**Description**

I want to develop a java syntax highlighter that will take a sentence and highlight the configured keywords into system.

**Scenario #1:** Highlight given sentence if matches configured keywords

**Given** asyntax highlighter into place

**When** I have input following text to system

I am going to join java mentoring program to learn cool stuff in fun way.

**Then** the system should return me highlighted text as

I [bold] am [/bold] going [italic] to [/italic] join java mentoring program [italic] to [/italic] learn cool stuff [underline] in [/underline] fun way.

**Scenario #2:** Don't highlight given sentence doesn't contain any configured keyword

**Given** asyntax highlighter into place

**When** I have input following text to system

I know everything, so it's waste of effort.

**Then** the system should return me highlighted text as

I know everything, so it's waste of effort.

**Configuration information**

Current following words can be configured as special keywords, those need highlighting:

**am 🡪 [bold]**

**to 🡪 [italic]**

**in 🡪 [underline]**

**Technical information**

The existing application provides SyntaxHighlighter interface; which can be used as initial point of developing solution. Moreover, there should be appropriate unit test cases for all implemented classes.

**Review**

More testing would reveal some edge cases.

1. What exactly defines the start of a word. And what defines the end? Could test for

“To the end” -> “To the end” //improve?

“Go (to the end)” -> “Go (to the end)” //improve?

“Go to, the end” -> “Go to, the end” //improve?

“Go [to the] end” -> “Go [to the] end” //improve?

“Go [bold]to the end” -> “Go [bold]to the end” //improve?

“Go [italic]to[/italic]” -> “Go [italic]to[/italic]” //improve?

“Go [bold] to [/bold]” -> “Go [bold] [italic]to[/bold] [/italic]”

I may want to adjust the encoded output to insert a space. This would allow application of two formats to one word. And match Pankaj’s exact intro.

“To the end” -> “[italic] To [/italic] the end”

“Go (to the end)” -> “Go (to the end)”

“Go to, the end” -> “Go to, the end”

“Go [to the] end” -> “Go [to the] end”

“Go [bold]to the end” -> “Go [bold]to the end”

“Go [italic]to[/italic]” -> “Go [italic]to[/italic]”

“Go [bold] to [/bold]” -> “Go [bold] [italic] to [/italic] [/bold]” //improve?