# [https://avatars2.githubusercontent.com/u/4156894?v=3&s=100](http://www.calstatela.edu/centers/hipic) CIS5560 Term Project Tutorial

#### Authors: Tony Lieu; Kevin Chea

#### Instructor: [Jongwook Woo](https://www.linkedin.com/in/jongwook-woo-7081a85)

#### Date: 12/07/2018

**Lab Tutorial**

Tony Lieu ([tlieu@calstatela.edu](mailto:tlieu@calstatela.edu))

Kevin Chea ([kchea2@calstatela.edu](mailto:kchea2@calstatela.edu))

12/07/2018

**Chicago Crimes Data Analysis using Hive**

**Objectives**

In this hands-on lab, you will learn how to:

* Import data into HDFS
* Create tables using Hive and Beeline
* SQL commands to perform the analysis.
* Export data on your computer
* Use data to create visualizations

**Platform Spec**

* Oracle Big Data Compute Edition: 5 nodes
* OCPUs: 10
* CPU Speed: Intel(R) Xeon(R) CPU E5-2699C v4 @ 2.20GHz
* Memory: 150GB
* Storage: 678 GB
* HDFS Capacity: 147 GB

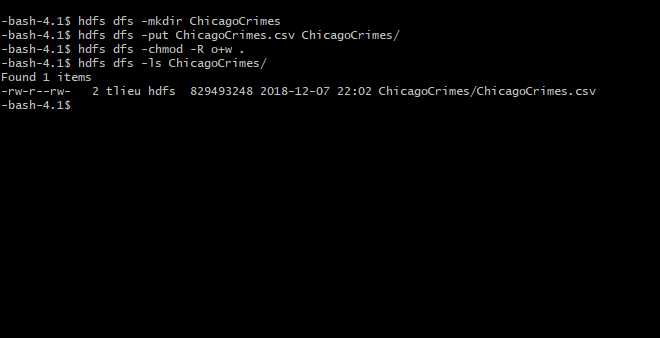
Step 1: Uploading the dataset into HDFS

This step is to download the dataset from the cityofchicago.org directly into the HDFS server.

1. $ ssh [username]@129.150.205.28
2. $ wget -O ChicagoCrimes.csv https://data.cityofchicago.org/api/views/ijzp-q8t2/rows.csv
3. $ hdfs dfs -mkdir ChicagoCrimes

$ hdfs dfs -put ChicagoCrimes.csv ChicagoCrimes/

$ hdfs dfs -chmod -R o+w .

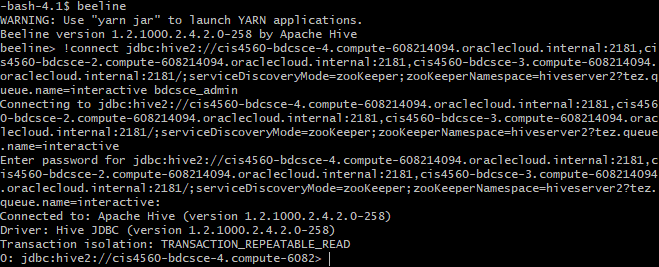


Step 2: Create Tables and Run Queries

This step is to import the dataset into Beeline and run queries to get information from the data.

* 1. $ beeline

!connect jdbc:hive2://cis4560-bdcsce-4.compute-608214094.oraclecloud.internal:2181,cis4560-bdcsce-2.compute-608214094.oraclecloud.internal:2181,cis4560-bdcsce-3.compute-608214094.oraclecloud.internal:2181/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=hiveserver2?tez.queue.name=interactive bdcsce\_admin



* 1. Use [username];
  2. Create a table of the dataset.

CREATE EXTERNAL TABLE IF NOT EXISTS chicagocrimes(

id BIGINT,

case\_number STRING,

`date` STRING,

block STRING,

iucr STRING,

primary\_type STRING,

description STRING,

location\_description STRING,

arrest STRING,

domestic STRING,

beat STRING,

district STRING,

ward STRING,

community\_area INT,

fbi\_code STRING,

x\_coordinate BIGINT,

y\_coordinate BIGINT,

year INT,

updated\_on STRING,

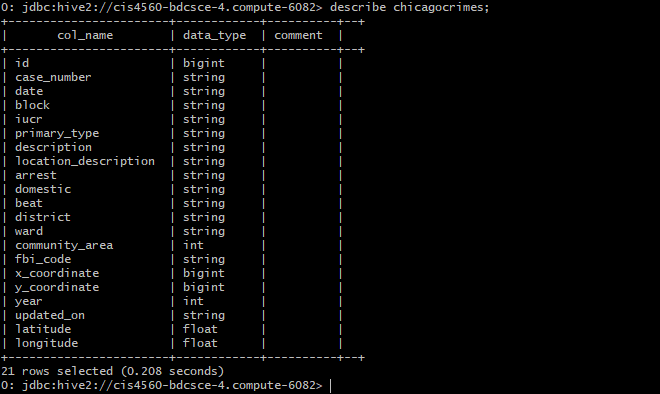
latitude FLOAT,

longitude FLOAT)

ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

STORED AS TEXTFILE LOCATION '/user/tlieu/ChicagoCrimes';

* 1. DESCRIBE chicagocrimes;



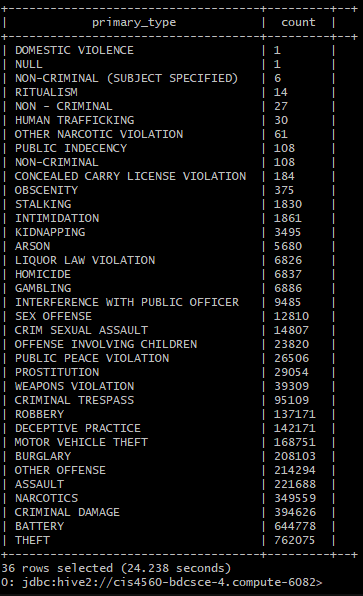
* 1. Run query to find count of each crime. (results may vary)

SELECT primary\_type, COUNT(\*) as `count`

FROM chicagocrimes

GROUP BY primary\_type

ORDER BY `count`;



* 1. Run query for each type of theft. (results may vary)

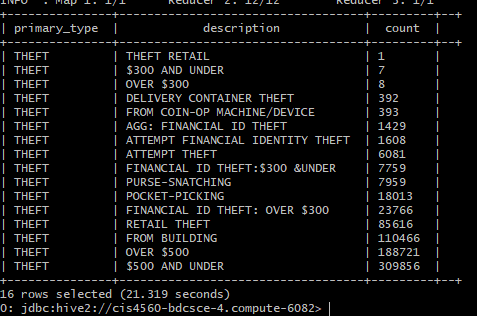
SELECT primary\_type, description, COUNT(\*) AS `count`

FROM chicagocrimes

WHERE primary\_type ='THEFT'

GROUP BY primary\_type, description

ORDER BY `count`;



* 1. Create a temporary table to change `date` format and data type.

CREATE TABLE time\_cc AS SELECT

id,

case\_number,

FROM\_UNIXTIME(UNIX\_TIMESTAMP(`date`, 'dd/mm/yyyy h:mm:ss a')) AS `date`,

block,

iucr,

primary\_type,

description,

location\_description,

arrest,

domestic,

beat,

district,

ward,

community\_area,

fbi\_code,

x\_coordinate,

y\_coordinate,

year,

updated\_on,

latitude,

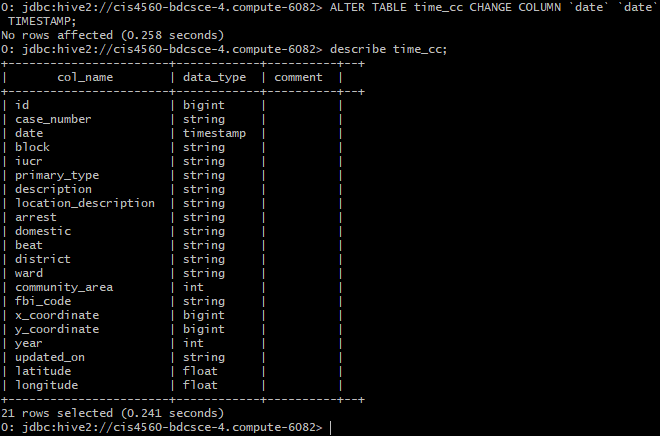
longitude

FROM chicagocrimes;

* 1. Change `date` from STRING to TIMESTAMP.

ALTER TABLE time\_cc CHANGE COLUMN `date` `date` TIMESTAMP;

DESCRIBE time\_cc;



* 1. Run query to find the count of thefts in each hour. (results may vary)

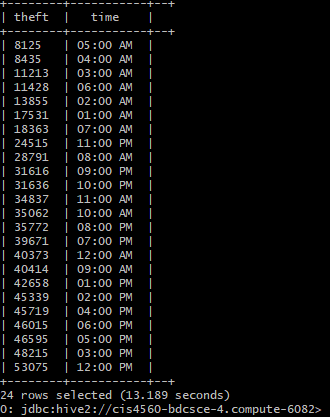
SELECT count(primary\_type) as theft, date\_format(`date`, "hh:00 aa") as time

FROM time\_cc

WHERE primary\_type ='THEFT'

GROUP BY date\_format(`date`, "hh:00 aa")

ORDER BY theft;



* 1. Run query to find the date with the most thefts commited. (Results my vary)

SELECT date\_format(`date`, 'YYYY-MM-DD') as `date`, COUNT(\*) AS `count`

FROM time\_cc

WHERE date\_format(`date`, "HH:mm") BETWEEN '09:00' and '21:00'

AND primary\_type ='THEFT'

GROUP BY date\_format(`date`, 'YYYY-MM-DD')

ORDER BY `count`;

* 1. Create a table that shows all theft commited on that specific day and in a time range.

CREATE TABLE time\_theft

ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

STORED AS TEXTFILE LOCATION '/user/tlieu/ChicagoCrimes'

AS SELECT

id,

case\_number,

`date`,

block,

iucr,

primary\_type AS crime,

description,

location\_description,

arrest,

domestic,

beat,

district,

ward,

community\_area,

fbi\_code,

x\_coordinate,

y\_coordinate,

year,

updated\_on,

latitude,

longitude

FROM time\_cc

WHERE date\_format(`date`, "yyyy-MM-dd HH:mm:ss") BETWEEN '2001-01-08 09:00:00' and '2001-01-08 21:00:00'

AND primary\_type ='THEFT';

Step 3: Create 3D Map with the new dataset

In This step we will export the new table to create a visualization of the data.

1. Export the dataset into the local cluster and create a csv file.

$ hdfs dfs -get /user/[username]/ChicagoCrimes/0000\*\_0

$ cat 0000\*\_0 > theftrange.csv

$ rm 0000\*\_0

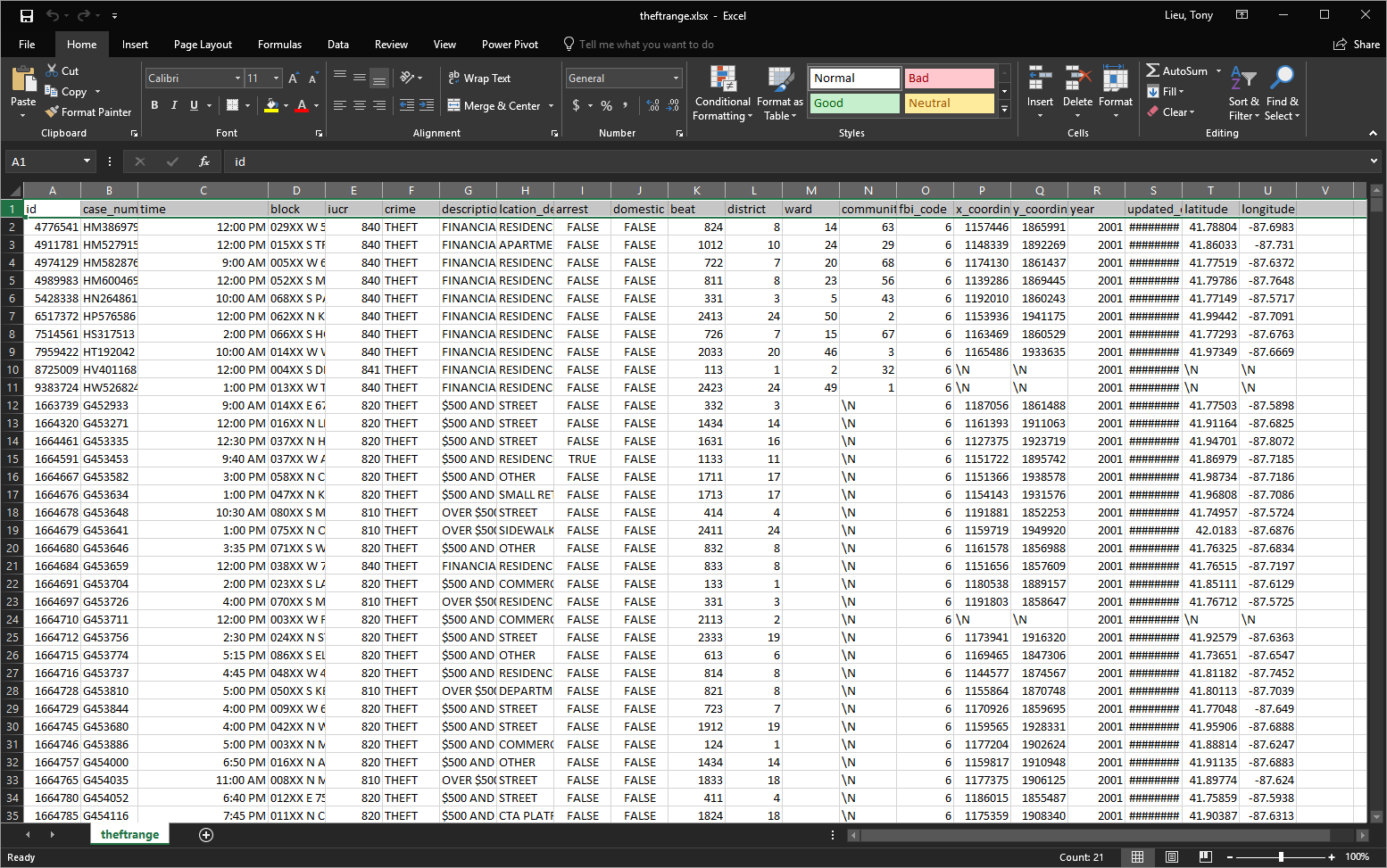
1. Logout of the cluster and download the csv file into your desktop.

$ logout

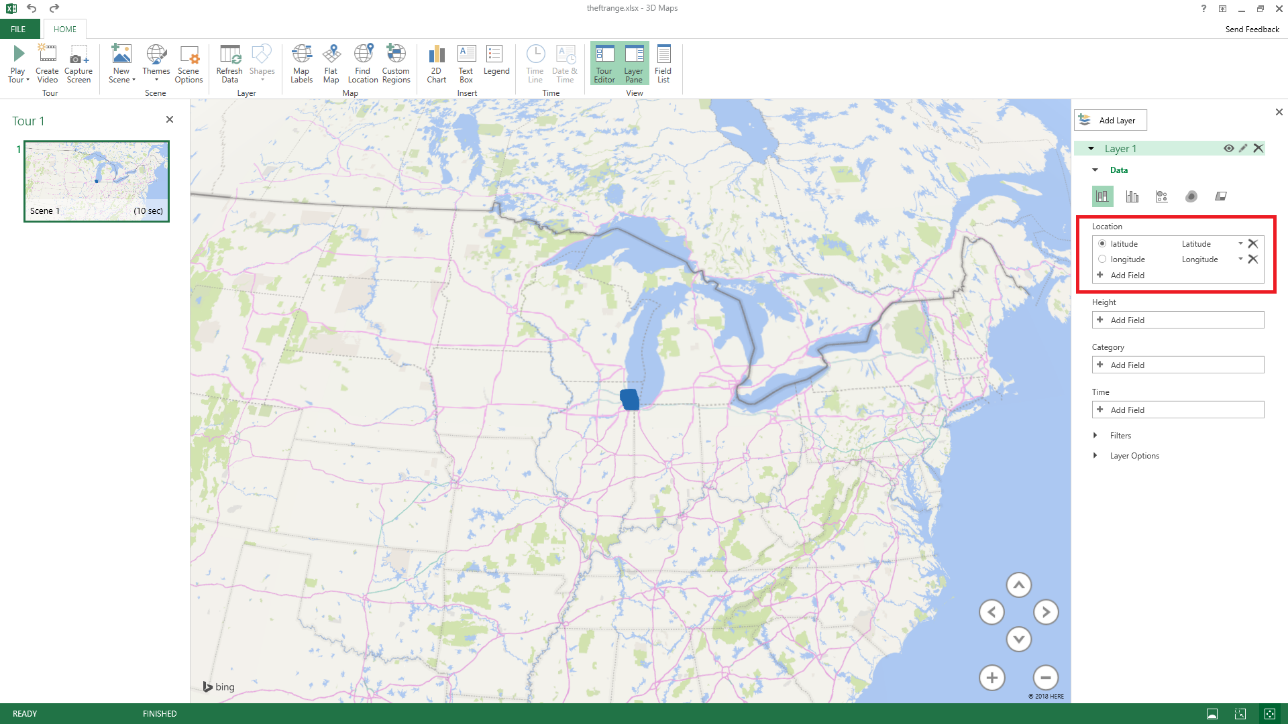
$ cd Desktop/

$ scp [username]@129.150.205.28:/home/[username]/theftrange.csv theftrange.csv

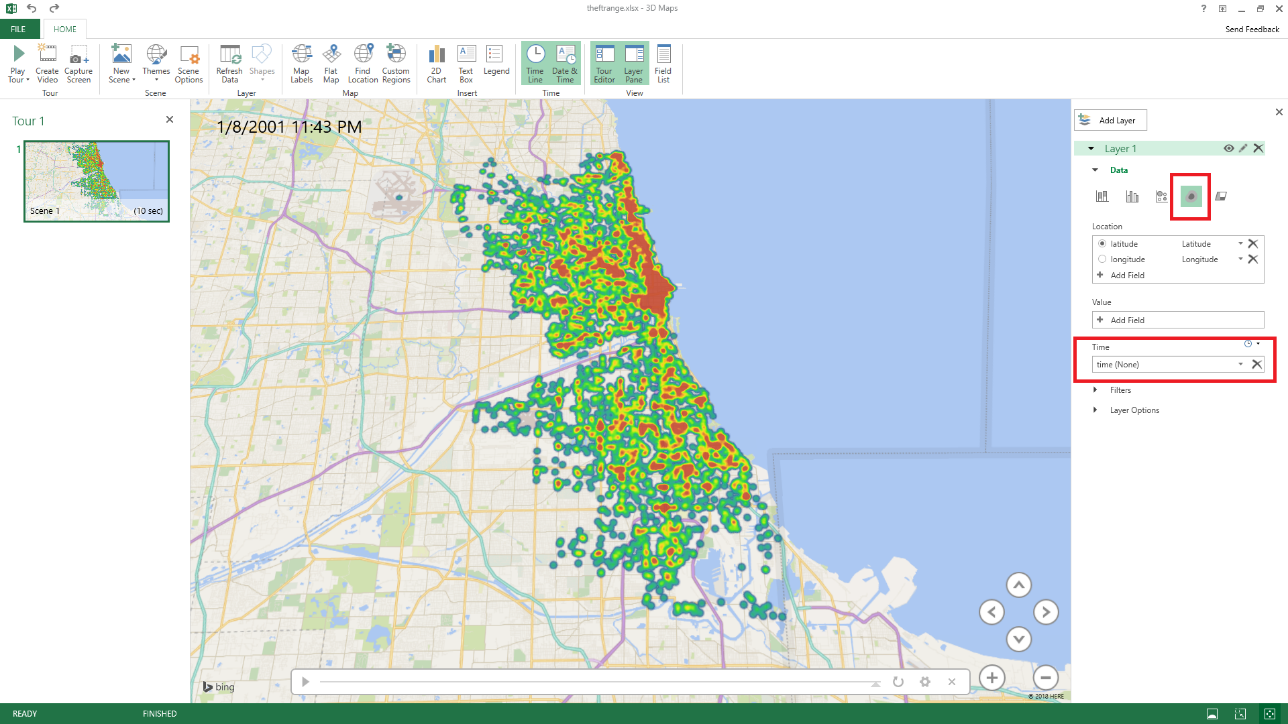
1. Open the csv file with excel, save it as a xlsx file, and create headers for each column.



1. Go to the insert tab and click on the 3D Map button to open a 3D representation of Earth. If the map does not have any blue dots then add the longitude and latitude columns on the right side of the map.



1. Next, add the time to the map and click on the heatmap option.



References

* 1. Data Source: <https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2>
  2. Github: <https://github.com/Tonylieu833/CIS-4560>
  3. Apache Hive: <https://cwiki.apache.org/confluence/display/Hive/Home>
  4. Orcale Cloud: <https://cloud.oracle.com/big-data-cloud>