Kingston University London

Faculty of Science, Engineering and Computing

CI 6110 (Programming III): Music meta-data assignment

This is a summative assessment (it **does** count towards your final mark)

There will be a progress assessment during the period 24th November – 5th December, with Feedback provided at the same time.

The Hand-in date is 23:59 on Monday 9th January 2017.
You should submit your work via studyspace in two parts:

a) A zip file containing all Java projects developed as part of this assignment
b) A pdf document (2-page max!), submitted via turnitin, that describes:
   i. your target user stories
   ii. which of these stories your program(s) demonstrate
   iii. instructions that allow us to run the program(s) to see this demo

Final Feedback provided by Monday 30th January 2017.

Your assignment brief is to make a Desktop Application that helps users to manage their music collections more effectively. It should be written in Java, and preferably using the NetBeans IDE.

The exact set of functions performed by your Application is left up to you. Here are some example functions that you may wish to consider:

- Listing all the music tracks contained in selected folders in the user's computer hard-drives and removable storage media
- Make a playlist of the some of the user's music, somehow selected, to transfer onto a mobile device
- Compile a list of duplicate items, and albums with missing tracks, and perform filesystem operations to re-organise the items
- Allow the user to specify which music is most important to them, to allow selective backups
- Using web-services to locate additional information about the music stored on the user's hard drive and removable storage
- Use Libraries to provide information to the user about their music collection, and allow them to process it is useful ways.
- Make lists of music that the user may want to acquire, based on their collection
- Compare two users' music collections to show the similarities and differences
- Acoustic fingerprinting using Echoprint to identify tracks by an audio sample.
- Retrieving cover art for albums using Discogs or Cover Art Archive API's.
- Using web services to locate information and store in ID3 tags.
- Getting a list of events an artist has played at using Last.fm and plotting locations on a map.

Test Data:

- Your programme may work with audio files (e.g. mp3), or json files containing e.g. Lists of Objects about Music Metadata, or indeed other relevant data that is put to good use.
- **Please DON'T add any extra audio data to your submission zip – this will make it too big**
- A test data repository is available at K:\Teaching Materials\CI\CI6110\test-data – you can use this test data from this location, in your programs, if you are inside the University network, or you can copy test-data.zip for use when not at the University. It represents two music collections, A and B, with some overlap.

Technology that you may consider using, includes (but is not limited to):

- The Files and Paths packages to inspect and manipulate the filesystem
- Storage of data in any of text, JSON or xml formats.
- Libraries for reading ID3Tags (such as JAudioTagger and M3Agic)
- A command line user interface that allows the user to get things done by issuing parameterised commands, possibly through a menu system
- A graphical user interface that allows the user to access the functionality you provide with a mouse (and keyboard).
- Components that consume music meta-data web services, such as Music Brainz, Discogs, Decibel, 7Digital, Last.fm and the Echo Nest.
- Java Media Framework components for analysis of media files.

Marks will be assigned as follows:

Final In-class Demonstration 5 marks

- 1 mark for showing understanding of the relevant IDE controls.
- 2 marks for showing understanding of the source code components that you have authored or adapted.
- 2 marks for showing understanding of the extent and limitations of the test resources used by this application.

Final Submission: 10 marks

- 3 Marks for achieving the stated objective (helping users manage their music)
    - Full (3) marks will be awarded if you provide instructions that can be followed to run your program, which implements one or more of the user stories you have written, that clearly help users manage their music collection.
    - Marks will be deducted for: lack of instructions, programs that can't be executed, programs that don't follow any of your stories, stories that don't meet requirements
- 5 Marks for the extent of the functionality you have been able to include
    - 1 mark: Basic command line program
    - 2 marks: Advanced command line program, with attention paid to user experience
    - Extra marks awarded for GUI interface (in addition or as alternative), extraction of information from mp3 files, refactoring of M3UTools libray, use of web services
- 2 Marks for the inclusion of test code that validates the correctness of your solution.
    - 1 mark for workable unit tests for at least two methods you have written
    - 2 marks for an approach that aims to provide complete unit testing (excluding user interface components)

Timescale:

Original Assignment publication date:	Sunday 6th November
Progress report:	during the period 24th November – 5th December, with Feedback provided at the same time.
Deadline to submit the assignment to blackboard:	**Monday 9th January**
Feedback and Mark:	30th January

Notes:

- As mentioned above, don't include additional audio files in your submission, but instead use the test data on S:\Teaching Materials\CI\CI6110, or use data on your H: drive / laptop etc.

- All source code that is not your own (e.g. copied from the internet, written by a friend) should be clearly marked as such, with links as applicable. Failure to do so may lead to accusations of academic misconduct.