Fiserv Test Developer

Using an *HTTP GET* method, retrieve information from Wikipedia using a given topic. Query *https://en.wikipedia.org/w/api.php?action=parse&section=0&prop=text&format=json&page=[topic]* to get the *topic* Wikipedia article. Return the total number of times that the string *[topic]* appears in the article's *text* field.

Note: the search is case-sensitive

The query response from the website is a JSON object described below:

- *parse*: A JSON object representing the article's parsed web page. It has the following three fields:
    1. *title*: The article's title, as specified by the argument passed as *topic*
    2. *pageid*: The article's Page ID
    3. *text*: A JSON object that contains the Wikipedia article as an HTML dump

Function Description

Returns:

int: an integer, the number of times the search term *topic* appears in the returned *text* field

Input Format for Custom Testing

Input from stdin will be processed as follows and passed to the function.

Sample Input

```
STDIN    Function
-----    -----
pizza  →  topic = 'pizza'
```

Sample Output*

```
149
```

*Note that because this question is dynamically getting the data from Wikipedia, the actual number of occurrences may have changed. 149 is only used as an example.

Explanation

The query is *https://en.wikipedia.org/w/api.php?action=parse&section=0&prop=text&format=json&page=pizza* and the response is:


```
{
 "parse": {
  "title": "Pizza",
  "pageid": 24768,
  "text": {
   "*": "<div role=\"note\" class=\"hatnote\">For other uses, see <a
href=\"\/wiki\/Pizza_(disambiguation)\" class=\"mw-disambig\" title=\"Pizza
(disambiguation)\">Pizza (disambiguation)<\/a>.<\/div>\n<table class=\"infobox hrecipe
adr\" style=\"width:22em\">\n<caption
class=\"fn\"><span>Pizza<\/span><\/caption>\n<tr>\n<td colspan=\"2\" style=\"text-
align:center\"><a href=\"\/wiki\/File:Pepperoni_pizza.jpg\" class=\"image\"><img
alt=\"Pepperoni pizza.jpg\"
src=\"\/\/upload.wikimedia.org\/wikipedia\/commons\/thumb\/d\/d1\/Pepperoni_pizza.jp
g\/220px-Pepperoni_pizza.jpg\" width=\"220\" height=\"139\"
srcset=\"\/\/upload.wikimedia.org\/wikipedia\/commons\/thumb\/d\/d1\/Pepperoni_pizz
a.jpg\/330px-Pepperoni_pizza.jpg 1.5x,
\/\/upload.wikimedia.org\/wikipedia\/commons\/thumb\/d\/d1\/Pepperoni_pizza.jpg\/44
0px-Pepperoni_pizza.jpg 2x\" data-file-width=\"959\" data-file-height=\"606\"
\/><\/a>\n<div style=\"padding-bottom:0.25em;border-bottom:1px solid #aaa;\">Pizza
topped with <a href=\"\/wiki\/Pepperoni\"
title=\"Pepperoni\">pepperoni<\/a><\/div>\n<\/td>\n<\/tr>\n<tr>\n<th scope=\"row\"
style=\"padding-top:0.245em;line-height:1.15em; padding-
right:0.65em;\">Type<\/th>\n<td><a href=\"\/wiki\/Flatbread\"
title=\"Flatbread\">Flatbread<\/a><\/td>\n<\/tr>\n<tr>\n<th scope=\"row\"
style=\"padding-top:0.245em;line-height:1.15em; padding-
right:0.65em;\">Course<\/th>\n<td>Lunch or dinner<\/td>\n<\/tr>\n<tr
class=\"note\">\n<th scope=\"row\" style=\"padding-top:0.245em;line-height:1.15em;
padding-right:0.65em;\">Place of origin<\/th>\n<td class=\"country-name\"><a
href=\"\/wiki\/Naples\" title=\"Naples\">Naples<\/a>, <a href=\"\/wiki\/Campania\"
title=\"Campania\">Campania<\/a>, <a href=\"\/wiki\/Italy\"
title=\"Italy\">Italy<\/a><\/td>\n<\/tr>\n<tr>\n<th scope=\"row\" style=\"padding-
top:0.245em;line-height:1.15em; padding-right:0.65em;\">Serving
temperature<\/th>\n<td>Hot or warm<\/td>\n<\/tr>\n<tr>\n<th scope=\"row\"
style=\"padding-top:0.245em;line-height:1.15em; padding-right:0.65em;\">Main
ingredients<\/th>\n<td class=\"ingredient\">Dough, often <a href=\"\/wiki\/Tomato_sauce\"
title=\"Tomato sauce\">tomato sauce<\/a>, <a href=\"\/wiki\/Cheese\"
title=\"Cheese\">cheese<\/a><\/td>\n<\/tr>\n<tr>\n<th scope=\"row\" style=\"padding-
top:0.245em;line-height:1.15em; padding-right:0.65em;\">Variations<\/th>\n<td><a
href=\"\/wiki\/Calzone\" title=...
  }
 }
}
```

As of April, 2020, the word '*pizza'* occurs 149 times in the HTML dump of the response article (i.e., *text*) from Wikipedia.

1.) Write the program
2.) Describe, develop or show how to test the code (JUnit) / validate solution
3.) Make it user friendly – we always expect a valid response (Input validation)
4.) Pls. upload the code into GitHub