TextMarkupParser

Introduction

TextMarkupParser is a class that can parse markup language. Markup Language is a type of syntax that identifies specific substrings into different tags, HTML and XML are Markup Languages. The string below can also be considered a Markup Language:

Lorem <color value=red>ipsum</color> dolor sit <mytag>amet</mytag>, <outer>consectetur adipiscing <inner>elit</inner></outer>

Here, we can categorize the text into 4 “*words*”.

A *word* is a subset of the text that is enclosed by a tag. Words can have tags or not, for example, the word *“dolor sit”* has no tags, while*”amet”* has one tag, (*mytag*), and *“elit”* has two tags (*outer* and *inner*)

Tags can also have parameters attributed to them, as for example the word “*ipsum*”, which has the tag “*color*”, has a parameter called “*value*” attached to it

How to use

TextMarkupParser is a simple class that needs to be instanced to be used, and each instance is associated with a string:

```
TextMarkupParser parser = new TextMarkupParser();
TextMarkupParser.TaggedText taggedText = parser.Parse(rawText);
```

Objects

TextMarkupParser

class
Represents the object the is responsible from parsing the markup text

Constructors

TextMarkupParser()
Creates a new TextMarkupParser.

Methods

TextMarkupParser.TaggedText Parse(string text)
Parses a string

TextMarkupParser.TaggedText

class
Represents the entire tagged text with all the words and tags

Constructors

TextMarkupParser.TaggedText(List<TextMarkupParser.TaggedWord> words)
Creates a new TaggedText object with the specified words.

Parameters

List<TextMarkupParser.TaggedWord> words
The list of all words in the text

Methods

string GetText()
Returns the raw text without any markup string

List<TextMarkupParser.TagData> GetDataFromIndex(int charIndex)
Returns all the tags associated with the word that contains the character defined by *charIndex*

TextMarkupParser.TagData GetTag(List<TextMarkupParser.TagData> tagData, string name)
Returns the tag with specified name from a list of tags

TextMarkupParser.TagData GetTagFromIndex(int charIndex, string name)
Returns the tag with the specified *name* associated with the word that contains the character defined by *charIndex*

TextMarkupParser.TagData

class
Represents a single tag that can be in a word, and all its parameters

Constructors

TextMarkupParser.TagData()
Default constructor

Parameters

string name
The tag name

Dictionary<string, string> pms
All the parameters and values of this tag

Methods

string GetValue(string key, string defaultValue)
Returns the parameters value for the specified parameter name, or *defaultValue* if there is no parameter with the specified name

TextMarkupParser.TaggedWord

struct
Represents a single word that has a list of tags

Fields

string text
The raw word text with no markup syntax

List<TextMarkupParser.TagData> data
All the markup data translated into *TextMarkupParser.TagData* objects