DB0201EN-Week4-1-1-RealDataPractice-v5

May 20, 2022

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

Estaimted time needed: **30** minutes

1.1 Objectives

After complting this lab you will be able to:

- Understand the dataset for Chicago Public School level performance
- Store the dataset in an Db2 database on IBM Cloud instance
- 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

1.2 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.

1.2.1 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.

While it is easier to read the dataset into a Pandas dataframe and then PERSIST it into the database as we saw in the previous lab, it results in mapping to default datatypes which may not be optimal for SQL querying. For example a long textual field may map to a CLOB instead of a VARCHAR.

Therefore, **it is highly recommended to manually load the table using the database console LOAD tool, as indicated in Week 2 Lab 1 Part II**. The only difference with that

lab is that in Step 5 of the instructions you will need to click on create "(+) New Table" and specify the name of the table you want to create and then click "Next".

Now open the Db2 console, open the LOAD tool, Select / Drag the .CSV file for the CHICAGO PUBLIC SCHOOLS dataset and load the dataset into a new table called SCHOOLS.

1.2.2 Connect to the database

Let us now load the ipython-sql extension and establish a connection with the database

The following modules are pre-installed in the Skills Network Labs environment. However if you run this notebook commands in a different Jupyter environment (e.g. Watson Studio or Ananconda) you may need to install these libraries by removing the `#` sign before `!pip` in the code cell below.

[3]:
```
# These libraries are pre-installed in SN Labs. If running in another␣
 ↪environment please uncomment lines below to install them:
# !pip install --force-reinstall ibm_db==3.1.0 ibm_db_sa==0.3.3
# Ensure we don't load_ext with sqlalchemy>=1.4 (incompadible)
# !pip uninstall sqlalchemy==1.4 -y && pip install sqlalchemy==1.3.24
# !pip install ipython-sql
```

[4]:
```
%load_ext sql
```

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

[5]:
```
import pandas as pd
```

[ ]:
```
# Enter the connection string for your Db2 on Cloud database instance below
# %sql ibm_db_sa://my-username:my-password@my-hostname:my-port/my-db-name?
 ↪security=SSL
%sql ibm_db_sa://syt31110:RRONCjbst1xrtQ4k@98538591-7217-4024-b027-8baa776ffad1.
 ↪c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:30875/bludb?
 ↪authSource=admin&replicaSet=replset
```

1.2.3 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

[ ]:
```
# type in your query to retrieve list of all tables in the database for your␣
 ↪db2 schema (username)
# my username is "syt31110"
%sql select TABSCHEMA, TABNAME, CREATE_TIME from SYSCAT.TABLES where␣
 ↪TABSCHEMA='syt31110'
```

Double-click **here** for a hint

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

1.2.4 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?

```
[ ]: # type in your query to retrieve the number of columns in the SCHOOLS table
%sql select count(*) from SYSCAT.COLUMNS where TABNAME = 'SCHOOLS'
```

Double-click **here** for a hint

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

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

```
[ ]: # type in your query to retrieve all column names in the SCHOOLS table along␣
  ↪with their datatypes and length
%sql select COLNAME, TYPENAME, LENGTH from SYSCAT.COLUMNS where TABNAME =␣
  ↪'SCHOOLS'
```

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

1.2.5 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 paranthesis (round brackets) have been replaced by the underscore character "_"?

1.3 Problems

1.3.1 Problem 1

How many Elementary Schools are in the dataset?

```
[ ]: %sql select count(*) from SCHOOLS where "Elementary, Middle, or High School" =␣
  ↪'ES'
```

Double-click **here** for a hint

Double-click **here** for another hint

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

1.3.2 Problem 2

What is the highest Safety Score?

```
[ ]: %sql select max(Safety_Score) AS MAX_SAFETY_SCORE from SCHOOLS
```

Double-click **here** for a hint

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

1.3.3 Problem 3

Which schools have highest Safety Score?

```
[ ]: %sql select Name_of_School, Safety_Score from SCHOOLS where \
      Safety_Score = (select Max Safety_Score from SCHOOLS)
```

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

1.3.4 Problem 4

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

```
[ ]: %sql select Name_of_School, Average_Student_Attendance from SCHOOLS \
     order by Average_Student_Attendance desc nulls last limit 10
```

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

1.3.5 Problem 5

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

```
[ ]: %sql select Name_of_School, Average_Student_Attendance from SCHOOLS \
     order by Average_Student_Attendance asc nulls last limit 5
```

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

1.3.6 Problem 6

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

```
[ ]: %sql SELECT Name_of_School, REPLACE(Average_Student_Attendance, '%', '') \
          from SCHOOLS \
          order by Average_Student_Attendance \
          fetch first 5 rows only
```

Double-click **here** for a hint

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

1.3.7 Problem 7

Which Schools have Average Student Attendance lower than 70%?

```
[ ]: %sql SELECT Name_of_School, Average_Student_Attendance  \
          from SCHOOLS \
          where CAST ( REPLACE(Average_Student_Attendance, '%', '') AS DOUBLE ) < 70␣
      ↪\
          order by Average_Student_Attendance
```

Double-click **here** for a hint

Double-click **here** for another hint

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

1.3.8 Problem 8

Get the total College Enrollment for each Community Area

```
[ ]: %sql select Community_Area_Name, sum(Community_Area_Name) as␣
      ↪Total_College_Enrollment from SCHOOLS \
    group by Community_Area_Name \
    order by Total_College_Enrollment desc
```

Double-click **here** for a hint

Double-click **here** for another hint

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

1.3.9 Problem 9

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

```
[ ]: %sql select Community_Area_Name, sum(Community_Area_Name) as␣
      ↪Total_College_Enrollment from SCHOOLS \
    group by Community_Area_Name \
    order by Total_College_Enrollment asc \
    limit 5
```

Double-click **here** for a hint

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

1.3.10 Problem 10

List 5 schools with lowest safety score.

```
[ ]: %sql select Name_of_School, safety_score from SCHOOLS \
    order by safety_score asc \
    limit 5
```

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

1.3.11 Problem 11

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

```
[ ]: %%sql
    select hardship_index
       from chicago_socioeconomic_data CD, schools CPS
       where CD.ca = CPS.community_area_number
          and college_enrollment = 4368
```

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

1.3.12 Problem 12

Get the hardship index for the community area which has the school with the highest enrollment.

```
[ ]: %sql select ca, community_area_name, hardship_index from␣
     ↪chicago_socioeconomic_data \
       where ca in \
       ( select community_area_number from schools order by college_enrollment desc␣
     ↪limit 1 )
```

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

1.4 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.

1.5 Author

Rav Ahuja

1.6 Change Log

Date (YYYY-MM-DD)	Version	Changed By	Change Description
2021-07-09	2.4	Malika	Updated connection string
2021-05-19	2.3	Lakshmi Holla	Updated question
2021-04-20	2.2	Malika	Added the libraries
2020-11-27	2.1	Sannareddy Ramesh	Modified data sets and added new problems
2020-08-28	2.0	Lavanya	Moved lab to course repo in GitLab

##