**LAPORAN**

**RENCANA TUGAS MANDIRI (RTM) Ke-4**

**MATA KULIAH BIG DATA C**

**“QUERY STATISTIK DESKRIPTIF MENGGUNAKAN HIVE DAN DATA XQUERY”**



**DISUSUN OLEH:**

Mohamad Ibnu Fajar Maulana (21083010106)

**DOSEN PENGAMPU:**

Tresna Maulana Fahrudin S.ST., M.T. (NIP. 199305012022031007)

**PROGRAM STUDI SAINS DATA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN” JAWA TIMUR**

**2022**

1. Analisis pada dataset NOAA menggunakan Hive

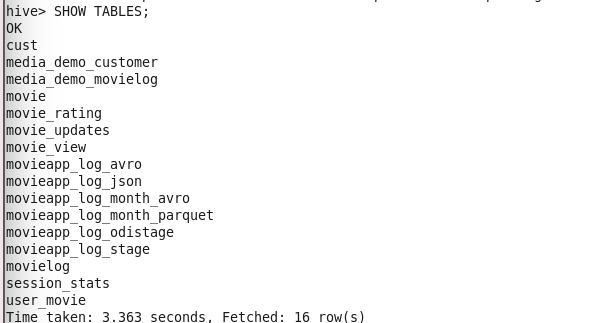
* Jalankan Hive dengan seperti kode dibawah ini

$ hive



* SHOW TABLES; untuk melihat tabel tabel yang ada di Hive

hive> SHOW TABLES;



* Membuat tabel suhu

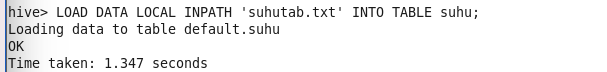
hive> create table suhu(tahun int, suhu int, kualitas int);



* Load data local ‘suhutab.txt

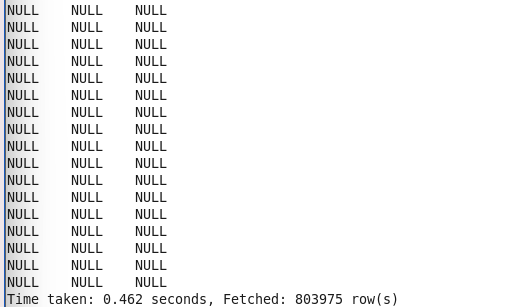
hive> LOAD DATA LOCAL INPATH ‘suhutab.txt’ INTO TABLE

suhu;



* Melihat keseluruhan tabel suhu

hive> select \* from suhu;



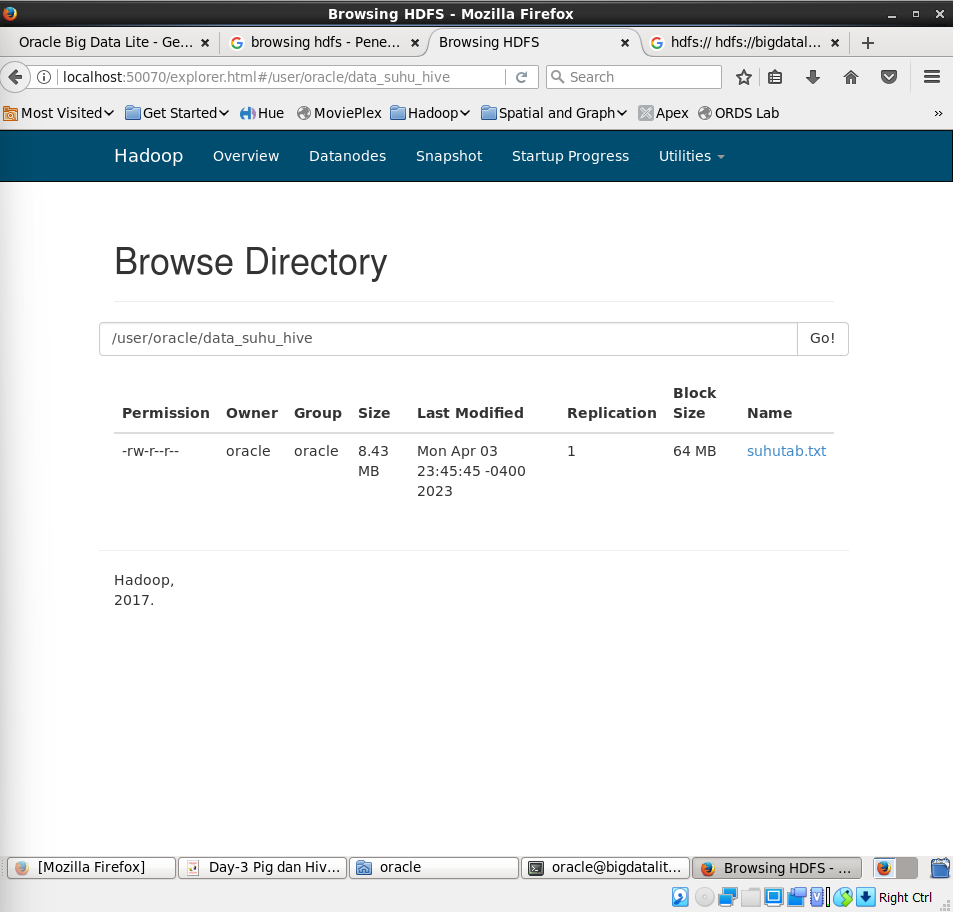
* Truncate tabel suhu

hive> truncate from suhu;



* Membuat folder data\_suhu\_hive pada hadoop

$ fs -mkdir data\_suhu\_hive

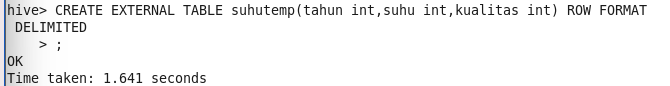


* Copy data suhutab.txt ke hadoop

$ fs -copyFromLocal suhutab.txt data\_suhu\_hive/suhutab.txt

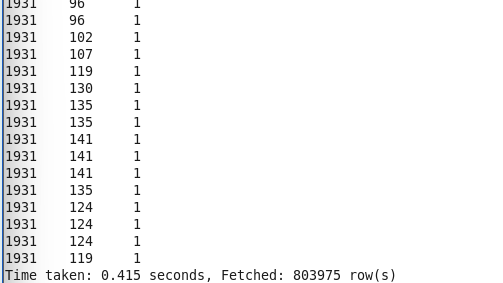
* Mebuat external table yaitu suhutemp pada Hive

hive> CREATE EXTERNAL TABLE suhutemp(tahun int,suhu int,kualitas int) ROW FORMAT DELIMITED FIELDS TERMINATED BY ‘\t’ STORED AS TEXTFILE LOCATION ‘/user/oracle/data\_suhu\_hive’;



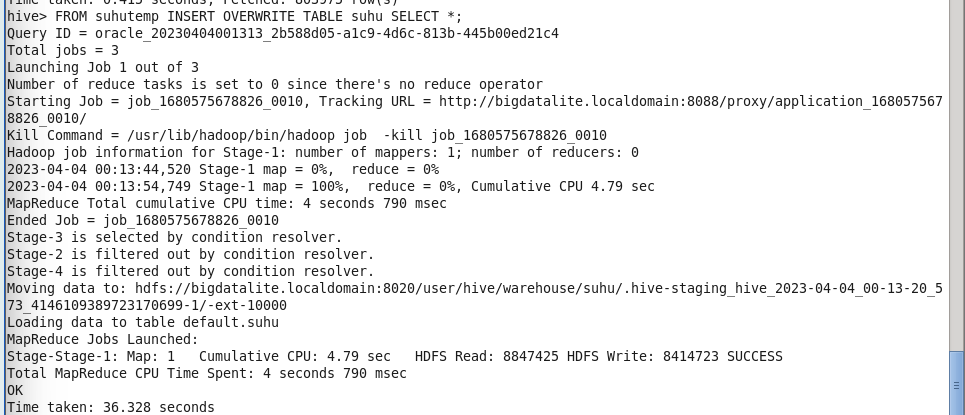
* Melihat keseluruhan tabel suhutemp

hive> SELECT \* FROM suhutemp;



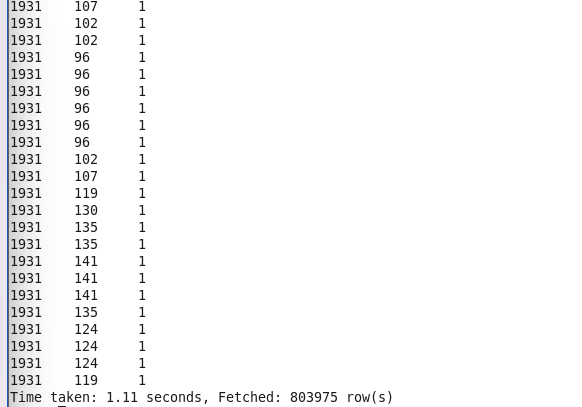
* Memindahkan suhutemp ke tabel utama

hive> FROM suhutemp INSERT OVERWRITE TABLE suhu SELECT \*;



* Melihat keseluruhan tabel suhu

hive> SELECT \* FROM suhu;



1. Melakukan Statistika deskriptif (suhu maksimum, minimum, rata-rata, varian, deviasi standar, dan persentil) yang dikelompokkkan berdasarkan masing-masing tahun

* Mencari nilai suhu maksimum, dengan kode berikut

hive> CREATE TABLE nilai\_max (

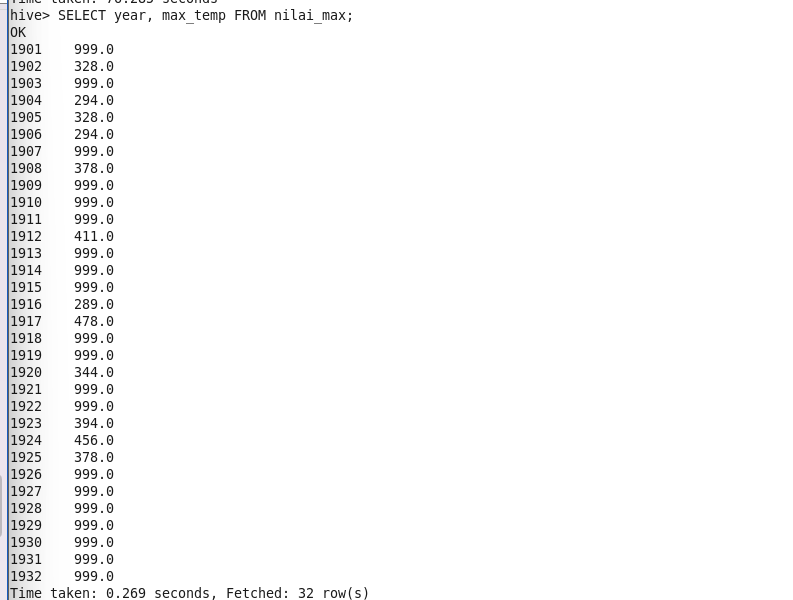
* year INT,
* max\_temp FLOAT
* );

hive> INSERT INTO nilai\_max

* SELECT tahun, MAX(suhu)
* FROM suhu
* GROUP BY tahun;

hive> SELECT year, max\_temp FROM nilai\_max;

Mendapatkan Output nilai maksimum:



* Mencari nilai suhu minimum, dengan kode berikut ini

hive> CREATE TABLE nilai\_min (

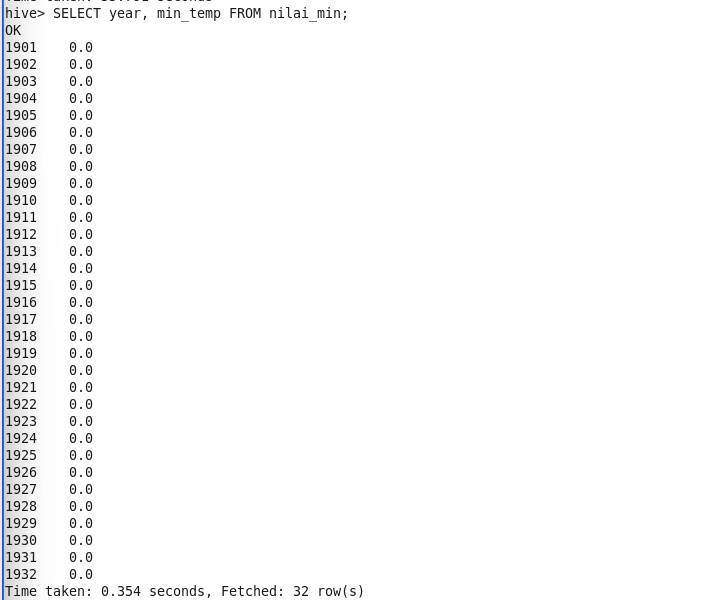
* year INT,
* min\_temp FLOAT
* );

hive> INSERT INTO nilai\_min

* SELECT tahun, MIN(suhu)
* FROM suhu
* GROUP BY tahun;

hive> SELECT year, min\_temp FROM nilai\_min;

Mendapatkan output nilai suhu minimum pertahunnya:



* Mencari nilai rata-rata pada suhu pertahunnya, dengan kode berikut ini:

hive> CREATE TABLE nilai\_avg (

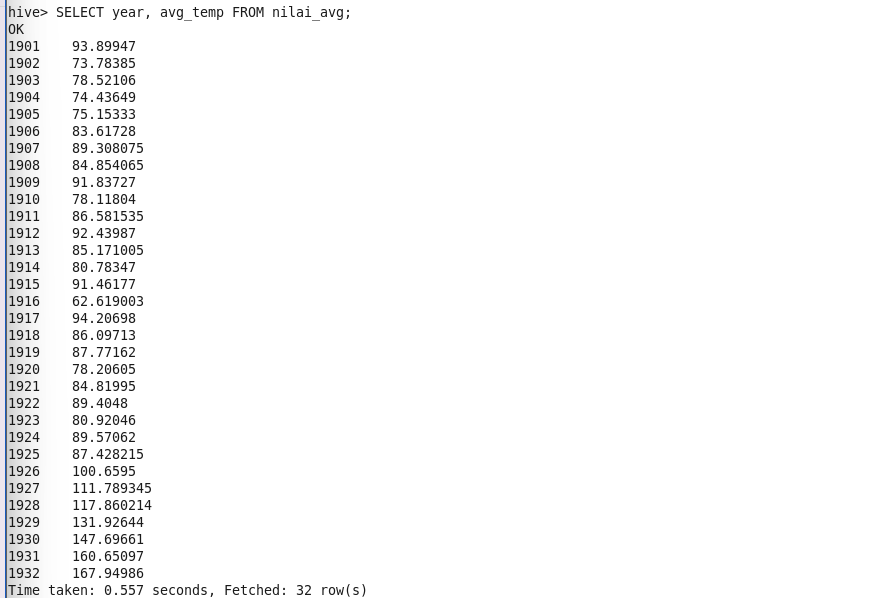
* year INT,
* avg\_temp FLOAT
* );

hive> INSERT INTO nilai\_avg

* SELECT tahun, avg(suhu)
* FROM suhu
* GROUP BY tahun;

hive> SELECT year, avg\_temp FROM nilai\_rata;

Mendapatkan output nilai rata-rata:



* Mencari nilai Varian pada suhu, dengan kode berikut:

hive> CREATE TABLE nilai\_var (

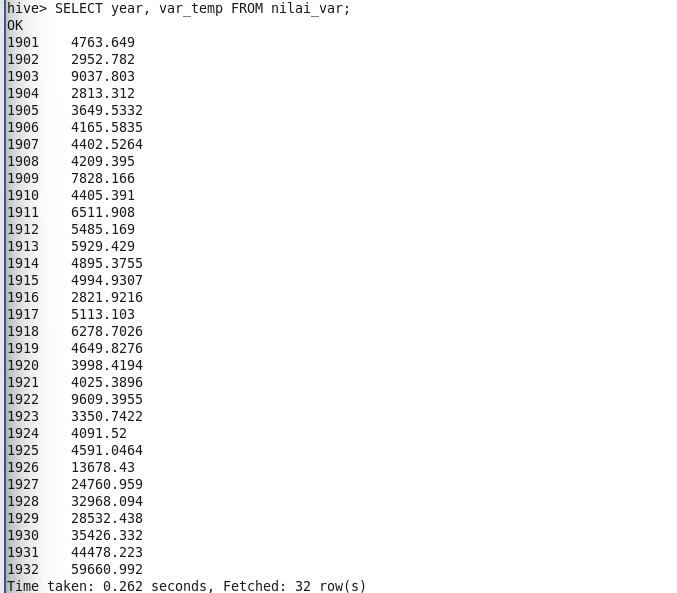
* year INT,
* var\_temp FLOAT
* );

hive> INSERT INTO nilai\_var

* SELECT tahun, VAR\_POP(suhu)
* FROM suhu
* GROUP BY tahun;

hive> SELECT year, var\_temp FROM nilai\_var;

Mendapatkan output nilai varian pada suhu:



* Mencari nilai standar deviasi pada suhu, dengan kode berikut:

hive> CREATE TABLE nilai\_std (

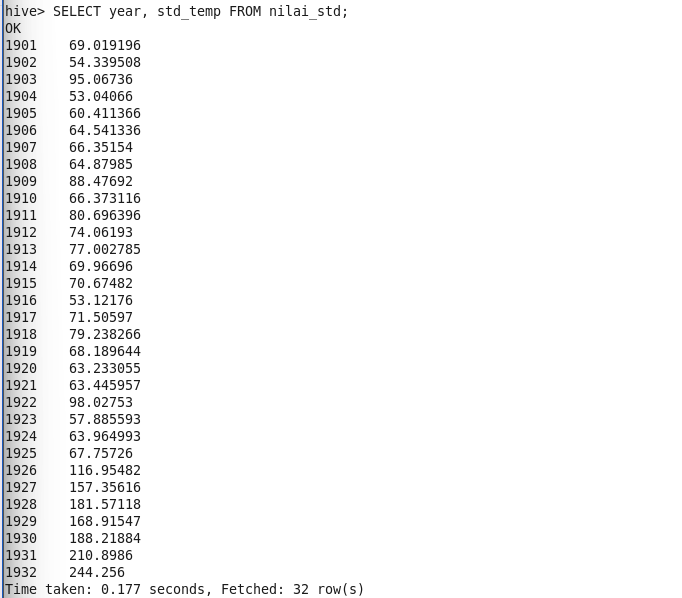
* year INT,
* std\_temp FLOAT
* );

hive> INSERT INTO nilai\_std

* SELECT tahun, STDDEV\_POP(suhu)
* FROM suhu
* GROUP BY tahun;

hive> SELECT year, std\_temp FROM nilai\_std;

Mendapatkan output:

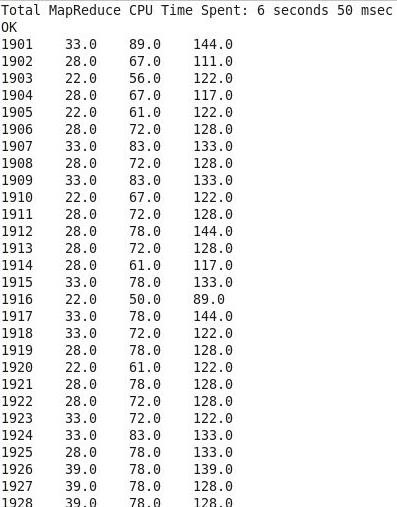


* Mencari nilai presentil suhu

hive> SELECT tahun,

* PERCENTILE(suhu, 0.25) AS suhu\_p25,
* PERCENTILE(suhu, 0.50) AS suhu\_p50,
* PERCENTILE(suhu, 0.75) AS suhu\_p75
* FROM suhutemp
* GROUP BY tahun;

Mendapatkan nilai presentil suhu:



1. Perubahan rata – rata suhu diantara 2 tahun, misalnya tahun 1912-1913

* Melakukan koding seperti dibawah ini:

hive> SELECT ((avg\_suhu\_tahun\_2 – avg\_suhu\_tahun\_1) / avg\_suhu\_tahun\_1) \* 100 AS presentase\_perubahan\_suhu

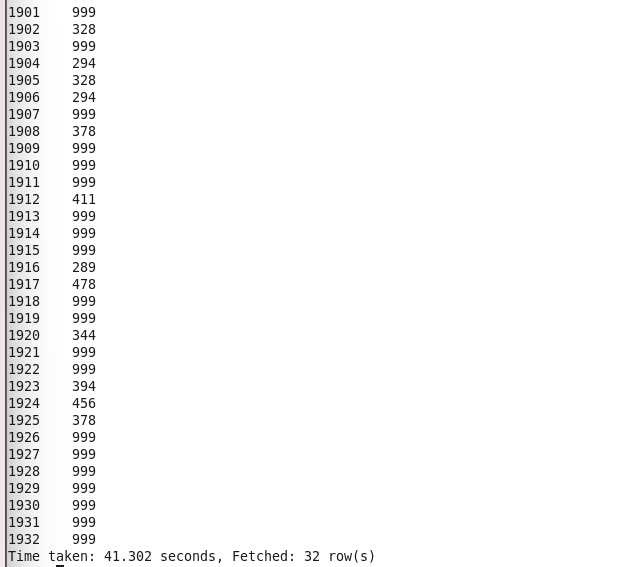
* FROM
* (SELECT AVG(suhu) AS avg\_suhu\_tahun\_1
* FROM suhu
* WHERE tahun = 1912) t1,
* (SELECT AVG(suhu) AS avg\_suhu\_tahun\_2
* FROM suhu
* WHERE tahun = 1913) t2;

Mendapatkan output perubahan rata-rata suhu diantara 2 tahun pada tahun 1912-1913:



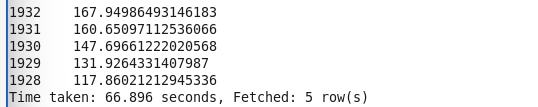
1. Membuat 3 pertanyaan analisi berdasarkan dataset NOAA
2. Berapa selisih suhu maksimum dan minimum pada pertahunnya?

Jawab: Selisih suhu maksimum dan minimum pada pertahunnya sangatlah bervariasi jika ingin melihat, bisa dilihat pada gambar dibawah ini:



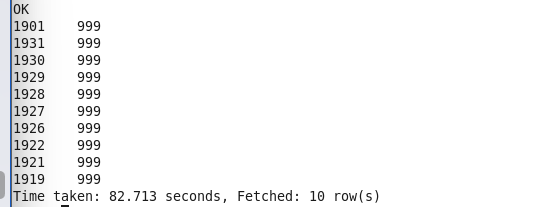
1. Sebutkan 5 saja tahun yang memiliki rata-rata suhu tertinggi?

Jawab: Tahun-tahun yang memiliki rata-rata suhu tertinggi ialah pada tahun 1932 yang memiliki rata-rata suhu , kemudian rata-rata suhu tertinggi kedua yakni pada tahun 1931 dengan rata-rata , selanjutanya rata-rata suhu tertinggi ketiga takni pada tahun 1930 dengan rata-rata , setelah itu disusul oleh tahun 1929 dengan rata-rata suhu selanjutnya yang rata-rata suhu yang kelima terjadi pada tahun 1928 dengan rata-rata, bisa dilihat pada gambar dibawah ini:



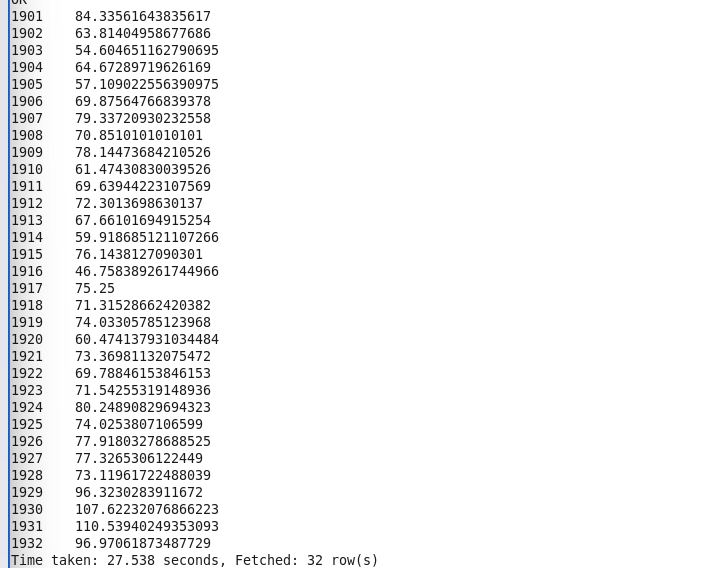
1. Bagaiman jika selisih suhu maksimum dan minimum terbesar ditampilkan dalam kurun waktu 10 tahun?

Jawab: Selisih suhu maksimum dan minimum terbesar dalam kurun waktu 10 tahun terjadi pada tahun bisa dilihat pada gambar berikut ini:



1. Berapa nilai median pada suhu dalam pertahunnya?

Jawab: berikut gambar dibawah ini merupakan nilai-nilai median pada suhu dalam pertahunnya.

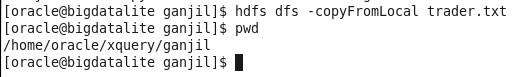
****

1. **Experimen 1 XQuery sederhana (integrating files)**

Melakukan analisis data xquery dengan 2 file dummy(Data Saham(ganjil) & Data Sembako(genap)) dengan beberapa metode:

[oracle@bigdatalite ganjil]$ hdfs dfs -copyFromLocal trader.txt

Kemudian pwd:



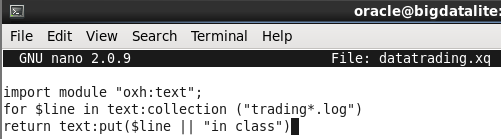
Selanjutnya membuat file trader.xq dengan kode berikut:

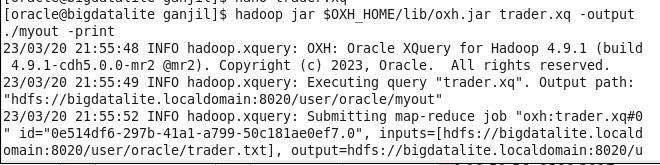
[oracle@bigdatalite ganjil]$ hadoop jar

$OXH\_HOME/lib/oxh.jar datatrader.xq -output

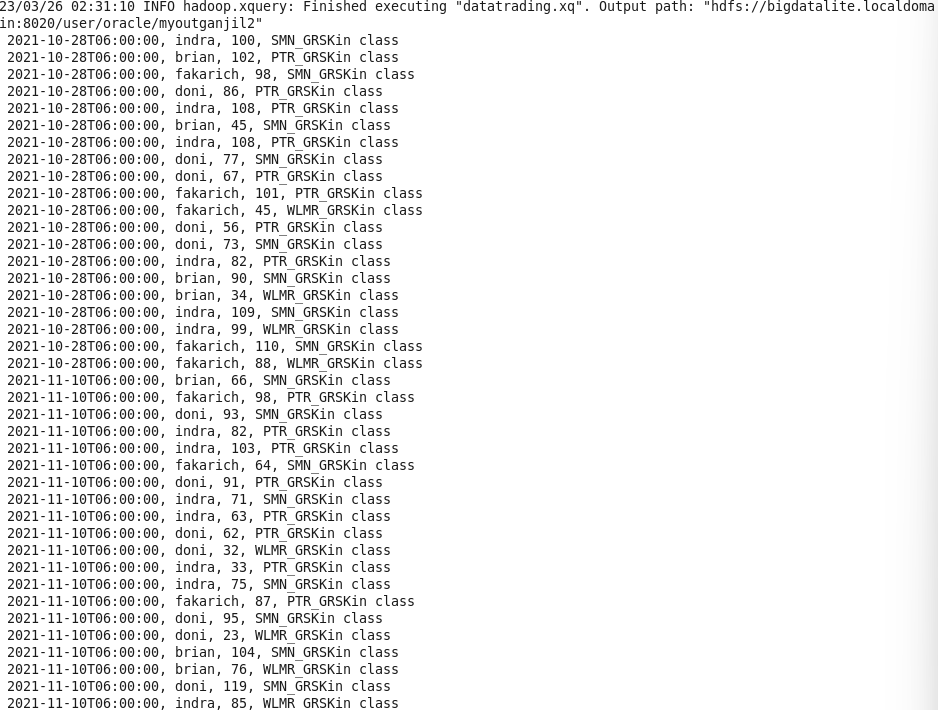
./myoutganjil1 -print

Mendapatkan output:



Lalu trader.xq dapat diupload ke hadoop:

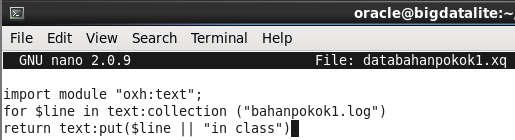
Kemudian diupload dengan cara berikut:



1. **Experimen 2 XQuery (basic filtering)**

Membuat file lognya terlebih dahulu dengan folder genap:

[oracle@bigdatalite genap]$ sudo nano databahanpokok1.xq



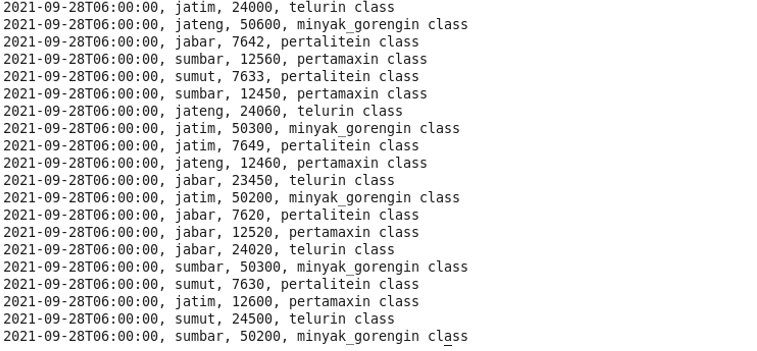
Kemudian menampilkan bahan pokok

[oracle@bigdatalite genap]$ hadoop jar

$OXH\_HOME/lib/oxh.jar databahanpokok1.xq -output .

-print

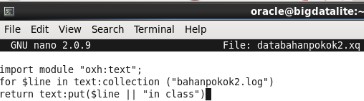
Mendapatkan output:



Selanjutnya membuat bahan pokok kedua dengan:

[oracle@bigdatalite genap]$ sudo nano databahanpokok2.xq

Mendapatkan output:



Selanjutnya melakukan filter untuk kedua bahan tersebut dengan

[oracle@bigdatalite genap]$ sudo nano basicfiltergenap.xq

1. **Experimen 3 (Group by and aggregation)**

Melakukan Group by and aggregation (data bahan pokok tiap hari atau per tanggal)

Membuat file xq pada folder genap

[oracle@bigdatalite genap]$ sudo nano groupbyaggregationgenap.xq

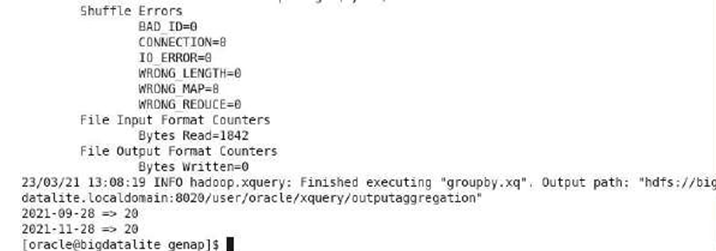
Kemudian jalankan group by.xq dengan jar

[oracle@bigdatalite genap]$ hadoop jar

$OXH\_HOME/lib/oxh.jar groupbyaggregationgenap.xq -output

./

Mendapatkan output:



1. **Experimen 4 (Inner joins)**

Buat file dengan kode berikut ini:

[oracle@bigdatalite genap]$ sudo nano innerjoingenap.xq

Kemudian jalankan dengan inner joins

[oracle@bigdatalite genap]$ hadoop jar

$OXH\_HOME/lib/oxh.jar innerjoingenap.xq -output.



1. **Experimen 5 (Left Outer Joins)**

Membuat file xq dengan kode berikut ini:

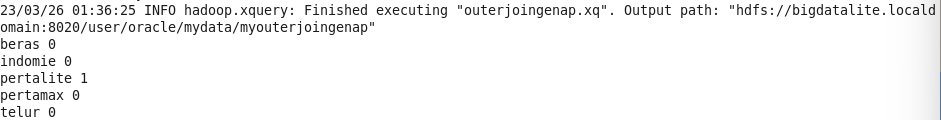
[oracle@bigdatalite genap]$ sudo nano outerjoingenap.xq

Kemudian jalankan dengan left outer joins:

[oracle@bigdatalite genap]$ hadoop jar

$OXH\_HOME/lib/oxh.jar outerjoingenap.xq -output.

Mendapatkan output:



1. **Experimen 6 (Semi Joins)**

Membuat file xq dengan kode berikut ini:

[oracle@bigdatalite genap]$ sudo nano semijoinsgenap.xq

Selanjutnya jalankan file dengan semi joins

[oracle@bigdatalite genap]$ hadoop jar

$OXH\_HOME/lib/oxh.jar semijoinsgenap.xq -output.



1. **Experimen 7 (Multiple Output)**

Membuat file xq dengan kode berikut ini:

[oracle@bigdatalite genap]$ sudo nano multipleoutputs.xq

Kemudian jalankan file dengan multiple output:

[oracle@bigdatalite genap]$ hadoop jar

$OXH\_HOME/lib/oxh.jar multipleoutputs.xq -output.

Mendapatkan output:



1. **Experimen 8 (Accessing auxiliary input data)**

Membuat file xq dengan kode berikut ini:

[oracle@bigdatalite genap]$ sudo nano accessingauxgenap.xq

Kemudian jalankan file dengan accessing auxiliary

[oracle@bigdatalite genap]$ hadoop jar

$OXH\_HOME/lib/oxh.jar -files cust.txt accessingauxgenap.xq -output. -print

Mendapatkan output:



1. **Experimen 9 (Calling a custom java function from XQuery)**

[oracle@bigdatalite RTM4]$ sudo nano accessingauxgenap.xq

[oracle@bigdatalite RTM4]$ sudo nano accessingauxgenap.xq

Membuat file xq dengan kode berikut:

[oracle@bigdatalite genap]$ sudo nano javafunctiongenap.xq

Kemudian lakukan pengkodean berikut:

[oracle@bigdatalite genap]$ hadoop jar

$OXH\_HOME/lib/oxh.jar javafunctiongenap.xq -output .

Mendapatkan output:



1. **Experimen 10**

**(Using user-defined XQuery library modules and XML schemas   
(membuat modul sendiri))**

Membuat file xq dengan kode berikut ini:

[oracle@bigdatalite mytools]$ sudo nano mytoolsgenap.xq

Kemudian jalankan file dengan kode berikut:

[oracle@bigdatalite mytools]$ hadoop jar

$OXH\_HOME/lib/oxh.jar -files mytoolsgenap.xq mytools2genap.xq -output .-print

Mendapatkan output:

