

Setup:

You will be creating a new Visual Studio project named "FirstnameLastname_CE08" using the C# Console Application template. You will also need to download the files found [here](#), these are the files that you will be reading in data from.

Grading:

Please refer to the CE:08 tab of [this](#) rubric to understand the grading procedures for this assignment.

Deliverables:

You will compress and upload a file named "FirstnameLastname_CE08.zip" with the following contents with the following names:

- Project - Folder containing your entire project and all files necessary to build the project(csproj, .cs files, App.config, and Properties folder).

Be sure to upload the correct project and all required files the first time as only one submission will be allowed. No extra time or consideration will be given if the wrong files are uploaded.

Instructions:

Data and layout file structure

Inside each of the data files is 102 lines of information. The first and last lines are just a header and footer that don't contain a relevant record. Each of the other 100 lines contains one record each. These records each have 150 fields separated by | . The final field is just a field indicating the end of a record.

The data layout file has one field name per line corresponding to each of the 150 fields that each record in the data files have. These field names and the fields in the data files are all in the same order.

For this assignment you can have all of your code in the Program class if you choose to. You will need to give the user the choice of reading in all of DataFile1, 2, or 3. The data in the chosen file should be read in and converted to the JSON format. It's important to note that the data files only have the values of each field, the DataFieldsLayout file contains the name of each field. The JSON formatted data then needs to be written out to a valid JSON file. The code that converts the data to the JSON format must be written by you. Do not use a serializer or 3rd party framework like Newtonsoft.Json.

Make sure your project meets the following requirements:

File I/O Reading

- The field names are read in from the DataFieldsLayout file and stored in a variable.
- The data fields are able to be read in from at least one of the DataFile files and stored in a variable.

File I/O Writing

- All relevant data from a DataFile is written out to a single file.

JSON

- Records are converted to valid JSON and written to a valid JSON file. (ex. The file extension should be .json and not .txt).

User Input

- The user is able to select which of the three data files they would like to process.

Extra Information

Tips and hints:

- Each line of the data files is a separate record made up of key:value pairs.
- Figuring out how to convert one record into JSON and then repeating that process for each record will be easier than trying to just lump convert the whole data set.
- Test your JSON using the JSON validator found [here](#).
- This is an array of data, make sure that your output JSON file accurately represents this fact. The data should not just be a bunch of JSON objects in a file with no structure.

Go back through your code and check for the following:

- All variables and methods are named appropriately.
- Member variables should have either protected or private access.
- Anytime the user is asked for input the expected input is made clear to the user.
- Any information being output to the user should be clear and concise.
- Make sure nothing accesses an object that doesn't exist.