**Sqoop Import**

**Simple Sqoop Import**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root \

--password cloudera \

--table customers

**Specifying Mappers**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers

-m 2

**Managing destination directory**

**Defining warehouse directory**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--warehouse-dir /user/cloudera/new-warehouse

**Defining target directory**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--target-dir /user/cloudera/customer-new

**Delete target directory if already exists**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--target-dir /user/cloudera/customer-new \

--delete-target-dir

**Working with File Formats**

**Importing as avro files**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--target-dir /user/cloudera/customer-avro \

--as-avrodatafile

**Importing as parquet files**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--target-dir /user/cloudera/customer-parquet \

--as-parquetfile

**Importing as Sequence files**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--target-dir /user/cloudera/customer-sequence \

--as-sequencefile

**Gzip Compressed**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--target-dir /user/cloudera/customer\_gzip \

--compress

**Working with Compression Types**

**Snappy Compressed**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--target-dir /user/cloudera/customer\_snappy \

--compress \

--compression-codec snappy

**Deflate Compressed**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--target-dir /user/cloudera/customer\_deflate \

--compress \

--compression-codec deflate

**Bzip Compressed**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--target-dir /user/cloudera/customer\_bzip \

--compress \

--compression-codec bzip2

**Lz4 Compressed**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--target-dir /user/cloudera/customer\_lz4 \

--compress \

--compression-codec lz4

**Conditional/Selective Imports**

**Conditional Imports**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--target-dir /user/cloudera/customer-name-m \

--where “customer\_fname=’Mary’”

**Selective Column Imports**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--target-dir /user/cloudera/customer-selected \

--columns “customer\_fname,customer\_lname,customer\_city’”

**Using query**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--target-dir /user/cloudera/customer-queries \

--query “Select \* from customers where customer\_id > 100 AND $CONDITIONS” \

--split-by “customer\_id”

**Split-by/Boundary Query**

**Split-By**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table product \

--target-dir /user/cloudera/products\_split \

--split-by “product\_id”

**Boundary-query**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table products \

--target-dir /user/cloudera/customer-boundary \

--boundary-query 'Select min(product\_id),max(product\_id) from products where product\_id>100' \

--split-by product\_id

**Handling Null**

**Handling Null**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customer\_new \

--target-dir /user/cloudera/customer-new \

--null-string “xxx” \

--null-non-string “yyy”

**Field Delimeters**

**Field Delimeters**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--table customers \

--target-dir /user/cloudera/customer-delimeted \

--columns 'customer\_fname,customer\_lname,customer\_city' \

--fields-terminated-by '|'

**Incremental Appends**

**Simple Import**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--target-dir /user/cloudera/orders-incremental \

--table orders

**Inserts**

insert into orders (order\_id,order\_date,order\_status)

values(100004,'2017-11-07 10:02:00','CLOSED');

insert into orders (order\_id,order\_date,order\_status)

values(100005,'2017-11-07 10:02:00','CLOSED');

insert into orders (order\_id,order\_date,order\_status)

values(100006,'2017-11-07 10:02:00','CLOSED');

**Incremental Append**

sqoop import \

--connect jdbc:mysql://localhost/retail\_db \

--username root --password cloudera \

--target-dir /user/cloudera/orders-incremental \

--table orders \

--incremental append \

--check-column order\_id \

--last-value 100003

**Hive Import**

**Hive Import**

sqoop import \

--connect "jdbc:mysql://localhost/retail\_db" \

--username root \

--password cloudera \

--table customers \

--hive-import \

--create-hive-table \

--hive-database default \

--hive-table customer\_mysql

**Sqoop JOB command**

sqoop job --create order\_update -- import --connect jdbc:mysql://localhost:3306/retail\_db --username root --password password –table orders --target-dir /user/sample/orders --incremental append --check-column order\_id --last-value 100009

**Hive Import change field delimeter**

sqoop import \

--connect "jdbc:mysql://localhost/retail\_db" \

--username root \

--password cloudera \

--table customers \

--fields-terminated-by '|' \

--hive-import \

--create-hive-table \

--hive-database default \

--hive-table customer\_mysql\_new

Sqoop job continuously check if there is new row inserted to table and retrieve this information accordingly.

To execute sqoop job:

Sqoop job --exec order-update