Part 3: Data Munging and Injection

Check incomplete files

```
$echo $'#!/bin/bash
files=/opt/dataset/twmask/*.csv
for f in $files;
do
  num=$(cat $f | cut -d',' -f1 | tail -n +2 | wc -l)
  if [[ $num == 0 ]]
  then
    echo "$f"
  fi;
done;' > num.sh

$chmod +x num.sh
$./num.sh > zero.txt
```

Remove incomplete files

```
$mkdir /tmp/twmask
$cp /opt/dataset/twmask/*.csv /tmp/twmask
$ls /tmp/twmask | wc -l
#5582

$nano zero.txt
^\ /opt/dataset /tmp

$echo $'#!/bin/bash
files=/tmp/twmask/*.csv
for f in $files;
do
  if [[ "$f" =~ $(echo ^\($(paste -sd'|' /home/bigred/zero.txt)\)
$) ]]; then
    rm $f
  fi;
done;' > rmzero.sh

$chmod +x rmzero.sh
$./rmzero.sh
$ls /tmp/twmask | wc -l
#5396
```

Remove header

```
##### one way
$echo $'#!/bin/bash
files=/tmp/twmask/*.csv
```

```
for f in $files;
do
  tail -n +2 $f > /tmp/123.csv
  mv /tmp/123.csv $f
done;' > rmheader.sh

##### another way
$echo $'#!/bin/bash
mkdir /tmp/twmask2
files=/tmp/twmask/*.csv
for f in $files;
do
  name=$(echo $f | cut -d"/" -f4)
  tail -n +2 $f > /tmp/twmask2/$name
done;' > rmheader2.sh

$chmod +x rmheader.sh
$./rmheader.sh


### Create month variable

$sudo apt-get install tofrodos

$echo $'#!/bin/bash
files=/tmp/twmask/*.csv
for f in $files;
do
  fromdos $f
  awk -F, -v OFS=, 'NR>0{$8=substr($7,7,1)}1' $f > /tmp/123.csv
  mv /tmp/123.csv $f
done;' > createmonth.sh

$chmod +x createmonth.sh
$./createmonth.sh


### Upload data to hdfs

$hdfs dfs -mkdir -p /raw/twmask
$hdfs dfs -put /tmp/twmask/*.csv /raw/twmask
$hdfs dfs -ls /raw/twmask | wc -l


### Create daily_mask table

$echo $'CREATE DATABASE IF NOT EXISTS twmask;
USE twmask;
DROP TABLE IF EXISTS daily_mask;
CREATE EXTERNAL TABLE daily_mask (
```

```
  code string,
  name string,
  address string,
  tel string,
  adult INT,
  child INT,
  dt string,
  month string
)
ROW FORMAT DELIMITED FIELDS TERMINATED BY \',\'
STORED AS TEXTFILE LOCATION \'/raw/twmask\'; ' > twmask.hsql

$hive -S -f twmask.hsql 2>/dev/null
$hive -S -e 'show databases' 2>/dev/null
$hive -S -e 'show tables in twmask' 2>/dev/null
$hive -S -e 'describe database twmask' 2>/dev/null
#twmask          hdfs://nna:8020/user/bigred/hive/twmask.db
bigred  USER


### Create twmask_month table

$echo $'USE twmask;
DROP TABLE IF EXISTS twmask_month;
CREATE EXTERNAL TABLE twmask_month (
  code string,
  name string,
  address string,
  tel string,
  adult INT,
  child INT,
  dt string
)
PARTITIONED BY (month string)
STORED AS TEXTFILE LOCATION \'/dataset/twmask_month\';' >
twmask_month.hsql

$hive -f twmask_month.hsql  2>/dev/null
$hive -e 'show tables in twmask' 2>/dev/null

$echo 'SET hive.exec.dynamic.partition.mode = nonstrict;
USE twmask;
INSERT OVERWRITE TABLE twmask_month
PARTITION (month)
SELECT code,name,address,tel,adult,child,dt,month
FROM daily_mask; ' > mm_insert.hsql

$hive -f mm_insert.hsql 2>/dev/null

$hdfs dfs -ls /dataset/twmask_month
```

```
### Create twmpar table

$echo $'USE twmask;
DROP TABLE IF EXISTS twmpar;
CREATE EXTERNAL TABLE twmpar (
  code string,
  name string,
  address string,
  tel string,
  adult INT,
  child INT,
  dt string
)
PARTITIONED BY (month string)
STORED AS Parquet LOCATION \'/dataset/twmpar\'; '> twmpar.hsql

$hive -f twmpar.hsql  2>/dev/null

$echo 'SET hive.exec.dynamic.partition.mode = nonstrict;
USE twmask;
INSERT OVERWRITE TABLE twmpar
PARTITION (month)
SELECT code,name,address,tel,adult,child,dt,month
FROM daily_mask; ' > mmm_insert.hsql

$hive -f mmm_insert.hsql 2>/dev/null

$hdfs dfs -ls /dataset/twmpar


### Check file size

$hdfs dfs -du -s -h /raw/twmask
$hdfs dfs -du -s -h /dataset/twmask_month
$hdfs dfs -du -s -h /dataset/twmpar


######################################################################

### Create twmcompact table

$nano twmcompact.pig

a = load '/raw/twmask' using PigStorage(',');
b = filter a by $0!='醫事機構代碼';
rmf /dataset/twmcompact;
store b into '/dataset/twmcompact' using PigStorage(',');
```

```
$pig twmcompact.pig
$hdfs dfs -ls /dataset/twmcompact

$echo $'USE twmask;
DROP TABLE IF EXISTS twmcompact;
CREATE EXTERNAL TABLE twmcompact (
  code string,
  name string,
  address string,
  tel string,
  adult INT,
  child INT,
  dt string
)
ROW FORMAT DELIMITED FIELDS TERMINATED BY \',\'
STORED AS TEXTFILE LOCATION \'/dataset/twmcompact\'; ' >
twmcompact.hsql

$hive -S -f twmcompact.hsql


### Create twmparquet table

$nano twmparquet.pig

a = load '/dataset/twmcompact' using PigStorage(',') as
(code:chararray, name:chararray, address: chararray, tel:chararray,
adult:int, child:int, dt:chararray);
rmf /dataset/twmparquet;
store a into '/dataset/twmparquet' using parquet.pig.ParquetStorer();

$pig twmparquet.pig
$hdfs dfs -ls /dataset/twmparquet

$nano twmparquet.hsql

USE twmask;
DROP TABLE IF EXISTS twmparquet;
CREATE EXTERNAL TABLE twmparquet (
  code string,
  name string,
  address string,
  tel string,
  adult INT,
  child INT,
  dt string
)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS Parquet LOCATION '/dataset/twmparquet';
```

```
$hive –S –f twmparquet.hsql
```