Sqoop Assignment

Question 1) Suppose we have a test_db database in mysql. We have an input table.Customers inside test_db. (SQL Commands are given)

Cust_Id	Customer_Name	Purchase_Date	Item	City	Price	Cust_Type
100	Rishi	2020-08-16	Mobile	Kanpur	10000	Regular
200	Venu	2019-05-04	Laptop	Bangalore	61000	Premium
300	Priya	2018-06-25	Mobile	Jaipur	20000	Premium
400	Rini	2019-01-30	Handbag	Pune	1000	Regular
700	Deepu	2019-12-12	Appliances	Mumbai	25000	Premium

The table has a Primary key on the Price column (which of course is not the right choice as prices may repeat when data grows).

```
mysql> desc customer;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| cust_id       | int         | YES  |     | NULL    |       |
| customer_name | varchar(50) | YES  |     | NULL    |       |
| purchase_date | date        | YES  |     | NULL    |       |
| item          | varchar(30) | YES  |     | NULL    |       |
| city          | varchar(30) | YES  |     | NULL    |       |
| price         | int         | NO   | PRI | NULL    |       |
| cust_type     | varchar(30) | YES  |     | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
7 rows in set (0.08 sec)
```

```
mysql> select * from customer;
+---------+---------------+---------------+------------+-----------+-------+-----------+
| cust_id | customer_name | purchase_date | item       | city      | price | cust_type |
+---------+---------------+---------------+------------+-----------+-------+-----------+
|     400 | Rini          | 2019-01-30    | Handbag    | Pune      |  1000 | Regular   |
|     100 | Rishi         | 2020-08-16    | Mobile     | Kanpur    | 10000 | Regular   |
|     300 | Priya         | 2018-06-25    | Mobile     | Jaipur    | 20000 | Premium   |
|     700 | Deepu         | 2019-12-12    | Appliances | Mumbai    | 25000 | Premium   |
|     200 | Venu          | 2019-05-04    | Laptop     | Bangalore | 61000 | Premium   |
+---------+---------------+---------------+------------+-----------+-------+-----------+
5 rows in set (0.02 sec)
```

1) Before performing the sqoop import, using the sqoop command display the data present in mysql Customers table . The output of the command should not display on the console,rather should be redirected to log file named ‘query.output’. Display the contents of the query.output file , share the Snapshot of the command and the output .

Displaying data in MySQL using Sqoop Command in query.output file:

```
[hduser@localhost ~]$ sqoop eval --connect jdbc:mysql://localhost/test_db --username root --password Root123$ --query "select * from customer" 1>query.output;
```

```
[hduser@localhost ~]$ cat query.output
Warning: /usr/local/sqoop/../hcatalog does not exist! HCatalog jobs will fail.
Please set $HCAT_HOME to the root of your HCatalog installation.
Warning: /usr/local/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
-----------------------------------------------------------------------------------------------------------------
| cust_id    | customer_name        | purchase_date | item                 | city                 | price      | cust_type            |
-----------------------------------------------------------------------------------------------------------------
| 400        | Rini                 | 2019-01-30 | Handbag              | Pune                 | 1000       | Regular              |
| 100        | Rishi                | 2020-08-16 | Mobile               | Kanpur               | 10000      | Regular              |
| 300        | Priya                | 2018-06-25 | Mobile               | Jaipur               | 20000      | Premium              |
| 700        | Deepu                | 2019-12-12 | Appliances           | Mumbai               | 25000      | Premium              |
| 200        | Venu                 | 2019-05-04 | Laptop               | Bangalore            | 61000      | Premium              |
-----------------------------------------------------------------------------------------------------------------
```

2) Perform a single sqoop import inside the directory in hdfs named sqoop_importdir, considering all the following points:

- Import all the columns except Cust_Type in hdfs.
- Include only the purchases made after 2019-01-01
- The output data generated should have fields separated by | and rows separated by ; (semicolon)
- While importing, Nulls in the data, should be overridden with 'NA'
- Redirect the log messages generated on screen to the files log_out1 and log_out2.Display the contents of the log_out2 file, when sqoop import is successful,share the snapshot of the number of records retrieved.
- Display the contents of the sqoop_importdir

Importing data using Sqoop:

```
[hduser@localhost ~]$ sqoop import --connect jdbc:mysql://localhost/test_db --username root --password Root123$ --table customer --columns cust_id,customer_na
me,purchase_date,item,city,price --where "purchase_date > '2019-01-01' " --null-string "NA" --target-dir /user/hduser/sqoop_importdir --fields-terminated-by '
|' --lines-terminated-by ';' --delete-target-dir 1>log_out1 2>log_out2
```

Displaying log_out2 file:

```
[hduser@localhost ~]$ cat log_out2
22/06/25 09:57:11 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6
22/06/25 09:57:11 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
22/06/25 09:57:11 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
22/06/25 09:57:11 INFO tool.CodeGenTool: Beginning code generation
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/local/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7.10.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/local/hbase/lib/phoenix-4.11.0-HBase-0.98-client.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/local/hbase/lib/slf4j-log4j12-1.6.4.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.Log4jLoggerFactory]
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
SPI and manual loading of the driver class is generally unnecessary.
22/06/25 09:57:12 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `customer` AS t LIMIT 1
22/06/25 09:57:12 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `customer` AS t LIMIT 1
22/06/25 09:57:12 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/local/hadoop
Note: /tmp/sqoop-hduser/compile/30bbfd4dc7f8dbe6f67f5ce5196a5e0d/customer.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
```

Displaying the data:

```
[hduser@localhost ~]$ hdfs dfs -ls /user/hduser/sqoop_importdir
22/06/25 09:59:16 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 5 items
-rw-r--r--   1 hduser supergroup          0 2022-06-25 09:58 /user/hduser/sqoop_importdir/_SUCCESS
-rw-r--r--   1 hduser supergroup         79 2022-06-25 09:58 /user/hduser/sqoop_importdir/part-m-00000
-rw-r--r--   1 hduser supergroup         45 2022-06-25 09:58 /user/hduser/sqoop_importdir/part-m-00001
-rw-r--r--   1 hduser supergroup          0 2022-06-25 09:58 /user/hduser/sqoop_importdir/part-m-00002
-rw-r--r--   1 hduser supergroup         43 2022-06-25 09:58 /user/hduser/sqoop_importdir/part-m-00003
[hduser@localhost ~]$ hdfs dfs -cat /user/hduser/sqoop_importdir/part-m-00000
22/06/25 09:59:32 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
400|Rini|2019-01-30|Handbag|Pune|1000;100|Rishi|2020-08-16|Mobile|Kanpur|10000;[hduser@localhost ~]$
```

Importing data by using customer Id as split column:

```
[hduser@localhost ~]$ sqoop import --connect jdbc:mysql://localhost/test_db --username root --password Root123$ --table customer --columns cust_id,customer_na
me,purchase_date,item,city,price --where "purchase_date > '2019-01-01' " --null-string "NA" --split-by 'cust_id' --target-dir /user/hduser/sqoop_importdir --f
ields-terminated-by '|' -m 1 --delete-target-dir;
```

Displaying the data:

```
[hduser@localhost ~]$ hdfs dfs -ls /user/hduser/sqoop_importdir
22/06/25 10:51:45 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 2 items
-rw-r--r--   1 hduser supergroup          0 2022-06-25 10:48 /user/hduser/sqoop_importdir/_SUCCESS
-rw-r--r--   1 hduser supergroup        167 2022-06-25 10:48 /user/hduser/sqoop_importdir/part-m-00000
[hduser@localhost ~]$ hdfs dfs -cat /user/hduser/sqoop_importdir/part-m-00000 | head;
22/06/25 10:52:02 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
400|Rini|2019-01-30|Handbag|Pune|1000
100|Rishi|2020-08-16|Mobile|Kanpur|10000
700|Deepu|2019-12-12|Appliances|Mumbai|25000
200|Venu|2019-05-04|Laptop|Bangalore|61000
```

The new record inserted is:

Cust_Id	Customer_Name	Purchase_Date	Item	City	Price	Cust_Type
10000	Raman	2019/09/04	Misc	Cochin	9000	Regular

Mention the sqoop import command you will frame from your end to deal with such a situation to ensure even work distribution among mappers, using customized bounding val query.

Note: you got to know that cust_id 10000 is erroneous record and should not be taken care.

Displaying contents of the table:

```
mysql> select * from customer;
+---------+---------------+---------------+------------+-----------+-------+-----------+
| cust_id | customer_name | purchase_date | item       | city      | price | cust_type |
+---------+---------------+---------------+------------+-----------+-------+-----------+
|     400 | Rini          | 2019-01-30    | Handbag    | Pune      |  1000 | Regular   |
|   10000 | Raman         | 2019-09-04    | Misc       | Cochin    |  9000 | Regular   |
|     100 | Rishi         | 2020-08-16    | Mobile     | Kanpur    | 10000 | Regular   |
|     300 | Priya         | 2018-06-25    | Mobile     | Jaipur    | 20000 | Premium   |
|     700 | Deepu         | 2019-12-12    | Appliances | Mumbai    | 25000 | Premium   |
|     200 | Venu          | 2019-05-04    | Laptop     | Bangalore | 61000 | Premium   |
+---------+---------------+---------------+------------+-----------+-------+-----------+
6 rows in set (0.01 sec)
```

Import using Custom Boundary query:

```
[hduser@localhost ~]$ sqoop import --connect jdbc:mysql://localhost/test_db \
> --username root --password Root123$ \
> --table customer --boundary-query "select min(cust_id),max(cust_id) from customer where cust_id < 10000" --target-dir /user/hduser/sqoop_importdir \
> --delete-target-dir --split-by 'cust_id';
```

Displaying the Data:

```
[hduser@localhost ~]$ hdfs dfs -ls /user/hduser/sqoop_importdir
22/06/25 11:18:36 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 5 items
-rw-r--r--   1 hduser supergroup          0 2022-06-25 11:14 /user/hduser/sqoop_importdir/_SUCCESS
-rw-r--r--   1 hduser supergroup        100 2022-06-25 11:14 /user/hduser/sqoop_importdir/part-m-00000
-rw-r--r--   1 hduser supergroup         49 2022-06-25 11:14 /user/hduser/sqoop_importdir/part-m-00001
-rw-r--r--   1 hduser supergroup         46 2022-06-25 11:14 /user/hduser/sqoop_importdir/part-m-00002
-rw-r--r--   1 hduser supergroup         53 2022-06-25 11:14 /user/hduser/sqoop_importdir/part-m-00003
[hduser@localhost ~]$ hdfs dfs -cat /user/hduser/sqoop_importdir/part-m-00000
22/06/25 11:19:11 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Kanpur,100,Regular,Rishi,Mobile,10000,2020-08-16
Bangalore,200,Premium,Venu,Laptop,61000,2019-05-04
[hduser@localhost ~]$ hdfs dfs -cat /user/hduser/sqoop_importdir/part-m-00001
22/06/25 11:19:17 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Jaipur,300,Premium,Priya,Mobile,20000,2018-06-25
[hduser@localhost ~]$ hdfs dfs -cat /user/hduser/sqoop_importdir/part-m-00002
22/06/25 11:19:22 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Pune,400,Regular,Rini,Handbag,1000,2019-01-30
[hduser@localhost ~]$ hdfs dfs -cat /user/hduser/sqoop_importdir/part-m-00003
22/06/25 11:19:29 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Mumbai,700,Premium,Deepu,Appliances,25000,2019-12-12
```

Question 2) Suppose we have a database named test_new_db in mysql. We have three tables inside it:

City_Tbl (Consider this is the bigger table)
State_Tbl (Consider this is the smaller table)
Country_Tbl (Smaller Table)

City_Tbl: City_ID is the Primary Key Column

City_Name City_ID
Bangalore 1000
Mumbai 1001
Chennai 1002
Kolkata 1003
Delhi 1004
Pune 1005
Nagpur 1006
Surat 1007
Kochi 1008

State_Tbl: No Primary Key Column

State_Name Districts
Karnataka 30
TamilNadu 32
Goa 2
Kerala 14
Assam 33

Country_Tbl: No Primary Key Column

Name Country_Code
Belgium 32
Brazil 55
France 33
Iran 98
India 91

Table description:

```
mysql> desc city_tbl;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| city_name | varchar(20) | YES  |     | NULL    |       |
| city_id   | int         | NO   | PRI | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
2 rows in set (0.10 sec)

mysql> create table state_tbl(state_name varchar(20),districts int);
Query OK, 0 rows affected (0.11 sec)

mysql> desc state_tbl;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| state_name | varchar(20) | YES  |     | NULL    |       |
| districts  | int         | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
2 rows in set (0.03 sec)

mysql> create table country_tbl(name varchar(30),country_code int);
Query OK, 0 rows affected (0.10 sec)

mysql> desc country_tbl;
+--------------+-------------+------+-----+---------+-------+
| Field        | Type        | Null | Key | Default | Extra |
+--------------+-------------+------+-----+---------+-------+
| name         | varchar(30) | YES  |     | NULL    |       |
| country_code | int         | YES  |     | NULL    |       |
+--------------+-------------+------+-----+---------+-------+
```

Table Data:

```
mysql> select * from country_tbl;
+---------+--------------+
| name    | country_code |
+---------+--------------+
| Belgium |           32 |
| Brazil  |           55 |
| France  |           33 |
| Iran    |           98 |
| India   |           91 |
+---------+--------------+
5 rows in set (0.01 sec)
```

```
mysql> select * from state_tbl;
+-------------+------------+
| state_name | districts |
+-------------+------------+
| Karnataka  |         30 |
| TamilNadu  |         32 |
| Goa        |          2 |
| Kerala     |         14 |
| Assam      |         33 |
+-------------+------------+
5 rows in set (0.00 sec)
```

```
mysql> select * from city_tbl;
+-----------+---------+
| city_name | city_id |
+-----------+---------+
| Bangalore |    1000 |
| Mumbai    |    1001 |
| Chennai   |    1002 |
| Kolkata   |    1003 |
| Delhi     |    1004 |
| Pune      |    1005 |
| Nagpur    |    1006 |
| Surat     |    1007 |
| Kochi     |    1008 |
+-----------+---------+
9 rows in set (0.00 sec)
```

A) Using a single sqoop import command, Import all the tables present in test_new_db to hdfs excluding the Country_Tbl You have to do it with a single sqoop command.

Also, City_Tbl should have 3 output files generated in hdfs. All the output files should be stored inside sqoop_all_tbl directory in hdfs, with sub-directories of each table name created inside the main directory. Share the snapshot of the command.

```
[hduser@localhost ~]$ sqoop import-all-tables --connect jdbc:mysql://localhost/test_new_db --username root --password Root123$ --exclude-tables \
> country_tbl --warehouse-dir /user/hduser/sqoop_all_tbl -m 3 --autoreset-to-one-mapper;
```

B) Show the contents of the output directory: (Share Snapshot)

```
[hduser@localhost ~]$ hdfs dfs -ls /user/hduser/sqoop_all_tbl
22/06/25 11:51:41 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 2 items
drwxr-xr-x   - hduser supergroup          0 2022-06-25 11:49 /user/hduser/sqoop_all_tbl/city_tbl
drwxr-xr-x   - hduser supergroup          0 2022-06-25 11:49 /user/hduser/sqoop_all_tbl/state_tbl
[hduser@localhost ~]$ hdfs dfs -ls /user/hduser/sqoop_all_tbl/city_tbl
22/06/25 11:51:50 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 4 items
-rw-r--r--   1 hduser supergroup          0 2022-06-25 11:49 /user/hduser/sqoop_all_tbl/city_tbl/_SUCCESS
-rw-r--r--   1 hduser supergroup         40 2022-06-25 11:49 /user/hduser/sqoop_all_tbl/city_tbl/part-m-00000
-rw-r--r--   1 hduser supergroup         34 2022-06-25 11:49 /user/hduser/sqoop_all_tbl/city_tbl/part-m-00001
-rw-r--r--   1 hduser supergroup         34 2022-06-25 11:49 /user/hduser/sqoop_all_tbl/city_tbl/part-m-00002
[hduser@localhost ~]$ hdfs dfs -ls /user/hduser/sqoop_all_tbl/state_tbl
22/06/25 11:51:57 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 2 items
-rw-r--r--   1 hduser supergroup          0 2022-06-25 11:49 /user/hduser/sqoop_all_tbl/state_tbl/_SUCCESS
-rw-r--r--   1 hduser supergroup         51 2022-06-25 11:49 /user/hduser/sqoop_all_tbl/state_tbl/part-m-00000
```

Question 3) We have a Categories Table in test_db in Mysql. On this table both inserts and updates are performed from ti me to time.

```
mysql> desc categories;
+------------------------+-------------+------+-----+---------+----------------+
| Field                  | Type        | Null | Key | Default | Extra          |
+------------------------+-------------+------+-----+---------+----------------+
| category_id            | int         | NO   | PRI | NULL    | auto_increment |
| category_department_id | int         | YES  |     | NULL    |                |
| category_name          | varchar(45) | YES  |     | NULL    |                |
| inclusion_date         | datetime    | NO   |     | NULL    |                |
+------------------------+-------------+------+-----+---------+----------------+
4 rows in set (0.12 sec)
```

```
mysql> select * from categories;
+-------------+------------------------+----------------------+---------------------+
| category_id | category_department_id | category_name        | inclusion_date      |
+-------------+------------------------+----------------------+---------------------+
|           1 |                      2 | FootBall             | 2020-04-30 00:00:00 |
|           2 |                      2 | HandBall             | 2020-05-01 00:00:00 |
|           3 |                      2 | Baseball and Softball | 2020-05-01 00:00:00 |
|           4 |                      2 | Basketball           | 2020-04-30 00:00:00 |
|           5 |                      3 | Tennins              | 2020-04-30 00:00:00 |
|           6 |                      3 | Hockey               | 2020-05-01 00:00:00 |
|           7 |                      3 | Swimming             | 2020-05-01 00:00:00 |
|           8 |                      3 | Cardio Equipment     | 2020-05-01 00:00:00 |
|           9 |                      4 | Strength Training    | 2020-05-01 00:00:00 |
|          10 |                      4 | Athletics            | 2020-05-02 00:00:00 |
|          11 |                   NULL | Cycling              | 2020-02-02 00:00:00 |
|          12 |                      5 | NULL                 | 2020-01-15 00:00:00 |
+-------------+------------------------+----------------------+---------------------+
12 rows in set (0.12 sec)
```

Do the following:

A) Import the Categories table in hdfs but during the import,do proper Null value handling:
- String Columns nulls should be replaced with '\N' (so that in file it should be read as \n and Non-string column nulls should be replaced with -1
- Use a warehouse directory
- We also want to see the query run by each mapper internally

Share the import command you will use, keeping in mind all of the above. Initially all records to be pulled in.

Import command:

```
[hduser@localhost ~]$ sqoop import --connect jdbc:mysql://localhost/test_db \
> --username root --password Root123$ --table categories \
> --null-string '\n' --null-non-string -1 \
> --warehouse-dir /user/hduser/sqoop_importdir2 --verbose
```

Query ran by each mapper:

```
22/06/25 18:03:39 INFO db.DataDrivenDBInputFormat: BoundingValsQuery: SELECT MIN(`category_id`), MAX(`category_id`) FROM `categories`
22/06/25 18:03:39 DEBUG db.IntegerSplitter: Splits: [                          1 to                          12] into 4 parts
22/06/25 18:03:39 DEBUG db.IntegerSplitter:                           1
22/06/25 18:03:39 DEBUG db.IntegerSplitter:                           4
22/06/25 18:03:39 DEBUG db.IntegerSplitter:                           7
22/06/25 18:03:39 DEBUG db.IntegerSplitter:                          10
22/06/25 18:03:39 DEBUG db.IntegerSplitter:                          12
22/06/25 18:03:39 DEBUG db.DataDrivenDBInputFormat: Creating input split with lower bound '`category_id` >= 1' and upper bound '`category_id` < 4'
22/06/25 18:03:39 DEBUG db.DataDrivenDBInputFormat: Creating input split with lower bound '`category_id` >= 4' and upper bound '`category_id` < 7'
22/06/25 18:03:39 DEBUG db.DataDrivenDBInputFormat: Creating input split with lower bound '`category_id` >= 7' and upper bound '`category_id` < 10'
22/06/25 18:03:39 DEBUG db.DataDrivenDBInputFormat: Creating input split with lower bound '`category_id` >= 10' and upper bound '`category_id` <= 12'
22/06/25 18:03:39 INFO mapreduce.JobSubmitter: number of splits:4
```

Displaying the content:

```
[hduser@localhost ~]$ hdfs dfs -ls /user/hduser/sqoop_importdir2
22/06/25 18:08:55 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 1 items
drwxr-xr-x   - hduser supergroup          0 2022-06-25 18:04 /user/hduser/sqoop_importdir2/categories
[hduser@localhost ~]$ hdfs dfs -ls /user/hduser/sqoop_importdir2/categories
22/06/25 18:09:05 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 5 items
-rw-r--r--   1 hduser supergroup          0 2022-06-25 18:04 /user/hduser/sqoop_importdir2/categories/_SUCCESS
-rw-r--r--   1 hduser supergroup        118 2022-06-25 18:04 /user/hduser/sqoop_importdir2/categories/part-m-00000
-rw-r--r--   1 hduser supergroup        104 2022-06-25 18:04 /user/hduser/sqoop_importdir2/categories/part-m-00001
-rw-r--r--   1 hduser supergroup        122 2022-06-25 18:04 /user/hduser/sqoop_importdir2/categories/part-m-00002
-rw-r--r--   1 hduser supergroup        102 2022-06-25 18:04 /user/hduser/sqoop_importdir2/categories/part-m-00003
[hduser@localhost ~]$ hdfs dfs -cat /user/hduser/sqoop_importdir2/categories/part-m-00003
22/06/25 18:09:20 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
10,4,Athletics,2020-05-02 00:00:00.0
11,-1,Cycling,2020-02-02 00:00:00.0
12,5,
,2020-01-15 00:00:00.0
```

B) New Records are added to the table and also existing records are updated, (refer the mysql_commands text file for the insert and update commands), so import only those newly inserted/updated records from Categories table to hdfs.
The delta records should get appended to existing directory.

Share the import command you will use this time, to get only delta records.

Insertion and Updating:

```
mysql> select * from categories;
+-------------+-----------------------+-----------------------+---------------------+
| category_id | category_department_id | category_name         | inclusion_date      |
+-------------+-----------------------+-----------------------+---------------------+
|           1 |                     2 | FootBall              | 2020-04-30 00:00:00 |
|           2 |                     2 | Walking               | 2020-07-15 00:00:00 |
|           3 |                     2 | Baseball and Softball | 2020-05-01 00:00:00 |
|           4 |                     2 | Basketball            | 2020-04-30 00:00:00 |
|           5 |                     3 | Tennins               | 2020-04-30 00:00:00 |
|           6 |                     3 | Hockey                | 2020-05-01 00:00:00 |
|           7 |                     3 | Swimming              | 2020-05-01 00:00:00 |
|           8 |                     3 | Cardio Equipment      | 2020-05-01 00:00:00 |
|           9 |                     4 | Strength Training     | 2020-05-01 00:00:00 |
|          10 |                     4 | Athletics             | 2020-05-02 00:00:00 |
|          11 |                  NULL | Cycling               | 2020-02-02 00:00:00 |
|          12 |                     5 | NULL                  | 2020-01-15 00:00:00 |
|          13 |                     6 | Running               | 2020-08-15 00:00:00 |
+-------------+-----------------------+-----------------------+---------------------+
13 rows in set (0.00 sec)
```

Category Id 2 is updated and category id 13 is newly added.

Sqoop Command:

```
[hduser@localhost ~]$ sqoop import --connect jdbc:mysql://localhost/test_db --username root --password Root123$ --table categories --null-string '\n' --null-n
on-string -1 --incremental lastmodified --check-column 'inclusion_date' --last-value '2020-06-15' --warehouse-dir /user/hduser/sqoop_importdir2 --append
```

```
22/06/25 18:23:54 INFO mapreduce.ImportJobBase: Retrieved 2 records.
22/06/25 18:23:54 INFO util.AppendUtils: Appending to directory categories
22/06/25 18:23:54 INFO util.AppendUtils: Using found partition 4
22/06/25 18:23:54 INFO tool.ImportTool: Incremental import complete! To run another incremental import of all data following this import, supply the following
 arguments:
22/06/25 18:23:54 INFO tool.ImportTool:  --incremental lastmodified
22/06/25 18:23:54 INFO tool.ImportTool:   --check-column inclusion_date
22/06/25 18:23:54 INFO tool.ImportTool:   --last-value 2022-06-25 18:22:59.0
22/06/25 18:23:54 INFO tool.ImportTool: (Consider saving this with 'sqoop job --create')
[hduser@localhost ~]$ hdfs dfs -ls /user/hduser/sqoop_importdir2/categories
22/06/25 18:24:11 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 9 items
-rw-r--r--   1 hduser supergroup          0 2022-06-25 18:04 /user/hduser/sqoop_importdir2/categories/_SUCCESS
-rw-r--r--   1 hduser supergroup        118 2022-06-25 18:04 /user/hduser/sqoop_importdir2/categories/part-m-00000
-rw-r--r--   1 hduser supergroup        104 2022-06-25 18:04 /user/hduser/sqoop_importdir2/categories/part-m-00001
-rw-r--r--   1 hduser supergroup        122 2022-06-25 18:04 /user/hduser/sqoop_importdir2/categories/part-m-00002
-rw-r--r--   1 hduser supergroup        102 2022-06-25 18:04 /user/hduser/sqoop_importdir2/categories/part-m-00003
-rw-r--r--   1 hduser supergroup         34 2022-06-25 18:23 /user/hduser/sqoop_importdir2/categories/part-m-00004
-rw-r--r--   1 hduser supergroup          0 2022-06-25 18:23 /user/hduser/sqoop_importdir2/categories/part-m-00005
-rw-r--r--   1 hduser supergroup          0 2022-06-25 18:23 /user/hduser/sqoop_importdir2/categories/part-m-00006
-rw-r--r--   1 hduser supergroup         35 2022-06-25 18:23 /user/hduser/sqoop_importdir2/categories/part-m-00007
```

C) After this second import, how many records do you see in the hdfs folder now? Did you find any duplicate record s, give details if any.

We can see total 14 records in hdfs folder. 12 are previously imported, one is newly added and one is updated record.

```
[hduser@localhost ~]$ hdfs dfs -cat /user/hduser/sqoop_importdir2/categories/part-m-00000
22/06/25 18:27:57 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
1,2,FootBall,2020-04-30 00:00:00.0
2,2,HandBall,2020-05-01 00:00:00.0
3,2,Baseball and Softball,2020-05-01 00:00:00.0
[hduser@localhost ~]$ hdfs dfs -cat /user/hduser/sqoop_importdir2/categories/part-m-00004
22/06/25 18:28:07 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2,2,Walking,2020-07-15 00:00:00.0
```

The record with customer id 2 is present twice. One is the older record and one is the newly updated record.

D) Create a new table in test_db named Categories_new.

This newly created table does not have a Primary key.

New table without primary key:

```
mysql> desc categories_new;
+-------------------------+-------------+------+-----+---------+-------+
| Field                   | Type        | Null | Key | Default | Extra |
+-------------------------+-------------+------+-----+---------+-------+
| category_id             | int         | NO   |     | 0       |       |
| category_department_id  | int         | YES  |     | NULL    |       |
| category_name           | varchar(45) | YES  |     | NULL    |       |
| inclusion_date          | datetime    | NO   |     | NULL    |       |
+-------------------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

Data in categories_new table:

```
mysql> select * from categories_new;
+-------------+-----------------------+----------------------+---------------------+
| category_id | category_department_id | category_name        | inclusion_date      |
+-------------+-----------------------+----------------------+---------------------+
|           1 |                     2 | FootBall             | 2020-04-30 00:00:00 |
|           2 |                     2 | Walking              | 2020-07-15 00:00:00 |
|           3 |                     2 | Baseball and Softball | 2020-05-01 00:00:00 |
|           4 |                     2 | Basketball           | 2020-04-30 00:00:00 |
|           5 |                     3 | Tennins              | 2020-04-30 00:00:00 |
|           6 |                     3 | Hockey               | 2020-05-01 00:00:00 |
|           7 |                     3 | Swimming             | 2020-05-01 00:00:00 |
|           8 |                     3 | Cardio Equipment     | 2020-05-01 00:00:00 |
|           9 |                     4 | Strength Training    | 2020-05-01 00:00:00 |
|          10 |                     4 | Athletics            | 2020-05-02 00:00:00 |
|          11 |                  NULL | Cycling              | 2020-02-02 00:00:00 |
|          12 |                     5 | NULL                 | 2020-01-15 00:00:00 |
|          13 |                     6 | Running              | 2020-08-15 00:00:00 |
+-------------+-----------------------+----------------------+---------------------+
13 rows in set (0.05 sec)
```

We want to do periodic imports and updates in this mysql table. But we do not want any duplicate records in the hdfs post import.
Also we want to automate the process of import & want a good way to manage the password. Choose a different warehouse directory this time.

Using Encrypted Password for the Job:

Creation of the password file with encryption:

```
[hduser@localhost ~]$ hadoop credential create mysql.root.password -provider jceks://hdfs/user/hduser/mysql.password.file
22/06/26 18:09:14 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Enter password:
Enter password again:
mysql.root.password has been successfully created.
```

Share the commands you will use when:

- First time we need to pull all records in hdfs
- Second time to pull only the delta records, but without duplicates in hdfs

First time pulling all records:

```
[hduser@localhost ~]$ sqoop import --connect jdbc:mysql://localhost/test_db --username root --password Root123$ --table categories_new --warehouse-dir sqoop_i
mportdir2 --split-by category_id --null-string "NA" --null-non-string -1;
```

Records in the table added and updated:

```
mysql> select * from categories_new;
+-------------+------------------------+----------------------+---------------------+
| category_id | category_department_id | category_name        | inclusion_date      |
+-------------+------------------------+----------------------+---------------------+
|           1 |                      2 | FootBall             | 2020-04-30 00:00:00 |
|           2 |                      2 | Walking              | 2020-07-15 00:00:00 |
|           3 |                      2 | Baseball and Softball | 2020-05-01 00:00:00 |
|           4 |                      2 | Basketball           | 2020-04-30 00:00:00 |
|           5 |                      3 | Cricket              | 2020-10-20 00:00:00 |
|           6 |                      3 | Hockey               | 2020-05-01 00:00:00 |
|           7 |                      3 | Swimming             | 2020-05-01 00:00:00 |
|           8 |                      3 | Cardio Equipment     | 2020-05-01 00:00:00 |
|           9 |                      4 | Strength Training    | 2020-05-01 00:00:00 |
|          10 |                      4 | Athletics            | 2020-05-02 00:00:00 |
|          11 |                   NULL | Cycling              | 2020-02-02 00:00:00 |
|          12 |                      5 | NULL                 | 2020-01-15 00:00:00 |
|          13 |                      6 | Running              | 2020-08-15 00:00:00 |
|          14 |                      5 | Yoga                 | 2020-11-15 00:00:00 |
+-------------+------------------------+----------------------+---------------------+
14 rows in set (0.00 sec)
```

Category Id 14 is newly added and category id 5 is updated.

Importing new records without duplicates:

With password:

```
[hduser@localhost ~]$ sqoop job --create sqoop_lastmodfied -- import --connect jdbc:mysql://localhost/test_db --username root --password Root123$ --table cate
gories_new --warehouse-dir sqoop_importdir2 --split-by category_id --incremental lastmodified --check-column "inclusion_date" --null-string "NA" --null-non-st
ring -1 --last-value "2020-09-15" --merge-key "category_id";
```

Without password:

```
[hduser@localhost ~]$ sqoop job -Dhadoop.security.credential.provider.path=jceks://hdfs/user/hduser/mysql.password.file --create sqoop_lastmodfied3 -- import
--connect jdbc:mysql://localhost/test_db --username root --password-alias mysql.root.password --table categories_new --warehouse-dir sqoop_importdir3 --split-
by category_id --incremental lastmodified --check-column "inclusion_date" --null-string "NA" --null-non-string -1 --last-value "2020-09-15" --merge-key "categ
ory_id";
```

Executing the job: sqoop job –exec sqoop_lastmodifed;

E) How many records do you see this time in hdfs post second import? Do you see any duplicate records now?

No, we cannot see any duplicate recrords.
We can see total 14 records.

```
[hduser@localhost ~]$ hdfs dfs -cat /user/hduser/sqoop_importdir2/categories_new/part-r-00000
22/06/25 22:26:21 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2,1,FootBall,2020-04-30 00:00:00.0
4,10,Athletics,2020-05-02 00:00:00.0
-1,11,Cycling,2020-02-02 00:00:00.0
5,12,NA,2020-01-15 00:00:00.0
6,13,Running,2020-08-15 00:00:00.0
5,14,Yoga,2020-11-15 00:00:00.0
2,2,Walking,2020-07-15 00:00:00.0
2,3,Baseball and Softball,2020-05-01 00:00:00.0
2,4,Basketball,2020-04-30 00:00:00.0
3,5,Cricket,2020-10-20 00:00:00.0
3,6,Hockey,2020-05-01 00:00:00.0
3,7,Swimming,2020-05-01 00:00:00.0
3,8,Cardio Equipment,2020-05-01 00:00:00.0
4,9,Strength Training,2020-05-01 00:00:00.0
```

F) Are any mapper files generated in hdfs this time after the second import? Explain.

After the second import we will use merge key to overcome the duplicate records. The output generated will be a reduced file. There will be no mapper files.

G) Share the command you will use to see the last value of a Saved Sqoop Job.

Command Used: sqoop job –show jobname;

Sqoop Quiz

1. Sqoop written in?

A. C
B. C++
C. Java
D. hadoop

2. Sqoop stands for?

A. SQL to Hadoop
B. SQL to Hbase
C. MySQL to Hadoop
D. SQL Hadoop

3. Is Apache Sqoop is an open-source tool?

A. TRUE
B. FALSE
C. Can be true or false
D. Can not say

4. Data processed by Scoop can be used for?

A. Hbase
B. HDFS
C. **Mapreduce**
D. MahOut

5. _____ tool can list all the available database schemas

A. sqoop-list-tables
B. **sqoop-list-databases**
C. sqoop-list-schema
D. sqoop-list-columns

6. The active Hadoop configuration is loaded from $HADOOP_HOME/conf/, unless the $HADOOP_CONF_DIR environment variable is unset.

A. TRUE
B. FALSE
C. Can be true or false
D. Can not say

7. Data can be imported in maximum _____ file formats.

A. 2
B. 3
C. 4
D. 5

8. If you set the inline LOB limit to _____ all large objects will be placed in external storage.

A. 0
B. 2
C. 3
D. 1

9. The import-tables tool imports a set of tables from an RDBMS to?

A. Hive
B. Sqoop
C. HDFS
D. Mapreduce
.

10. Sqoop can also import the data into Hive by generating and executing a _____ statement to define the data's layout in Hive.

A. SET TABLE
B. CREATE TABLE
C. INSERT TABLE
D. All of the above

11. The following tool imports a set of tables from an RDBMS to HDFS
A. export-all-tables
B. **import**-all-**tables**
C. import-tables
D. none of the mentioned

12. With the –staging-table parameter, the data is moved from staging to final table

A. Automatically if staging load is successful
B. Has to be done by user after verifying the data in staging
C. Depends on the data size
D. Depends on the memory available to move the data