**Task 1**

**Data:** Dataset1

**Instructions:**

* Create database targetdb at HDFS location /data/target/targetdb

hadoop fs -mkdir /data/target

hadoop fs -mkdir /data/target/targetdb

CREATE DATABASE targetdb LOCATION '/data/target/targetdb';

use targetdb;

* Create tables: movies, ratings, tags in targetdb database

create table movies (movieId int, title string, genres string) row format delimited fields terminated by ',';

create table ratings (userId int,movieId int,rating double,t\_stamp decimal(12,0)) row format delimited fields terminated by ',';

create table tags (userId int,movieId int,tag string,t\_stamp decimal(12,0)) row format delimited fields terminated by ',';

* Load the relevant data files to those tables

hadoop fs -copyFromLocal /home/cloudera/Desktop/Exercise\_7/Dataset1/\* /data/source/dataset1

hadoop fs -copyFromLocal /home/cloudera/Desktop/Exercise\_7/Dataset2/\* /data/source/dataset2

hadoop fs -copyFromLocal /home/cloudera/Desktop/Exercise\_7/Dataset3/\* /data/source/dataset3

impala-shell

invalidate metadata;

truncate table movies;

truncate table ratings;

truncate table tags;

LOAD DATA INPATH '/data/source/dataset1/movies.csv' INTO TABLE movies;

LOAD DATA INPATH '/data/source/dataset1/ratings.csv' INTO TABLE ratings;

LOAD DATA INPATH '/data/source/dataset1/tags.csv' INTO TABLE tags;

* Prepare a SQL to load movie\_stats table.

Following is the column detail for movie\_stats table.

movieid, movie\_name,avg\_rating,hash\_tag,hash\_tag\_cnt

**Note**: Choose appropriate column data type as per your knowledge.

truncate table movies\_stats;

=======================================================================

**Task 2**

Let’s apply some integrity constraints to our table

* Modify movies table to make column movieId UPI and NOT NULL

=======================================================================

**Task 3**

**Data:** Dataset2

**Instructions:**

* Create new database stagedb at location /data/staging/stagedb
* Create tables movies, ratings, tags under database stagedb with same details as targetdb tables
* Prepare three different scripts, one for each to load stage tables with given data files considering all possible integrity checks as
  + Movie ID is unique and not null field. Hence have to check for any unique key as well as not null violation
  + Rating should be for a valid movie. Hence movieId in rating data file should be present in Movie table
  + If there is any exact duplicate record then we must discard one.

=======================================================================

**Task 4**

**Instructions:**

* Prepare full refresh load script to refresh movie\_stats table

=======================================================================

**Task 5**

**Data:** Dataset 3

**Instructions:**

* Movies table is Slowly Changing Dimension (SCD) type 2 table.
* Generally, we receive Insert/Update/Delete flag with source file to load Slowly Changing Dimensions (SCD)
* Modify the movies table load script to implement IUD logic

=======================================================================

**Task 6**

**Instructions:**

We have some clients using MySQL database to generate reports. Hence we need to push our movie\_stats table to that MySQL database.