PROJECT “DATABASE AND SQL FOR DATA SCIENCE WITH PYTHON” by Kate Rogatina

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**

* Here I will be loading the csv files into the pandas Dataframe and then loading the data into the above mentioned sqlite tables.

* Next I will be connecting to the sqlite database **FinalDB**.

```
!pip install --force-reinstall ibm_db ibm_db_sa
!pip install sqlalchemy==1.3.9
```

```
Collecting ibm_db
  Downloading ibm_db-3.2.0-cp37-cp37m-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl (43.4 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 43.4/43.4 MB
43.2 MB/s eta 0:00:0000:0100:01
Collecting ibm_db_sa
  Downloading ibm_db_sa-0.4.0-py3-none-any.whl (31 kB)
Collecting sqlalchemy>=0.7.3 (from ibm_db_sa)
  Downloading SQLAlchemy-2.0.22-cp37-cp37m-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl (3.0 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 3.0/3.0 MB
81.0 MB/s eta 0:00:00:00:01
Collecting typing-extensions>=4.2.0 (from
sqlalchemy>=0.7.3->ibm_db_sa)
```

```
  Downloading typing_extensions-4.7.1-py3-none-any.whl (33
kB)
Collecting greenlet!=0.4.17 (from sqlalchemy>=0.7.3-
>ibm_db_sa)
  Downloading greenlet-3.0.1-cp37-cp37m-
manylinux_2_24_x86_64.manylinux_2_28_x86_64.whl (568 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 569.0/569.0 kB
62.0 MB/s eta 0:00:00
Collecting importlib-metadata (from sqlalchemy>=0.7.3-
>ibm_db_sa)
  Downloading importlib_metadata-6.7.0-py3-none-any.whl (22
kB)
Collecting zipp>=0.5 (from importlib-metadata-
>sqlalchemy>=0.7.3->ibm_db_sa)
  Downloading zipp-3.15.0-py3-none-any.whl (6.8 kB)
Installing collected packages: ibm_db, zipp, typing-
extensions, greenlet, importlib-metadata, sqlalchemy,
ibm_db_sa
  Attempting uninstall: ibm_db
    Found existing installation: ibm-db 3.1.0
    Uninstalling ibm-db-3.1.0:
      Successfully uninstalled ibm-db-3.1.0
  Attempting uninstall: zipp
    Found existing installation: zipp 3.15.0
    Uninstalling zipp-3.15.0:
      Successfully uninstalled zipp-3.15.0
  Attempting uninstall: typing-extensions
    Found existing installation: typing_extensions 4.5.0
    Uninstalling typing_extensions-4.5.0:
      Successfully uninstalled typing_extensions-4.5.0
  Attempting uninstall: importlib-metadata
    Found existing installation: importlib-metadata 4.11.4
    Uninstalling importlib-metadata-4.11.4:
      Successfully uninstalled importlib-metadata-4.11.4
  Attempting uninstall: sqlalchemy
    Found existing installation: SQLAlchemy 1.3.24
    Uninstalling SQLAlchemy-1.3.24:
      Successfully uninstalled SQLAlchemy-1.3.24
  Attempting uninstall: ibm_db_sa
    Found existing installation: ibm-db-sa 0.3.3
    Uninstalling ibm-db-sa-0.3.3:
      Successfully uninstalled ibm-db-sa-0.3.3
```

```
ERROR: pip's dependency resolver does not currently take
into account all the packages that are installed. This
behaviour is the source of the following dependency
conflicts.
dash 2.9.3 requires dash-core-components==2.0.0, which is
not installed.
dash 2.9.3 requires dash-html-components==2.0.0, which is
not installed.
dash 2.9.3 requires dash-table==5.0.0, which is not
installed.
Successfully installed greenlet-3.0.1 ibm_db-3.2.0
ibm_db_sa-0.4.0 importlib-metadata-6.7.0 sqlalchemy-2.0.22
typing-extensions-4.7.1 zipp-3.15.0
Collecting sqlalchemy==1.3.9
  Downloading SQLAlchemy-1.3.9.tar.gz (6.0 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 6.0/6.0 MB
82.6 MB/s eta 0:00:00:00:0100:01
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: sqlalchemy
  Building wheel for sqlalchemy (setup.py) ... done
  Created wheel for sqlalchemy: filename=SQLAlchemy-1.3.9-
cp37-cp37m-linux_x86_64.whl size=1159121
sha256=2cb4e50ce70f99ad08238c6085a95a25b45ba5950a437b538775
0e2daf88e155
  Stored in directory: /home/jupyterlab/.cache/pip/wheels/
03/71/13/010faf12246f72dc76b4150e6e599d13a85b4435e06fb9e51f
Successfully built sqlalchemy
Installing collected packages: sqlalchemy
  Attempting uninstall: sqlalchemy
    Found existing installation: SQLAlchemy 2.0.22
    Uninstalling SQLAlchemy-2.0.22:
      Successfully uninstalled SQLAlchemy-2.0.22
Successfully installed sqlalchemy-1.3.9
```

```
import csv, sqlite3
```

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

```
!pip install -q pandas==1.1.5
%load_ext sql
```

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

```
'Connected: @Chicago_Assignment.db'
```

```
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/ChicagoCensusData.csv")
df.to_sql("CENSUS_DATA", con, if_exists='replace', index=False,method="multi")

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

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")
```

Problems:

Problem 1

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

```
%sql SELECT COUNT(*) FROM CHICAGO_CRIME_DATA;
```

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

COUNT(*)
533

Problem 2¶

List community areas with per capita income less than 11000.

```
%sql SELECT COMMUNITY_AREA_NAME FROM CENSUS_DATA WHERE
PER_CAPITA_INCOME<11000
```

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

COMMUNITY_AREA_NAME

West Garfield Park
South Lawndale
Fuller Park
Riverdale

Problem 3

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

```
%sql select CASE_NUMBER from CHICAGO_CRIME_DATA where
PRIMARY_TYPE = 'OFFENSE INVOLVING CHILDREN';
```

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

CASE_NUMBER
HN567387
HR391350
HM768251
HT394616

Problem 4

List all kidnapping crimes involving a child?

```
%sql SELECT DISTINCT CASE_NUMBER, PRIMARY_TYPE, DATE,
DESCRIPTION FROM CHICAGO_CRIME_DATA \
WHERE PRIMARY_TYPE='KIDNAPPING'
```

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

CASE_NUMBE R	PRIMARY_TYP E	DATE	DESCRIPTION
HN144152	KIDNAPPING	2007-01-26	CHILD ABDUCTION/ STRANGER

Problem 5

What kinds of crimes were recorded at schools?

```
%sql SELECT DISTINCT(PRIMARY_TYPE), LOCATION_DESCRIPTION
FROM CHICAGO_CRIME_DATA \
WHERE LOCATION_DESCRIPTION LIKE '%SCHOOL%'
```

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

PRIMARY_TYPE	LOCATION_DESCRIPTION
BATTERY	SCHOOL, PUBLIC, GROUNDS
BATTERY	SCHOOL, PUBLIC, BUILDING
CRIMINAL DAMAGE	SCHOOL, PUBLIC, GROUNDS
NARCOTICS	SCHOOL, PUBLIC, GROUNDS

NARCOTICS	SCHOOL, PUBLIC, BUILDING
ASSAULT	SCHOOL, PUBLIC, GROUNDS
CRIMINAL TRESPASS	SCHOOL, PUBLIC, GROUNDS
PUBLIC PEACE VIOLATION	SCHOOL, PRIVATE, BUILDING
PUBLIC PEACE VIOLATION	SCHOOL, PUBLIC, BUILDING

Problem 6

List the average safety score for each type of school.

```
%sql SELECT "Elementary, Middle, or High School",
AVG(SAFETY_SCORE) \
FROM CHICAGO_PUBLIC_SCHOOLS_DATA GROUP BY "Elementary,
Middle, or High School";
```

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

Elementary, Middle, or High School	AVG(SAFETY_SCORE)
ES	49.5203836930456
HS	49.6235294117647
MS	48.0

Problem 7

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

```
%sql SELECT COMMUNITY_AREA_NAME,
PERCENT_HOUSEHOLDS_BELOW_POVERTY FROM CENSUS_DATA ORDER BY
PERCENT_HOUSEHOLDS_BELOW_POVERTY DESC LIMIT 5;
```

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

COMMUNITY_AREA_NAM	PERCENT_HOUSEHOLDS_BELOW_POVERTY
Riverdale	56.5
Fuller Park	51.2
Englewood	46.6
North Lawndale	43.1
East Garfield Park	42.4

Problem 8

Which community area is most crime prone?

```
%sql select community_area_number, count(*) as
Number_of_Crime from CHICAGO_CRIME_DATA group by
COMMUNITY_AREA_NUMBER\
order by Number_of_Crime desc Limit 1;
```

```
 * sqlite:///Chicago_Assignment.db
```

Done.

COMMUNITY_AREA_NUMBER	Number_of_Crime
25.0	43

Problem 9

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

```
%sql select community_area_name from CENSUS_DATA\
where hardship_index = (select max(hardship_index) from
CENSUS_DATA)
```

 * sqlite:///Chicago_Assignment.db
Done.

COMMUNITY_AREA_NAME
Riverdale

Problem 10

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

```
%%sql SELECT community_area_name FROM CENSUS_DATA WHERE
COMMUNITY_AREA_NUMBER = (SELECT COMMUNITY_AREA_NUMBER FROM
CHICAGO_CRIME_DATA
GROUP BY COMMUNITY_AREA_NUMBER ORDER BY
COUNT(COMMUNITY_AREA_NUMBER) DESC LIMIT 1) LIMIT 1;
```

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

COMMUNITY_AREA_NAME
Austin