1. **Place irregularities\_head.csv file in Hadoop**

hdfs dfs -mkdir class\_project

hdfs dfs -mkdir class\_project/irregularities

mv irregularities-head.csv irregularities\_head.csv

hdfs dfs -put irregularities\_head.csv class\_project/irregularities/

hdfs dfs -chmod -R 777 .

2. **Connect to Hive and use your own database.**

-bash-4.1$ beeline

WARNING: Use "yarn jar" to launch YARN applications.

Beeline version 1.2.1000.2.4.2.0-258 by Apache Hive

beeline> !connect jdbc:hive2://cis5200s3-bdcsce-4.compute-608214094.oraclecloud.internal:2181,cis5200s3-bdcsce-2.compute-608214094.oraclecloud.internal:2181,cis5200s3-bdcsce-3.compute-608214094.oraclecloud.internal:2181/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=hiveserver2?tez.queue.name=interactive bdcsce\_admin

Connecting to jdbc:hive2://cis5200s3-bdcsce-4.compute-608214094.oraclecloud.internal:2181,cis5200s3-bdcsce-2.compute-608214094.oraclecloud.internal:2181,cis5200s3-bdcsce-3.compute-608214094.oraclecloud.internal:2181/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=hiveserver2?tez.queue.name=interactive

Enter password for jdbc:hive2://cis5200s3-bdcsce-4.compute-608214094.oraclecloud.internal:2181,cis5200s3-bdcsce-2.compute-608214094.oraclecloud.internal:2181,cis5200s3-bdcsce-3.compute-608214094.oraclecloud.internal:2181/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=hiveserver2?tez.queue.name=interactive:

use rjoshi5;

show tables;

1. **Create schema and use values from ‘irregularities\_head.csv’ file.**

CREATE EXTERNAL TABLE IF NOT EXISTS irregularities\_head\_raw\_data(id INT,detection\_date STRING,detection\_utc\_date STRING,update\_date STRING,update\_utc\_time STRING,street STRING,city STRING,state STRING,country STRING,is\_highway STRING,speed DOUBLE,regular\_speed DOUBLE,delay\_seconds SMALLINT,seconds SMALLINT,length SMALLINT,trend SMALLINT,type STRING,severity DOUBLE,jam\_level INT,drivers\_count INT,alerts\_count INT,n\_thumbs\_up INT,line\_x1 STRING,line\_y1 STRING,line\_x2 STRING,line\_y2 STRING)

ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

STORED AS TEXTFILE LOCATION '/user/rjoshi5/class\_project/irregularities';

**4. Create new directory for clean table in Hadoop and create cleaned table**

Create new table and insert cleaned data “irregularities\_head\_clean”

-bash-4.1$ hdfs dfs -mkdir /user/rjoshi5/class\_project/irregularities/clean\_irregularities/

-bash-4.1$ hdfs dfs -chmod 777 /user/rjoshi5/class\_project/irregularities/clean\_irregularities/

CREATE EXTERNAL TABLE IF NOT EXISTS irregularities\_head\_clean

( id INT,

weekday STRING,

data\_time STRING,

street STRING,

city STRING,

state STRING,

country STRING,

is\_highway STRING,

speed DOUBLE,

regular\_speed DOUBLE,

delay\_seconds SMALLINT,

seconds SMALLINT,

length SMALLINT,

trend SMALLINT,

type STRING,

severity DOUBLE,

jam\_level INT,

drivers\_count INT,

alerts\_count INT,

n\_thumbs\_up INT,

x1 STRING,

y1 STRING,

x2 STRING,

y2 STRING

)

STORED AS TEXTFILE LOCATION '/user/rjoshi5/class\_project/irregularities/clean\_irregularities/';

INSERT INTO TABLE irregularities\_head\_clean

SELECT id ,SUBSTRING(detection\_date, 0, 3) ,detection\_utc\_date ,street ,city ,state ,country ,IF(is\_highway == 't', 'highway', 'not highway') ,speed ,regular\_speed ,delay\_seconds ,seconds ,length ,trend ,type ,severity ,jam\_level ,drivers\_count ,alerts\_count ,n\_thumbs\_up ,REGEXP\_EXTRACT(line\_x1,'([-+]?\\d+(\\.\\d+))') as x1 ,REGEXP\_EXTRACT(line\_y1,'([-+]?\\d+(\\.\\d+))') as y1 ,REGEXP\_EXTRACT(line\_x2,'([-+]?\\d+(\\.\\d+))') as x2,REGEXP\_EXTRACT(line\_y2,'([-+]?\\d+(\\.\\d+))') as y2

FROM irregularities\_head\_raw\_data

**5.Create next table for second file ‘alerts\_head.csv’**

hdfs dfs -mkdir ./class\_project/alerts

hdfs dfs -chmod 777 ./class\_project/\*

hdfs dfs -put alerts\_head\_raw\_data.csv ./class\_project/alerts/

hdfs dfs -chmod 777 ./class\_project/alerts/\*

CREATE EXTERNAL TABLE IF NOT EXISTS alerts\_head\_raw\_data(id STRING,uuid INT,pub\_millis STRING,pub\_utc\_date STRING,road\_type SMALLINT,location\_x STRING,location\_y STRING,street STRING,city STRING,state STRING,country STRING,magvar SMALLINT,reliability SMALLINT,report\_description STRING,report\_rating SMALLINT,confidence SMALLINT,type STRING,subtype STRING,report\_by\_municipality\_user SMALLINT,thumbs\_up SMALLINT,jam\_uuid SMALLINT,datafile\_id SMALLINT) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE LOCATION '/user/rjoshi5/class\_project/alerts';

**6. Create additional table with name for road types** (source for the names: <https://github.com/CityOfLosAngeles/aqueduct/blob/master/waze/schema-waze.sql>)

CREATE TABLE road\_type (value SMALLINT,name STRING);

INSERT INTO road\_type (value, name) VALUES (1, 'Streets');

INSERT INTO road\_type (value, name) VALUES (2, 'Primary Street');

INSERT INTO road\_type (value, name) VALUES (3, 'Freeways');

INSERT INTO road\_type (value, name) VALUES (4, 'Ramps');

INSERT INTO road\_type (value, name) VALUES (5, 'Trails');

INSERT INTO road\_type (value, name) VALUES (6, 'Primary');

INSERT INTO road\_type (value, name) VALUES (7, 'Secondary');

INSERT INTO road\_type (value, name) VALUES (8, '4X4 Trails');

INSERT INTO road\_type (value, name) VALUES (9, 'Walkway');

INSERT INTO road\_type (value, name) VALUES (10, 'Pedestrian');

INSERT INTO road\_type (value, name) VALUES (11, 'Exit');

INSERT INTO road\_type (value, name) VALUES (12, '?');

INSERT INTO road\_type (value, name) VALUES (13, '?');

INSERT INTO road\_type (value, name) VALUES (14, '4X4 Trails');

INSERT INTO road\_type (value, name) VALUES (15, 'Ferry crossing');

INSERT INTO road\_type (value, name) VALUES (16, 'Stairway');

INSERT INTO road\_type (value, name) VALUES (17, 'Private road');

INSERT INTO road\_type (value, name) VALUES (18, 'Railroads');

INSERT INTO road\_type (value, name) VALUES (19, 'Runway/Taxiway');

INSERT INTO road\_type (value, name) VALUES (20, 'Parking lot road');

INSERT INTO road\_type (value, name) VALUES (21, 'Service road');

**7. Create new table and insert cleaned data “alerts\_head\_clean ”**

-bash-4.1$ hdfs dfs -mkdir /user/rjoshi5/class\_project/alerts/clean\_alerts/

-bash-4.1$ hdfs dfs -chmod 777 /user/rjoshi5/class\_project/alerts/clean\_alerts/

CREATE EXTERNAL TABLE IF NOT EXISTS alerts\_head\_clean

( id STRING,

date\_time STRING,

road\_type STRING,

location\_x STRING,

location\_y STRING,

street STRING,

city STRING,

state STRING,

country STRING,

reliability SMALLINT,

report\_description STRING,

report\_rating SMALLINT,

type STRING,

subtype STRING

)

STORED AS TEXTFILE LOCATION '/user/rjoshi5/class\_project/alerts/clean\_alerts/';

INSERT INTO TABLE alerts\_head\_clean

SELECT al.id ,al.pub\_utc\_date, r.name as road\_type,REGEXP\_EXTRACT(al.location\_x,'([-+]?\\d+(\\.\\d+))') as location\_x,REGEXP\_EXTRACT(al.location\_y,'([-+]?\\d+(\\.\\d+))') as location\_y,al.street,al.city,al.state,al.country,al.reliability,al.report\_description,al.report\_rating,al.type,al.subtype

FROM alerts\_head\_raw\_data al

LEFT JOIN road\_type r

on r.value = al.road\_type;

**8. Create a table with summary info from alerts**

CREATE EXTERNAL TABLE IF NOT EXISTS alerts\_sum

( no\_of\_alerts SMALLINT,

date\_ STRING,

time\_ STRING,

city STRING,

road\_type STRING,

type STRING,

subtype STRING

)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

STORED AS TEXTFILE LOCATION '/user/rjoshi5/class\_project/summary';

INSERT INTO TABLE alerts\_sum

SELECT count(\*) as no\_of\_alerts,SUBSTRING(date\_time, 0, 10), SUBSTRING(date\_time, 12, 5), city,road\_type, type, subtype

FROM alerts\_head\_clean

GROUP BY date\_time, city, road\_type, type, subtype;

**9. Create table - jams\_head\_raw\_data for third file ‘jams\_head.csv’**

-bash-4.1$ hdfs dfs -put jams\_head\_raw\_data.csv ./class\_project/jams

-bash-4.1$ hdfs dfs -chmod 777 ./class\_project/jams

-bash-4.1$ hdfs dfs -chmod 777 ./class\_project/jams/\*

CREATE EXTERNAL TABLE IF NOT EXISTS jams\_head\_raw\_data (pid INT,id INT,uuid STRING,pub\_millis STRING,pub\_utc\_date STRING,start\_node STRING,road\_type STRING,street SMALLINT,city STRING,state STRING,country STRING,delay SMALLINT,speed FLOAT,speed\_kmh FLOAT,length SMALLINT,turn\_type STRING,level SMALLINT,blocking\_alert\_id STRING,location\_x1 STRING,location\_y1 STRING,location\_x2 STRING,location\_y2 STRING)

ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

STORED AS TEXTFILE LOCATION '/user/rjoshi5/class\_project/jams';

**10. Take output of cleaned tables and rename them as per proper naming convention**

-bash-4.1$ hdfs dfs -get /user/rjoshi5/class\_project/irregularities/clean\_irregularities/\*

-bash-4.1$ mv 000000\_0 irregularities\_head\_clean.csv

-bash-4.1$ hdfs dfs -get /user/rjoshi5/class\_project/alerts/clean\_alerts/000000\_0

-bash-4.1$ mv 000000\_0 alerts\_head\_clean.csv

-bash-4.1$ ls -l

-bash-4.1$ chmod 744 \*