Find the minimum length word from a given dictionary words, which has all the letters from the string licensePlate. Such a word is said to *complete* the given string licensePlate

Here, for letters we ignore case. For example, "P" on the licensePlate still matches "p" on the word.

It is guaranteed an answer exists. If there are multiple answers, return the one that occurs first in the array.

The license plate might have the same letter occurring multiple times. For example, given a licensePlate of "PP", the word "pair" does not complete the licensePlate, but the word "supper" does.

**Example 1:**

**Input:** licensePlate = "1s3 PSt", words = ["step", "steps", "stripe", "stepple"]

**Output:** "steps"

**Explanation:** The smallest length word that contains the letters "S", "P", "S", and "T".

Note that the answer is not "step", because the letter "s" must occur in the word twice.

Also note that we ignored case for the purposes of comparing whether a letter exists in the word.

**Example 2:**

**Input:** licensePlate = "1s3 456", words = ["looks", "pest", "stew", "show"]

**Output:** "pest"

**Explanation:** There are 3 smallest length words that contains the letters "s".

We return the one that occurred first.

**Note:**

1. licensePlate will be a string with length in range [1, 7].
2. licensePlate will contain digits, spaces, or letters (uppercase or lowercase).
3. words will have a length in the range [10, 1000].
4. Every words[i] will consist of lowercase letters, and have length in range [1, 15].