Homework #11

Complete By: Friday November 15th @ class
Policy: Individual work only, late work *not* accepted
Assignment: SQL queries
Submission: submit electronically on Gradescope

Overview

In HW #11 you're going to write SQL queries to retrieve data from a Chicago Crimes database running in the Azure cloud. The database has been setup for you, your job is to write the queries to retrieve the required data in the required format. These queries will be saved in .sql files and submitted to gradescope for testing.

Learning SQL: reading and sample exercises

Let's start our introduction to declarative programming with a short, 2-page article that compares imperative programming (say C++) to declarative programming (in SQL). Just read the first 2 pages at the following web site (you will need to click "Continue to site" in upper right to reach underlying article):

http://www.databasejournal.com/sqletc/article.php/1408491/Beginning-SQL-Programming-Pt-2.htm

Next, let's learn some SQL basics. Here's a good site that features an integrated SQL database for practice:

http://sqlzoo.net/

Work through the following exercises on this site:

0 SELECT Basics:	exercises 1-4
1 SELECT name:	exercises 1-2
2 SELECT from World:	exercises 1-5
3 SELECT from Nobel:	exercises 1-6
5 SUM and COUNT:	exercises 1-5

Looking forward, here's a good reference site for SQL: http://www.w3schools.com/sql/default.asp

Executing SQL: Azure Data Studio

Azure Data Studio is a free, cross-platform tool for accessing queries against databases in the Azure Cloud. For download instructions, see

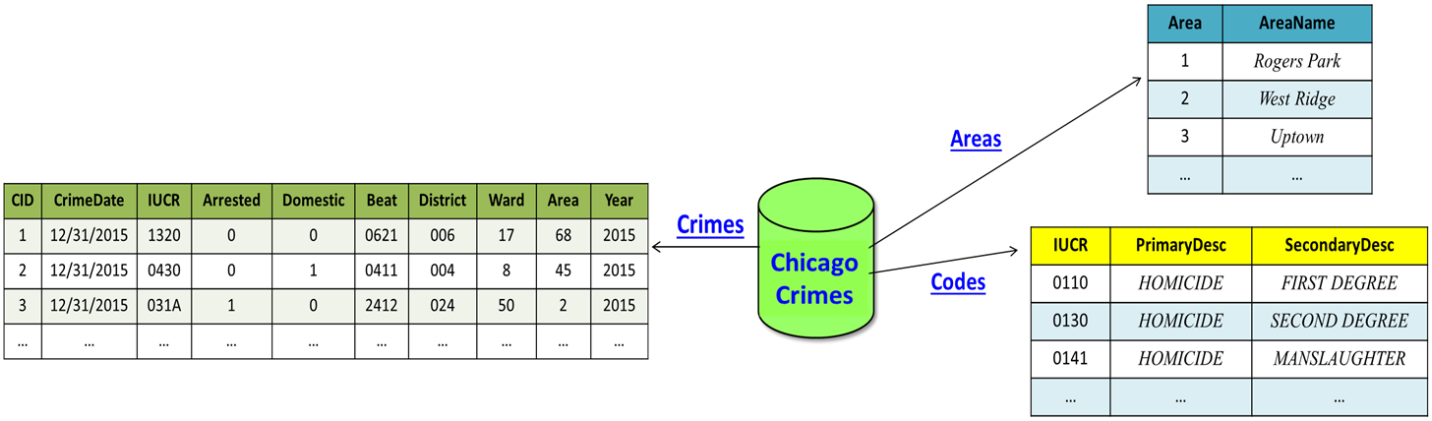
https://docs.microsoft.com/en-us/sql/azure-data-studio

Install, run, and then use the File to open a "New Query". Click the **connect** button at the top of the new window, and enter the following connection information:

server: uiccs341.database.windows.net
authentication type: SQL Login
user: student
pwd: cs341!uic

Once you enter the above, select "ChicagoCrimes" from the database drop-down list. Click **connect**. Once connected, you'll be able to enter queries and execute via the **run** button. Use File >> Save to save your work for submission to Gradescope.

Note: clicking **run** will execute all queries in the window. To execute a subset of queries (or just one), select to highlight, and click **run**.



CID	CrimeDate	IUCR	Arrested	Domestic	Beat	District	Ward	Area	Year
1	12/31/2015	1320	0	0	0621	006	17	68	2015
2	12/31/2015	0430	0	1	0411	004	8	45	2015
3	12/31/2015	031A	1	0	2412	024	50	2	2015
...	...	...	...	...	...	...	...	...	...

Area	AreaName
1	*Rogers Park*
2	*West Ridge*
3	*Uptown*
...	...

IUCR	PrimaryDesc	SecondaryDesc
0110	*HOMICIDE*	*FIRST DEGREE*
0130	*HOMICIDE*	*SECOND DEGREE*
0141	*HOMICIDE*	*MANSLAUGHTER*
...	...	...

SQL Exercises

Exercise #01: Total number of crimes in area 4

Write a single SQL query to retrieve the total # of crimes that have occurred in area 4 of Chicago (which includes UIC). Use the "As" clause to name this result "TotalNumCrimes". Save your work to the file "hw1101.sql" for submission to Gradescope.

[Hint: use Count function. Answer: 430.]

Exercise #02: Which crime code entities have ROBBERY as their Primary Description?

Write a single SQL query to retrieve the IUCR, Primary Description, and Secondary Description of all crime code entities that have ROBBERY as their primary description. The data should be ordered by IUCR in ascending order. Save your work to the file "hw1102.sql" for submission to Gradescope.

[Answer should contain 14 rows.]

Exercise #03: How many crimes occurred on December 1st, 2015?

Write a single SQL query to retrieve the number # of crimes that have occurred on 12/01/2015. Use the "As" clause to name this result "NumCrimes". Save your work to the file "hw1103.sql" for submission to Gradescope.

[Hint: CONVERT(date, CrimeDate) will translate a datetime object to a date.
SQL dates are of the form 'yyyy-mm-dd'.
Answer: 689.]

Exercise #04: How many crimes have occurred for each crime code?

Write a single SQL query to retrieve the # of crimes that have occurred for each IUCR code. Retrieve the IUCR code and the # of crimes; use the "As" clause to name the latter column "NumCrimes". Order the data in descending order by the number of crimes; this way the most common crime is shown first. Save your work to the file "hw1104.sql" for submission to Gradescope.

[Hint: group by. Answer: first crime is 0820 with 5861 crimes. Second is 0460 with 5650 crimes.]

Exercise #05: What crime is 0820? 0486? 0460?

Crimes 0820, 0486, and 0460 are the top-3 crimes in Chicago. What are these crimes? Write a single SQL query to retrieve the IUCR, Primary Description, and Secondary Description for these 3 IUCR codes. Order the results in ascending order by IUCR. NOTE: the IUCR codes are strings, not integers, which impacts how you write the WHERE clause. The Boolean operators OR and AND are available in addition to = and <> (not equal). Save your work to the file "hw1105.sql" for submission to Gradescope.

Grading and Electronic Submission

When you're ready, submit your individual query files "hw1101.sql", "hw1102.sql", etc. on Gradescope under "HW11".

Policy

Late work is not accepted for this assignment. All work is to be done individually — group work is not allowed. While we encourage you to talk to your peers and learn from them (e.g. your "iClicker teammates"), this interaction must be superficial with regards to all work submitted for grading. This means you *cannot* work in teams, you cannot work side-by-side, you cannot submit someone else's work (partial or complete) as your own. The University's policy is available here:

https://dos.uic.edu/conductforstudents.shtml .

In particular, note that you are guilty of academic dishonesty if you extend or receive any kind of unauthorized assistance. Absolutely no transfer of program code between students is permitted (paper or electronic), and you may not solicit code from family, friends, or online forums. Other examples of academic dishonesty include emailing your program to another student, copying-pasting code from the internet, working in a group on a homework assignment, and allowing a tutor, TA, or another individual to write an answer for you. It is also considered academic dishonesty if you click someone else's iClicker with the intent of answering for that student, whether for a quiz, exam, or class participation. Academic dishonesty is unacceptable, and penalties range from failure to expulsion from the university; cases are handled via the official student conduct process described at https://dos.uic.edu/conductforstudents.shtml .