Problem 2129: Capitalize the Title

Problem Information

Difficulty: Easy
Acceptance Rate: 0.00%
Paid Only: No

Problem Description

You are given a string

title

consisting of one or more words separated by a single space, where each word consists of English letters.

Capitalize

the string by changing the capitalization of each word such that:

If the length of the word is

1

or

2

letters, change all letters to lowercase.

Otherwise, change the first letter to uppercase and the remaining letters to lowercase.

Return

the

capitalized

title

.

Example 1:

Input:

title = "capiTalIze tHe titLe"

Output:

"Capitalize The Title"

Explanation:

Since all the words have a length of at least 3, the first letter of each word is uppercase, and the remaining letters are lowercase.

Example 2:

Input:

title = "First leTTeR of EACH Word"

Output:

"First Letter of Each Word"

Explanation:

The word "of" has length 2, so it is all lowercase. The remaining words have a length of at least 3, so the first letter of each remaining word is uppercase, and the remaining letters are lowercase.

Example 3:

Input:

title = "i lOve leetcode"

Output:

"i Love Leetcode"

Explanation:

The word "i" has length 1, so it is lowercase. The remaining words have a length of at least 3, so the first letter of each remaining word is uppercase, and the remaining letters are lowercase.

Constraints:

1 <= title.length <= 100

title

consists of words separated by a single space without any leading or trailing spaces.

Each word consists of uppercase and lowercase English letters and is

non-empty

.

Code Snippets

C++:

```
class Solution {
public:
string capitalizeTitle(string title) {

}
};
```

Java:

```java
class Solution {
public String capitalizeTitle(String title) {

}
}
```

Python3:

```python
class Solution:
def capitalizeTitle(self, title: str) -> str:
```

Python:

```python
class Solution(object):
def capitalizeTitle(self, title):
"""
:type title: str
:rtype: str
"""
```

JavaScript:

```javascript
/**
* @param {string} title
* @return {string}
*/
var capitalizeTitle = function(title) {

};
```

TypeScript:

```typescript
function capitalizeTitle(title: string): string {

};
```

C#:

```csharp
public class Solution {
public string CapitalizeTitle(string title) {
}
}
```

C:

```
char* capitalizeTitle(char* title) {

}
```

Go:

```
func capitalizeTitle(title string) string {

}
```

Kotlin:

```
class Solution {
fun capitalizeTitle(title: String): String {

}
}
```

Swift:

```
class Solution {
func capitalizeTitle(_ title: String) -> String {

}
}
```

Rust:

```
impl Solution {
pub fn capitalize_title(title: String) -> String {

}
}
```

Ruby:

```
# @param {String} title
# @return {String}
def capitalize_title(title)

end
```

PHP:

```
class Solution {

/**
* @param String $title
* @return String
*/
function capitalizeTitle($title) {

}
}
```

Dart:

```
class Solution {
String capitalizeTitle(String title) {

}
}
```

Scala:

```
object Solution {
def capitalizeTitle(title: String): String = {

}
}
```

Elixir:

```
defmodule Solution do
@spec capitalize_title(title :: String.t) :: String.t
def capitalize_title(title) do

end
end
```

Erlang:

```
-spec capitalize_title(Title :: unicode:unicode_binary()) ->
unicode:unicode_binary().
capitalize_title(Title) ->
.
```

Racket:

```
(define/contract (capitalize-title title)
(-> string? string?)
)
```

Solutions

C++ Solution:

```
/*
* Problem: Capitalize the Title
* Difficulty: Easy
* Tags: string
*
* Approach: String manipulation with hash map or two pointers
* Time Complexity: O(n) or O(n log n)
* Space Complexity: O(1) to O(n) depending on approach
*/

class Solution {
public:
string capitalizeTitle(string title) {

}
};
```

Java Solution:

```
/**
* Problem: Capitalize the Title
* Difficulty: Easy
* Tags: string
*
```

```
* Approach: String manipulation with hash map or two pointers
* Time Complexity: O(n) or O(n log n)
* Space Complexity: O(1) to O(n) depending on approach
*/

class Solution {
public String capitalizeTitle(String title) {

}
}
```

Python3 Solution:

```
"""
Problem: Capitalize the Title
Difficulty: Easy
Tags: string

Approach: String manipulation with hash map or two pointers
Time Complexity: O(n) or O(n log n)
Space Complexity: O(1) to O(n) depending on approach
"""

class Solution:
def capitalizeTitle(self, title: str) -> str:
# TODO: Implement optimized solution
pass
```

Python Solution:

```
class Solution(object):
def capitalizeTitle(self, title):
"""
:type title: str
:rtype: str
"""
```

JavaScript Solution:

```
/**
* Problem: Capitalize the Title
* Difficulty: Easy
* Tags: string
*
* Approach: String manipulation with hash map or two pointers
* Time Complexity: O(n) or O(n log n)
* Space Complexity: O(1) to O(n) depending on approach
*/

/**
* @param {string} title
* @return {string}
*/
var capitalizeTitle = function(title) {

};
```

TypeScript Solution:

```
/**
* Problem: Capitalize the Title
* Difficulty: Easy
* Tags: string
*
* Approach: String manipulation with hash map or two pointers
* Time Complexity: O(n) or O(n log n)
* Space Complexity: O(1) to O(n) depending on approach
*/

function capitalizeTitle(title: string): string {

};
```

C# Solution:

```
/*
* Problem: Capitalize the Title
* Difficulty: Easy
* Tags: string
*
* Approach: String manipulation with hash map or two pointers
* Time Complexity: O(n) or O(n log n)
* Space Complexity: O(1) to O(n) depending on approach
*/

public class Solution {
public string CapitalizeTitle(string title) {

}
}
```

C Solution:

```c
/*
* Problem: Capitalize the Title
* Difficulty: Easy
* Tags: string
*
* Approach: String manipulation with hash map or two pointers
* Time Complexity: O(n) or O(n log n)
* Space Complexity: O(1) to O(n) depending on approach
*/

char* capitalizeTitle(char* title) {

}
```

Go Solution:

```go
// Problem: Capitalize the Title
// Difficulty: Easy
// Tags: string
//
// Approach: String manipulation with hash map or two pointers
// Time Complexity: O(n) or O(n log n)
// Space Complexity: O(1) to O(n) depending on approach

func capitalizeTitle(title string) string {

}
```

Kotlin Solution:

```
class Solution {
fun capitalizeTitle(title: String): String {

}
}
```

Swift Solution:

```
class Solution {
func capitalizeTitle(_ title: String) -> String {

}
}
```

Rust Solution:

```
// Problem: Capitalize the Title
// Difficulty: Easy
// Tags: string
//
// Approach: String manipulation with hash map or two pointers
// Time Complexity: O(n) or O(n log n)
// Space Complexity: O(1) to O(n) depending on approach

impl Solution {
pub fn capitalize_title(title: String) -> String {

}
}
```

Ruby Solution:

```
# @param {String} title
# @return {String}
def capitalize_title(title)

end
```

PHP Solution:

```
class Solution {
```

```
/**
* @param String $title
* @return String
*/
function capitalizeTitle($title) {

}
}
```

Dart Solution:

```
class Solution {
String capitalizeTitle(String title) {

}
}
```

Scala Solution:

```
object Solution {
def capitalizeTitle(title: String): String = {

}
}
```

Elixir Solution:

```
defmodule Solution do
@spec capitalize_title(title :: String.t) :: String.t
def capitalize_title(title) do

end
end
```

Erlang Solution:

```
-spec capitalize_title(Title :: unicode:unicode_binary()) ->
unicode:unicode_binary().
capitalize_title(Title) ->
.
```

Racket Solution:

```
(define/contract (capitalize-title title)
(-> string? string?)
)
```