



**Skills
Network**

Working with a real world data-set using SQL and Python

Estimated time needed: **30** minutes

Objectives

After completing this lab you will be able to:

- Understand the dataset for Chicago Public School level performance
- Store the dataset in SQLite database.
- Retrieve metadata about tables and columns and query data from mixed case columns
- Solve example problems to practice your SQL skills including using built-in database functions

Chicago Public Schools - Progress Report Cards (2011-2012)

The city of Chicago released a dataset showing all school level performance data used to create School Report Cards for the 2011-2012 school year. The dataset is available from the Chicago Data Portal: <https://data.cityofchicago.org/Education/Chicago-Public-Schools-Progress-Report-Cards-2011-/9xs2-f89t>

This dataset includes a large number of metrics. Start by familiarizing yourself with the types of metrics in the database:
<https://data.cityofchicago.org/api/assets/AAD41A13-BE8A-4E67-B1F5-86E711E09D5F?download=true>

NOTE:

Do not download the dataset directly from City of Chicago portal. Instead download a static copy which is a more database friendly version from this link.

Now review some of its contents.

```
In [1]: import csv, sqlite3  
  
con = sqlite3.connect("RealWorldData.db")  
cur = con.cursor()
```

```
In [2]: !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 124.9 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 147.6 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-sq
l) (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->ipyth
on->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->ipyth
on-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
```

```
In [3]: !pip install ipython-sql
%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) (3.16.0)
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)
```

In [4]: %load_ext sql

The sql extension is already loaded. To reload it, use:

```
%reload_ext sql
```

```
In [5]: %sql sqlite:///RealWorldData.db
```

```
In [6]: %sql SELECT name FROM sqlite_master WHERE type='table'
```

```
* sqlite:///RealWorldData.db
```

Done.

```
Out[6]: name
```

Store the dataset in a Table

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

To analyze the data using SQL, it first needs to be stored in the database.

We will first read the csv files from the given url into pandas dataframes

Next we will be using the df.to_sql() function to convert each csv file to a table in sqlite with the csv data loaded in it.

```
In [ ]:
```

Double-click **here** for the solution.

```
import pandas as pd
```

```
df = pandas.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDriverSkillsNetwork-DB0201EN-SkillsNetwork/labs/FinalModule_Coursera_V5/data/ChicagoPublicSchools.csv")
```

```
df.to_sql("CHICAGO_PUBLIC_SCHOOLS_DATA", con, if_exists='replace', index=False, method="multi")
```

Query the database system catalog to retrieve table metadata

You can verify that the table creation was successful by retrieving the list of all tables in your schema and checking whether the SCHOOLS table was created

```
In [7]: # type in your query to retrieve list of all tables in the database
%sql SELECT name FROM sqlite_master WHERE type='table'
```

```
# how know type = table  
* sqlite:///RealWorldData.db  
Done.
```

Out[7]: [name](#)

Double-click **here** for a hint

Double-click **here** for the solution.

```
In [8]: %load_ext sql  
%sql sqlite:///RealWorldData.db  
%sql SELECT name FROM sqlite_master WHERE type='table'
```

The sql extension is already loaded. To reload it, use:

```
%reload_ext sql  
(sqlite3.OperationalError) near "orلدData": syntax error  
[SQL: orلدData.db]  
(Background on this error at: https://sqlalche.me/e/20/e3q8)  
* sqlite:///RealWorldData.db  
Done.
```

Out[8]: [name](#)

Query the database system catalog to retrieve column metadata

The SCHOOLS table contains a large number of columns. How many columns does this table have?

```
In [ ]: # type in your query to retrieve the number of columns in the SCHOOLS table
```

Double-click **here** for the solution.

Now retrieve the the list of columns in SCHOOLS table and their column type (datatype) and length.

```
In [ ]: # type in your query to retrieve all column names in the SCHOOLS table along with their datatypes and length
```

Double-click **here** for the solution.

Questions

1. Is the column name for the "SCHOOL ID" attribute in upper or mixed case?
2. What is the name of "Community Area Name" column in your table? Does it have spaces?
3. Are there any columns in whose names the spaces and parenthesis (round brackets) have been replaced by the underscore character "_"?

Problems

Problem 1

How many Elementary Schools are in the dataset?

In []:

Double-click **here** for a hint

Double-click **here** for another hint

Double-click **here** for the solution.

Problem 2

What is the highest Safety Score?

In []:

Double-click **here** for a hint

Double-click **here** for the solution.

Problem 3

Which schools have highest Safety Score?

In []:

Double-click **here** for the solution.

Problem 4

What are the top 10 schools with the highest "Average Student Attendance"?

In []:

Double-click **here** for the solution.

Problem 5

Retrieve the list of 5 Schools with the lowest Average Student Attendance sorted in ascending order based on attendance

In []:

Double-click **here** for the solution.

Problem 6

Now remove the '%' sign from the above result set for Average Student Attendance column

In []:

Double-click **here** for a hint

Double-click **here** for the solution.

Problem 7

Which Schools have Average Student Attendance lower than 70%?

In []:

Double-click **here** for a hint

Double-click **here** for another hint

Double-click **here** for the solution.

Problem 8

Get the total College Enrollment for each Community Area

In []:

Double-click **here** for a hint

Double-click **here** for another hint

Double-click **here** for the solution.

Problem 9

Get the 5 Community Areas with the least total College Enrollment sorted in ascending order

In []:

Double-click **here** for a hint

Double-click **here** for the solution.

Problem 10

List 5 schools with lowest safety score.

In []:

Double-click **here** for the solution.

Problem 11

Get the hardship index for the community area of the school which has College Enrollment of 4368

In []:

Double-click **here** for the solution.

Problem 12

Get the hardship index for the community area which has the highest value for College Enrollment

In []:

Double-click **here** for the solution.

Summary

In this lab you learned how to work with a real word dataset using SQL and Python. You learned how to query columns with spaces or special characters in their names and with mixed case names. You also used built in database functions and practiced how to sort, limit, and order result sets, as well as used sub-queries and worked with multiple tables.

Author

Rav Ahuja