Lesson_5
Lesson_5 ICP

Name: Andrew Poitras
email: ap3h7@mail.umkc.edu

Description:
1.
I began by following the lecture to create a table in mysql and import it into hdfs through sqoop then export it from hdfs to mysql.

1-1. I started mysql and created a database named db1 to put my table into.
Query:

```
mysql> create database db1;
Query OK, 1 row affected (0.00 sec)

mysql> use db1;
Database changed
mysql>
```

1-2. I then created a table called acad and inserted values into it and displayed the table to make sure that it worked.
Query+Result:

```
mysql> create table acad(emp_id int not null auto_increment, emp_name varchar(10
0), emp_sal int, primary key(emp_id));
Query OK, 0 rows affected (0.00 sec)

mysql> insert into acad values (5, "sanam", 50000),(6,"opra",600000),(7,"yella",
700000);
Query OK, 3 rows affected (0.00 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> select * from acad;
+--------+----------+---------+
| emp_id | emp_name | emp_sal |
+--------+----------+---------+
|      5 | sanam    |   50000 |
|      6 | opra     |  600000 |
|      7 | yella    |  700000 |
+--------+----------+---------+
3 rows in set (0.00 sec)

mysql>
```

1-3. I then imported acad into hdfs and displayed it to make sure that it worked.
Query:

```
[cloudera@quickstart ~]$ sqoop import --connect jdbc:mysql://localhost/db1 --use
rname root --password cloudera --table acad --m 1
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
21/05/08 09:10:29 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
21/05/08 09:10:29 WARN tool.BaseSqoopTool: Setting your password on the command-
line is insecure. Consider using -P instead.
21/05/08 09:10:30 INFO manager.MySQLManager: Preparing to use a MySQL streaming
resultset.
21/05/08 09:10:30 INFO tool.CodeGenTool: Beginning code generation
21/05/08 09:10:30 INFO manager.SqlManager: Executing SQL statement: SELECT t.* F
ROM `acad` AS t LIMIT 1
21/05/08 09:10:30 INFO manager.SqlManager: Executing SQL statement: SELECT t.* F
ROM `acad` AS t LIMIT 1
```

Result:

```
[cloudera@quickstart ~]$ cd Downloads
[cloudera@quickstart Downloads]$ hadoop fs -ls
Found 5 items
drwx------   - cloudera cloudera          0 2021-05-08 09:10 .staging
drwxr-xr-x   - cloudera cloudera          0 2021-05-06 23:18 DataBase
drwxr-xr-x   - cloudera cloudera          0 2021-05-08 09:10 acad
drwxr-xr-x   - cloudera cloudera          0 2021-05-06 16:20 matrices
drwxr-xr-x   - cloudera cloudera          0 2021-05-06 12:50 wordcount
[cloudera@quickstart Downloads]$ hadoop fs -ls acad/
Found 2 items
-rw-r--r--   1 cloudera cloudera          0 2021-05-08 09:10 acad/_SUCCESS
-rw-r--r--   1 cloudera cloudera         43 2021-05-08 09:10 acad/part-m-00000
[cloudera@quickstart Downloads]$ hadoop fs -cat acad/*
5,sanam,50000
6,opra,600000
7,yella,700000
[cloudera@quickstart Downloads]$
```

1-4. I then practiced by exporting acad back to mysql

Query:

```
[cloudera@quickstart Downloads]$ sqoop export --connect jdbc:mysql://localhost/d
b1 --username root --password cloudera --table acad --export-dir queryresult/par
t-m-00000
```

2. For part 2 I had to create a table with an hql script and run some queries on it.

2-1. I ran the hql script in hive and created the table

Query:

```
[cloudera@quickstart Downloads]$ hive -f tables-schemas.hql

Logging initialized using configuration in jar:file:/usr/lib/hive/lib/hive-commo
n-1.1.0-cdh5.13.0.jar!/hive-log4j.properties
OK
Time taken: 1.669 seconds
OK
Time taken: 0.204 seconds
OK
employees
movies
olympic
petrol
ratings
users
Time taken: 0.115 seconds, Fetched: 6 row(s)
ls: cannot access /home/hadoop/thinkbig-hive-tutorial/data/employees/input: No s
uch file or directory
Command failed with exit code = 2
WARN: The method class org.apache.commons.logging.impl.SLF4JLogFactory#release()
 was invoked.
WARN: Please see http://www.slf4j.org/codes.html#release for an explanation.
```

Result:

```
hive> use work
    > ;
OK
Time taken: 1.366 seconds
hive> show tables
    > ;
OK
employees
movies
olympic
petrol
ratings
users
Time taken: 0.128 seconds, Fetched: 6 row(s)
hive> describe employees;
OK
name                    string
salary                  float
subordinates            array<string>
deductions              map<string,float>
address                 struct<street:string,city:string,state:string,zip:int>

Time taken: 0.082 seconds, Fetched: 5 row(s)
```

2-2. I then created a simple table called emp and loaded the data from acad from question 1 into it.

Queries+Results:

```
hive> create table emp (empid int, emp_name string)
    > row format delimited
    > fields terminated by ','
    > lines terminated by '\n'
    > stored as textfile;
OK
Time taken: 0.04 seconds
hive> load data inpath 'acad/'
    > into table emp;
Loading data to table work.emp
Table work.emp stats: [numFiles=1, totalSize=43]
OK
Time taken: 0.363 seconds
hive> select * from emp;
OK
5       sanam
6       opra
7       yella
Time taken: 0.201 seconds, Fetched: 3 row(s)
```

2-3. Then I made a new table in mysql that matched emp called newEmp and exported emp into it and viewed newEmp to make sure that it worked.
Queries:

```
mysql> create table empNew (empid int, emp_name varchar(100));
Query OK, 0 rows affected (0.02 sec)
```

```
[cloudera@quickstart ~]$ sqoop export --connect jdbc:mysql://localhost/db1 --use
rname root --password cloudera --table empNew --export-dir /user/hive/warehouse/
work.db/emp -m 1
```

Results:

```
mysql> select * from empNew;
+-------+----------+
| empid | emp_name |
+-------+----------+
|     5 | sanam    |
|     6 | opra     |
|     7 | yella    |
+-------+----------+
3 rows in set (0.00 sec)
```

3. For part 3 I used the dividends.csv data and imported it into a hive table.

3-1. I created a hive table called dividends and loaded dividends.csv into it.
Query:

```
hive> create table dividends (date DATE, dividend float)
    > row format delimited
    > fields terminated by ','
    > stored as textfile;
OK
Time taken: 0.102 seconds
hive> load data local inpath '/home/cloudera/Downloads/dividends.csv' into table
 dividends;
Loading data to table work.dividends
Table work.dividends stats: [numFiles=1, totalSize=3247]
OK
Time taken: 0.381 seconds
```

3-2. I then exported this table into mysql to run queries from there. I started this by making a similar table in mysql and then exporting into there.
Queries:

```
mysql> create table newDividends (date DATE, dividend float);
Query OK, 0 rows affected (0.02 sec)
```

```
[cloudera@quickstart ~]$ sqoop export --connect jdbc:mysql://localhost/db1 --use
rname root --password cloudera --table newDividends --export-dir /user/hive/ware
house/work.db/dividends -m 1
```

Results:

```
mysql> select * from newDividends;
+------------+----------+
| date       | dividend |
+------------+----------+
| 2010-02-08 |     0.55 |
| 2009-11-06 |     0.55 |
| 2009-08-06 |     0.55 |
| 2009-05-06 |     0.55 |
| 2009-02-06 |      0.5 |
| 2008-11-06 |      0.5 |
| 2008-08-06 |      0.5 |
| 2008-05-07 |      0.5 |
| 2008-02-06 |      0.4 |
| 2007-11-07 |      0.4 |
| 2007-08-08 |      0.4 |
| 2007-05-08 |      0.4 |
| 2007-02-07 |      0.3 |
| 2006-11-08 |      0.3 |
| 2006-08-08 |      0.3 |
| 2006-05-08 |      0.3 |
| 2006-02-08 |      0.2 |
```

3-3. I chose to run a query that grouped the data by dividend and ordered by dividend descending to see what dates had the highest dividends and when the dividends would change.
Query+Results:

```
mysql> select dividend,
    -> date
    -> from newDividends
    -> group by dividend
    -> order by dividend desc;
+----------+------------+
| dividend | date       |
+----------+------------+
|     0.55 | 2010-02-08 |
|      0.5 | 2009-02-06 |
|      0.4 | 2008-02-06 |
|   0.3025 | 1992-11-05 |
|      0.3 | 2007-02-07 |
|    0.275 | 1989-02-02 |
|   0.2375 | 1984-05-04 |
|    0.215 | 1983-02-03 |
|      0.2 | 2006-02-08 |
|     0.18 | 2005-02-08 |
|     0.16 | 2004-02-06 |
|  0.15625 | 1977-11-02 |
|     0.15 | 2003-02-06 |
|  0.14062 | 1976-11-04 |
|     0.14 | 2002-02-06 |
|    0.135 | 1993-05-06 |
```