

go documentation

build passing

# CamelCase

CamelCase is a Golang (Go) package to split the words of a camelcase type string into a slice of words. It can be used to convert a camelcase word (lower or upper case) into any type of word.

## Splitting rules:

1. If string is not valid UTF-8, return it without splitting as single item array.
2. Assign all unicode characters into one of 4 sets: lower case letters, upper case letters, numbers, and all other characters.
3. Iterate through characters of string, introducing splits between adjacent characters that belong to different sets.
4. Iterate through array of split strings, and if a given string is upper case:
  - if subsequent string is lower case:
    - move last character of upper case string to beginning of lower case string

## Install

```
go get github.com/fatih/camelcase
```

## Usage and examples

```
splitted := camelcase.Split("GolangPackage")

fmt.Println(splitted[0], splitted[1]) // prints: "Golang", "Package"
```

Both lower camel case and upper camel case are supported. For more info please check:

<http://en.wikipedia.org/wiki/CamelCase>

Below are some example cases:

```
" " => []
"lowercase" => ["lowercase"]
"Class" => ["Class"]
"MyClass" => ["My", "Class"]
"MyC" => ["My", "C"]
"HTML" => ["HTML"]
"PDFLoader" => ["PDF", "Loader"]
"AString" => ["A", "String"]
"SimpleXMLParser" => ["Simple", "XML", "Parser"]
"vimRPCPlugin" => ["vim", "RPC", "Plugin"]
"GL11Version" => ["GL", "11", "Version"]
"99Bottles" => ["99", "Bottles"]
"May5" => ["May", "5"]
"BFG9000" => ["BFG", "9000"]
"BöseÜberraschung" => ["Böse", "Überraschung"]
```

```
"Two spaces" => ["Two", " ", "spaces"]  
"BadUTF8\xe2\xe2\xa1" => ["BadUTF8\xe2\xe2\xa1"]
```