

IT UNIVERSITY OF COPENHAGEN

BDSA 2014

Assignment 37

Anders Wind Steffensen - awis@itu.dk
Christopher Blundell - mail@itu.dk
Pierre Mandas - mail@itu.dk

Generated by Doxygen 1.8.8

Fri Sep 19 2014 22:00

Contents

1	Namespace Index	1
1.1	Packages	1
2	Class Index	3
2.1	Class List	3
3	Namespace Documentation	5
3.1	Package TextFileReaderProject	5
4	Class Documentation	7
4.1	TextFileReaderProject.TextSearcher Class Reference	7
4.1.1	Detailed Description	7
4.1.2	Constructor & Destructor Documentation	7
4.1.2.1	TextSearcher	8
4.1.2.2	TextSearcher	8
4.1.3	Member Function Documentation	8
4.1.3.1	createRegex	8
4.1.3.2	print	8
4.2	TextFileReaderProject.TextSearcherTests Class Reference	8
4.2.1	Detailed Description	9
4.2.2	Member Function Documentation	9
4.2.2.1	TestCreateRegexEmptyInput	9
4.2.2.2	TestCreateRegexPlusCommand	9
4.2.2.3	TestCreateRegexStarPrefixCommand	9
4.2.2.4	TestCreateRegexStarSuffixCommand	9
4.2.2.5	TestCreateRegexTwoCommandsInput	9
4.2.2.6	TestDateRegex	9
4.2.2.7	TestURLRegex	10
	Index	11

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

TextFileReaderProject	5
--	-------------------

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

TextFileReaderProject.TextSearcher

A utility class that allows the user to search for a single keyword in the text, and then print it in the terminal 7

TextFileReaderProject.TextSearcherTests

A testclass that tests the TestSearcher class' methods. 8

Chapter 3

Namespace Documentation

3.1 Package TextFileReaderProject

Classes

- class **TextFileReader**
Utility class for reading text files into a single string.
- class **TextSearcher**
A utility class that allows the user to search for a single keyword in the text, and then print it in the terminal
- class **TextSearcherTests**
A testclass that tests the TestSearcher class' methods.

Chapter 4

Class Documentation

4.1 TextFileReaderProject.TextSearcher Class Reference

A utility class that allows the user to search for a single keyword in the text, and then print it in the terminal

Public Member Functions

- **TextSearcher** (string inputFile)

*The constructor tries to load the textfile 'inputfile' in the debug folder. If inputfile is invalid, the default text file 'TestFile.txt' will be loaded. While running, the user can enter new keywords to search for. Commands: + between two words, * prefix, * suffix.*

- **TextSearcher** ()

Empty constructor

- string **createRegex** (string input)

*Generates a regular expression string based on the user input. Three different commands are available, but there can only be one command per regular expression. Commands: + between two words, * prefix, * suffix.*

- void **print** (string content, string keyword)

Finds matches for URL, DATES and KEYWORD matches. Prints URLs with blue font. Prints DATES with green font. Prints KEYWORDS with yellow background. Cannot handle multiple formattions eg. url and keyword matches.

Static Public Member Functions

- static void **Main** (string[] args)

Properties

- string **urlRegex** [get]
- string **dateRegex** [get]

4.1.1 Detailed Description

A utility class that allows the user to search for a single keyword in the text, and then print it in the terminal

4.1.2 Constructor & Destructor Documentation

4.1.2.1 TextFileReaderProject.TextSearcher.TextSearcher (string *inputFile*)

The constructor tries to load the textfile 'inputfile' in the debug folder. If inputfile is invalid, the default text file 'TestFile.txt' will be loaded. While running, the user can enter new keywords to search for. Commands: + between two words, * prefix, * suffix.

Parameters

<i>inputFile</i>	The text file the class will print and search for keywords in
------------------	---

4.1.2.2 TextFileReaderProject.TextSearcher.TextSearcher ()

Empty constructor

4.1.3 Member Function Documentation

4.1.3.1 string TextFileReaderProject.TextSearcher.createRegex (string *input*)

Generates a regular expression string based on the user input. Three different commands are available, but there can only be one command per regular expression. Commands: + between two words, * prefix, * suffix.

- means that two adjacent keywords should be present one after the other, e.g. text + file matches "text file" but does not match "text big file"
- after a keyword means search for words starting with keyword e.g. inter* matches "internet"
- before a keyword means search for words ending with keyword e.g. *net matches "internet" and "intranet"

Parameters

<i>input</i>	The keyword the user searches after
--------------	-------------------------------------

Returns

A regular expression string created from the user input

4.1.3.2 void TextFileReaderProject.TextSearcher.print (string *content*, string *keyword*)

Finds matches for URL, DATES and KEYWORD matches. Prints URLs with blue font. Prints DATES with green font. Prints KEYWORDS with yellow background. Cannot handle multiple formattions eg. url and keyword matches.

Parameters

<i>content</i>	The text that needs to be printed
<i>keyword</i>	The search keyword in a regular expression format

The documentation for this class was generated from the following file:

- C:/Users/Christopher/Documents/BDSA-Exercises/BDSA2014/TextFileReader/TextSearcher.cs

4.2 TextFileReaderProject.TextSearcherTests Class Reference

A testclass that tests the TestSearcher class' methods.

Public Member Functions

- void **TestCreateRegexEmptyInput** ()
Test the CreateEgex method for empty input. Expected "";
- void **TestCreateRegexTwoCommandsInput** ()
Tests the CreateRegex method for a few inputs which has multiple keywords or commands.
- void **TestCreateRegexPlusCommand** ()
Tests the CreateRegex method's ability to create plus command regex when input is correct.
- void **TestCreateRegexStarPrefixCommand** ()
*Tests the CreateRegex method's ability to create * prefix command regex when input is correct.*
- void **TestCreateRegexStarSuffixCommand** ()
*Tests the CreateRegex method's ability to create * suffix command regex when input is correct.*
- void **TestURLRegex** ()
Tests the Regex which is supposed to find urls in a text. Only URLs starting with http(s):// are accepted.
- void **TestDateRegex** ()
Tests the Regex which is supposed to find dates in a text. The only date format that it supports is the one given in the TestFile.txt

4.2.1 Detailed Description

A testclass that tests the TestSearcher class' methods.

4.2.2 Member Function Documentation

4.2.2.1 void TextFileReaderProject.TextSearcherTests.TestCreateRegexEmptyInput ()

Test the CreateEgex method for empty input. Expected "";

4.2.2.2 void TextFileReaderProject.TextSearcherTests.TestCreateRegexPlusCommand ()

Tests the CreateRegex method's ability to create plus command regex when input is correct.

4.2.2.3 void TextFileReaderProject.TextSearcherTests.TestCreateRegexStarPrefixCommand ()

Tests the CreateRegex method's ability to create * prefix command regex when input is correct.

4.2.2.4 void TextFileReaderProject.TextSearcherTests.TestCreateRegexStarSuffixCommand ()

Tests the CreateRegex method's ability to create * suffix command regex when input is correct.

4.2.2.5 void TextFileReaderProject.TextSearcherTests.TestCreateRegexTwoCommandsInput ()

Tests the CreateRegex method for a few inputs which has multiple keywords or commands.

4.2.2.6 void TextFileReaderProject.TextSearcherTests.TestDateRegex ()

Tests the Regex which is supposed to find dates in a text. The only date format that it supports is the one given in the TestFile.txt

4.2.2.7 void TextFileReaderProject.TextSearcherTests.TestURLRegex ()

Tests the Regex which is supposed to find urls in a text. Only URLs starting with http(s):// are accepted.

The documentation for this class was generated from the following file:

- C:/Users/Christopher/Documents/BDSA-Exercises/BDSA2014/TextFileReader/TextSearcher.tests.cs

Index

- createRegex
 - TextFileReaderProject::TextSearcher, [8](#)
- print
 - TextFileReaderProject::TextSearcher, [8](#)
- TestCreateRegexEmptyInput
 - TextFileReaderProject::TextSearcherTests, [9](#)
- TestCreateRegexPlusCommand
 - TextFileReaderProject::TextSearcherTests, [9](#)
- TestCreateRegexStarPrefixCommand
 - TextFileReaderProject::TextSearcherTests, [9](#)
- TestCreateRegexStarSuffixCommand
 - TextFileReaderProject::TextSearcherTests, [9](#)
- TestCreateRegexTwoCommandsInput
 - TextFileReaderProject::TextSearcherTests, [9](#)
- TestDateRegex
 - TextFileReaderProject::TextSearcherTests, [9](#)
- TestURLRegex
 - TextFileReaderProject::TextSearcherTests, [9](#)
- TextFileReaderProject, [5](#)
- TextFileReaderProject.TextSearcher, [7](#)
- TextFileReaderProject.TextSearcherTests, [8](#)
- TextFileReaderProject::TextSearcher
 - createRegex, [8](#)
 - print, [8](#)
 - TextSearcher, [7](#), [8](#)
- TextFileReaderProject::TextSearcherTests
 - TestCreateRegexEmptyInput, [9](#)
 - TestCreateRegexPlusCommand, [9](#)
 - TestCreateRegexStarPrefixCommand, [9](#)
 - TestCreateRegexStarSuffixCommand, [9](#)
 - TestCreateRegexTwoCommandsInput, [9](#)
 - TestDateRegex, [9](#)
 - TestURLRegex, [9](#)
- TextSearcher
 - TextFileReaderProject::TextSearcher, [7](#), [8](#)