FinalSQL1

August 8, 2025

[10]:
```
!pip install pandas
!pip install ipython-sql prettytable

import prettytable

prettytable.DEFAULT = 'DEFAULT'
```

```
Collecting pandas
  Downloading
pandas-2.3.1-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata
(91 kB)
Collecting numpy>=1.26.0 (from pandas)
  Downloading
numpy-2.3.2-cp312-cp312-manylinux_2_27_x86_64.manylinux_2_28_x86_64.whl.metadata
(62 kB)
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)
Collecting tzdata>=2022.7 (from pandas)
  Downloading tzdata-2025.2-py2.py3-none-any.whl.metadata (1.4 kB)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.12/site-
packages (from python-dateutil>=2.8.2->pandas) (1.17.0)
Downloading
pandas-2.3.1-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (12.0
MB)
                           12.0/12.0 MB
148.5 MB/s eta 0:00:00
Downloading
numpy-2.3.2-cp312-cp312-manylinux_2_27_x86_64.manylinux_2_28_x86_64.whl (16.6
MB)
                           16.6/16.6 MB
173.4 MB/s eta 0:00:00
Downloading tzdata-2025.2-py2.py3-none-any.whl (347 kB)
Installing collected packages: tzdata, numpy, pandas
Successfully installed numpy-2.3.2 pandas-2.3.1 tzdata-2025.2
Collecting ipython-sql
  Downloading ipython_sql-0.5.0-py3-none-any.whl.metadata (17 kB)
```

```
Collecting prettytable
  Downloading prettytable-3.16.0-py3-none-any.whl.metadata (33 kB)
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)
Collecting sqlparse (from ipython-sql)
  Downloading sqlparse-0.5.3-py3-none-any.whl.metadata (3.9 kB)
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)
Downloading ipython_sql-0.5.0-py3-none-any.whl (20 kB)
Downloading prettytable-3.16.0-py3-none-any.whl (33 kB)
Downloading sqlparse-0.5.3-py3-none-any.whl (44 kB)
Installing collected packages: sqlparse, prettytable, ipython-sql
Successfully installed ipython-sql-0.5.0 prettytable-3.16.0 sqlparse-0.5.3
```

[11]:
```
# Load the SQL magic extension
%load_ext sql

# Connect the %sql magic to the SQLite database (this is needed for %sql cells)
%sql sqlite:///FinalDB.db

# Import libraries
import pandas as pd
import sqlite3

# Regular Python SQLite connection (optional for pandas)
con = sqlite3.connect("FinalDB.db")
```

[12]:
```
# Load CSVs into DataFrames
df1 = pd.read_csv('https://cf-courses-data.s3.us.cloud-object-storage.appdomain.
 ↪cloud/IBMDeveloperSkillsNetwork-DB0201EN-SkillsNetwork/labs/
 ↪FinalModule_Coursera_V5/data/ChicagoCensusData.csv')
df2 = pd.read_csv('https://cf-courses-data.s3.us.cloud-object-storage.appdomain.
 ↪cloud/IBMDeveloperSkillsNetwork-DB0201EN-SkillsNetwork/labs/
 ↪FinalModule_Coursera_V5/data/ChicagoPublicSchools.csv')
df3 = pd.read_csv('https://cf-courses-data.s3.us.cloud-object-storage.appdomain.
 ↪cloud/IBMDeveloperSkillsNetwork-DB0201EN-SkillsNetwork/labs/
 ↪FinalModule_Coursera_V5/data/ChicagoCrimeData.csv')

# Save DataFrames to SQL tables
df1.to_sql('CENSUS_DATA', con, if_exists='replace', index=False)
df2.to_sql('CHICAGO_PUBLIC_SCHOOLS', con, if_exists='replace', index=False)
df3.to_sql('CHICAGO_CRIME_DATA', con, if_exists='replace', index=False)
```

[12]:
```
533
```

#P1 Find the total number of crimes recorded in the CRIME table.

[51]:
```
%sql SELECT COUNT(*) FROM CHICAGO_CRIME_DATA;
```

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

[51]:
```
[(533,)]
```

#P2List community area names and numbers with per capita income less than 11000

[54]:
```
%sql SELECT COMMUNITY_AREA_NAME, COMMUNITY_AREA_NUMBER FROM CENSUS_DATA WHERE␣
↪PER_CAPITA_INCOME < 11000;
```

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

[54]:
```
[('West Garfield Park', 26.0),
 ('South Lawndale', 30.0),
 ('Fuller Park', 37.0),
 ('Riverdale', 54.0)]
```

#P3 List all case numbers for crimes involving minors?(children are not considered minors for the purposes of crime analysis)

[50]:
```
%sql SELECT DISTINCT CASE_NUMBER FROM CHICAGO_CRIME_DATA WHERE DESCRIPTION LIKE␣
↪'%MINOR%';
```

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

[50]:
```
[('HL266884',), ('HK238408',)]
```

#Code Explanation #2. SELECT DISTINCT CASE_NUMBER: DISTINCT removes duplicates — so if the same CASE_NUMBER appears multiple times in the results, you only see it once. CASE_NUMBER is a column in your table — probably the unique ID for each reported crime.

P4: List all kidnapping crimes involving a child?

[16]:
```
%sql SELECT DISTINCT CASE_NUMBER, PRIMARY_TYPE FROM CHICAGO_CRIME_DATA WHERE␣
↪(DESCRIPTION LIKE '%CHILD%') AND (PRIMARY_TYPE LIKE '%KIDNAPPING%');
```

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

[16]:
```
[('HN144152', 'KIDNAPPING')]
```

[55]:
```
%sql SELECT * FROM CHICAGO_CRIME_DATA WHERE (DESCRIPTION LIKE '%CHILD%') AND␣
↪(PRIMARY_TYPE = 'KIDNAPPING');
```

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

[55]:
```
[(5276766, 'HN144152', '2007-01-26', '050XX W VAN BUREN ST', '1792',
'KIDNAPPING', 'CHILD ABDUCTION/STRANGER', 'STREET', 0, 0, 1533, 15, 29.0, 25.0,
'20', 1143050.0, 1897546.0, 2007, 41.87490841, -87.75024931, '(41.874908413,
-87.750249307)')]
```

P5: List the kind of crimes that were recorded at schools. (No repetitions)

[35]:
```
%sql SELECT DISTINCT PRIMARY_TYPE FROM CHICAGO_CRIME_DATA WHERE␣
↪(LOCATION_DESCRIPTION LIKE '%SCHOOL%');
```

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

[35]:
```
[('BATTERY',),
 ('CRIMINAL DAMAGE',),
 ('NARCOTICS',),
 ('ASSAULT',),
 ('CRIMINAL TRESPASS',),
 ('PUBLIC PEACE VIOLATION',)]
```

P6: List the type of schools along with the average safety score for each type.

[37]:
```
%%sql SELECT "Elementary, Middle, or High School",
    AVG(SAFETY_SCORE) AS average_safety_score
FROM CHICAGO_PUBLIC_SCHOOLS
GROUP BY "Elementary, Middle, or High School";
```

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

[37]:
```
[('ES', 49.52038369304557), ('HS', 49.62352941176471), ('MS', 48.0)]
```

P7: List 5 community areas with highest % of households below poverty line

[39]:
```
%sql SELECT COMMUNITY_AREA_NAME, PERCENT_HOUSEHOLDS_BELOW_POVERTY FROM␣
↪CENSUS_DATA ORDER BY PERCENT_HOUSEHOLDS_BELOW_POVERTY DESC LIMIT 5;
```

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

[39]:
```
[('Riverdale', 56.5),
 ('Fuller Park', 51.2),
 ('Englewood', 46.6),
 ('North Lawndale', 43.1),
 ('East Garfield Park', 42.4)]
```

P8: Which community area is most crime prone? Display the coumminty area number only.

[46]:
```
%%sql
SELECT COMMUNITY_AREA_NUMBER
FROM CHICAGO_CRIME_DATA
GROUP BY COMMUNITY_AREA_NUMBER
ORDER BY COUNT(*) DESC
LIMIT 1;
```

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

[46]:
```
[(25.0,)]
```

P9: Use a sub-query to find the name of the community area with highest hardship index

[41]:
```
#Option 1:
%sql SELECT COMMUNITY_AREA_NAME FROM CENSUS_DATA WHERE HARDSHIP_INDEX = (␣
 ↪SELECT MAX(HARDSHIP_INDEX) from CENSUS_DATA);
```

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

[41]:
```
[('Riverdale',)]
```

[60]:
```
#Option 2:

%sql SELECT COMMUNITY_AREA_NAME FROM CENSUS_DATA WHERE HARDSHIP_INDEX IN␣
 ↪(SELECT MAX(HARDSHIP_INDEX) FROM CENSUS_DATA);
```

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

[60]:
```
[('Riverdale',)]
```

P10: Use a sub-query to determine the Community Area Name with most number of crimes?

[49]:
```
%sql SELECT COMMUNITY_AREA_NAME, COMMUNITY_AREA_NUMBER FROM CENSUS_DATA WHERE␣
 ↪COMMUNITY_AREA_NUMBER = (SELECT COMMUNITY_AREA_NUMBER FROM␣
 ↪CHICAGO_CRIME_DATA GROUP BY COMMUNITY_AREA_NUMBER ORDER BY COUNT(*) DESC␣
 ↪LIMIT 1);
```

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

[49]:
```
[('Austin', 25.0)]
```

[ ]:
```

```