# CO1401 Coursework Assignment (BAE) 2019/20

# JET (Java Editor for Text)

## Deadline: 1st May 2020

In this coursework assignment you will create an editor for plain text files in Java using SWING. Your GUI design should be similar to existing basic text editors such as Windows Notepad or Wordpad. The focus of the assignment is on the GUI and utilizing all you have learned on CO1401.

The key features required for a pass mark of 40% are:

* Ability to create and edit new files.
* Ability to save files (note).
* Ability to load files.

You must use objects wherever possible in your implementation. For example, use an object to represent the file currently being edited and its state.

Advanced Features include:

* File menu (with load/save/exit commands minimally).
* Menu bar with icons for load/save.
* Support for multiple editing windows.
* Ability to search and replace.
* Ability to copy/paste to clipboard[[1]](#footnote-1).

# Submission

You must submit your source code application online through the appropriate link on Blackboard. You should submit a .zip file of your assignment Eclipse project folder. You must also submit a video demonstration of your application and source code as appropriate. Use a screen recorder (e.g. <https://camstudio.org/>) to record a demonstration of the features in the marking criteria that you have implemented in your app, along with a voice-over explanation. Please do not submit video files, instead please host your videos online (e.g. using OneDrive or YouTube/Unlisted etc). Use the second submission link to submit the URL/instructions to access your video demo.

# Marking Criteria

**To gain 40% you must have:**

* Created a Java SWING program using a JTextArea or similar to allow the user to input and edit text over multiple lines.
* In addition to the above, the program allows the user to successfully save the text they have input to a plain text file.
* In addition to the above, the program prompts the user to save the file if the user attempts to close the program without saving their most recent changes. Program may contain bugs, but is largely functional
* In addition to the above, the program allows the user to load in text from an existing file which is then visible to the user, the program also makes use of an objects to represent the text/file within the program.

**To gain 50% you must have** *(in addition to the above):*

* Add a File menu, with load/save/exit commands minimally, which works correctly.
* Code is correctly indented and appropriately commented.
* Program allows users to adjust preferences e.g. text display size.
* Program uses a suitable layout manager to control layout for different size windows.

**To gain 60% you must have** *(in addition to the above):*

* Add a ‘menu’ bar with icons to the top of the GUI, with icons for load/save/exit commands minimally, which works correctly.
* Code demonstrates use of Inheritance.
* Application demonstrates robustness through the appropriate throwing or handling of Java Exceptions.
* Chosen preferences persist when the application is restarted.

**To gain 70% you must have** *(in addition to the above):*

* Program supports multiple simultaneous editing windows
* Code written demonstrates Polymorphic behavior.
* Program is bug free upon testing by tutors
* Good code structure providing an elegant solution

**To gain 80%+ you must have** *(in addition to the above):*

* Program allows the user to search and replace strings within the document
* Program can paste text from the system clipboard into the document.
* Program can copy text from the document to the system clipboard.
* Program contains at least one extra feature that would be of use in a light weight text editor.
* Excellent code structure providing an elegant solution.

**Late work:** Except where an extension of the hand-in deadline date has been approved, work that is handed in within 5 working days late will receive a maximum mark of 40%. Work handed in later than this will receive 0%.

**Cheating:** The consequences of cheating in assessments are serious - you will fail the module. Cheating is using or attempting to use unfair means to enhance performance. This includes plagiarism (presenting someone else's work, or work found online as if it was your own), collusion (working with others on an individual assignment) and allowing other students to access your work. Make sure that you do not give someone the opportunity to steal your work (e.g. by asking them to print it out for you). If you have any doubt about what cheating is or how to reference material properly, please ask a tutor.

# Known Bugs

* Cannot un-bold text on default font

# Possible Features

* Add preview on font dialog and make it not live
* Make toolbar buttons sync with the prefs
* Maybe remove the bold preference

1. Look at <https://docs.oracle.com/javase/7/docs/api/java/awt/datatransfer/Clipboard.html> and Toolkit.getDefaultToolkit().getSystemClipboard() [↑](#footnote-ref-1)