Data Ingestion (Task 2 & 3)

Data Ingestion from RDS to HBase Table using Sqoop (Task 2)

It is assumed that this Task is done in continuation of Task 1, and common steps already done during Task 1, are already performed. Few of common steps are:

- Dataset Files are already downloaded through 'wget' to folder /root/tripdata/.
- RDS – MySQL database has been created and table is populated as mentioned in Task 1. Database service is up and running and connectivity between EMR cluster and RDS is configured.
- Shell variables DNS_EMR and DNS_RDS are already set to desired URLs.

Step 1. Switch to root user

HBase shell require root privileges to perform its actions. Therefore, before using hbase shell, switch to 'root' user using sudo command.

```
sudo -i
```

Screenshot of switching as 'root' user

Step 2. Start HBase shell

For executing any command on HBase database, start HBase shell.

```
hbase shell
```

Screenshot of starting HBase

```
[root@ip-172-31-24-56 tripdata]# hbase shell
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/lib/hadoop/lib/slf4j-reload4j-1.7.36.jar!
/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/lib/hbase/lib/client-facing-thirdparty/sl
f4j-reload4j-1.7.33.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.slf4j.impl.Reload4jLoggerFactory]
HBase Shell
Use "help" to get list of supported commands.
Use "exit" to quit this interactive shell.
For Reference, please visit: http://hbase.apache.org/2.0/book.html#shell
Version 2.4.15-amzn-0.1, rUnknown, Fri Jun 23 16:31:13 UTC 2023
Took 0.0024 seconds
hbase:001:0>
```

Step 3. Create HBase Table

The yellow tripdata contains information related to trips that are done on yellow taxi for months January to June for year 2017. Dataset only contains information for single category 'trips'. Therefore, creating HBase table using single column family 'trips'. Following commands are used:

- create – for creating HBase table. Table name and column family name are provided as input.
- list – it is used to list all the tables in HBase database. Here, it is used to validate table has created in HBase.
- describe – to describe a table in HBase, it shows the table information including all column families associated with it. Table name is provided as input.

```
create 'yellow_tripdata_full', 'trips'
list
describe 'yellow_tripdata_full'
```

```
quit()
```

Screenshot of creating HBase table

```
hbase:001:0> create 'yellow_tripdata_full', 'trips'
Created table yellow_tripdata_full
Took 2.7342 seconds
=> Hbase::Table - yellow_tripdata_full
hbase:002:0> list
TABLE
yellow_tripdata_full
1 row(s)
Took 0.0338 seconds
=> ["yellow_tripdata_full"]
hbase:003:0> describe 'yellow_tripdata_full'
Table yellow_tripdata_full is ENABLED
yellow_tripdata_full
COLUMN FAMILIES DESCRIPTION
{NAME => 'trips', BLOOMFILTER => 'ROW', IN_MEMORY => 'false', VERSIONS => '1', K
EEP_DELETED_CELLS => 'FALSE', DATA_BLOCK_ENCODING => 'NONE', COMPRESSION => 'NON
E', TTL => 'FOREVER', MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '6
5536', REPLICATION_SCOPE => '0'}

1 row(s)
Quota is disabled
Took 0.1694 seconds
hbase:004:0> quit()
[root@ip-172-31-24-56 tripdata]#
```

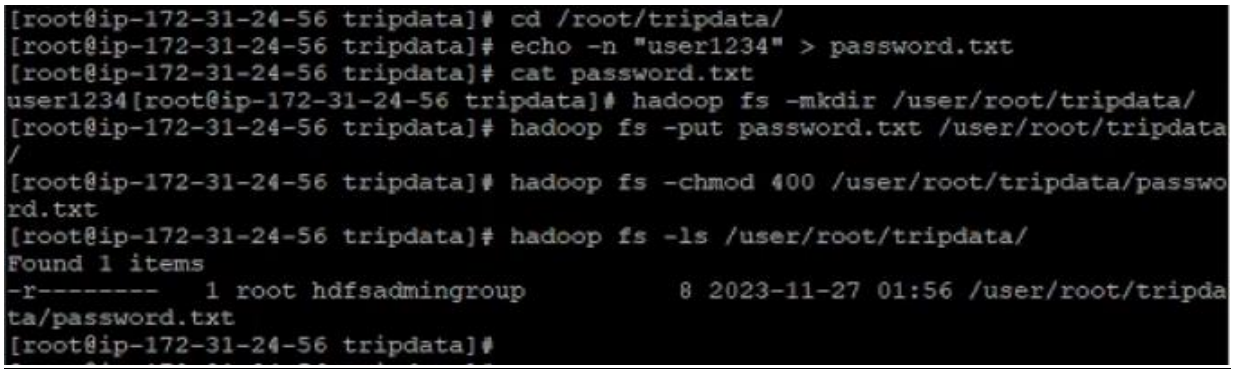
Step 4. Store password in file for security

Passwords are usually not passed as plain text in commands, to avoid their exposure to security threat and tracking commands containing password in case of change in password by administrator. Therefore, saving password in separate file on Hadoop, and later access directly from file while using Sqoop command. Following steps are performed:

- Creating password file – using echo command along with redirection operator to create 'password.txt'.
- Verifying password – using cat command to validate password file.
- Transfer password file to Hadoop file system (HDFS) – using Hadoop fs -put to transfer file from local file system to HDFS.
- Changing rights – use chmod option to grants only the 'root' (current user) of the file read permission while restricting everyone else entirely.
- Verifying file creation – using Hadoop fs -ls to validate successful creation of password file.

```
cd /root/tripdata/
echo -n "user1234" > password.txt
cat password.txt
hadoop fs -mkdir /user/root/tripdata/
hadoop fs -put password.txt /user/root/tripdata/
hadoop fs -chmod 400 /user/root/tripdata/password.txt
hadoop fs -ls /user/root/tripdata/
```

Screenshot of creating password file



Step 5. Installing MySQL connector for Sqoop

Sqoop is a tool that help in efficiently transferring bulk data between Relational Databases and Hadoop/HBase. Following properties are set while using Sqoop utility to transfer data from RDS to HBase:

```
wget https://de-mysql-connector.s3.amazonaws.com/mysql-connector-java-
8.0.25.tar.gz
tar -xvf mysql-connector-java-8.0.25.tar.gz
cd mysql-connector-java-8.0.25/
sudo cp mysql-connector-java-8.0.25.jar /usr/lib/sqoop/lib/
```

Screenshot of installing MySQL connector

```
[root@ip-172-31-24-56 tripdata]# wget https://de-mysql-connector.s3.amazonaws.co
m/mysql-connector-java-8.0.25.tar.gz
--2023-11-27 01:56:20--  https://de-mysql-connector.s3.amazonaws.com/mysql-conne
ctor-java-8.0.25.tar.gz
Resolving de-mysql-connector.s3.amazonaws.com (de-mysql-connector.s3.amazonaws.c
om)... 52.217.228.177, 52.217.229.49, 54.231.135.105, ...
Connecting to de-mysql-connector.s3.amazonaws.com (de-mysql-connector.s3.amazona
ws.com)|52.217.228.177|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 4079310 (3.9M) [application/x-gzip]
Saving to: 'mysql-connector-java-8.0.25.tar.gz'

100%[===================================>] 4,079,310   --.-K/s   in 0.1s

2023-11-27 01:56:20 (40.4 MB/s) - 'mysql-connector-java-8.0.25.tar.gz' saved [40
79310/4079310]

[root@ip-172-31-24-56 tripdata]# tar -xvf mysql-connector-java-8.0.25.tar.gz
mysql-connector-java-8.0.25/
mysql-connector-java-8.0.25/src/
mysql-connector-java-8.0.25/src/build/
mysql-connector-java-8.0.25/src/build/java/
mysql-connector-java-8.0.25/src/build/java/documentation/
mysql-connector-java-8.0.25/src/build/java/instrumentation/
mysql-connector-java-8.0.25/src/build/misc/
```

```
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/package-info.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/InternalXBaseTest
Case.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/MysqlxSessionTest
.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/XProtocolAsyncTes
t.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/XProtocolAuthTest
.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/XProtocolTest.jav
a
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/package-info.java
[root@ip-172-31-24-56 tripdata]# cd mysql-connector-java-8.0.25/
[root@ip-172-31-24-56 mysql-connector-java-8.0.25]# sudo cp mysql-connector-java
-8.0.25.jar /usr/lib/sqoop/lib/
[root@ip-172-31-24-56 mysql-connector-java-8.0.25]# cd ..
[root@ip-172-31-24-56 tripdata]#
```

Step 6. Setting permission for connection to MySQL (This step not required for AWS RDS)

Before executing Sqoop command, one need to ensure that permission to connect to MySQL database is provided. In MySQL AWS RDS this permission is default for DB user. Otherwise, set it using following commands:

Login to RDS instance of MySQL using the credentials used to create the database service:

Username : admin
Password : user1234

```
mysql -h $DNS_RDS -P 3306 -u admin -p
GRANT ALL PRIVILEGES ON *.* TO 'admin'@'%' identified by 'user1234' WITH
GRANT OPTION;
GRANT ALL PRIVILEGES ON tripdata.* TO 'admin'@'%';
flush privileges;
quit;
```

Step 7. Loading dataset from RDS to HBase using Sqoop

Sqoop is a tool that help in efficiently transferring bulk data between Relational Databases and Hadoop/HBase. Following properties are set while using Sqoop utility to transfer data from RDS to HBase:

- --connect : to provide connection string to MySQL (RDS instance).
- --table : table name in relational database (MySQL).
- --username : username of relational database (MySQL).
- --password-file : password file in hadoop file system (HDFS) containing password of relational database (MySQL).
- --null-string : to indicate how to handle 'null' in string data.
- --null-non-string : to indicate how to handle 'null' in non-string data.
- --hbase-table : to indicate name of table in HBase database
- --column-family : to indicate name of column-family in table of HBase database
- --hbase-row-key : to indicate primary-key (unique composite key) in dataset to uniquely identify each record in dataset.
- —split-by : to indicate the key by which records can be split across multiple mappers.
- -m : number of mappers to be used for Sqoop command.

In Sqoop import from MySQL to HBase table, we are using Primary Key which is on String Type (single string which is combination of Source, Year, Month and Row Index), therefore we need to set parameter org.apache.sqoop.splitter.allow_text_splitter to true.

a. Using Sqoop --import

Following command is used for directly importing data from RDS to hbase:

```
sqoop import -D org.apache.sqoop.splitter.allow_text_splitter=true \
--connect jdbc:mysql://$DNS_RDS:3306/tripdata \
--username admin --password-file /user/root/tripdata/password.txt \
--table yellow_tripdata \
--split-by RECORD_ID \
--null-string '\\N' --null-non-string '\\N' \
--hbase-table yellow_tripdata_full \
--column-family trips \
--hbase-row-key RECORD_ID \
-m 20
```

b. Using Sqoop Job

Following command is used to create sqoop job that can be used to import data from RDS to hbase:

```
sqoop job –create tripdata_import \
-- import -D org.apache.sqoop.splitter.allow_text_splitter=true \
--connect jdbc:mysql://$DNS_RDS:3306/tripdata \
--username admin --password-file /user/root/tripdata/password.txt \
--table yellow_tripdata \
--split-by RECORD_ID \
--null-string '\\N' --null-non-string '\\N' \
--hbase-table yellow_tripdata_full \
--column-family trips \
--hbase-row-key RECORD_ID \
-m 20
```

Following command is used to verify the sqoop job created through listing all the sqoop jobs available.

```
sqoop job --list
```

Following command is used to run the sqoop job.

```
sqoop job --exec tripdata_import
```

Sqoop job is useful in scenarios where sqoop command is to run periodically. As for assignment purpose, we need to perform sqoop import job only once, therefore using direct sqoop - -import to keep it simple.

Screenshot of sqoop command

```
[root@ip-172-31-24-56 tripdata]# sqoop import -D org.apache.sqoop.splitter.allow
_text_splitter=true \
> --connect jdbc:mysql://$DNS_RDS:3306/tripdata \
> --username admin --password-file /user/root/tripdata/password.txt \
> --table yellow_tripdata \
> --split-by RECORD_ID \
> --null-string '\\N' --null-non-string '\\N' \
> --hbase-table yellow_tripdata_full \
> --column-family trips \
> --hbase-row-key RECORD_ID \
> -m 20
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/lib/hadoop/lib/slf4j-reload4j-1.7.36.jar!
/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/lib/hive/lib/log4j-slf4j-impl-2.17.1.jar!
/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/lib/hbase/lib/client-facing-thirdparty/sl
f4j-reload4j-1.7.33.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.slf4j.impl.Reload4jLoggerFactory]
2023-11-27 01:57:31,880 INFO sqoop.Sqoop: Running Sqoop version: 1.4.7
2023-11-27 01:57:33,332 INFO manager.MySQLManager: Preparing to use a MySQL stre
aming resultset.
2023-11-27 01:57:33,332 INFO tool.CodeGenTool: Beginning code generation
Loading class `com.mysql.jdbc.Driver'. This is deprecated. The new driver class
is `com.mysql.cj.jdbc.Driver'. The driver is automatically registered via the SP
I and manual loading of the driver class is generally unnecessary.
2023-11-27 01:57:34,049 INFO manager.SqlManager: Executing SQL statement: SELECT
 t.* FROM `yellow_tripdata` AS t LIMIT 1
2023-11-27 01:57:34,224 INFO manager.SqlManager: Executing SQL statement: SELECT
 t.* FROM `yellow_tripdata` AS t LIMIT 1
2023-11-27 01:57:34,249 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/
lib/hadoop-mapreduce
2023-11-27 01:57:38,952 INFO orm.CompilationManager: Writing jar file: /tmp/sqoo
p-root/compile/735c5e84d1925c76cf6e8784a4dc1393/yellow_tripdata.jar
2023-11-27 01:57:38,976 WARN manager.MySQLManager: It looks like you are importi
ng from mysql.
2023-11-27 01:57:38,976 WARN manager.MySQLManager: This transfer can be faster!
Use the --direct
2023-11-27 01:57:38,976 WARN manager.MySQLManager: option to exercise a MySQL-sp
ecific fast path.
2023-11-27 01:57:38,976 INFO manager.MySQLManager: Setting zero DATETIME behavio
r to convertToNull (mysql)
2023-11-27 01:57:39,376 INFO mapreduce.ImportJobBase: Beginning import of yellow
_tripdata
2023-11-27 01:57:39,383 INFO Configuration.deprecation: mapred.jar is deprecated
```

```
. Instead, use mapreduce.job.jar
2023-11-27 01:57:39,405 INFO Configuration.deprecation: mapred.map.tasks is depr
ecated. Instead, use mapreduce.job.maps
2023-11-27 01:57:43,304 WARN mapreduce.TableMapReduceUtil: The addDependencyJars
(Configuration, Class<?>...) method has been deprecated since it is easy to use
incorrectly. Most users should rely on addDependencyJars(Job) instead. See HBASE
-8386 for more details.
2023-11-27 01:57:43,580 INFO client.DefaultNoHARMFailoverProxyProvider: Connecti
ng to ResourceManager at ip-172-31-24-56.ec2.internal/172.31.24.56:8032
2023-11-27 01:57:43,755 INFO client.AHSProxy: Connecting to Application History
server at ip-172-31-24-56.ec2.internal/172.31.24.56:10200
2023-11-27 01:57:44,524 INFO mapreduce.JobResourceUploader: Disabling Erasure Co
ding for path: /tmp/hadoop-yarn/staging/root/.staging/job_1701048046021_0001
2023-11-27 01:57:59,299 INFO db.DBInputFormat: Using read commited transaction i
solation
2023-11-27 01:57:59,300 INFO db.DataDrivenDBInputFormat: BoundingValsQuery: SELE
CT MIN(`RECORD_ID`), MAX(`RECORD_ID`) FROM `yellow_tripdata`
2023-11-27 01:57:59,319 WARN db.TextSplitter: Generating splits for a textual in
dex column.
2023-11-27 01:57:59,319 WARN db.TextSplitter: If your database sorts in a case-i
nsensitive order, this may result in a partial import or duplicate records.
2023-11-27 01:57:59,319 WARN db.TextSplitter: You are strongly encouraged to cho
ose an integral split column.
2023-11-27 01:57:59,411 INFO mapreduce.JobSubmitter: number of splits:22
2023-11-27 01:57:59,670 INFO Configuration.deprecation: yarn.resourcemanager.sys
tem-metrics-publisher.enabled is deprecated. Instead, use yarn.system-metrics-pu
blisher.enabled
2023-11-27 01:57:59,837 INFO mapreduce.JobSubmitter: Submitting tokens for job:
job_1701048046021_0001
2023-11-27 01:57:59,837 INFO mapreduce.JobSubmitter: Executing with tokens: []
2023-11-27 01:58:00,096 INFO conf.Configuration: resource-types.xml not found
2023-11-27 01:58:00,097 INFO resource.ResourceUtils: Unable to find 'resource-ty
pes.xml'.
2023-11-27 01:58:00,765 INFO impl.YarnClientImpl: Submitted application applicat
ion_1701048046021_0001
2023-11-27 01:58:00,834 INFO mapreduce.Job: The url to track the job: http://ip-
172-31-24-56.ec2.internal:20888/proxy/application_1701048046021_0001/
2023-11-27 01:58:00,835 INFO mapreduce.Job: Running job: job_1701048046021_0001
2023-11-27 01:58:10,954 INFO mapreduce.Job: Job job_1701048046021_0001 running i
n uber mode : false
2023-11-27 01:58:10,956 INFO mapreduce.Job:  map 0% reduce 0%
2023-11-27 01:58:21,066 INFO mapreduce.Job:  map 9% reduce 0%
2023-11-27 01:58:29,136 INFO mapreduce.Job:  map 27% reduce 0%
2023-11-27 01:58:30,141 INFO mapreduce.Job:  map 50% reduce 0%
2023-11-27 01:58:38,182 INFO mapreduce.Job:  map 59% reduce 0%
2023-11-27 01:58:40,192 INFO mapreduce.Job:  map 64% reduce 0%
2023-11-27 01:58:41,197 INFO mapreduce.Job:  map 68% reduce 0%
2023-11-27 01:58:46,218 INFO mapreduce.Job:  map 73% reduce 0%
```

```
2023-11-27 01:58:10,956 INFO mapreduce.Job:  map 0% reduce 0%
2023-11-27 01:58:21,066 INFO mapreduce.Job:  map 9% reduce 0%
2023-11-27 01:58:29,136 INFO mapreduce.Job:  map 27% reduce 0%
2023-11-27 01:58:30,141 INFO mapreduce.Job:  map 50% reduce 0%
2023-11-27 01:58:38,182 INFO mapreduce.Job:  map 59% reduce 0%
2023-11-27 01:58:40,192 INFO mapreduce.Job:  map 64% reduce 0%
2023-11-27 01:58:41,197 INFO mapreduce.Job:  map 68% reduce 0%
2023-11-27 01:58:46,218 INFO mapreduce.Job:  map 73% reduce 0%
2023-11-27 01:58:49,230 INFO mapreduce.Job:  map 77% reduce 0%
2023-11-27 01:58:50,235 INFO mapreduce.Job:  map 82% reduce 0%
2023-11-27 01:58:51,243 INFO mapreduce.Job:  map 91% reduce 0%
2023-11-27 04:29:51,047 INFO mapreduce.Job:  map 95% reduce 0%
2023-11-27 04:34:52,553 INFO mapreduce.Job:  map 100% reduce 0%
2023-11-27 04:34:52,556 INFO mapreduce.Job: Job job_1701048046021_0001 completed
 successfully
2023-11-27 04:34:52,668 INFO mapreduce.Job: Counters: 34
        File System Counters
                FILE: Number of bytes read=0
                FILE: Number of bytes written=7403478
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=3753
                HDFS: Number of bytes written=0
                HDFS: Number of read operations=22
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=0
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Killed map tasks=2
                Launched map tasks=24
                Other local map tasks=24
                Total time spent by all maps in occupied slots (ms)=899432016
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=18738167
                Total vcore-milliseconds taken by all map tasks=18738167
                Total megabyte-milliseconds taken by all map tasks=28781824512
        Map-Reduce Framework
                Map input records=18880595
                Map output records=18880595
                Input split bytes=3753
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=35080
                CPU time spent (ms)=1835550
```

```
        Map-Reduce Framework
                Map input records=18880595
                Map output records=18880595
                Input split bytes=3753
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=35080
                CPU time spent (ms)=1835550
                Physical memory (bytes) snapshot=8673615872
                Virtual memory (bytes) snapshot=68638040064
                Total committed heap usage (bytes)=7598505984
                Peak Map Physical memory (bytes)=687874048
                Peak Map Virtual memory (bytes)=3196575744
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=0
2023-11-27 04:34:52,672 INFO mapreduce.ImportJobBase: Transferred 0 bytes in 9,4
29.3634 seconds (0 bytes/sec)
2023-11-27 04:34:52,673 INFO mapreduce.ImportJobBase: Retrieved 18880595 records
.
[root@ip-172-31-24-56 tripdata]#
```

Step 8. Verification of Loaded dataset from RDS to HBase

Dataset loaded has been verified using 'count' command on HBase Table. As dataset is huge, therefore using interval size of 1 million rows while counting. Commands are:

```
hbase shell
count 'yellow_tripdata_full', INTERVAL => 1000000
```

Screenshot of data upload verification

```
[root@ip-172-31-24-56 tripdata]# hbase shell
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/lib/hadoop/lib/slf4j-reload4j-1.7.36.jar!
/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/lib/hbase/lib/client-facing-thirdparty/sl
f4j-reload4j-1.7.33.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.slf4j.impl.Reload4jLoggerFactory]
HBase Shell
Use "help" to get list of supported commands.
Use "exit" to quit this interactive shell.
For Reference, please visit: http://hbase.apache.org/2.0/book.html#shell
Version 2.4.15-amzn-0.1, rUnknown, Fri Jun 23 16:31:13 UTC 2023
Took 0.0041 seconds
hbase:001:0> count 'yellow_tripdata_full', INTERVAL => 1000000
Current count: 1000000, row: MYSQL-2017-01-0001000000
Current count: 2000000, row: MYSQL-2017-01-0002000000
Current count: 3000000, row: MYSQL-2017-01-0003000000
Current count: 4000000, row: MYSQL-2017-01-0004000000
Current count: 5000000, row: MYSQL-2017-01-0005000000
Current count: 6000000, row: MYSQL-2017-01-0006000000
Current count: 7000000, row: MYSQL-2017-01-0007000000
Current count: 8000000, row: MYSQL-2017-01-0008000000
Current count: 9000000, row: MYSQL-2017-01-0009000000
Current count: 10000000, row: MYSQL-2017-02-0010000000
Current count: 11000000, row: MYSQL-2017-02-0011000000
Current count: 12000000, row: MYSQL-2017-02-0012000000
Current count: 13000000, row: MYSQL-2017-02-0013000000
Current count: 14000000, row: MYSQL-2017-02-0014000000
Current count: 15000000, row: MYSQL-2017-02-0015000000
Current count: 16000000, row: MYSQL-2017-02-0016000000
Current count: 17000000, row: MYSQL-2017-02-0017000000
Current count: 18000000, row: MYSQL-2017-02-0018000000
18880595 row(s)
Took 910.1501 seconds
=> 18880595
hbase:002:0>
```

Step 9. Additional helpful commands for monitoring Sqoop process

Yarn is responsible for distributing and monitoring of MapReduce jobs. As Sqoop, execute import/export using map function, therefore execution of Sqoop command monitored with Yarn. Below command is helpful in monitoring the job while its executing:

```
yarn application -list
```

Screenshot of yarn command

```
[root@ip-172-31-24-56 ~]# yarn application -list
2023-11-27 04:35:22,134 INFO client.DefaultNoHARMFailoverProxyProvider: Connecti
ng to ResourceManager at ip-172-31-24-56.ec2.internal/172.31.24.56:8032
2023-11-27 04:35:22,504 INFO client.AHSProxy: Connecting to Application History
server at ip-172-31-24-56.ec2.internal/172.31.24.56:10200
Total number of applications (application-types: [], states: [SUBMITTED, ACCEPTE
D, RUNNING] and tags: []):0
                Application-Id      Application-Name        Application-Type
     User           Queue                   State             Final-State
      Progress                       Tracking-URL
[root@ip-172-31-24-56 ~]#
```

********** END OF TASK-2 **********

Data Ingestion from CSV file to HBase in Batch mode (Task 3)

It is assumed that this Task is done in continuation of Task 1 and Task 2, and common steps already done during Task 1 and 2, are already performed. Few of common steps are:

- Dataset Files are already downloaded through 'wget' to folder /root/tripdata/.
- HBase table is already created as desired.
- Python files are transferred to /home/Hadoop/tasks/ folder. And execution permission has been added.
- Shell variables DNS_EMR and DNS_RDS are already set to desired URLs.

Step 1. Setup "Happybase" API

Following steps are performed for setting-up "Happybase":

- **sudo -i:** switch to root user.
- **sudo yum install gcc:** install or update gcc on machine.
- **sudo yum install python3-devel:** to incorporates modules, exceptions, dynamic typing, very high level dynamic data types, and classes
- **pip install --use-feature=2020-resolver happybase:** to install happybase API.
- **pip install mrjob:** Although for data ingestion task mrjob is not required, it will be required to perform the tasks to be performed on dataset.
- **jps:** to check whether "ThriftServer" is running.
- **hbase thrift start:** If thrift server is not running use this command to start thrift server.

```
sudo -i
sudo yum install gcc
sudo yum install python3-devel
pip install --use-feature=2020-resolver happybase
pip install mrjob
jps
hbase thrift start
python -c "import happybase"
```

Screenshot of setting-up Happybase

```
[root@ip-172-31-24-227 tripdata]# sudo -i

EEEEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRRR
E::::::::::::::::::E M:::::::M         M:::::::M R::::::::::::::R
EE::::::EEEEEEEEE:::E M::::::::M       M::::::::M R:::::RRRRRR:::::R
  E::::E       EEEEE M:::::::::M     M:::::::::M RR::::R      R::::R
  E::::E             M::::::M:::M   M:::M::::::M   R:::R      R::::R
  E::::::EEEEEEEEEE   M:::::M M:::M M:::M M:::::M   R:::RRRRRR:::::R
  E::::::::::::::::E  M:::::M  M:::M:::M  M:::::M   R::::::::::::RR
  E::::::EEEEEEEEEE   M:::::M   M:::::M   M:::::M   R:::RRRRRR::::R
  E::::E             M:::::M    M:::M    M:::::M   R:::R      R::::R
  E::::E       EEEEE M:::::M     MMM     M:::::M   R:::R      R::::R
EE::::::EEEEEEEE:::E M:::::M             M:::::M   R:::R      R::::R
E::::::::::::::::::E M:::::M             M:::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR      RRRRRR

[root@ip-172-31-24-227 ~]# sudo yum install gcc
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                | 3.6 kB     00:00
14 packages excluded due to repository priority protections
Package gcc-7.3.1-17.amzn2.x86_64 already installed and latest version
Nothing to do
[root@ip-172-31-24-227 ~]# sudo yum install python3-devel
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
14 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
---> Package python3-devel.x86_64 0:3.7.16-1.amzn2.0.4 will be installed
--> Processing Dependency: python3-rpm-macros for package: python3-devel-3.7.16-
1.amzn2.0.4.x86_64
--> Processing Dependency: system-rpm-config for package: python3-devel-3.7.16-1
.amzn2.0.4.x86_64
--> Running transaction check
---> Package python3-rpm-macros.noarch 0:3-60.amzn2.0.1 will be installed
--> Processing Dependency: python-srpm-macros >= 3-38 for package: python3-rpm-m
acros-3-60.amzn2.0.1.noarch
--> Processing Dependency: python-rpm-macros for package: python3-rpm-macros-3-6
0.amzn2.0.1.noarch
---> Package system-rpm-config.noarch 0:9.1.0-76.amzn2.0.14 will be installed
--> Processing Dependency: dwz >= 0.4 for package: system-rpm-config-9.1.0-76.am
zn2.0.14.noarch
--> Processing Dependency: go-srpm-macros for package: system-rpm-config-9.1.0-7
6.amzn2.0.14.noarch
--> Processing Dependency: perl-srpm-macros for package: system-rpm-config-9.1.0
-76.amzn2.0.14.noarch
--> Running transaction check
---> Package dwz.x86_64 0:0.11-3.amzn2.0.3 will be installed
---> Package go-srpm-macros.noarch 0:3.0.15-23.amzn2.0.2 will be installed
```

```
---> Package go-srpm-macros.noarch 0:3.0.15-23.amzn2.0.2 will be installed
---> Package perl-srpm-macros.noarch 0:1-8.amzn2.0.1 will be installed
---> Package python-rpm-macros.noarch 0:3-60.amzn2.0.1 will be installed
---> Package python-srpm-macros.noarch 0:3-60.amzn2.0.1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                Arch       Version                  Repository      Size
================================================================================
Installing:
 python3-devel          x86_64     3.7.16-1.amzn2.0.4       amzn2-core     244 k
Installing for dependencies:
 dwz                    x86_64     0.11-3.amzn2.0.3         amzn2-core      98 k
 go-srpm-macros         noarch     3.0.15-23.amzn2.0.2      amzn2-core      23 k
 perl-srpm-macros       noarch     1-8.amzn2.0.1            amzn2-core     4.7 k
 python-rpm-macros      noarch     3-60.amzn2.0.1           amzn2-core      14 k
 python-srpm-macros     noarch     3-60.amzn2.0.1           amzn2-core      18 k
 python3-rpm-macros     noarch     3-60.amzn2.0.1           amzn2-core      12 k
 system-rpm-config      noarch     9.1.0-76.amzn2.0.14      amzn2-core      90 k

Transaction Summary
================================================================================
Install  1 Package (+7 Dependent packages)

Total download size: 505 k
Installed size: 1.2 M
Is this ok [y/d/N]: y
Downloading packages:
(1/8): go-srpm-macros-3.0.15-23.amzn2.0.2.noarch.rpm       |  23 kB   00:00
(2/8): dwz-0.11-3.amzn2.0.3.x86_64.rpm                     |  98 kB   00:00
(3/8): perl-srpm-macros-1-8.amzn2.0.1.noarch.rpm           | 4.7 kB   00:00
(4/8): python-srpm-macros-3-60.amzn2.0.1.noarch.rpm        |  18 kB   00:00
(5/8): python3-devel-3.7.16-1.amzn2.0.4.x86_64.rpm         | 244 kB   00:00
(6/8): python-rpm-macros-3-60.amzn2.0.1.noarch.rpm         |  14 kB   00:00
(7/8): python3-rpm-macros-3-60.amzn2.0.1.noarch.rpm        |  12 kB   00:00
(8/8): system-rpm-config-9.1.0-76.amzn2.0.14.noarch.rpm    |  90 kB   00:00
--------------------------------------------------------------------------------
Total                                              1.9 MB/s | 505 kB  00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : perl-srpm-macros-1-8.amzn2.0.1.noarch                        1/8
  Installing : dwz-0.11-3.amzn2.0.3.x86_64                                  2/8
  Installing : go-srpm-macros-3.0.15-23.amzn2.0.2.noarch                    3/8
  Installing : system-rpm-config-9.1.0-76.amzn2.0.14.noarch                 4/8
```

```
  Installing : system-rpm-config-9.1.0-76.amzn2.0.14.noarch                 4/8
  Installing : python-srpm-macros-3-60.amzn2.0.1.noarch                      5/8
  Installing : python-rpm-macros-3-60.amzn2.0.1.noarch                       6/8
  Installing : python3-rpm-macros-3-60.amzn2.0.1.noarch                      7/8
  Installing : python3-devel-3.7.16-1.amzn2.0.4.x86_64                       8/8
  Verifying  : python-srpm-macros-3-60.amzn2.0.1.noarch                      1/8
  Verifying  : system-rpm-config-9.1.0-76.amzn2.0.14.noarch                  2/8
  Verifying  : python-rpm-macros-3-60.amzn2.0.1.noarch                       3/8
  Verifying  : dwz-0.11-3.amzn2.0.3.x86_64                                   4/8
  Verifying  : python3-rpm-macros-3-60.amzn2.0.1.noarch                      5/8
  Verifying  : go-srpm-macros-3.0.15-23.amzn2.0.2.noarch                     6/8
  Verifying  : python3-devel-3.7.16-1.amzn2.0.4.x86_64                       7/8
  Verifying  : perl-srpm-macros-1-8.amzn2.0.1.noarch                         8/8

Installed:
  python3-devel.x86_64 0:3.7.16-1.amzn2.0.4

Dependency Installed:
  dwz.x86_64 0:0.11-3.amzn2.0.3
  go-srpm-macros.noarch 0:3.0.15-23.amzn2.0.2
  perl-srpm-macros.noarch 0:1-8.amzn2.0.1
  python-rpm-macros.noarch 0:3-60.amzn2.0.1
  python-srpm-macros.noarch 0:3-60.amzn2.0.1
  python3-rpm-macros.noarch 0:3-60.amzn2.0.1
  system-rpm-config.noarch 0:9.1.0-76.amzn2.0.14

Complete!
[root@ip-172-31-24-227 ~]# pip install --use-feature=2020-resolver happybase
WARNING: Running pip install with root privileges is generally not a good idea.
Try `pip3 install --user` instead.
Collecting happybase
  Downloading happybase-1.2.0.tar.gz (40 kB)
     |████████████████████████████████| 40 kB 9.1 MB/s
Requirement already satisfied: six in /usr/local/lib/python3.7/site-packages (fr
om happybase) (1.13.0)
Collecting thriftpy2>=0.4
  Downloading thriftpy2-0.4.17.tar.gz (519 kB)
     |████████████████████████████████| 519 kB 37.2 MB/s
  Installing build dependencies ... done
  WARNING: Missing build requirements in pyproject.toml for thriftpy2>=0.4 from
https://files.pythonhosted.org/packages/1d/5c/852a627317a75e0ec19f42b955ef115b09
06c43ee4c7595c112a652f0b20/thriftpy2-0.4.17.tar.gz#sha256=190f35c32da9146d1fdd82
2f46b6a0ad543572ea405ca6853b4ec7b128efbc0d (from happybase).
  WARNING: The project does not specify a build backend, and pip cannot fall bac
k to setuptools without 'wheel'.
  Getting requirements to build wheel ... done
  Installing backend dependencies ... done
    Preparing wheel metadata ... done
```

```
    Preparing wheel metadata ... done
Collecting six
  Downloading six-1.16.0-py2.py3-none-any.whl (11 kB)
Collecting ply<4.0,>=3.4
  Downloading ply-3.11-py2.py3-none-any.whl (49 kB)
     |                                | 49 kB 10.7 MB/s
Using legacy 'setup.py install' for happybase, since package 'wheel' is not inst
alled.
Building wheels for collected packages: thriftpy2
  Building wheel for thriftpy2 (PEP 517) ... done
  Created wheel for thriftpy2: filename=thriftpy2-0.4.17-cp37-cp37m-linux_x86_64
.whl size=1309802 sha256=ddd6757d3b920988f95d82a1a1f6475f845a83f26aa16edec44e1a5
885f5c7ae
  Stored in directory: /root/.cache/pip/wheels/6a/f3/2b/9a4b02cc3cdff27f9afc9101
d3df6de13e975caef66c7dcb77
Successfully built thriftpy2
Installing collected packages: six, ply, thriftpy2, happybase
  Attempting uninstall: six
    Found existing installation: six 1.13.0
    Uninstalling six-1.13.0:
      Successfully uninstalled six-1.13.0
    Running setup.py install for happybase ... done
Successfully installed happybase-1.2.0 ply-3.11 six-1.16.0 thriftpy2-0.4.17
[root@ip-172-31-24-227 ~]# pip install mrjob
WARNING: Running pip install with root privileges is generally not a good idea.
Try `pip3 install --user` instead.
Collecting mrjob
  Downloading mrjob-0.7.4-py2.py3-none-any.whl (439 kB)
     |                                | 439 kB 35.5 MB/s
Requirement already satisfied: PyYAML>=3.10 in /usr/local/lib64/python3.7/site-p
ackages (from mrjob) (5.4.1)
Installing collected packages: mrjob
  WARNING: The scripts mrjob, mrjob-3 and mrjob-3.7 are installed in '/usr/local
/bin' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warn
ing, use --no-warn-script-location.
Successfully installed mrjob-0.7.4
[root@ip-172-31-24-227 ~]# jps
28578 RunJar
31236 RunJar
17125 Bootstrap
20966 WebAppProxyServer
23239 HMaster
20746 JobHistoryServer
32524 Main
21999 QuorumPeerMain
9298 Main
9331 Main
```

```
21999 QuorumPeerMain
9298 Main
9331 Main
13747 Log4jHotPatch
20533 ApplicationHistoryServer
19031 NameNode
21687 KMSWebServer
9497 Main
21212 ResourceManager
22524 ThriftServer
13662 Jps
22847 RESTServer
[root@ip-172-31-24-227 ~]# python -c "import happybase"
[root@ip-172-31-24-227 ~]#
```

Step 2. Create Python code using Happybase API

Create file batch_ingest.py file. Code is provided in separate file as desired:

```
vi batch_ingest.py <or Transfer through WinSCP, if already created on
local machine>
```

Step 3. Execution of Python Script to load Batch data to HBase

- Please enter following arguments.
  - DNS (URL) for RDS (MySQL) Instance.
  - Valid filenames with full path to load to HBase (any numbers).

```
python /home/hadoop/tasks/batch_ingest.py $DNS_EMR
/root/tripdata/yellow_tripdata_2017-03.csv
/root/tripdata/yellow_tripdata_2017-04.csv
```

Screenshot of execution of batch script

```
[root@ip-172-31-24-227 ~]# python /home/hadoop/tasks/batch_ingest.py $DNS_EMR /r
oot/tripdata/yellow_tripdata_2017-03.csv /root/tripdata/yellow_tripdata_2017-04.
csv
Connection to HBase... ec2-3-80-83-172.compute-1.amazonaws.com
Opening the file...
2023-11-28 07:25:36.979036 : Rows inserted : 0
2023-11-28 07:25:36.979088 : Rows inserted : 0
2023-11-28 07:28:09.858522 : Rows inserted : 1000000
2023-11-28 07:30:42.449262 : Rows inserted : 2000000
2023-11-28 07:33:18.313306 : Rows inserted : 3000000
2023-11-28 07:35:52.799631 : Rows inserted : 4000000
2023-11-28 07:38:26.871479 : Rows inserted : 5000000
2023-11-28 07:41:03.028671 : Rows inserted : 6000000
2023-11-28 07:43:28.247667 : Rows inserted : 7000000
2023-11-28 07:45:55.710578 : Rows inserted : 8000000
2023-11-28 07:48:21.600354 : Rows inserted : 9000000
2023-11-28 07:51:06.711071 : Rows inserted : 10000000
Close Connection...
File Name:  /root/tripdata/yellow_tripdata_2017-03.csv
Total Rows inserted:  10295441
Time taken to insert  0:26:20.229256
Connection to HBase... ec2-3-80-83-172.compute-1.amazonaws.com
Opening the file...
2023-11-28 07:53:53.059134 : Rows inserted : 11000000
2023-11-28 07:56:37.152327 : Rows inserted : 12000000
2023-11-28 07:59:10.886844 : Rows inserted : 13000000
2023-11-28 08:01:47.508329 : Rows inserted : 14000000
2023-11-28 08:04:36.488507 : Rows inserted : 15000000
2023-11-28 08:07:10.597460 : Rows inserted : 16000000
2023-11-28 08:09:43.411230 : Rows inserted : 17000000
2023-11-28 08:12:26.341877 : Rows inserted : 18000000
2023-11-28 08:15:19.403279 : Rows inserted : 19000000
2023-11-28 08:18:10.830598 : Rows inserted : 20000000
Close Connection...
File Name:  /root/tripdata/yellow_tripdata_2017-04.csv
Total Rows inserted:  20342576
Time taken to insert  0:27:13.898490
All files loaded successfully.
[root@ip-172-31-24-227 ~]#
```

Step 4. Verification of data inserted in HBase

As per instructions 4 datasets for month Jan (01), Feb (02), Mar (03) and Apr (04) of year 2017 have been inserted in HBase table. Total rows inserted in HBase are as follows:

SN	File Name	Year-Month	Inserted through	Total Rows (remove 1 header row)
1	yellow_tripdata_2017-01.csv	2017-01	Sqoop with source data at MySQL	9710820

2	yellow_tripdata_2017-02.csv	2017-02	Sqoop with source data at MySQL	9169775
3	yellow_tripdata_2017-03.csv	2017-03	Batch Ingestion using Happybase	10295441
4	yellow_tripdata_2017-04.csv	2017-04	Batch Ingestion using Happybase	10047135
Total rows in HBase Table				**39223171**

Dataset loaded has been verified using 'count' command on HBase Table. As dataset is huge, therefore using interval size of 1 million rows while counting. Commands are:

```
hbase shell
count 'yellow_tripdata_full', INTERVAL => 1000000
```

Screenshot of data upload verification

```
[root@ip-172-31-24-227 ~]# hbase shell
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/lib/hadoop/lib/slf4j-reload4j-1.7.36.jar!
/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/lib/hbase/lib/client-facing-thirdparty/sl
f4j-reload4j-1.7.33.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.slf4j.impl.Reload4jLoggerFactory]
HBase Shell
Use "help" to get list of supported commands.
Use "exit" to quit this interactive shell.
For Reference, please visit: http://hbase.apache.org/2.0/book.html#shell
Version 2.4.15-amzn-0.1, rUnknown, Fri Jun 23 16:31:13 UTC 2023
Took 0.0033 seconds
```

```
hbase:001:0> count 'yellow_tripdata_full', INTERVAL => 1000000
Current count: 1000000, row: CSV-2017-03-0001000000
Current count: 2000000, row: CSV-2017-03-0002000000
Current count: 3000000, row: CSV-2017-03-0003000000
Current count: 4000000, row: CSV-2017-03-0004000000
Current count: 5000000, row: CSV-2017-03-0005000000
Current count: 6000000, row: CSV-2017-03-0006000000
Current count: 7000000, row: CSV-2017-03-0007000000
Current count: 8000000, row: CSV-2017-03-0008000000
Current count: 9000000, row: CSV-2017-03-0009000000
Current count: 10000000, row: CSV-2017-03-0010000000
Current count: 11000000, row: CSV-2017-04-0011000000
Current count: 12000000, row: CSV-2017-04-0012000000
Current count: 13000000, row: CSV-2017-04-0013000000
Current count: 14000000, row: CSV-2017-04-0014000000
Current count: 15000000, row: CSV-2017-04-0015000000
Current count: 16000000, row: CSV-2017-04-0016000000
Current count: 17000000, row: CSV-2017-04-0017000000
Current count: 18000000, row: CSV-2017-04-0018000000
Current count: 19000000, row: CSV-2017-04-0019000000
Current count: 20000000, row: CSV-2017-04-0020000000
Current count: 21000000, row: MYSQL-2017-01-0000657424
Current count: 22000000, row: MYSQL-2017-01-0001657424
Current count: 23000000, row: MYSQL-2017-01-0002657424
Current count: 24000000, row: MYSQL-2017-01-0003657424
Current count: 25000000, row: MYSQL-2017-01-0004657424
Current count: 26000000, row: MYSQL-2017-01-0005657424
Current count: 27000000, row: MYSQL-2017-01-0006657424
Current count: 28000000, row: MYSQL-2017-01-0007657424
Current count: 29000000, row: MYSQL-2017-01-0008657424
Current count: 30000000, row: MYSQL-2017-01-0009657424
Current count: 31000000, row: MYSQL-2017-02-0010657424
Current count: 32000000, row: MYSQL-2017-02-0011657424
Current count: 33000000, row: MYSQL-2017-02-0012657424
Current count: 34000000, row: MYSQL-2017-02-0013657424
Current count: 35000000, row: MYSQL-2017-02-0014657424
Current count: 36000000, row: MYSQL-2017-02-0015657424
Current count: 37000000, row: MYSQL-2017-02-0016657424
Current count: 38000000, row: MYSQL-2017-02-0017657424
Current count: 39000000, row: MYSQL-2017-02-0018657424
39223171 row(s)
Took 1869.4295 seconds
=> 39223171
hbase:002:0>
```

Line count in CSV files (has 1 row for header in each file)

```
[root@ip-172-31-28-99 tripdata]# wc -l yellow_tripdata_2017-0[1-4].csv
   9710821 yellow_tripdata_2017-01.csv
   9169776 yellow_tripdata_2017-02.csv
  10295442 yellow_tripdata_2017-03.csv
  10047136 yellow_tripdata_2017-04.csv
  39223175 total
```

********** END OF TASK-3 **********