Mock assessment (based on an old PCS2 exam) MOCK

Date: Wednesday, June 19, 2019
Time: 12:45 – 15:15 h. (150 minutes)

Rules:

Admitted resources:

- You are allowed to use everything on paper (books, notes, etc.) and on your laptop, but only what you bring in: you are not allowed to borrow something from someone else.

- During the exam it is not allowed to use the network. You should make the assessment yourself: so, no communication with other students via Facebook, e-mail, Skype, mobile phone or whatever.

Way of working during the assessment and handing in your assessment:

During the assessment you will use a program that blocks your internet connection, takes care of downloading the assessment (this description and a start-up project) and lets you upload your solution.

Preliminary remarks:

- Whenever this assessment paper suggests the use of a certain name for a method, variable or anything else, you are required to indeed use that name.

- If you think you need more class-members, you are allowed to add more members to the classes than specified in this assessment.
- In this assessment we use the word "app", which is an abbreviation for "application" or "program".

Introduction: An app for keeping track of news-items of a magazine.

With this app it should be possible to keep track of news-items of a magazine. Authors can write news-items for a magazine. It should be possible to add news-items, to update news-items, to remove news-items and some more functionality.
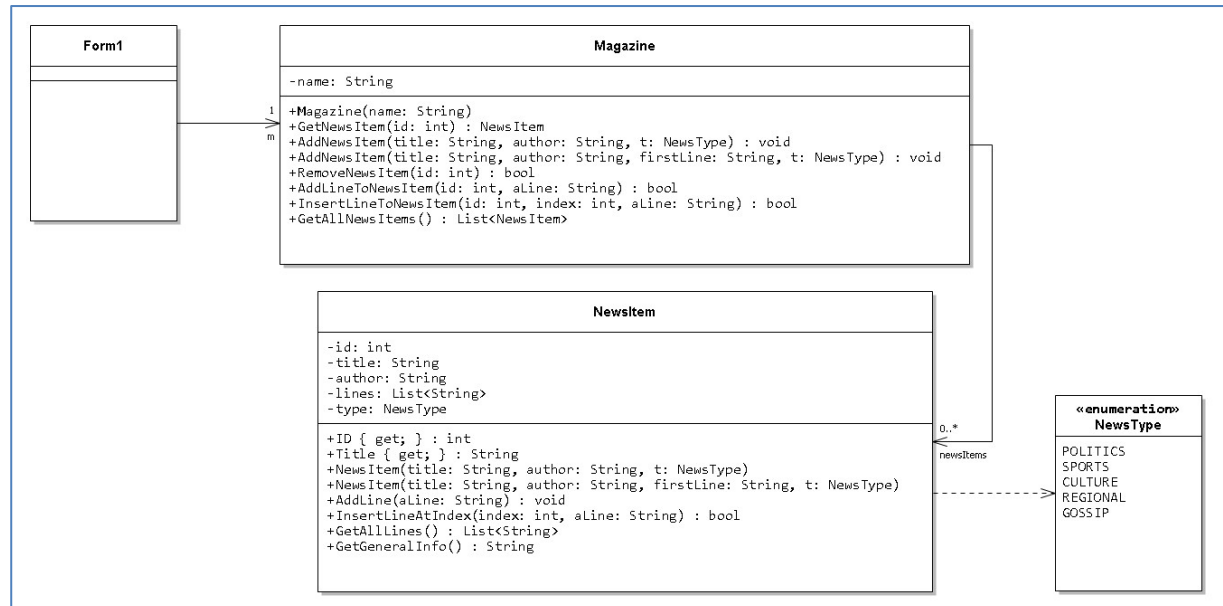For this app the user interface has already been made (see figure below).

About the code in the start-up project.

The start-up project contains some code, but it does not compile yet!
You have to implement some code to make it work.
More specific instructions will be provided in the assignments.

The class diagram for the desired solution is:

In this class diagram we use a List to represent a collection of objects. Whenever we talk about a collection, you are free to use a list or you may use an array: the choice is yours.

In Form1 there are several buttons.

- Clicking the button with text "fill magazine with some news-items" fills the magazine with some test data, which you are free to use. If you use the test data, after completing some code, it could result in the screendump above.
- The other buttons will be explained in the last assignment (blue buttons are already implemented; you have to implement the purple buttons).

The classes `Magazine` and `NewsItem` are empty.

The enum `Newstype` is missing.

Assignment 1 (3 points): About the enum NewsType.

Add the enum `NewsType` to the project according to the above class diagram.

Assignment 2 (4 + 4 + 8 + 10 + 4 = 30 points): About the class NewsItem.

Implement the class `NewsItem` according to the above class diagram. Some explanation and additional requirements:

a) Every news-item should have a title (for instance "Beyonce is back in town"), is written by an author (for instance "Medhat Riad"), has a unique identity-number and a collection of lines of text (some Strings, for instance the lines of text "Beyonce announced she will come to the Netherlands.", "She will perform in the Ziggo Dome." and "Tomorrow ticket-sale starts.").

b) The class `NewsItem` must have at least 2 properties: one for the identity-number and one for the title.

c) The class `NewsItem` must have 2 constructors: a constructor with 3 parameters and a constructor with 4 parameters.
One of these constructors has parameters for the title, the author and the newstype. This constructor creates a `NewsItem` –object with title, author and newstype as in the parameters and 0 lines of text.
The other constructor has 4 parameters: one for the title, one for the author, one for the first line of text and one for the newstype. This constructor creates a `NewsItem` –object with title, author, 1 line of text, newstype as in the parameters.
The news-item, that will be created first, should automatically get 101 as identity-number; the next news-item, that will be created, should get 102 as identity-number; the next gets 103; the next gets 104, etc.

d) There must be some methods:
- a method `AddLine` to add a line at the end of the lines;
- a method `InsertLineAtIndex` to insert a new line at a certain index, if possible. If possible the new line is inserted and the method returns true; otherwise nothing is inserted and the method returns false,
- a method `GetAllLines` that returns a collection of all lines.

e) There must be a method `GetGeneralInfo` that returns a string holding information about the id, title, author and the number of lines (as pictured in the screendump on page 1)

Assignment 3 (3+5+5+5+5+5+5+2 = 35 points): About the class Magazine

Implement the class `Magazine` according to the above class diagram. Some explanation and additional requirements:

a. A magazine has a name and a collection of news-items.

b. The constructor should create a `Magazine`–object with name as in the parameter and an empty collection of news-items.

c. The method `GetNewsItem` should return the news-item with the required id, if possible. If there is no such news-item, this method should return null.

d. The method `AddNewsItem` should add a news-item to the magazine. One of these methods should add a news-item with 0 lines of text, the other with 1 line of text.

e. The method `RemoveNewsItem` should, if possible, remove the news-item with that id from the magazine. In case it is successfully removed, this method should return true, otherwise it returns false.

f. The method `AddLineToNewsItem` should, if possible, add a line of text to the news-item with that id. In case it is successfully done, this method should return true, otherwise it returns false.

g. The method `InsertLineToNewsItem` should, if possible, insert a line of text to the news-item with that id. In case it is successfully done, this method should return true, otherwise it returns false.

h. The method `GetAllNewsItems` should return a collection of all news-items of the magazine.

Assignment 4 (6 + 6 + 10 + 10 = 32 points): The Form1-class

The button-handlers for the blue buttons are already implemented. You are not allowed to change it! It will be your task to implement the other buttons.

Assignments:

a. Clicking the button with text "add news-item" should add a news-item as specified in the orange groupbox.
   If the text in the textbox about "first line" is not equal to the empty string, the added news-item should have that string as the first line.
   If the text in the textbox about "first line" is equal to the empty string, the added news-item should have 0 lines.
   Make it work.

   Example: clicking this button in the situation as pictured in the screendump on page 1, a news-item with title "Summer in Egypt", author "Bert van Gestel" and newstype CULTURE will be added. It has "Summer starts soon" as the first line of text.
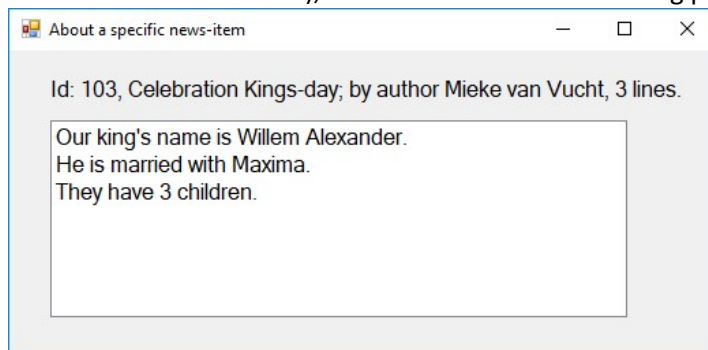
b. Clicking the button with text "show general info of all news-items" should show general information of all news-items in the listbox on the right-top. Make it work.

c. Clicking the button with text "show info of news-item" should check if there is a news-item with identity-number as specified in the textbox tbId.
   If so, it should show another window with a label and a listbox. In that label is shown general information of that news-item and in the listbox it shows all lines of text of that news-item. If not, show an appropriate message, telling the user that there is no such news-item.
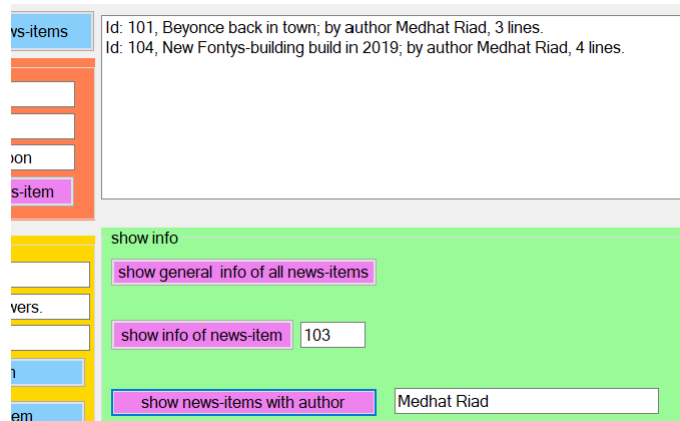
   Make it work.

   Example, clicking this button in the situation as pictured in the screendump on page 1, (so for 103 as possible id of a news-item), a window like in the following picture should be shown:

d. Clicking the button with text "show news-items with author" should check if there are news-item with the author inputted in the textbox tbFilterAuthor.
If so, it should show the general information of all news-items of that author in the listbox on the right-top.
If not, it should nothing in the listbox on the right-top.

Make it work by overloading the method GetAllNewsItems with an author. Hint: you will have to add code to the classes Magazine and NewsItem to implement this.

Example, clicking this button in the situation as pictured in the screendump on page 1, (so for Medhat Riad as possible author of a news-item), should show:

----- END of Mock Assessment -----