CS264 - Software Design

Assignment 01

Assignment Release Date:	19-09-2022
Submission Due Date:	07-10-2022
Feedback Due Date (estimated):	22-10-2022 (for assignments that make Due Date)
Support Laboratories	Labs 01-02 (Two Weeks)
Total Mark:	10%

This Assignment is worth 10% of the Software Design CA Component.

This is an open-book, graded assignment. You may use online resources for reference purposes only to help with the assignment. Please cite all references as comments in your submissions. You cannot directly reuse C# **solution code** from online sources. **You must not engage with another student, in person or electronically (phone, social media, etc.) to secure assistance with this assignment. If you do so you will receive an automatic fail (0%).** We will perform similarity checks on submitted assignments to check for collaborative efforts. A reasonable attempt at this assignment will gain you 5% of your continual assignment marks. It is possible to gain extra credit (up to a maximum of 5%) for this assignment.

Assignment 01 - Object Oriented Development with C#

You are required to develop an object-oriented console solution (app), called `tabconv`, for managing and converting Tablular markup information. This app will provide functionality to convert between different markup formats for tabular data, for example, it will convert between CSV (Comma Separated Values), MD (Markdown), JSON (JavaScript Object Notation, and HTML-TABLE (HTML <table> element) formats. Your app should handle errors in input, providing appropriate error information. Your solution should implement a CLI (command-line interface) user interface along the lines for the following:

```
tabconv  -v -i <file.ext> -o <file.ext>

    -v, —verbose               Verbose mode (debugging output to STDOUT)
    -o <file>, —output=<file>  Output file specified by <file>
    -l, —list-formats          List formats
    -h, —help                  Show usage message
    -i, —info                  Show version information

<.ext> will be one of [.html | .md | .csv | .json]


$ tabconv -i table.json -o table.html      (convert JSON table to <table>)
```

Assignment 01 - Requirements

Your solution does not have to handle nested tables. Your C# solution should include a single class that provides methods that perform the conversion between different formats. For example, here is a conversion between JSON, MD, HTML and CSV (from converters at https://csvjson.com/json2csv and https://www.convertcsv.com/csv-to-html.htm):

CSV	MD
``` "album","year","US_peak_chart_post" "The White Stripes",1999,"-" "De Stijl",2000,"-" "White Blood Cells",2001,61 ```	``` |album |year|US_peak_chart_post| |----------------|----|------------------| |The White Stripes|1999|- | |De Stijl |2000|- | |White Blood Cells|2001|61 | ```

CSV

```
"album","year","US_peak_chart_post"
"The White Stripes",1999,"-"
"De Stijl",2000,"-"
"White Blood Cells",2001,61
```

MD

```
|album           |year|US_peak_chart_post|
|----------------|----|------------------|
|The White Stripes|1999|-                 |
|De Stijl         |2000|-                 |
|White Blood Cells|2001|61                |
```

JSON

```
[
  {
    "album": "The White Stripes",
    "year": 1999,
    "US_peak_chart_post": "-"
  },
  {
    "album": "De Stijl",
    "year": 2000,
    "US_peak_chart_post": "-"
  },
  {
    "album": "White Blood Cells",
    "year": 2001,
    "US_peak_chart_post": 61
  },
]
```

HTML

```
<table>
    <tr>
        <th>album</th>
        <th>year</th>
        <th>US_peak_chart_post</th>
    </tr>
    <tr>
        <td>The White Stripes</td>
        <td align=""right"">1999</td>
        <td>-</td>
    </tr>
    <tr>
        <td>De Stijl</td>
        <td align=""right"">2000</td>
        <td>-</td>
    </tr>
    <tr>
        <td>White Blood Cells</td>
        <td align=""right"">2001</td>
        <td>61</td>
    </tr>
</table>
```

Please adhere to the following development requirements:

1. You must provide an object oriented solution (a tabular class with get/set methods, etc.) which is used by your main program (Program.cs).

2. You must comment your code, clearly indicating, how your code implements the solution described above in the "Assignment 01 - Requirements" section.

3. You must provide a proper C# CLI user interface using command-line arguments processing. You must use C# file handling for input and output.

Please note that there are many sample (C#, JS, C++, Python) solutions for implementing similar solutions functionality available online. While it is fine to consult these, and accompanying articles, for references, you may not re-use code from these projects. Please cite your reference sources in your codebase. We will search and identify online coding solutions to similar problems for the purposes of checking against submitted solutions in instances where we have concerns about code originality.

You can find information on Markdown Table encoding in this extended Markdown guide: https://www.markdownguide.org/extended-syntax/

There is an excellent Table Generator at: https://www.tablesgenerator.com/markdown_tables

Assignment 01 - Additional Requirements for Extra CA Credit (5%)

If you manage to develop and design and solution to this assignment before the deadline you may also wish to include (i) a conversion for LaTeX tables (https://www.tablesgenerator.com/latex_tables) and text tables (https://www.tablesgenerator.com/text_tables) together with (ii) specifying styling (where appropriate) for generated tables. Successfully completing this extra functionality will result in awarding an extra 5% towards your CA Total.

IMPORTANT SUBMISSION DETAILS

Please indicate the Operating System (Linux/Windows/MacOS/Online) and IDE (e.g. VS Code) version used for testing (as a comment in your submitted code).

All work must be submitted via Moodle (see "Assignments" section for submission). Work submitted via other means will not be accepted unless you have prior arrangements with the Head Demonstrator (Mark McCormack). All work MUST be submitted by the due-date deadline. Late submissions will not be accepted.

If you are attempting the Extra Credit option you must upload your solution with the extra-credit code as a separate upload using the submission link provided in the CS264 Moodle space.

*The assignment submission is a zip file named "**assignment-01-xxxxxxxxx.zip**" (where "xxxxxxxxx" is your student id) containing solution files, e.g. named "**tabular.cs**", "**Program.cs**", etc. together with any other resources used in the assignment solution. Please ensure that all external files use relative directory referencing, rather than hard-coding the files' location.*