

mod5_final_project

February 12, 2025

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=ExinfluSkillsNetwork-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=ExinfluSkillsNetwork-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=ExinfluSkillsNetwork-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.

Execute the below code cell to avoid prettytable default error.

```
[19]: import csv, sqlite3

      con = sqlite3.connect("FinalDB.db")
      cur = con.cursor()
```

```
[20]: !pip install ipython-sql prettytable
```

```
import prettytable
```

```
prettytable.DEFAULT = 'DEFAULT'
```

Requirement already satisfied: ipython-sql in /opt/conda/lib/python3.12/site-packages (0.5.0)

Requirement already satisfied: prettytable in /opt/conda/lib/python3.12/site-packages (3.14.0)

Requirement already satisfied: ipython in /opt/conda/lib/python3.12/site-packages (from ipython-sql) (8.31.0)

Requirement already satisfied: sqlalchemy>=2.0 in /opt/conda/lib/python3.12/site-packages (from ipython-sql) (2.0.37)

Requirement already satisfied: sqlparse in /opt/conda/lib/python3.12/site-packages (from ipython-sql) (0.5.3)

Requirement already satisfied: six in /opt/conda/lib/python3.12/site-packages (from ipython-sql) (1.17.0)

Requirement already satisfied: ipython-genutils in /opt/conda/lib/python3.12/site-packages (from ipython-sql) (0.2.0)

Requirement already satisfied: wcwidth in /opt/conda/lib/python3.12/site-packages (from prettytable) (0.2.13)

Requirement already satisfied: greenlet!=0.4.17 in /opt/conda/lib/python3.12/site-packages (from sqlalchemy>=2.0->ipython-sql) (3.1.1)

Requirement already satisfied: typing-extensions>=4.6.0 in /opt/conda/lib/python3.12/site-packages (from sqlalchemy>=2.0->ipython-sql) (4.12.2)

Requirement already satisfied: decorator in /opt/conda/lib/python3.12/site-packages (from ipython->ipython-sql) (5.1.1)

Requirement already satisfied: jedi>=0.16 in /opt/conda/lib/python3.12/site-packages (from ipython->ipython-sql) (0.19.2)

Requirement already satisfied: matplotlib-inline in /opt/conda/lib/python3.12/site-packages (from ipython->ipython-sql) (0.1.7)

Requirement already satisfied: pexpect>4.3 in /opt/conda/lib/python3.12/site-packages (from ipython->ipython-sql) (4.9.0)

Requirement already satisfied: prompt_toolkit<3.1.0,>=3.0.41 in /opt/conda/lib/python3.12/site-packages (from ipython->ipython-sql) (3.0.50)

Requirement already satisfied: pygments>=2.4.0 in /opt/conda/lib/python3.12/site-packages (from ipython->ipython-sql) (2.19.1)

Requirement already satisfied: stack_data in /opt/conda/lib/python3.12/site-packages (from ipython->ipython-sql) (0.6.3)

Requirement already satisfied: traitlets>=5.13.0 in /opt/conda/lib/python3.12/site-packages (from ipython->ipython-sql) (5.14.3)

Requirement already satisfied: parso<0.9.0,>=0.8.4 in /opt/conda/lib/python3.12/site-packages (from jedi>=0.16->ipython->ipython-sql) (0.8.4)

Requirement already satisfied: ptyprocess>=0.5 in

```

/opt/conda/lib/python3.12/site-packages (from pexpect>4.3->ipython->ipython-sql)
(0.7.0)
Requirement already satisfied: executing>=1.2.0 in
/opt/conda/lib/python3.12/site-packages (from stack_data->ipython->ipython-sql)
(2.1.0)
Requirement already satisfied: asttokens>=2.1.0 in
/opt/conda/lib/python3.12/site-packages (from stack_data->ipython->ipython-sql)
(3.0.0)
Requirement already satisfied: pure_eval in /opt/conda/lib/python3.12/site-
packages (from stack_data->ipython->ipython-sql) (0.2.3)

```

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

```

[21]: !pip install ipython-sql
      !pip install pandas
      %load_ext sql

```

```

Requirement already satisfied: ipython-sql in /opt/conda/lib/python3.12/site-
packages (0.5.0)
Requirement already satisfied: prettytable in /opt/conda/lib/python3.12/site-
packages (from ipython-sql) (3.14.0)
Requirement already satisfied: ipython in /opt/conda/lib/python3.12/site-
packages (from ipython-sql) (8.31.0)
Requirement already satisfied: sqlalchemy>=2.0 in
/opt/conda/lib/python3.12/site-packages (from ipython-sql) (2.0.37)
Requirement already satisfied: sqlparse in /opt/conda/lib/python3.12/site-
packages (from ipython-sql) (0.5.3)
Requirement already satisfied: six in /opt/conda/lib/python3.12/site-packages
(from ipython-sql) (1.17.0)
Requirement already satisfied: ipython-genutils in
/opt/conda/lib/python3.12/site-packages (from ipython-sql) (0.2.0)
Requirement already satisfied: greenlet!=0.4.17 in
/opt/conda/lib/python3.12/site-packages (from sqlalchemy>=2.0->ipython-sql)
(3.1.1)
Requirement already satisfied: typing-extensions>=4.6.0 in
/opt/conda/lib/python3.12/site-packages (from sqlalchemy>=2.0->ipython-sql)
(4.12.2)
Requirement already satisfied: decorator in /opt/conda/lib/python3.12/site-
packages (from ipython->ipython-sql) (5.1.1)
Requirement already satisfied: jedi>=0.16 in /opt/conda/lib/python3.12/site-

```

```

packages (from ipython->ipython-sql) (0.19.2)
Requirement already satisfied: matplotlib-inline in
/opt/conda/lib/python3.12/site-packages (from ipython->ipython-sql) (0.1.7)
Requirement already satisfied: pexpect>4.3 in /opt/conda/lib/python3.12/site-
packages (from ipython->ipython-sql) (4.9.0)
Requirement already satisfied: prompt_toolkit<3.1.0,>=3.0.41 in
/opt/conda/lib/python3.12/site-packages (from ipython->ipython-sql) (3.0.50)
Requirement already satisfied: pygments>=2.4.0 in
/opt/conda/lib/python3.12/site-packages (from ipython->ipython-sql) (2.19.1)
Requirement already satisfied: stack_data in /opt/conda/lib/python3.12/site-
packages (from ipython->ipython-sql) (0.6.3)
Requirement already satisfied: traitlets>=5.13.0 in
/opt/conda/lib/python3.12/site-packages (from ipython->ipython-sql) (5.14.3)
Requirement already satisfied: wcwidth in /opt/conda/lib/python3.12/site-
packages (from prettytable->ipython-sql) (0.2.13)
Requirement already satisfied: parso<0.9.0,>=0.8.4 in
/opt/conda/lib/python3.12/site-packages (from jedi>=0.16->ipython->ipython-sql)
(0.8.4)
Requirement already satisfied: ptyprocess>=0.5 in
/opt/conda/lib/python3.12/site-packages (from pexpect>4.3->ipython->ipython-sql)
(0.7.0)
Requirement already satisfied: executing>=1.2.0 in
/opt/conda/lib/python3.12/site-packages (from stack_data->ipython->ipython-sql)
(2.1.0)
Requirement already satisfied: asttokens>=2.1.0 in
/opt/conda/lib/python3.12/site-packages (from stack_data->ipython->ipython-sql)
(3.0.0)
Requirement already satisfied: pure_eval in /opt/conda/lib/python3.12/site-
packages (from stack_data->ipython->ipython-sql) (0.2.3)
Requirement already satisfied: pandas in /opt/conda/lib/python3.12/site-packages
(2.2.3)
Requirement already satisfied: numpy>=1.26.0 in /opt/conda/lib/python3.12/site-
packages (from pandas) (2.2.2)
Requirement already satisfied: python-dateutil>=2.8.2 in
/opt/conda/lib/python3.12/site-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /opt/conda/lib/python3.12/site-
packages (from pandas) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in /opt/conda/lib/python3.12/site-
packages (from pandas) (2025.1)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.12/site-
packages (from python-dateutil>=2.8.2->pandas) (1.17.0)
The sql extension is already loaded. To reload it, use:
    %reload_ext sql

```

```
[22]: %sql sqlite:///FinalDB.db
```

Load the SQL magic module

```
[23]: %reload_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.

```
[24]: import pandas

df = pandas.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.
    ↪appdomain.cloud/IBMDeveloperSkillsNetwork-DB0201EN-SkillsNetwork/labs/
    ↪FinalModule_Coursera_V5/data/ChicagoPublicSchools.csv")
df.to_sql("CHICAGO_PUBLIC_SCHOOLS_DATA", con, if_exists='replace', index=False,
    ↪method="multi")

import pandas

census_df = pandas.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.
    ↪appdomain.cloud/IBMDeveloperSkillsNetwork-DB0201EN-SkillsNetwork/labs/
    ↪FinalModule_Coursera_V5/data/ChicagoCensusData.csv")
schools_df = pandas.read_csv("https://cf-courses-data.s3.us.
    ↪cloud-object-storage.appdomain.cloud/
    ↪IBMDeveloperSkillsNetwork-DB0201EN-SkillsNetwork/labs/
    ↪FinalModule_Coursera_V5/data/ChicagoPublicSchools.csv")
crime_df = pandas.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.
    ↪appdomain.cloud/IBMDeveloperSkillsNetwork-DB0201EN-SkillsNetwork/labs/
    ↪FinalModule_Coursera_V5/data/ChicagoCrimeData.csv")

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

census_df.to_sql("ChicagoCensusData", conn, if_exists="replace", index=False,
    ↪method="multi")
schools_df.to_sql("ChicagoPublicSchools", conn, if_exists="replace",
    ↪index=False, method="multi")
crime_df.to_sql("ChicagoCrimeData", conn, if_exists="replace", index=False,
    ↪method="multi")

conn.close()
```

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

```
[25]: %sql SELECT name FROM sqlite_master WHERE type='table'
```

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

```
[25]: [('CENSUS_DATA',),
      ('CHICAGO_PUBLIC_SCHOOLS',),
      ('CHICAGO_CRIME_DATA',),
      ('CHICAGO_PUBLIC_SCHOOLS_DATA',),
```

```
('ChicagoCensusData',),  
( 'ChicagoPublicSchools',),  
( 'ChicagoCrimeData',)]
```

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.

```
[26]: %%sql  
SELECT COUNT(*) AS Total_Crimes FROM ChicagoCrimeData;  
  
* sqlite:///FinalDB.db  
Done.
```

```
[26]: [(533,)]
```

1.2.2 Problem 2

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

```
[27]: %%sql  
SELECT "Community Area Name", "Community Area Number", "Per Capita Income "  
FROM ChicagoCensusData  
WHERE "Per Capita Income " < 11000;  
  
* sqlite:///FinalDB.db  
Done.
```

```
[27]: []
```

1.2.3 Problem 3

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

```
[28]: %%sql  
SELECT "Case Number"  
FROM ChicagoCrimeData  
WHERE "Primary Type" LIKE '%MINOR%';  
  
* sqlite:///FinalDB.db  
Done.
```

```
[28]: []
```

1.2.4 Problem 4

List all kidnapping crimes involving a child?

```
[29]: %sql
SELECT "Case Number", "Primary Type", "Description"
FROM ChicagoCrimeData
WHERE "Primary Type" = 'KIDNAPPING' AND "Description" LIKE '%CHILD%';

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

[29]: []

1.2.5 Problem 5

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

```
[30]: %sql
SELECT DISTINCT "Primary Type"
FROM ChicagoCrimeData
WHERE "Location Description" LIKE '%SCHOOL%';

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

[30]: []

1.2.6 Problem 6

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

```
[31]: %sql
SELECT "School Type", AVG("Safety Score") AS Average_Safety_Score
FROM ChicagoPublicSchools
GROUP BY "School Type";

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

[31]: [('School Type', 0.0)]

1.2.7 Problem 7

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

```
[32]: %sql
SELECT "Community Area Name", "Community Area Number", "Percent of Households_
↳Below Poverty Line"
FROM ChicagoCensusData
ORDER BY "Percent of Households Below Poverty Line" DESC
LIMIT 5;

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



```
[32]: [('Community Area Name', 'Community Area Number', 'Percent of Households Below
Poverty Line'),
      ('Community Area Name', 'Community Area Number', 'Percent of Households Below
Poverty Line'),
      ('Community Area Name', 'Community Area Number', 'Percent of Households Below
Poverty Line'),
      ('Community Area Name', 'Community Area Number', 'Percent of Households Below
Poverty Line'),
      ('Community Area Name', 'Community Area Number', 'Percent of Households Below
Poverty Line')]
```

1.2.8 Problem 8

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

```
[33]: %%sql
SELECT "Community Area Number"
FROM ChicagoCrimeData
GROUP BY "Community Area Number"
ORDER BY COUNT(*) DESC
LIMIT 1;
```

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

```
[33]: [('Community Area Number',)]
```

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

```
[34]: %%sql
SELECT "Community Area Name"
FROM ChicagoCensusData
WHERE "Hardship Index" = (
    SELECT MAX("Hardship Index")
    FROM ChicagoCensusData
);
```

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

```
[34]: [('Community Area Name',),
      ('Community Area Name',),
      ('Community Area Name',),
      ('Community Area Name',),
      ('Community Area Name',),
      ('Community Area Name',)]
```

[illegible]

[illegible]

1.2.10 Problem 10

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

```
[35]: %%sql
SELECT "Community Area Name"
FROM ChicagoCensusData
WHERE "Community Area Number" = (
    SELECT "Community Area Number"
    FROM ChicagoCrimeData
    GROUP BY "Community Area Number"
    ORDER BY COUNT(*) DESC
    LIMIT 1
);
```

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

```
[35]: [('Community Area Name',),
      ('Community Area Name',),
      ('Community Area Name',),
      ('Community Area Name',),
```

[illegible]

[illegible]

1.3 Author(s)

Hima Vasudevan

Rav Ahuja

Ramesh Sannreddy

1.4 Contributor(s)

Malika Singla

Abhishek Gagneja

##

© IBM Corporation 2023. All rights reserved.