cView App – Phase 3

Instructor: Dr. Raman Adaikkalavan

Points: 50 Assigned: 6/10/15 @ 8 a.m. Due: 6/17/15 @ 8 a.m.

!!! START TODAY !!!

PROJECT DESCRIPTION

From Phase 1: Open source software spurs innovation and competition. Open data allows citizens to learn more about their city, state, country, and even more. It allows them to explore all services offered, status of various items, financial health, and much more. Currently, many government entities have made their data open (see http://www.data.gov), so that citizens can use them. As software developers, one of the main goal is to convert data to meaningful information and make them easily accessible to general public. For instance, look at some of the current apps that process open data @ http://www.data.gov/applications

Fortunately, City of South Bend has made its data open https://data.southbendin.gov/

This semester, we will build an app that allows us to explore some of this data, and present them in a meaningful way. Our main goal is to explore C# programming and we will use this app building experience to learn C# and many of its features. This app will be built via multiple phases throughout the semester.

From Phase 2: In phase 1, you visited South Bend Open Data website and explored various data sets available. You should have explored the following three data sets:

1) Business Licenses: https://data.southbendin.gov/Business/Business-Licenses/imxu-7m5i
2) Parks and locations: https://data.southbendin.gov/Parks-Recreation/Parks-Locations-and-Features/yf5x-7tkb
3) Public facilities: https://data.southbendin.gov/Health-Human-Services/Public-Facilities/jeef-dsq9

In phase 2, you created a C# app (Console or UI) by creating classes (no inheritance, interface, etc.) and Lists of objects of the classes to store one data set different from Phase 1 and 5 fields

PHASE 3 DESCRIPTION

For this phase, create a C# app (Console or UI) meeting the following requirements:

- Download the data for all THREE data sets as .csv
    - Name them as parks.csv, publicfacilities.csv, and businesses.csv
    - Clean up these data files so that each cell contain only one attribute value. Also make sure there is no comma within each cell.
        - For instance, split the location attribute to multiple columns such as street address, city, state, zip, latitude, and longitude.
        - Delete duplicate columns within each .csv file

- Download **Ph3.zip** from Oncourse Assignment Phase 3
    - Feel free to add more files, but you cannot remove my files or change their names
    - Feel free to add, delete, modify contents in any file, except Item.cs
    - You can add more in Item.cs but cannot modify my statements.
    - You must keep ItemList private and store your entire data using it
- You cView app should be able to do the following
    - Menu to be used
        - Option 0) Clear ItemList and Load data from csv to ItemList
        - Option 1) Add an item
            - Make sure to ask the type of item before adding
            - Call the Add method in the ItemDB class
        - Option 2) Modify an item through user input
            - Display all the stored items and ask the user to choose an item
            - Display all the fields and ask the user to choose a field to modify
            - Get the new value for that field and replace the old value
            - Call the Modify method in the ItemDB class
        - Option 3) Search based on user input and display the matched results
            - Display the possible item types and ask the user to choose an item type for searching
            - Display the possible fields and ask the user to choose a field for searching
            - Get the value for the search.
        - Option 4) Delete based on user input
            - Display all the stored items (item type, id, and name) and ask the user to choose an item (id)
            - Delete the chosen item id
            - Display all the stored items again
        - Option 5) Display all the stored items
            - Override ToString() method of each class and use it for displaying objects.
        - Option 6) Print Statistics
            - When this option is chosen, display each type and number of items in each type. This is based on Item.Type and not Item.ItemType
        - Option 7) exit
            - Save the entire ItemList back to the three .CSV Files based on ItemType
- **Only for Graduate Students:**
    - In Option 7: Save the entire ItemList back to one XML or JSON file
    - In Option 1: Allow user to read the input from three .csv files or from the one XML or JSON file that you saved in Option 7.

CHECKLIST

- Include program (Name, Assignment Info, Date, Your Info, Language Used, etc.) and function (input, output, purpose) headers
- Include comments for each logical block
- Use constants where required. Avoid using literals.

- Check for valid inputs
- You application must use only the C# components (or things closely related) discussed so far in this course. If you are not sure, send me a message well ahead of time for my inputs.

SUBMISSION

- Name the project as cView-P#-yourName (where # is the phase number)
- Run your program. Capture the screen shots to demonstrate the program's execution using Sniptool or other software and store the screen shots in a word document. Convert the word document to a PDF and name the PDF as cView-P#-yourName.pdf Submit the PDF file with screen shots.
- Zip the entire cView-P#-yourName Visual Studio folder and upload the zip file cView-P#-yourName.zip to Oncourse
- Upload cView-P#-yourName.pdf to Oncourse
- Upload the .csv files to Oncourse
- Submit printouts of the source (.cs) and snapshot (.pdf) files