Spreadsheet Menu Options

Cpt S 321 Homework Assignment, WSU

Submission Instructions:

Submit via Git. This project will be used in future assignments.
Directory name: Spreadsheet
Git tag for submission: Spreadsheet-v5.0

Important note: This is the framework for a spreadsheet application that you will build over the course of the semester. Almost ALL remaining homework assignments will build on top of this. So think about these instructions DEEPLY!

Assignment Instructions:

Read each step's instructions carefully before you write *any* code.

In this assignment, you will add some new features to our spreadsheet application. These include loading and saving capabilities to your spreadsheet application as well as an about window with your spreadsheet's version and your contact information. You'll need to add a Winforms menu to your application via the StatusStrip Control or other method.

Menu Options: File -> Load and File -> Save

In the Menu, create load and save menu options. These should call the standard windows dialog for loading and saving files. We'll be using simple XML to store our data in a flat text file.

Design an XML format for your spreadsheet data. At a high level, it will probably have a structure somewhat like:

```
<spreadsheet>
  <cell name="B1">
    <bgcolor>FF8000</bgcolor>
    <text>=A1+6</text>
  </cell>
</spreadsheet>
```

You'll obviously have more than one cell in most cases. Make sure you do the following:

- Provide saving and loading functions that take a stream as the parameter.

- Add menu options in the UI for saving and loading.
- Use the openFileDialog() WinForms tool for selecting a file.
- Make sure the saving and loading code is in the logic engine (why?).
- Use existing XML classes from the .NET framework.
- When saving, only write data from cells that have one or more non-default properties. This means that if a cell hasn't been changed in any way then you don't need to write data for it to the file.
- Clear all spreadsheet data before loading file data. The load-from-file action is NOT a merge with existing content.
- Make sure formulas are properly evaluated after loading.
- You may assume only valid XML files will be loaded, but make sure loading is resilient to XML that has different ordering from what your saving code produces as well as extra tags. As a simple example, if you're always writing the <bgcolor> tag first for each cell followed by the <text> tag, then your loader must still support files that have these two in the opposite order. Also, if you didn't write more than these two tags within the <cell> content, your loader should just ignore extra tags when loading. See the example below.
- Use XML reading/writing capabilities from .NET. Do not write your own XML parsers that do things manually down at the string level. We discussed several options in class, such as XDocument, XmlDocument, XmlReader, and XmlWriter. The LINQ style XML queries make parsing easier once you figure them out.

If you saved:

```
<spreadsheet>
  <cell name="B1">
    <bgcolor>FF8000</bgcolor>
    <text>=A1+6</text>
  </cell>
</spreadsheet>
```

then you must be able to load:

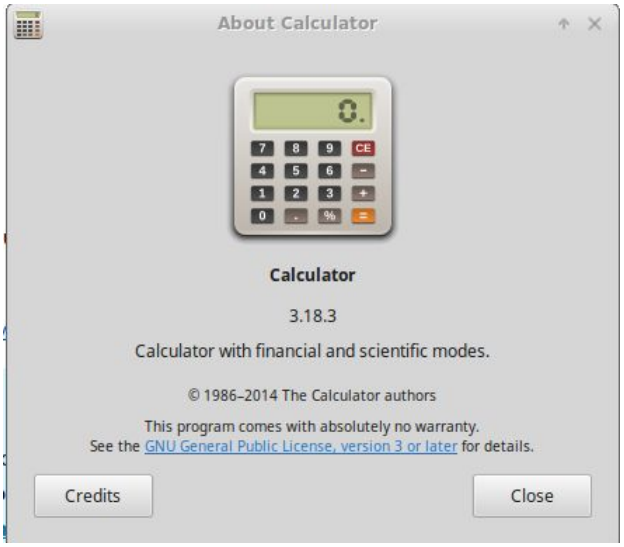
```
<spreadsheet>
  <cell unusedattr="abc" name="B1"> <text>=A1+6</text>
    <some_tag_you_didnt_write>blah</some_tag_you_didnt_write>
    <bgcolor>FF8000</bgcolor> <another_unused_tag>data</another_unused_tag>
  </cell>
</spreadsheet>
```

About Window in: Help -> About

In your menu bar, add a help drop down with an About option. This should bring up a second window with information about your application. The key components you need to have in the window are:

1. Program Title (like CptS 321 Spreadsheet or something cooler)
2. Program Version (currently 5.0) - this should be set in a static constant class variable
3. Your name & Contact email for your work
4. Copyright (this year)
5. Warranty and License (your choice on license, or all rights reserved)
    a. The LIcense should be generated at Creative Commons - License Generator
    b. Feel free top copy/paste their license imagery for your About window

Here's an example About window from the Linux Mint calculator app:



Requirement Details:

- Working menu (2 pts)
- Working Save to XML feature (5 pts)
- Working Load from XML feature (5pts)
- Working About window (3pts)