder$ hdfs dfs -ls /output_purchases/milestone1
Found 2 items
-rw-r--r-- 1 bigdata supergroup 0 2024-03-01 22:01 /output_purchases/milestone1/success
-rw-r--r-- 1 bigdata supergroup 68 2024-03-01 22:01 /output_purchases/milestone1/part-00000
(base) bigdata@bigdata:~/project-folder$ hdfs dfs -cat /output_purchases/milestone1/part-00000
Consumer Electronics 57452374.130001
Toys 57463477.10999886
(base) bigdata@bigdata:~/project-folder$
  
```

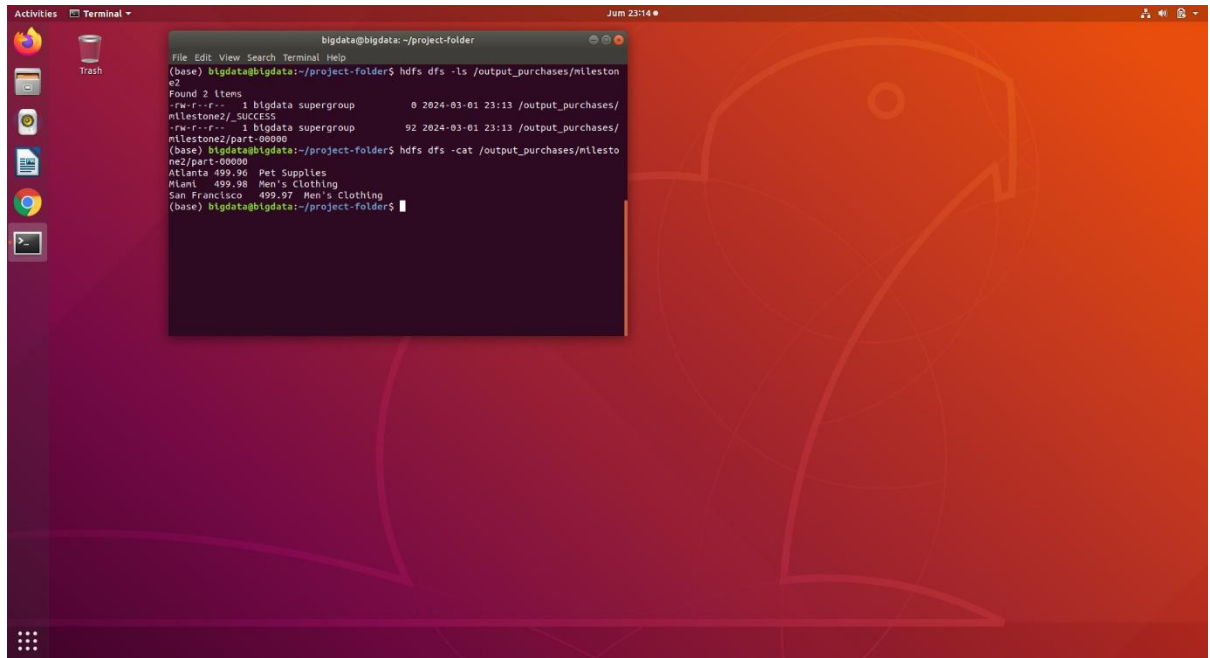
Catatan: semua file python mapper dan reducer yang digunakan selama HandsOn ini, dibutuhkan untuk disubmit. Baca detail format pengumpulannya di bagian paling bawah dari dokumen ini.

B. Milestone 2:

1. Tampilkan nilai penjualan tertinggi beserta item produknya¹ untuk masing-masing toko yang berada di kota: **Miami**, **San Francisco** dan **Atlanta**. Output dari milestone ini adalah sebagai berikut.

Atlanta	499.96	Pet Supplies
Miami	499.98	Men's Clothing
San Francisco	499.97	Men's Clothing

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



```

bigdata@bigdata: ~/project-folder
(base) bigdata@bigdata:~/project-folder$ hdfs dfs -ls /output_purchases/milestone2
Found 2 items
-rw-r--r-- 1 bigdata supergroup 0 2024-03-01 23:13 /output_purchases/milestone2/_SUCCESS
-rw-r--r-- 1 bigdata supergroup 92 2024-03-01 23:13 /output_purchases/milestone2/part-00000
(base) bigdata@bigdata:~/project-folder$ hdfs dfs -cat /output_purchases/milestone2/part-00000
Atlanta 499.96 Pet Supplies
Miami 499.98 Men's Clothing
San Francisco 499.97 Men's Clothing
(base) bigdata@bigdata:~/project-folder$

```

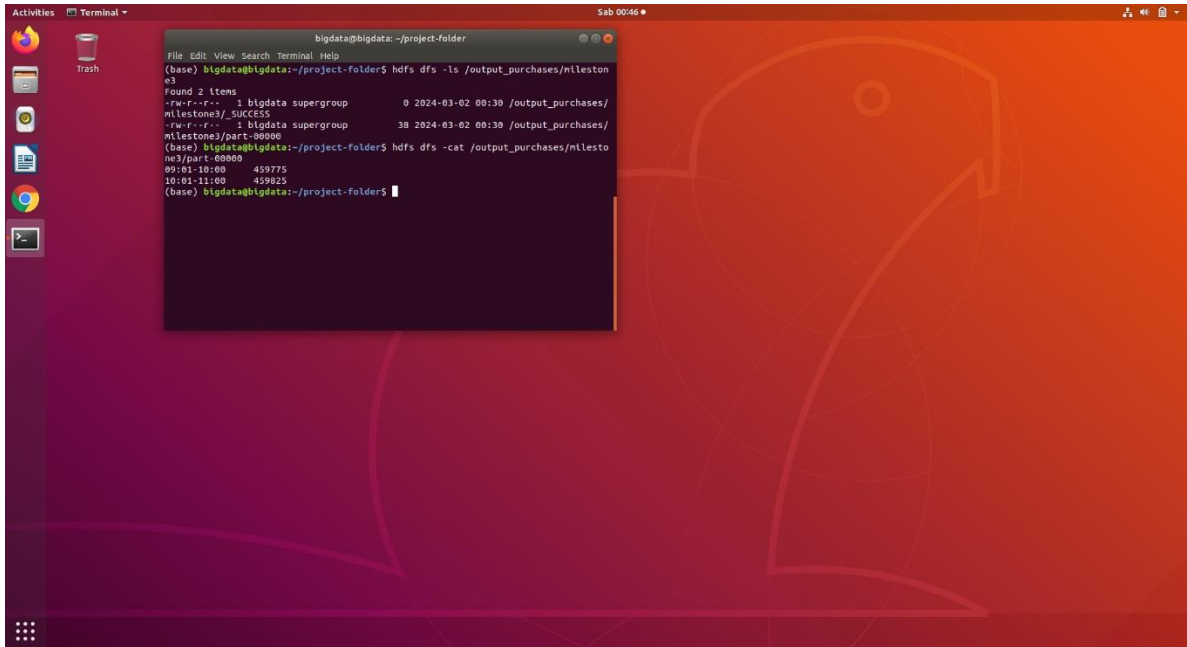
C. Milestone 3:

1. 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	459775
10:01-11:00	459825

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

¹ 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.

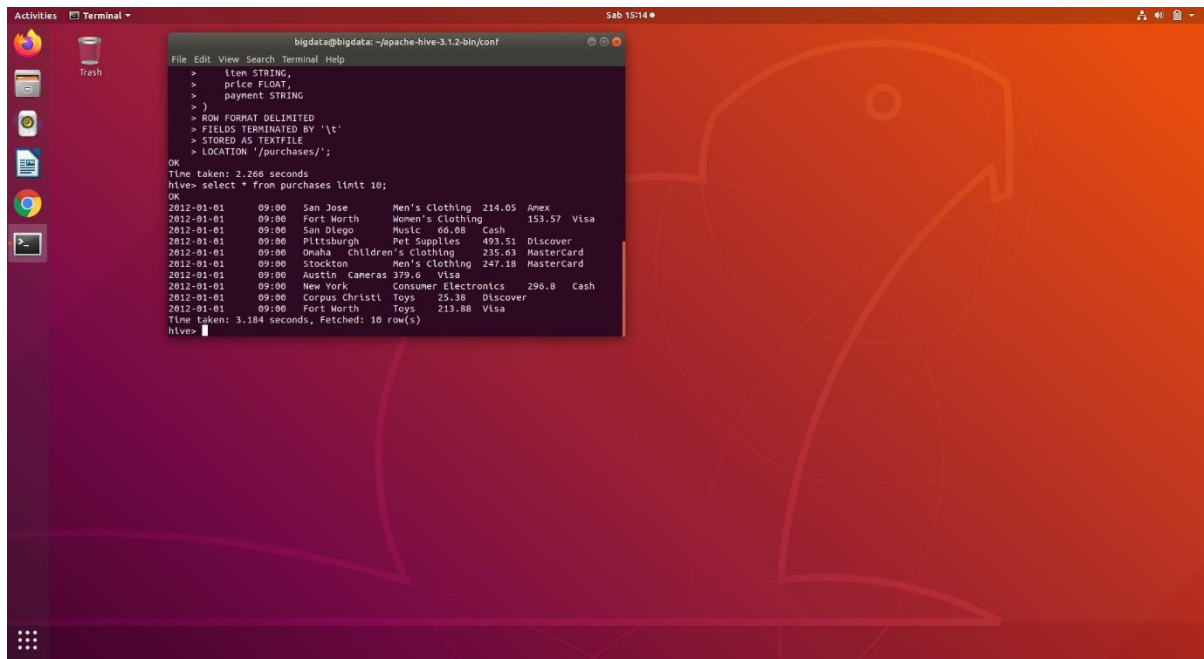
Di dalam ekosistem Hadoop, tersedia sebuah *tool* yang mengubah *SQL-like query* ke komputasi MapReduce yang kemudian dapat didistribusikan ke dalam kluster, yaitu Apache Hive. Dengan menggunakan Apache Hive, seorang programmer dapat mengolah data tabel yang tersimpan (terdistribusi) di kluster layaknya 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.



E. Milestone 5

1. Lakukan Milestone 1, akan tetapi menggunakan query Hive dari tabel “purchases” yang telah dibuat.²
2. Ambil screenshot (bagian ekspresi SQL dan hasil query-nya), dan pastekan di bawah ini. Hasil Milestone 1 dan 5 seharusnya memberikan keluaran yang sama³.

Ekspresi SQL:

```
SELECT
    SUM(price) AS total_sales
FROM
    purchases
WHERE
    item LIKE '%Consumer Electronics%';
```

² 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”.

³ Hiraukan perbedaan minor string “lowercase” dan “Capital Each Word”.

```

Activities Terminal Sab 17:29
bigdata@bigdata: ~/apache-hive-3.1.2-bin/conf

(base) bigdata@bigdata:~/apache-hive-3.1.2-bin/conf$ 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 = e401fddb-4fe7-446f-b788-95bfe456e7b0

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 1.X releases.
Hive Session ID = de4714b9-6b58-42ea-9707-63714f289e8e
hive> SELECT
> SUM(price) AS total_sales
> FROM
> purchases
> WHERE
> item LIKE 'NConsumer Electronics%';
Query ID = bigdata_20240302172810_84ad87ba-6ab7-4b4a-94cf-ee1dcf5d7b
Total jobs = 1
Launching Job 1 out of 1
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.reduces=<number>
Starting Job = job_1709325153574_0027, Tracking URL = http://bigdata:8088/proxy/application_1709325153574_0027/
Kill Command = /home/bigdata/hadoop-3.2.2/bin/mapred job -kill job_1709325153574_0027
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2024-03-02 17:28:35,521 Stage-1 map = 0%, reduce = 0%
2024-03-02 17:28:47,330 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 8.1 sec
2024-03-02 17:28:53,783 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 9.94 sec
MapReduce Total cumulative CPU time: 9 seconds 940 msec
Ended Job = job_1709325153574_0027
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 9.94 sec HDFS Read: 211332204 HDFS Write: 119 SUCCESS
Total MapReduce CPU Time Spent: 9 seconds 940 msec
OK
5.745237412163785E7
Time taken: 38.59 seconds, Fetched: 1 row(s)
hive>

```

```

SELECT
SUM(price) AS total_sales
FROM
purchases
WHERE
item LIKE '%Toys%';

```

```

Activities Terminal Sab 17:31
bigdata@bigdata: ~/apache-hive-3.1.2-bin/conf

(base) bigdata@bigdata:~/apache-hive-3.1.2-bin/conf$ 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 = fa784aba-7cbc-4ae5-85a5-538256399912

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 1.X releases.
Hive Session ID = 84db5cc0-c83-4794-bbf0-6674c9b9e1aa
hive> SELECT
> SUM(price) AS total_sales
> FROM
> purchases
> WHERE
> item LIKE 'NToys%';
Query ID = bigdata_20240302173032_90f3d364-7a63-4a35-8a62-91a3f15799e8
Total jobs = 1
Launching Job 1 out of 1
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.reduces=<number>
Starting Job = job_1709325153574_0028, Tracking URL = http://bigdata:8088/proxy/application_1709325153574_0028/
Kill Command = /home/bigdata/hadoop-3.2.2/bin/mapred job -kill job_1709325153574_0028
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2024-03-02 17:30:48,946 Stage-1 map = 0%, reduce = 0%
2024-03-02 17:30:59,896 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 6.57 sec
2024-03-02 17:31:06,880 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 10.01 sec
MapReduce Total cumulative CPU time: 10 seconds 10 msec
Ended Job = job_1709325153574_0028
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 10.01 sec HDFS Read: 211332133 HDFS Write: 119 SUCCESS
Total MapReduce CPU Time Spent: 10 seconds 10 msec
OK
5.746347711329821E7
Time taken: 36.825 seconds, Fetched: 1 row(s)
hive>

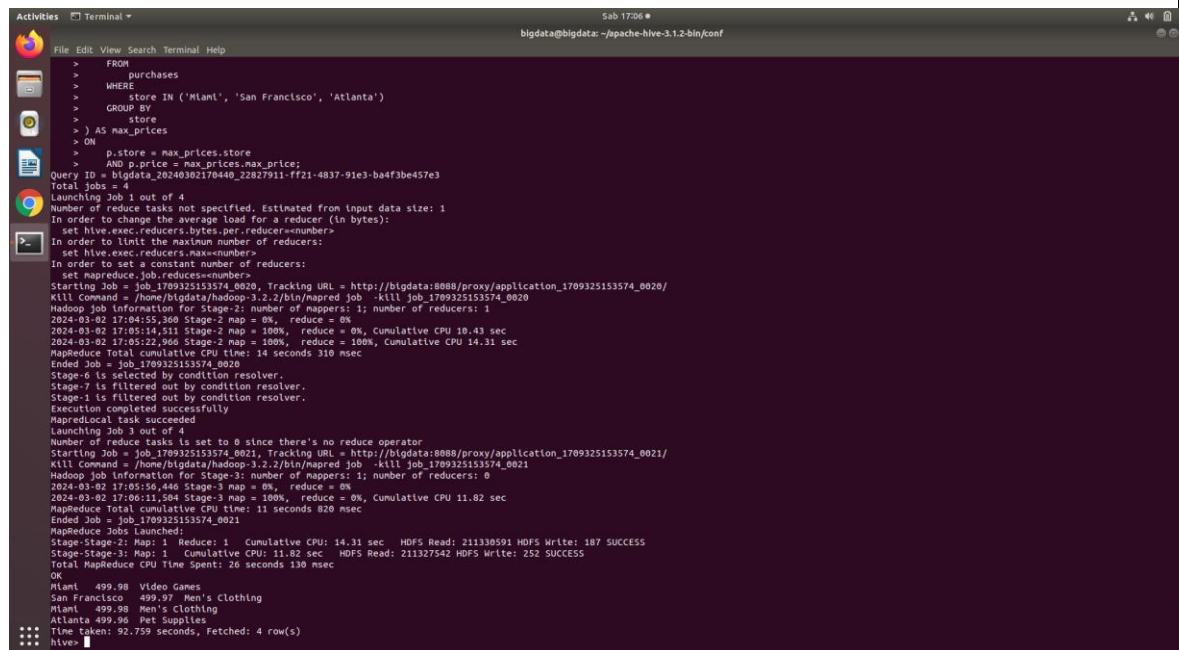
```

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.

Ekspresi SQL:

```
SELECT
    p.store,
    p.price AS max_price,
    p.item
FROM
    purchases p
JOIN (
    SELECT
        store,
        MAX(price) AS max_price
    FROM
        purchases
    WHERE
        store IN ('Miami', 'San Francisco', 'Atlanta')
    GROUP BY
        store
) AS max_prices
ON
    p.store = max_prices.store
    AND p.price = max_prices.max_price;
```



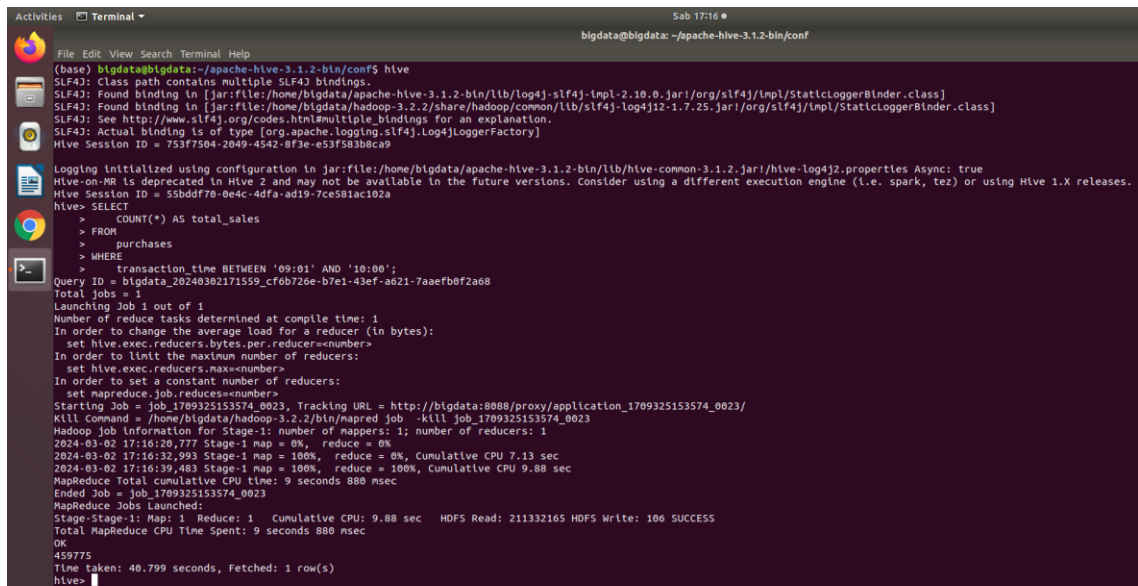
```
Activities Terminal
bigdata@bigdata: ~ -japche-hive-3.1.2-bin/conf
> FROM
>   purchases
> WHERE
>   store IN ('Miami', 'San Francisco', 'Atlanta')
> GROUP BY
>   store
> ) AS max_prices
> ON
>   p.store = max_prices.store
>   AND p.price = max_prices.max_price;
Query ID = bigdata_20240302170440_22827911-ff21-4037-91e3-ba4f3be457e3
Total jobs = 4
Launching Job 1 out of 4
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_1709325153574_0020, Tracking URL = http://bigdata:8080/proxy/application_1709325153574_0020/
Kill Command = /home/bigdata/hadoop-3.2.2/bin/mapred job -kill job_1709325153574_0020
Hadoop job information for Stage-2: number of mappers: 1; number of reducers: 1
2024-03-02 17:04:55,360 Stage-2 map = 0%, reduce = 0%, Cumulative CPU 10.43 sec
2024-03-02 17:05:14,511 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 10.43 sec
2024-03-02 17:05:22,960 Stage-2 map = 100%, reduce = 100%, Cumulative CPU 14.31 sec
MapReduce Total cumulative CPU time: 14 seconds 310 msec
Ended Job = Job_1709325153574_0020
Stage-6 is selected by condition resolver.
Stage-7 is filtered out by condition resolver.
Stage-1 is filtered out by condition resolver.
Execution completed successfully
MapReduce task succeeded
Launching Job 3 out of 4
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = Job_1709325153574_0021, Tracking URL = http://bigdata:8080/proxy/application_1709325153574_0021/
Kill Command = /home/bigdata/hadoop-3.2.2/bin/mapred job -kill job_1709325153574_0021
Hadoop job information for Stage-3: number of mappers: 1; number of reducers: 0
2024-03-02 17:05:56,446 Stage-3 map = 0%, reduce = 0%
2024-03-02 17:06:11,504 Stage-3 map = 100%, reduce = 0%, Cumulative CPU 11.02 sec
MapReduce Total cumulative CPU time: 11 seconds 920 msec
Ended Job = Job_1709325153574_0021
MapReduce Jobs Launched:
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 14.31 sec HDFS Read: 211330591 HDFS Write: 187 SUCCESS
Stage-Stage-3: Map: 1 Cumulative CPU: 11.02 sec HDFS Read: 211327542 HDFS Write: 252 SUCCESS
Total MapReduce CPU Time Spent: 26 seconds 130 msec
OK
Miami 499.98 Video Games
San Francisco 499.97 Men's Clothing
Miami 499.98 Men's Clothing
Atlanta 499.96 Pet Supplies
Time taken: 92.759 seconds, Fetched: 4 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.

Eksprei SQL:

```
SELECT
    COUNT(*) AS total_sales
FROM
    purchases
WHERE
    transaction_time BETWEEN '09:01' AND '10:00';
```



```
bigdata@bigdata: ~/apache-hive-3.1.2-bin/conf
(base) bigdata@bigdata:~/apache-hive-3.1.2-bin/conf$ 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 = 753f7504-2049-4542-8f3e-e53f583b8ca9
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 1.X releases.
Hive Session ID = 55bddf70-0e4c-4dfa-ad19-7ce581ac102a
hive> SELECT
>     COUNT(*) AS total_sales
> FROM
>     purchases
> WHERE
>     transaction_time BETWEEN '09:01' AND '10:00';
Query ID = bigdata_20240302171559_cf6b726e-b7e1-43ef-a621-7aaefb0f2a68
Total jobs = 1
Launching Job 1 out of 1
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_1709325153574_0023, Tracking URL = http://bigdata:8088/proxy/application_1709325153574_0023/
Kill Command = /home/bigdata/hadoop-3.2.2/bin/mapred job -kill job_1709325153574_0023
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2024-03-02 17:16:20.777 Stage-1 map = 0%, reduce = 0%
2024-03-02 17:16:32.993 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 7.13 sec
2024-03-02 17:16:39.483 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 9.88 sec
MapReduce Total cumulative CPU time: 9 seconds 880 msec
Ended Job = job_1709325153574_0023
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 9.88 sec HDFS Read: 211332165 HDFS Write: 106 SUCCESS
Total MapReduce CPU Time Spent: 9 seconds 880 msec
OK
459775
Time taken: 40.799 seconds, Fetched: 1 row(s)
hive>
```

```
SELECT
    COUNT(*) AS total_sales
FROM
    purchases
WHERE
    transaction_time BETWEEN '10:01' AND '11:00';
```



```

Activities Terminal ▾
bigdata@bigdata: ~/apache-hive-3.12-bin/conf
File Edit View Search Terminal Help
Launching Job 1 out of 1
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.reduces=<number>
Starting Job = job_1709325153574_0023, Tracking URL = http://bigdata:8088/proxy/application_1709325153574_0023/
Kill Command = /home/bigdata/hadoop-3.2.2/bin/mapred job -kill job_1709325153574_0023
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2024-03-02 17:16:20,777 Stage-1 map = 0%, reduce = 0%
2024-03-02 17:16:32,993 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 7.13 sec
2024-03-02 17:16:39,483 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 9.88 sec
MapReduce Total cumulative CPU time: 9 seconds 880 msec
Ended Job = job_1709325153574_0023
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 9.88 sec HDFS Read: 211332165 HDFS Write: 106 SUCCESS
Total MapReduce CPU Time Spent: 9 seconds 880 msec
OK
459775
Time taken: 40.799 seconds, Fetched: 1 row(s)
hive> SELECT
  > COUNT(*) AS total_sales
  > FROM
  > purchases
  > WHERE
  > transaction_time BETWEEN '10:01' AND '11:00';
Query ID = bigdata_20240302171815_786605e6-7be6-4c3b-8a5f-e4b04646123
Total jobs = 1
Launching Job 1 out of 1
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.reduces=<number>
Starting Job = job_1709325153574_0024, Tracking URL = http://bigdata:8088/proxy/application_1709325153574_0024/
Kill Command = /home/bigdata/hadoop-3.2.2/bin/mapred job -kill job_1709325153574_0024
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2024-03-02 17:18:30,679 Stage-1 map = 0%, reduce = 0%
2024-03-02 17:18:44,045 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 9.21 sec
2024-03-02 17:18:50,372 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 11.79 sec
MapReduce Total cumulative CPU time: 11 seconds 790 msec
Ended Job = job_1709325153574_0024
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 11.79 sec HDFS Read: 211332175 HDFS Write: 106 SUCCESS
Total MapReduce CPU Time Spent: 11 seconds 790 msec
OK
459825
Time taken: 37.213 seconds, Fetched: 1 row(s)
hive>

```

H. Milestone 8

1. Jika pada Milestone 5, 6 dan 7 hasil yang didapatkan tidak dapat menyamai⁴ 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.
 - Pada Milestone 1 dan 5 didapatkan hasil yang berbeda, hal ini dikarenakan perbedaan penanganan floating point pada Python dan Hive. Namun, kedua hasilnya saling mendekati sehingga apabila dilakukan pembulatan maka akan menghasilkan nilai yang sama.
 - Pada Milestone 2 dan 6 didapatkan hasil yang sama dimana harga tertinggi untuk Atlanta adalah Pet Supplies, harga tertinggi untuk Miami adalah Men's Clothing, dan harga tertinggi untuk San Fransisco adalah Men's Clothing. Meskipun pada pada Miami terdapat 2 item dengan harga tertinggi yang salah satunya adalah Video Games, namun pada spesifikasi diperbolehkan memilih salah satu item saja sehingga hal ini memenuhi.
 - Pada Milestone 3 dan 7 didapatkan hasil yang sama dimana jumlah penjualan di antara 09.01-10.00 adalah 459775 dan jumlah penjualan di antara 10.01-11.00 adalah 459825.

⁴ 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 ---