

Assignment 1 COSC 2347:

DUE 11:59 pm on 2/5/25. No late assignments.

Using correct C# style and naming conventions as detailed in the lectures and slides, create a C# program as described below.

Part 1 (60 points): I will provide you a test file of “user input” and you will process that file and will produce console output as detailed below.

Each of the items in brackets [] you will read in from the file and display correctly. Items in () are things that you will calculate, just follow those directions in the parentheses.

Reading the file will be easy, we'll use a feature called console input redirection and the console input will be read from the file. This is no different than the code that you've been seeing and writing to date. Here is your template to write to the console, filled in by you reading in the input items in brackets and following the directions in the parentheses:

Hi, my name is [string] [string], and I was born in [string]. It is polite to write my name like this: (print same first name and last name but in all upper case). My name is huge; it is (calculate and print the length of the first and last name) characters long!

Many of my friends were born in the country of [string]. Its capital is [string] (which will always start with the letters Para), but people call it: (print the capital name, but with the string “Para” removed).

My country is very wealthy, with a GDP of [decimal] (print in dollar and cents format). We have 100000 citizens, and each generates (previous GDP / 100000) of the GDP output. Each of us produces 30,000 widgets a year, for a total of (calculate and print total widgets) widgets per year! Many of the world's population of (create a variable to hold the population of the world and print it) people will buy them!

Here is my friend, her age, and how many widgets she produced per quarter last year:

Name: [string] Age: [byte] Q1: [int] Q2: [int] Q3: [int] Q4: [int]. Total: (add all quarter numbers, can exceed int size)

Here is another friend, and he is [byte] years old. They are a total of (add the ages of the friends) years old.

Is this string [string] a number? (It will be non-zero. If it converted to a number, display it. Otherwise print “No, didn't convert”).

Is this string [string] a number? (It will be non-zero. If it converted to a number, display it. Otherwise print “No, didn't convert”).

-----

Assignment 1, part 2 (40 points):

For this part of the assignment, just keep adding to the screen output from the last part of the assignment.

Create 3 tuples made up of your favorite movie or tv actors, along with their age, and a movie or tv show that they were in. Write each whole tuple (not tuple element), to the console.

Create 2 tuples of 2 integers and initialize each tuple with 2 numbers of your choice. Then write to the console tuple1 and tuple2 and the sum and product of your tuples 1 and 2.

Create a string array of cats and initialize it with 9 cat breeds.

If you can figure out how to write to the console with 1 WriteLine statement all of the cats in your array without specifying them individually, I'll give you 5 points of extra credit.

Using the range operator, write to the console just the 3<sup>rd</sup> through 5<sup>th</sup> elements of from your cats array.

NOTE: The 3<sup>rd</sup> through 5<sup>th</sup> elements of the array are NOT identical to cat[3], cat[4], and cat[5] because your array is 0 based.

-----

You will:

- 1) Read items from the file using the Console.ReadLine() function as if a user were typing the input
- 2) Perform the correct calculations and conversions.
- 3) Console output should be generated through interpolation, not manually constructed strings.
- 4) Output the program's results to the console using the standard Console.Write and Console.WriteLine commands.
- 5) Your program cannot throw exceptions from conversion results. If you are converting a string to a number, assume the number is 0 if it doesn't convert correctly.
- 6) Capture the output screen with the correct results. If your output screen is disappearing before you can capture it, put in a WriteLine statement at the end of the program and put a breakpoint on it, then you can capture the output window.
- 7) Turn your complete source code along with the screen capture of your complete program output into Blackboard by the assignment due date. Failure to turn in full source code (in text format) and screen output snapshots showing all of your program output = grade of 0.
- 8) Use MY data that I've supplied in the text file, don't use whatever data you want.
- 9) Formatting matters! This is an exercise in data manipulation and the entire point of the assignment is how closely you match my output. Use the instructions in this assignment as a checklist if you'd like to do well at it.

**Console redirection** will make your life easy with this program:

Setting up console redirection is easy; in VS go to the debug menu and choose the last option which should be your program's properties. The very first text input box is "command line arguments" and you will copy and paste in all characters after this next colon: < c:\temp\myinput.txt

You will then have to copy myinput.txt to your computer's c:\temp folder. If you don't have a c:\temp folder, then you will need to create one, and when you execute your code, instead of asking you to input the information from the console, it will read it from myinput.txt.

When you no longer wish to input the data from the file, simply remove that line from the command line arguments input box and the program will again accept input from your console.

Mac users running the Mac version of Visual Studio will quickly discover that it does not handle console redirection correctly. If you're running on a Mac with the Mac version of VS, contact me for some workaround code.

Notes:

If you are running on a Linux environment or running in another development environment other than VS, you will have to find the comparable options in your development environment.

THERE IS ZERO FILE CODE in this program. If you start down that path, you're heading the wrong direction!

Like all assignments in this class, this is NOT a group project. Writing your code with someone else or sharing code is academic dishonesty and will be dealt with accordingly.

Write the code for section 1 to produce the EXACT console output as demonstrated above. For section 2, follow the directions carefully and output what I told you to output. Your grade will drop the more that you stray from what I did above.

The BIGGEST RISK of this program to your grade is you not carefully following these directions. Check them, double check them, check them again when you're done; your grade will thank you.

As explained in the syllabus, these instructions are subject to verbal change in this class. If you are not paying attention or not here and miss any changes in instructions, that is on you.