**Problem #151. Reverse Words in a String (Medium)**

Given an input string s, reverse the order of the **words**.

A **word** is defined as a sequence of non-space characters. The **words** in s will be separated by at least one space.

Return a string of the words in reverse order concatenated by a single space.

**Note** that s may contain leading or trailing spaces or multiple spaces between two words. The returned string should only have a single space separating the words. Do not include any extra spaces.

**Example 1:**

**Input:** s = "the sky is blue"

**Output:** "blue is sky the"

**Example 2:**

**Input:** s = " hello world "

**Output:** "world hello"

**Explanation:** Your reversed string should not contain leading or trailing spaces.

**Example 3:**

**Input:** s = "a good example"

**Output:** "example good a"

**Explanation:** You need to reduce multiple spaces between two words to a single space in the reversed string.

**Constraints:**

* 1 <= s.length <= 104
* s contains English letters (upper-case and lower-case), digits, and spaces ' '.
* There is **at least one** word in s.

**Follow-up:**If the string data type is mutable in your language, can you solve it **in-place** with O(1) extra space?

**My Solution:**

1. Split the string into words based on white space separator and save the words in a list.
2. Join the words with whitespaces after reversing the words in the list.
3. Return the string obtained in the previous step.

class Solution:

def reverseWords(self, s: str) -> str:

words = s.split()

return ' '.join(words[::-1])