**DT228 OOP Lab Test**

**Rules of the Test!**

* **This is an open book test. You can use any books including ebooks and webcourses.**
* **General Internet usage is not allowed. Any use of Google, Facebook chat messaging etc will result of a mark of 0.**
* **You must submit a ZIP file of your Visual Studio solution folder through webcourses ONLY at the end of the test.**
* **This is an individual test. No talking or collaboration permitted.**
* **You should appropriately comment your code.**

The file hnr1.abc contains a list of music scores in the ABC music notation language. Each tune in the file consists of headers followed by the musical notes. Headers consist of a letter followed by a colon. For example T: indicates a tune title. Each new tune **always starts** with the X: header. The number following X: is called the tune index number. Tunes can have multiple titles, but we are only interested in the first and second title (known as the title and the alternative title). Here are the headers we are interested in for each tune:

X: The Index number  
T: The title. There can be multiple titles in each tune, but we are only interested in the first and second T header. Note not all tunes have more than one T header. The first T: will be the title and the second T header if there is one will become the altTitle.  
M: The time signature  
K: The key signature

Load up the file in Notepad and see if you can understand how it is organised before attempting the test.

Create a new C# Console Application and place this file into the bin/Debug folder of your Visual Studio Solution. For example, if you named your solution Tunes, the file would go into the folder:

Tunes\Tunes\bin\Debug

**Part 1**

Create a Tune class with the following fields, properties, a default constructor and a ToString method:

* index (an int)
* title (a string)
* altTitle (a string)
* keySig (a string)
* timeSig (a string)

Important! In the ToString method, you should return the properties separated by semicolons ; If a property is empty, then you should skip it. Here are some examples:

Belharbour Reel, The; Sailor's Return, The; G; C|

Ivy Leaf, The; Amix; C|

NOTE! The second tune has no value in the altTitle field, so we just skip the field.

**Part 2**

In your Program class, declare a static list of type Tune. If you missed Friday’s tutorial, you can have a look at the sample code (ListExample) for an example of how to use a list. Alternatively, use an array. Write a method:

static void LoadTunes(string filename)

This method should load the file and parse it into the list (or array) of Tune objects you declared. Set each tune property from the appropriate header in the tune in the file. For example:

X:105

T:Rolling in the Ryegrass

T:Shannon Breezes, The

R:reel

Z:id:hn-reel-105

M:C|

K:D

ABAF DFAF|G2BG dGBG|~A3F DFAF|GBAF EFDF|

~A3F DFAF|GFEF GBdB|ABAF DFAF|GBAF EFD2||

ABde ~f3d|~g3e fedB|ABde fefg|afdf (3efe dB|

ABde fAdf|~g3e fedB|ABde fefg|afdf efdB||

Results in a tune object where the index is 105, the title is “Rolling in the Ryegrass” the altTitle is “Shannon Breezes, The”, the timeSig is C| and the keySig is D.

**Part 3**

Write a static method in your program class:

static void PrintTunes(string keyword)

This method should print any tunes whose title or altTitle contains the keyword. This should be case insensitive. If keyword is null or empty, all tunes should be printed.

**Part 4**

In your main method, write code to first prompt the user to enter a keyword and receive the users input. When the user types a keyword you should print any tunes that contain the keyword in the title or altTitle properties by calling the PrintTunes method you wrote. When the user types quit, the program should quit. This should be in a loop. Below is a typical interaction:

Enter a keyword or quit to quit

maid

Dairy Maid, The; D; C|

Sligo Maid, The; Ador; C|

Mermaid of Mullaghmore, The; Maids of Mullach, The; Dmix; C|

Enter a keyword or quit to quit

breez

Rolling in the Ryegrass; Shannon Breezes, The; D; C|

Heather Breeze, The; Heathery Breeze; G; C|

Enter a keyword or quit to quit

quit

Goodbye

Marking Scheme:

Part 1: 20 marks

Part 2: 30 marks

Part 3: 20 marks

Part 4: 20 marks

Comments & style 10 marks