

Arizona State University  
CSE 360: Introduction to Software Engineering  
Fall 2019

## Text Formatter User Guide

Saeed Almazrouei

Benjamin Danek

Samira Said

Rachel White

## Table of Contents

|                              |        |
|------------------------------|--------|
| Table of Contents.....       | Page 2 |
| Introduction.....            | Page 3 |
| Overview of the Program..... | Page 3 |
| External Requirements.....   | Page 4 |
| Getting Started.....         | Page 5 |
| Error Conditions.....        | Page 6 |
| Ending.....                  | Page 7 |

## **Introduction**

The purpose of this document outlines the main overview of the Text Formatter program. This includes the pre-requirements to execute the program, an in depth set of instruction on how-to on get started, a successful sample run of the program with a look into the User Interface, as well as error messages and conditions for the program. It is important to read the user guide to understand the functionality of the Text Formatter program for the best results.

## **Overview of the Program**

The Text Formatter program takes in a given text file selected by the user, processes the file, and returns it back as an output file. While processing the file, the program analyzes the set of commands which are embedded within the text and translates the file into a formatted version based on the commands given.

At the start of the program, the user is initially greeted with an open button that allows the user to select a file for the program to process. The user's input file is then imported into the program and is properly formatted based on the commands embedded within the text which includes justify, breaks, and other text formatting options. If errors are found in the input file such as incorrect extension or unrecognized command, process the file as much as possible and then print an error message on the screen of the file's status. Once the program completely processes the file with little to no exceptions, the program then shows a button where the user is able to save the output file, which indicates the status of the file as complete.

Below is a starting list of commands covered by the Text Formatter program.

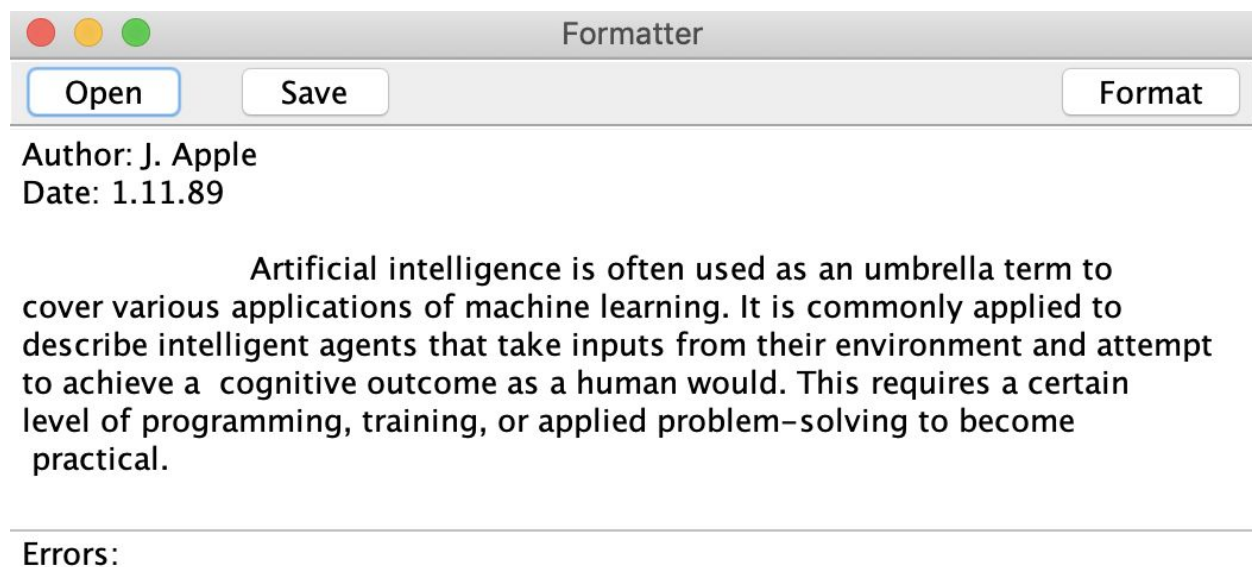
|     |                                 |
|-----|---------------------------------|
| -n# | (number of characters per line) |
| -r  | (right justified)               |
| -l  | (left justified)                |
| -c  | (center justified)              |
| -e  | (equal distribution)            |
| -w+ | (wrap on)                       |
| -w- | (wrap off)                      |
| -s  | (single space)                  |
| -d  | (double space)                  |
| -t  | (title)                         |
| -p# | (paragraph)                     |
| -b# | (blank line)                    |
| -a# | (number of columns)             |

## External requirements

This program is designed to work on macOS (Mojave or newer) or Windows (8 or newer) personal computers. To begin with the program, pre requirements must be met in order for installation to succeed. The Text Formatter is packaged as a JAR file (Java Archive), which indicates that users must have access to a JRE (Java Runtime Environment) on their device in order to run the program. It is also required that an input text file with a .txt extension is selected for the program to generate an output file. The program does not use any external applications.

## Getting Started

The first step to getting started is installing the source file from Github. Once the source file is downloaded onto your device, double click and open the file and run the executable: TextFormatter.exe. Once the program has been installed, the program will launch with a GUI as the User Interface of the program. Below is a sample of the User Interface of the project with an input file that has been opened and formatted into the text area.



To begin the process, proceed to the following steps to format a text file.

1. Initiate the process of the program by clicking on the 'Open' button and selecting a text (.txt) file from the computer directory. The now ingested text should be visible in the main textbox of the program

2. Clicking format will parse the source text for formatting flags, and reformat the source text to match the flags. The intended role of the user is to include the formatting flags as desired to describe where and how a formatting change belongs. Use the scroll bar down this section to see the entire text.
3. If any errors are seen, the program will display the error condition in the 'Error' box on the bottom section in red text.
4. Once the desired results are attained, click on the 'Save' button to generate a file that contains the results labeled as Output.txt in the same directory as the source file.
5. Select the Exit button displayed on the top left of the program to exit out of the program.

### **Error Conditions**

While using the Text Formatter Program, general conditions such as unrecognized commands will be handled by the program; however, all actions marked as unrecognized by the program will trigger an error message printed on the screen.

This section lists all errors and warnings users may encounter when using the Text Formatter Program.

| <b>Condition</b>             | <b>Definition</b>   |
|------------------------------|---|
| Unable to Open Input File    | The input file cannot be found or opened, or it is not a .txt file. |
| Unable to Create Output File | The output file could not be created.                               |

### **Ending**

The Text Formatter is a software program that allows users to input a text file which then analyzes the set of commands embedded in the text and converts the file into a formatted version based on the provided commands.

This user guide document for the Text Formatter program has covered the necessary pre-requirements, instructions on how-to get started, error conditions, and a successful sample run of the program with a look into the User Interface.