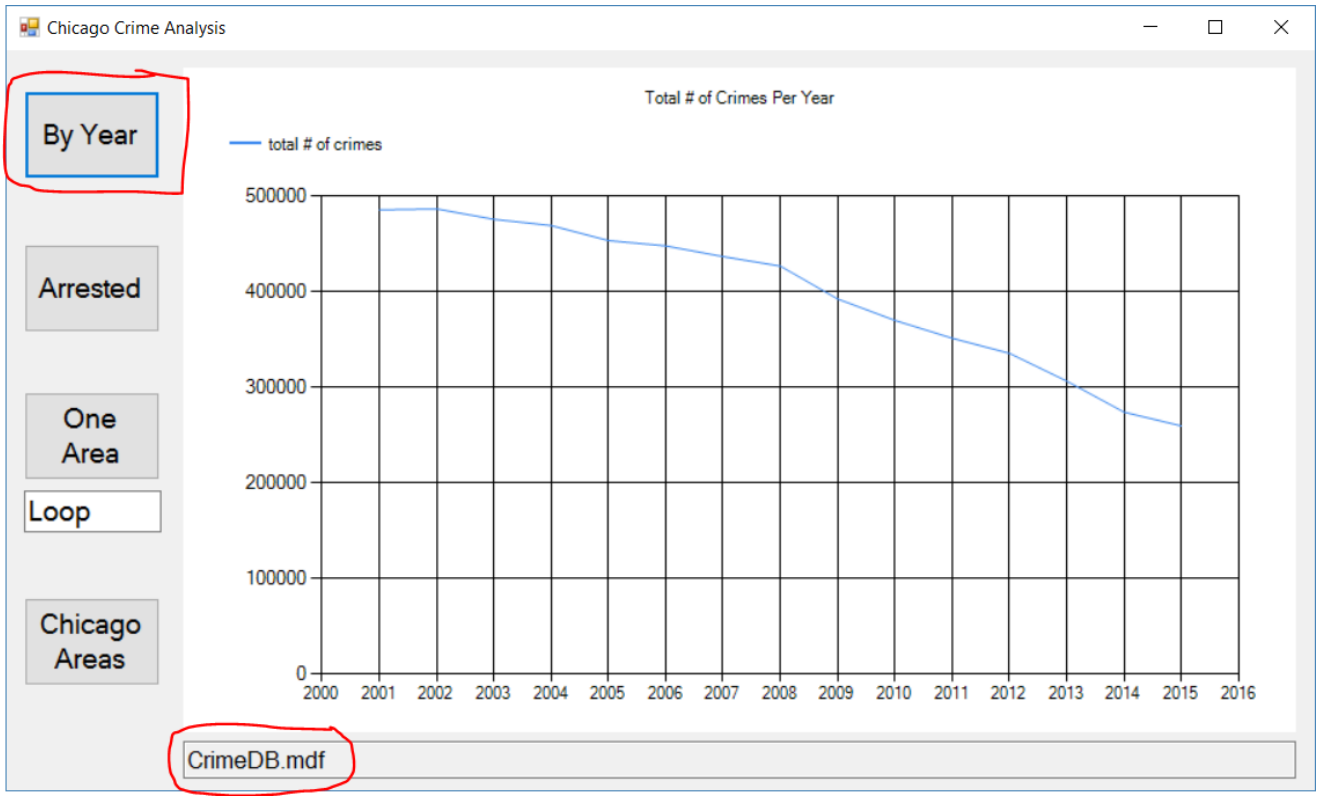
Project #07: Chicago Crime Analysis with SQL

Complete By: Friday, April 20th @ 11:59pm
Assignment: completion of following exercise
Policy: Individual work only, late work *is* accepted
Submission: electronic submission via Blackboard

Chicago Crime Analysis

The project is focused on retrieving crime data from a database and graphing it on the screen. We'll be using the Chicago Crime database we've been discussing in class (and HW). Here's a screenshot of the app you're going to build --- the user has clicked the button to display the # of crimes per year across Chicago:



Most of the GUI is written for you, your job is to add the SQL and ADO.NET needed to retrieve the data from the database.

Getting Started

Step 1 is to download the provided GUI app from the course web page. Look under "Projects", then "project07-files". Here's the link:

https://www.dropbox.com/s/jsr0m4ssi499qad/ChicagoCrimeAnalysis.zip?dl=0

After you download, open the .zip, extract the folder "ChicagoCrimeAnalysis" to your desktop, close the .zip file, and delete the .zip file. Go ahead and open the "ChicagoCrimeAnalysis" folder, and double-click on the Solution (.SLN) file to open in Visual Studio. Run (F5), and click one of the buttons --- you should get an error message about the file "CrimeDB.mdf" not found. Close the app, minimize Visual Studio, and let's fix this…

Step 2 is to install a copy of the 2 CrimeDB files (.mdf and .ldf) into the project's "bin\Debug" sub-folder. You probably have a copy of these files on your computer that you were using for the Homework --- it is preferred that you copy and install these files, since these have been upgraded to the version of Microsoft SQL Server installed on your machine. Try to make copies of the 2 database files (ctrl-C) and install (ctrl-V) in the project's "bin\Debug" sub-folder. If the files are locked and you are unable to copy: switch back to Visual Studio, open Server Explorer, right-click on the Data connection, and "disconnect" from the CrimeDB database. Now you should be able to copy the two database files. [*If* you have not yet installed a copy of the CrimeDB database files, do the following:

1. download a copy from the course web page (see Projects\project07-files, CrimeDB.zip)
2. Unzip, extract the 2 files (database and log file), save on desktop
3. Back in Visual Studio, view Server Explorer, and open a Data Connection to the database file --- upgrade if asked
4. Right-click on newly-created data connection, and "disconnnect"
5. Minimize Visual Studio, and copy the two database file down into the project's "bin\Debug" sub-folder.

Once you have the files installed into "bin\Debug", switch back to Visual Studio and run the app again --- clicking a button should not produce an error message. Instead, you'll see the chart change slightly (with a title and legend), but no data. At this point you're ready to start the assignment.

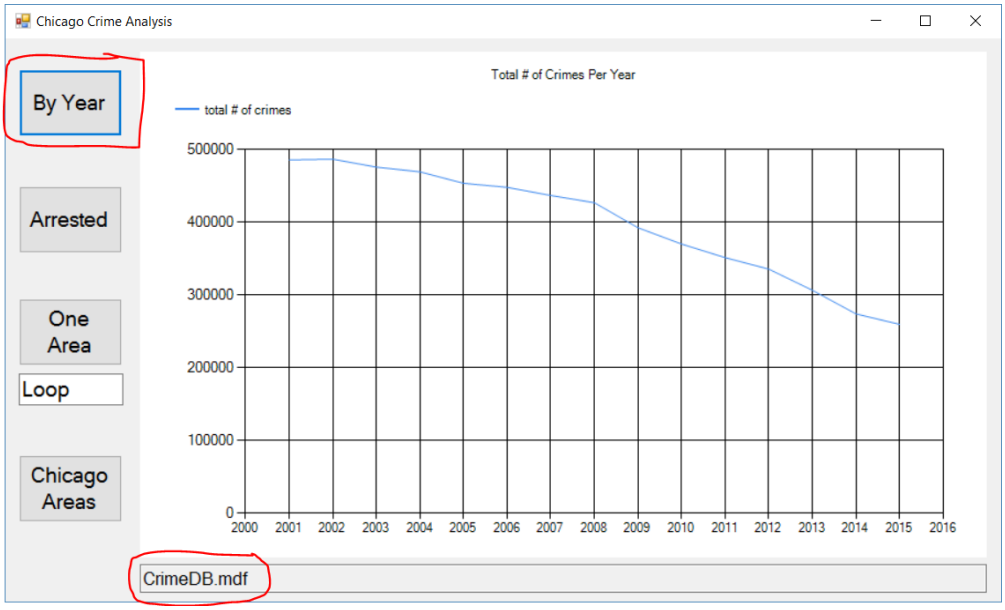
Assignment Details

The GUI has been built for you: 4 buttons, a couple text boxes, and a chart for graphing. The chart is built-in like the buttons and textboxes, it was simply dragged onto the form from the Toolbox. Your job is to implement the 4 buttons. Note that the screenshots shown on the following pages are from the larger 2001-2015 database.

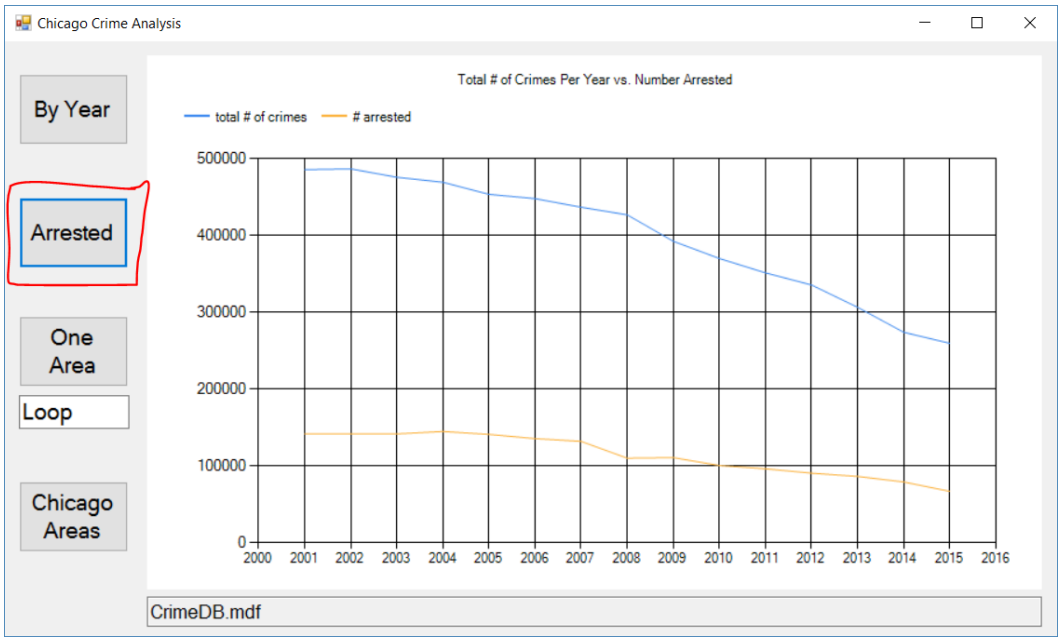
To implement the code behind these 4 buttons, you have to use ADO.NET to execute the SQL queries against the database. ADO.NET was the subject of HW #12, and class on Monday, April 16th. If you missed class, an "ADO.NET code summary" was handed out. You can find it here: PPT, 1-page PDF, 6-page PDF.

<< next page… >>

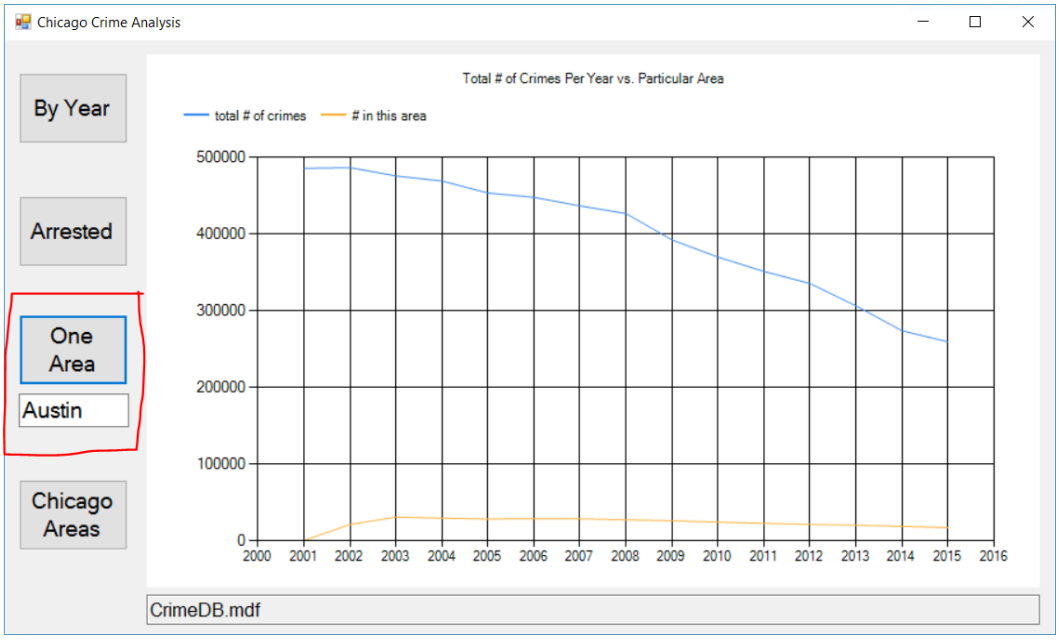
#1) By Year: total # of crimes reported each year. Plot year on the X axis, and total on the Y axis. Code is provided to create the plot the (x, y) points. You'll execute a single SQL query to retrieve the data, and then add the necessary values into two lists named **X** and **Y**. See the provided code for more details.



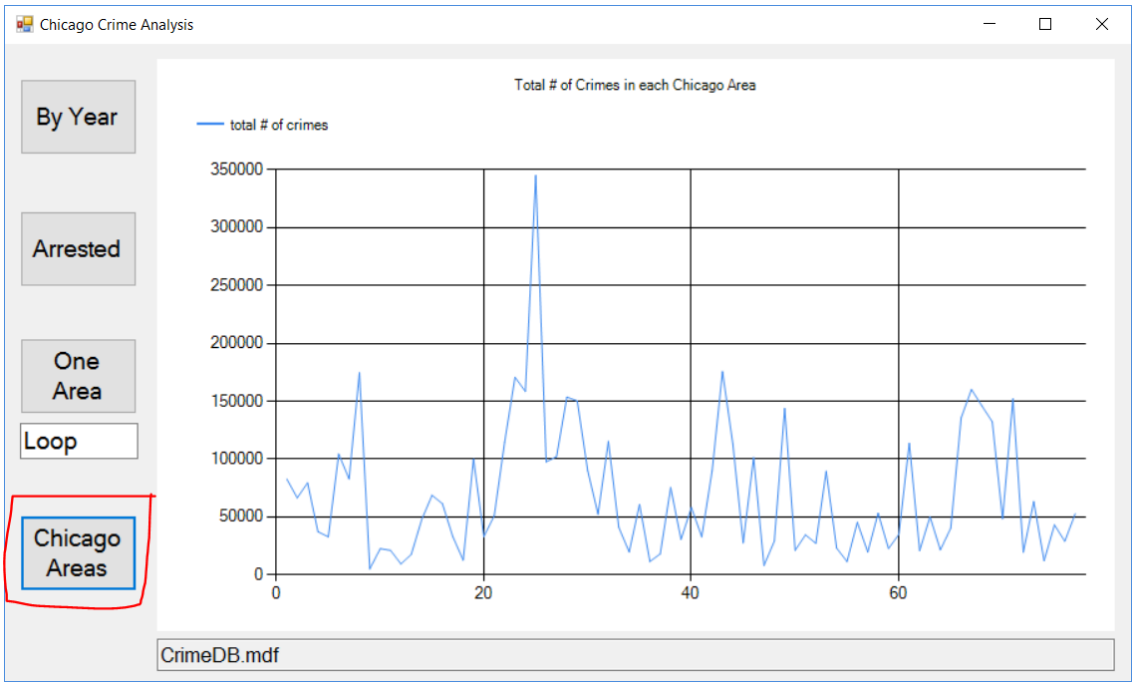
#2) Arrested: total # of crimes reported each year vs. # arrested each year. Plot year on the X axis; plot total and # arrested on the Y axis. You can execute two queries to retrieve the data you need, though it can be done in one query.



#3) One Area: total # of crimes reported each year vs. crimes in a particular area of the city. Plot year on the X axis; plot total and # in area on the Y axis. You may assume the user will enter a valid area name in the textbox, such as "Loop", "Austin", or "Bridgeport". A complete list of areas is available here. Retrieving the data you need may require executing three separate queries, which is fine; you can do this in two using a join.



#4) Chicago Areas: total crimes for each area of the city. This can be done using one SQL query. Note that areas range from 1..77, with 0 used as the "unknown" area for crimes reported in which the area was not specified. Ignore area 0 when you retrieve the data ("WHERE Area > 0").



Electronic Submission

First, add a header comment to the top of your main "Form1.cs" C# source code file, along the lines of:

```
//
// GUI app to analyze Chicago crime data, using SQL and ADO.NET
//
// <<YOUR NAME HERE>>
// U. of Illinois, Chicago
// CS341, Spring 2018
// Project 07
//
```

Exit out of Visual Studio, and find the project folder "ChicagoCrimeAnalysis". Drill down to the "bin\Debug" sub-folder, and delete the database files to save space. Then create an archive (.zip) of this entire folder for submission: right-click, Send To, and select "Compressed (zipped) folder". Submit the resulting .zip file on Blackboard (http://uic.blackboard.com/) under the assignment "P07: Crime DB GUI".

Your program should be readable with proper indentation and naming conventions; commenting is expected where appropriate. You may submit as many times as you want before the due date, but we grade the last version submitted. This implies that if you submit a version before the due date and then another after the due date, we will grade the version submitted after the due date — we will *not* grade both and then give you the better grade. We grade the last one submitted. In general, do not submit after the due date unless you had a non-working program before the due date.

Policy

Late work *is* accepted. You may submit as late as 24 hours after the deadline for a penalty of 25%. After 24 hours, no submissions will be accepted.

All work is to be done individually — group work is not allowed. While I encourage you to talk to your peers and learn from them (e.g. your "iClicker teammates" or Piazza), 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 described here:

http://www.uic.edu/depts/dos/docs/Student%20Disciplinary%20Policy.pdf .

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 http://www.uic.edu/depts/dos/studentconductprocess.shtml .