**SQL**

I did create a nytimes database and two tables in Hive - Cloudera VM. The scripts and results from HUE below:

* Find the solution in file named “DDL.sql”

Table “orders”

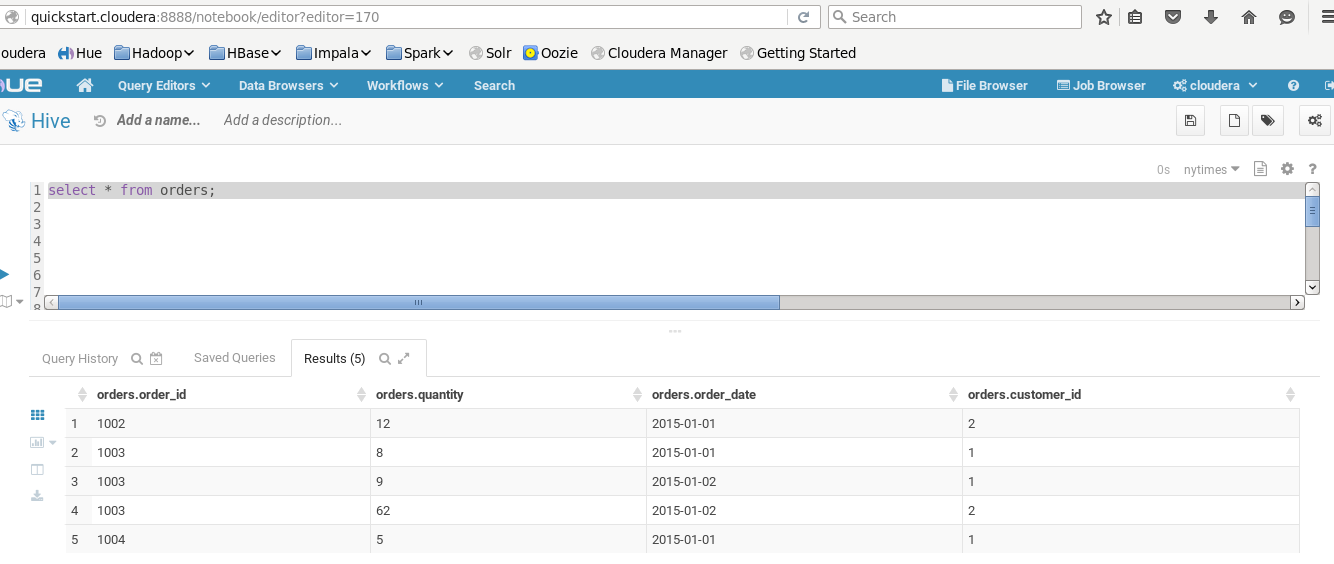
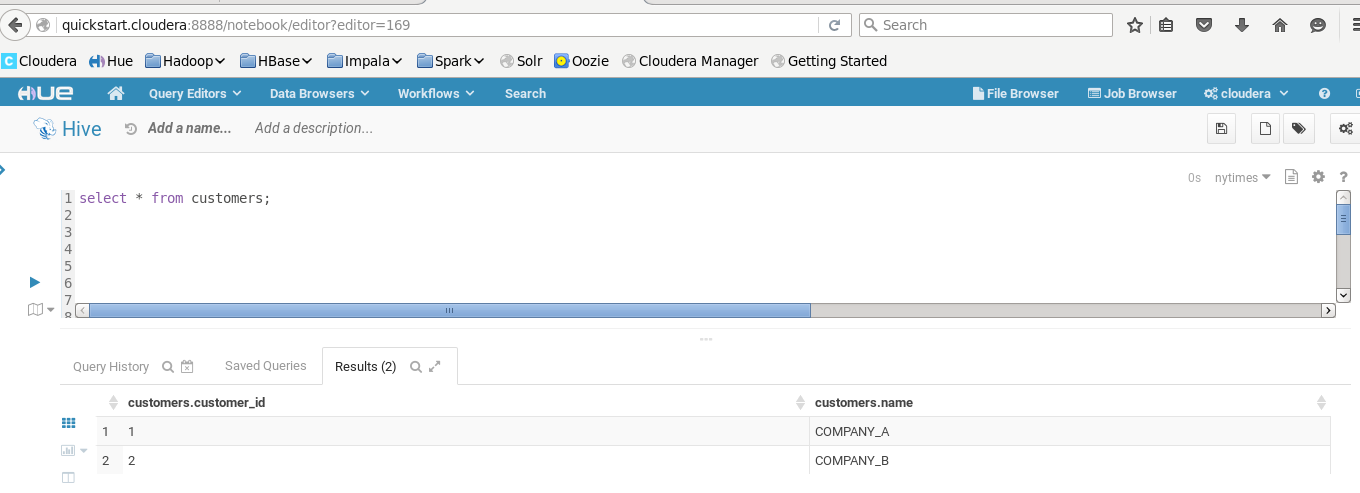


Table “customers”

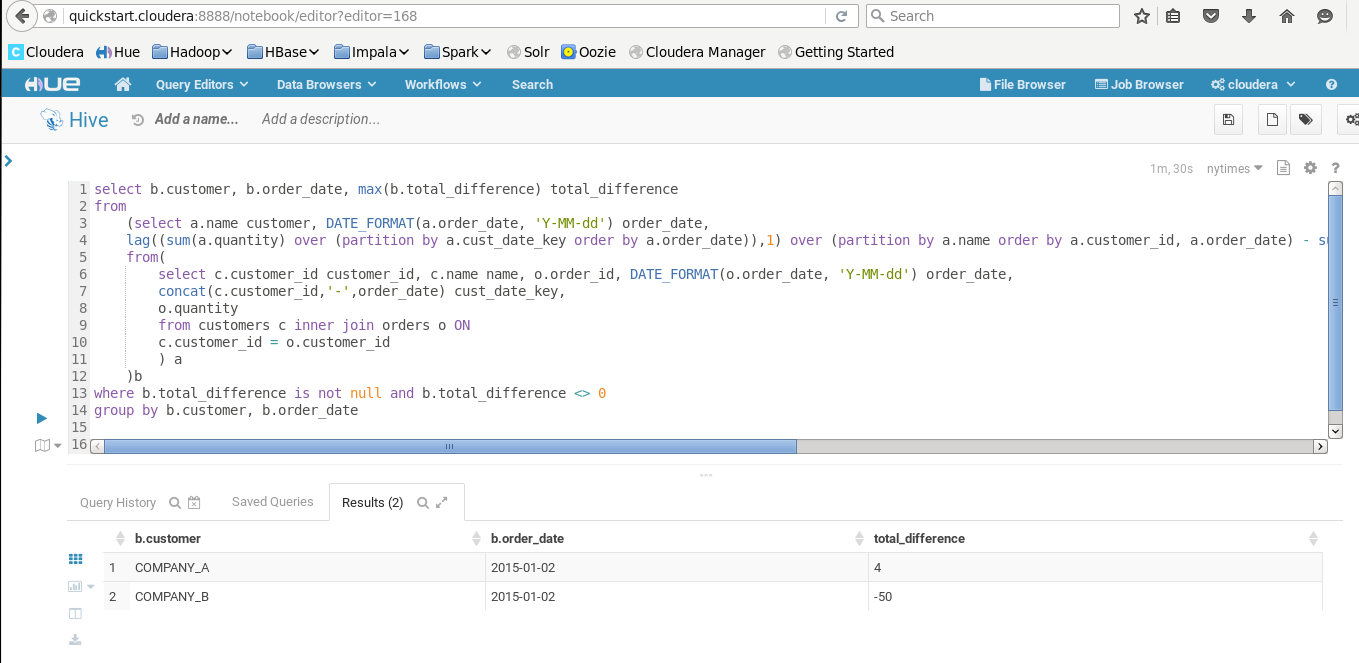


Solution of SQL Exercise 1

* Find the solution in file named “SQL\_EX1.sql”

Solution of SQL Exercise 2

* Find the solution in file named “SQL\_EX2.sql”



**Nearest Stars**

Exercise 1

* Find the solution in file named “findstars.py”

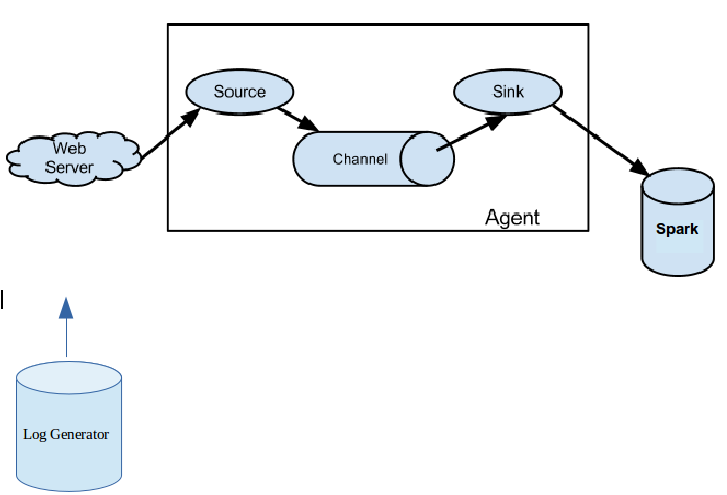
Exercise 2

* Find the solution in file named “findstars\_service.py”
* In order to create a solution that response quickly to a request for distance I would use Spark Machine Learning Library to process the job in parallel and using in-memory benefits.

**Solution Systems Design**

The solution has below components:

* Log generator app in python that will add the logs into the telnet localhost port 44445
  + File named “Streaming/gen\_logs.zip”
* Flume configuration that receives the logs and pass those to spark through a channel
  + File named “Streaming/netcatflumeModif.conf”
* Data pipeline in Spark Scala that receives the logs from flume, calculates statistics and store the results in HDFS.
  + File named “Streaming/testStreamFlume.scala”



After you present the design,comment​ ​ on​ ​ where​ ​ you​ ​ think​ ​ it​ ​ would​ ​ break​ ​ as​ ​ we​ ​ increase

utilization of the system​ by 10x,​ ​ 100x,​ ​ 1000x.​ ​ How​ ​ would​ ​ you​ ​ need​ ​ to​ ​ change​ ​ it​ ​ to​ ​ adapt​ ​ to​ ​ that

failure​ ​ mode​ ​ ?

* Configure the batch sizes equal to the sum of batches size of the streams. For example if you have a flume agent running a source with 5 upstream agend sending events and each one has a bath size of 10, you should start that downstream agent with a batch size of 50.
* If your bath size goes beyond 10,000 you should add a new Sink instead. This will increase the parallelism.

**Systems**

Exercise 1



Exercise 2

* Use saveAsHadoopFile and compression codec in Spark to upload from local system to S3

Exercise 3

* I would use EMR in the same region of the S3 bucket in order to ingest the files, check errors and store the new files back to S3.