Wet Pancake CLI 1.0

- Project URL: https://gitlab.com/craig.oates/Wet-Pancake
- Requires .Net 4.7+
- Tested on Windows 10 only

Wet Pancake CLI is a console program for generating text. Its main aim is to provide a direct way to interact with the Wet Pancake NuGet package -- outside of a coding environment. Please refer to the URL above for more information about the project.

Disclaimer: This is a technical document, written for software developers. Not for general consumption.

Note about File Paths

Standard console file path rules apply. When entering file paths, wrap them in the quotes (" ") if the path contains spaces. You can omit them if there are no spaces.

CleanText (sentences: int) (text: string)

Checks to see if the string matches the desired sentence count and removes any over that limit. If the string has less sentences than the number requested, it will not change. Sentences must be greater than 0 and the text must contain at least one "." "?" "!"

Example: `CleanText 1 "This is a test sentence. And, this one needs to be removed."`

Exit ()

Exits out of the program.

Example: `Exit`

Help ()

Lists out the commands this program offers.

Example: `Help`

MakeCleanText (gibberish-level: int) (sentences: int) (copy-to-clipboard: bool)

Makes text using the gibberish-level and number of sentences specified by the user. The result goes through an extra "cleaning" process to remove any artefact sentences. Use this if you cannot tolerate the odd extra sentence. With that said, it does mean it is slower than its MakeText counterpart. Gibberish-level must be between 2 and 20. Sentences must be greater than 0. Pass in "true" to copy the result straight to

the clipboard. Pass in "false" or leave blank to not copy the result.

Examples:

- `MakeCleanText 5 10 true`
- `MakeCleanText 3 7 false`
- `MakeCleanText 9 12`

MakeCleanTextFromFile (gibberish-level: int) (sentences: int) (file path: string) (copy-to-clipboard: bool)

Loads the specified .txt file and makes text based on it, using the gibberish-level and number of sentences specified by the user. The result goes through an extra "cleaning" process to remove any artefact sentences. Use this if you cannot tolerate the odd extra sentence. With that said, it does mean it is slower than its MakeTextFromFile counterpart. Gibberish-level must be between 2 and 20. Sentences must be greater than 0. Pass in "true" to copy the result straight to the clipboard. Pass in "false" or leave blank to not copy the result.

Examples:

- `MakeCleanTextFromFile 3 6 C:/yourfile.txt true`
- `MakeCleanTextFromFile 5 9 C:/yourfile.txt false`
- `MakeCleanTextFromFile 7 15 C:/yourfile.txt`

MakeRandomText (copy-to-clipboard: bool)

Makes random text, the number of sentences Maked is randomly determined. Pass in true to copy result straight to clipboard. Pass in "false" or leave blank to not copy the result.

Examples:

- `MakeRandomText true`
- `MakeRandomText false`
- `MakeRandomText`

MakeText (gibberish-level: int) (sentences: int) (copy-to-clipboard: bool)

Makes text using the gibberish-level and number of sentences specified by the user. This command does not run the result through the extra "cleaning" process like MakeCleanText. This means this command is faster but it might produce an extra sentence on the odd occasion. Use this if you prefer speed over accuracy. Gibberish-level must be between 2 and 20. Sentences must be greater than 0. Pass in "true" to copy the result straight to the clipboard. Pass in "false" or leave blank to not copy the result.

Examples:

- `MakeText 5 10 true`
- `MakeText 7 15 false`
- `MakeText 10 5`

MakeTextFromFile (gibberish-level: int) (sentences: int) (file path: string)

Loads the specified .txt file and makes text based on it, using the gibberish-level and number of sentencesspecifiedby the user. This command does not run the result through the extra "cleaning" process like MakeCleanTextFromFile. This means this command is faster but it might produce an extra sentence on the odd occasion. Use this if you prefer speed over accuracy. Gibberish-level must be between 2 and 20. Sentences must be greater than 0. Pass in "true" to copy the result straight to the clipboard. Pass in "false" or leave blank to not copy the result.

Examples:

- `MakeTextFromFile 3 6 C:/yourfile.txt true`
- `MakeTextFromFile 6 13 C:/yourfile.txt false`
- `MakeTextFromFile 12 8 C:/yourfile.txt`

QuickGuide (file path: string)

Writes a quick guide to the specified location or the desktop if no path was specified.

Examples:

- `QuickGuide C:/your/filePath`
- `QuickGuide`

RequestAllTemplateFiles ()

Returns a list of all the available .txt files built-in to Wet Pancake.

Example: `RequestAllTemplateFiles`

Test ()

Prints a test message to the console.

Example: `Test`

ValidateFile (file path: string)

Checks the text in the specified .txt file to see if it is compatible with this program. For a file to be compatible, it must be a .txt file and contain at least one '.', '!' or '?'.

Example: `ValidateFile "C:/your-file.txt"`