# mod5_final_project

October 13, 2024

Assignment: Notebook for Graded Assessment

# 1 Introduction

Using this Python notebook you will:

1. Understand three Chicago datasets
2. Load the three datasets into three tables in a SQLIte database
3. Execute SQL queries to answer assignment questions

## 1.1 Understand the datasets

To complete the assignment problems in this notebook you will be using three datasets that are available on the city of Chicago's Data Portal:

1. Socioeconomic Indicators in Chicago
2. Chicago Public Schools
3. Chicago Crime Data

### 1.1.1 1. Socioeconomic Indicators in Chicago

This dataset contains a selection of six socioeconomic indicators of public health significance and a "hardship index," for each Chicago community area, for the years 2008 – 2012.

A detailed description of this dataset and the original dataset can be obtained from the Chicago Data Portal at:

https://data.cityofchicago.org/Health-Human-Services/Census-Data-Selected-socioeconomic-indicators-in-C/kn9c-c2s2

### 1.1.2 2. Chicago Public Schools

This dataset shows all school level performance data used to create CPS School Report Cards for the 2011-2012 school year. This dataset is provided by the city of Chicago's Data Portal.

A detailed description of this dataset and the original dataset can be obtained from the Chicago Data Portal at:

https://data.cityofchicago.org/Education/Chicago-Public-Schools-Progress-Report-Cards-2011-/9xs2-f89t

### 1.1.3 3. Chicago Crime Data

This dataset reflects reported incidents of crime (with the exception of murders where data exists for each victim) that occurred in the City of Chicago from 2001 to present, minus the most recent seven days.

A detailed description of this dataset and the original dataset can be obtained from the Chicago Data Portal at:

https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2

### 1.1.4 Download the datasets

This assignment requires you to have these three tables populated with a subset of the whole datasets.

In many cases the dataset to be analyzed is available as a .CSV (comma separated values) file, perhaps on the internet.

Use the links below to read the data files using the Pandas library.

- Chicago Census Data

https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-
DB0201EN-SkillsNetwork/labs/FinalModule_Coursera_V5/data/ChicagoCensusData.csv?utm_medium=Exinflu
SkillsNetwork-Channel-SkillsNetworkCoursesIBMDeveloperSkillsNetworkDB0201ENSkillsNetwork20127838-
2021-01-01

- Chicago Public Schools

https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-
DB0201EN-SkillsNetwork/labs/FinalModule_Coursera_V5/data/ChicagoPublicSchools.csv?utm_medium=Exin
SkillsNetwork-Channel-SkillsNetworkCoursesIBMDeveloperSkillsNetworkDB0201ENSkillsNetwork20127838-
2021-01-01

- Chicago Crime Data

https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-
DB0201EN-SkillsNetwork/labs/FinalModule_Coursera_V5/data/ChicagoCrimeData.csv?utm_medium=Exinflu
SkillsNetwork-Channel-SkillsNetworkCoursesIBMDeveloperSkillsNetworkDB0201ENSkillsNetwork20127838-
2021-01-01

**NOTE:** Ensure you use the datasets available on the links above instead of directly from the Chicago Data Portal. The versions linked here are subsets of the original datasets and have some of the column names modified to be more database friendly which will make it easier to complete this assignment.

### 1.1.5 Store the datasets in database tables

To analyze the data using SQL, it first needs to be loaded into SQLite DB. We will create three tables in as under:

1. **CENSUS_DATA**
2. **CHICAGO_PUBLIC_SCHOOLS**
3. **CHICAGO_CRIME_DATA**

Load the pandas and sqlite3 libraries and establish a connection to FinalDB.db

[1]:

```
!pip install pandas --quiet
!pip install --user ipython-sql --quiet

import sqlite3
import pandas as pd

conn = sqlite3.connect("FinalDB.db")
```

Load the SQL magic module

[2]:

```
%load_ext sql
```

Use Pandas to load the data available in the links above to dataframes. Use these dataframes to load data on to the database FinalDB.db as required tables.

[3]:

```
censusData_URL = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.
 ↪cloud/IBMDeveloperSkillsNetwork-DB0201EN-SkillsNetwork/labs/
 ↪FinalModule_Coursera_V5/data/ChicagoCensusData.csv?
 ↪utm_medium=Exinfluencer&utm_source=Exinfluencer&utm_content=000026UJ&utm_term=10006555&utm_
chiPubEdu_URL = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.
 ↪cloud/IBMDeveloperSkillsNetwork-DB0201EN-SkillsNetwork/labs/
 ↪FinalModule_Coursera_V5/data/ChicagoPublicSchools.csv?
 ↪utm_medium=Exinfluencer&utm_source=Exinfluencer&utm_content=000026UJ&utm_term=10006555&utm_
chiCrime_URL = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.
 ↪cloud/IBMDeveloperSkillsNetwork-DB0201EN-SkillsNetwork/labs/
 ↪FinalModule_Coursera_V5/data/ChicagoCrimeData.csv?
 ↪utm_medium=Exinfluencer&utm_source=Exinfluencer&utm_content=000026UJ&utm_term=10006555&utm_

censusDF = pd.read_csv(censusData_URL)
eduDF = pd.read_csv(chiPubEdu_URL)
crimeDF = pd.read_csv(chiCrime_URL)

censusDF.to_sql('censusSQL', conn, if_exists='replace')
eduDF.to_sql('eduSQL', conn, if_exists='replace')
crimeDF.to_sql('crimeSQL', conn, if_exists='replace')
```

[3]:

```
533
```

Establish a connection between SQL magic module and the database FinalDB.db

[4]:

```
%sql sqlite:///FinalDB.db
```

You can now proceed to the the following questions. Please note that a graded assignment will follow this lab and there will be a question on each of the problems stated below. It can be from the answer you received or the code you write for this problem. Therefore, please keep a note of both your codes as well as the response you generate.

## 1.2 Problems

Now write and execute SQL queries to solve assignment problems

### 1.2.1 Problem 1

**Find the total number of crimes recorded in the CRIME table.**

```
[5]: %sql select COUNT(CASE_NUMBER) from crimeSQL
```

```
 * sqlite:///FinalDB.db
Done.
```

```
[5]: [(533,)]
```

### 1.2.2 Problem 2

**List community area names and numbers with per capita income less than 11000.**

```
[6]: %sql select COMMUNITY_AREA_NAME, COMMUNITY_AREA_NUMBER from censusSQL where␣
      ↪PER_CAPITA_INCOME<11000
```

```
 * sqlite:///FinalDB.db
Done.
```

```
[6]: [('West Garfield Park', 26.0),
      ('South Lawndale', 30.0),
      ('Fuller Park', 37.0),
      ('Riverdale', 54.0)]
```

### 1.2.3 Problem 3

**List all case numbers for crimes involving minors?(children are not considered minors for the purposes of crime analysis)**

```
[7]: %sql select CASE_NUMBER, DESCRIPTION from crimeSQL where DESCRIPTION like␣
      ↪'%MINOR%'
```

```
 * sqlite:///FinalDB.db
Done.
```

```
[7]: [('HL266884', 'SELL/GIVE/DEL LIQUOR TO MINOR'),
      ('HK238408', 'ILLEGAL CONSUMPTION BY MINOR')]
```

### 1.2.4 Problem 4

**List all kidnapping crimes involving a child?**

```
[8]: %sql select CASE_NUMBER, PRIMARY_TYPE, DESCRIPTION from crimeSQL where␣
      ↪PRIMARY_TYPE="KIDNAPPING" and DESCRIPTION like '%child%'
```

```
 * sqlite:///FinalDB.db
Done.
```

```
[8]: [('HN144152', 'KIDNAPPING', 'CHILD ABDUCTION/STRANGER')]
```

### 1.2.5 Problem 5

**List the kind of crimes that were recorded at schools. (No repetitions)**

```
[9]: %sql select distinct PRIMARY_TYPE from crimeSQL where LOCATION_DESCRIPTION like␣
      ↪'%SCHOOL%'
```

```
 * sqlite:///FinalDB.db
Done.
```

```
[9]: [('BATTERY',),
      ('CRIMINAL DAMAGE',),
      ('NARCOTICS',),
      ('ASSAULT',),
      ('CRIMINAL TRESPASS',),
      ('PUBLIC PEACE VIOLATION',)]
```

### 1.2.6 Problem 6

**List the type of schools along with the average safety score for each type.**

```
[10]: %sql select "Elementary, Middle, or High School", AVG(SAFETY_SCORE) from eduSQL␣
       ↪group by "Elementary, Middle, or High School"
```

```
 * sqlite:///FinalDB.db
Done.
```

```
[10]: [('ES', 49.52038369304557), ('HS', 49.62352941176471), ('MS', 48.0)]
```

### 1.2.7 Problem 7

**List 5 community areas with highest % of households below poverty line**

```
[11]: %sql select COMMUNITY_AREA_NAME, PERCENT_HOUSEHOLDS_BELOW_POVERTY from␣
       ↪censusSQL order by PERCENT_HOUSEHOLDS_BELOW_POVERTY desc limit 5
```

```
 * sqlite:///FinalDB.db
Done.
```

```
[11]: [('Riverdale', 56.5),
       ('Fuller Park', 51.2),
       ('Englewood', 46.6),
       ('North Lawndale', 43.1),
       ('East Garfield Park', 42.4)]
```

### 1.2.8 Problem 8

**Which community area is most crime prone? Display the coumminty area number only.**

```
[12]: %sql select COMMUNITY_AREA_NUMBER from crimeSQL group by COMMUNITY_AREA_NUMBER␣
       ↪order by count(*) desc limit 1
```

```
 * sqlite:///FinalDB.db
Done.
```

```
[12]: [(25.0,)]
```

Double-click **here** for a hint

### 1.2.9 Problem 9

**Use a sub-query to find the name of the community area with highest hardship index**

```
[13]: %sql select COMMUNITY_AREA_NAME, HARDSHIP_INDEX from censusSQL where␣
       ↪HARDSHIP_INDEX in (select max(HARDSHIP_INDEX) from censusSQL)
```

```
 * sqlite:///FinalDB.db
Done.
```

```
[13]: [('Riverdale', 98.0)]
```

### 1.2.10 Problem 10

**Use a sub-query to determine the Community Area Name with most number of crimes?**

```
[14]: %sql select COMMUNITY_AREA_NAME from censusSQL where COMMUNITY_AREA_NUMBER =␣
       ↪(select COMMUNITY_AREA_NUMBER from crimeSQL group by COMMUNITY_AREA_NUMBER␣
       ↪order by count(*) desc)
```

```
 * sqlite:///FinalDB.db
Done.
```

```
[14]: [('Austin',)]
```

## 1.3 Author(s)

Hima Vasudevan

Rav Ahuja

Ramesh Sannreddy

## 1.4 Contribtuor(s)

Malika Singla

Abhishek Gagneja

##