Search Suggestions System

Try to solve the Search Suggestions System problem.

> **We'll cover the following** ∧
>
> - Statement
> - Examples
> - Understand the problem
> - Figure it out!
> - Try it yourself

Statement

Given an array of strings called `products` and a word to search, design a system that, when each character of the searched word is typed, suggests at most three product names from `products`. Suggested products should share a common prefix with the searched word. If more than three products exist with a common prefix, return the three product names that appear first in lexicographical order.

Return the suggested products, which will be a list of lists after each character of searched word is typed.

Constraints:

- $1 \leq$ `products.length` ≤ 1000
- $1 \leq$ `products[i].length` ≤ 3000
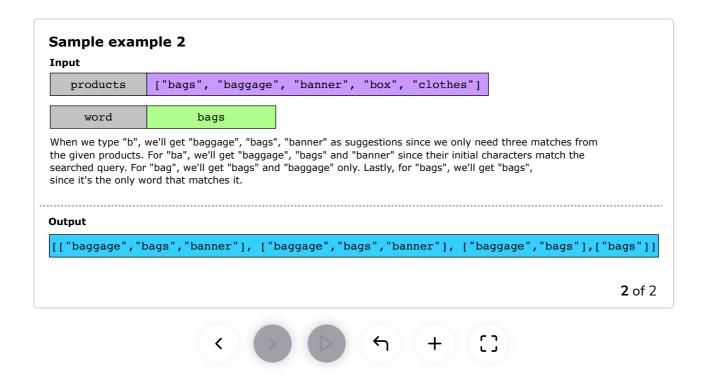- $1 \leq$ `sum(products[i].length)` $\leq 2 \times 10^3$
- All the strings of products are unique.
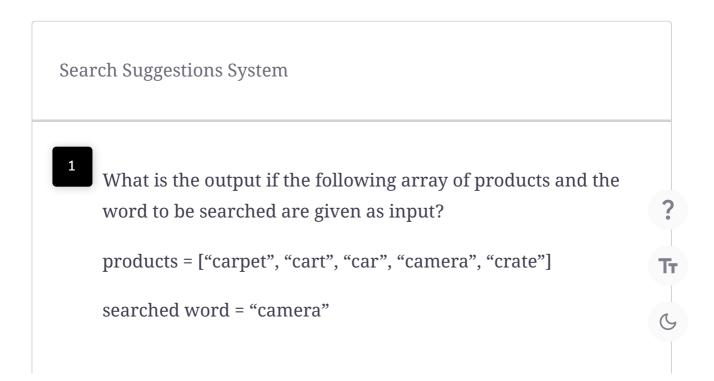- `products[i]` consists of lowercase English letters.

- $1 \leq$ `searched word.length` ≤ 1000
- The searched word consists of lowercase English letters.

Examples

Sample example 2

Input

products	["bags", "baggage", "banner", "box", "clothes"]
word	bags

When we type "b", we'll get "baggage", "bags", "banner" as suggestions since we only need three matches from the given products. For "ba", we'll get "baggage", "bags" and "banner" since their initial characters match the searched query. For "bag", we'll get "bags" and "baggage" only. Lastly, for "bags", we'll get "bags", since it's the only word that matches it.

Output

```
[["baggage","bags","banner"], ["baggage","bags","banner"], ["baggage","bags"],["bags"]]
```
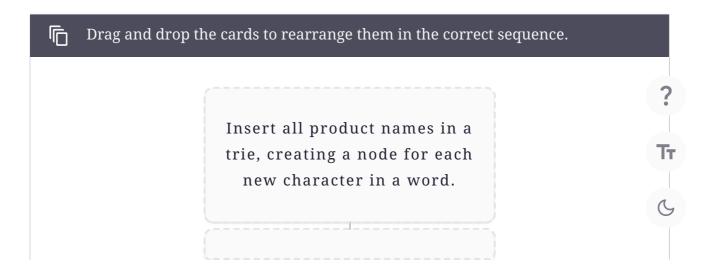
Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:
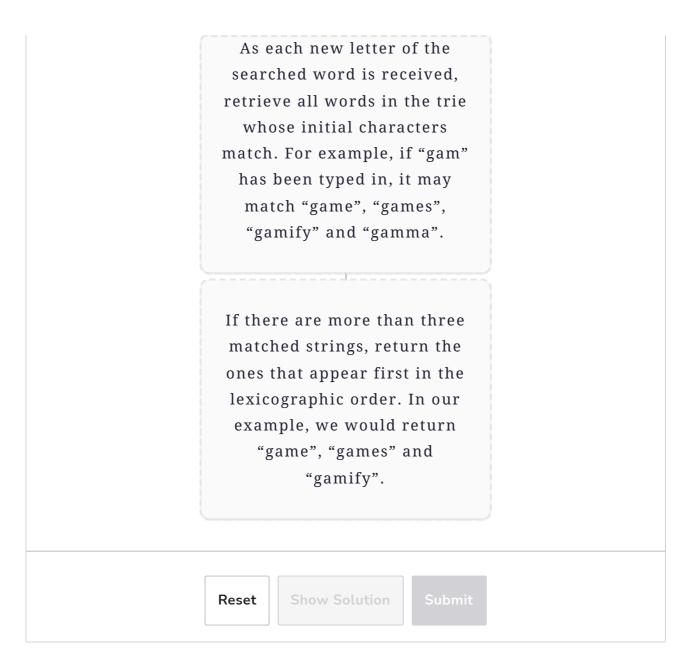
Search Suggestions System

1 What is the output if the following array of products and the word to be searched are given as input?

products = ["carpet", "cart", "car", "camera", "crate"]

searched word = "camera"

A) [["camera", "car", "carpet"], ["camera"], ["camera"], ["camera"], ["camera"]]

B) [["camera", "car", "carpet"], ["camera"], ["camera"], ["camera"], ["camera"]]

C) [["camera", "car", "carpet"], ["camera", "car", "carpet"], ["camera"], ["camera"], ["camera"], ["camera"]]

D) [["camera", "car", "carpet"], ["camera", "car", "carpet"], ["camera"]]

Submit Answer

<

Question 1 of 2
0 attempted

>

Reset Quiz ⟳

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.
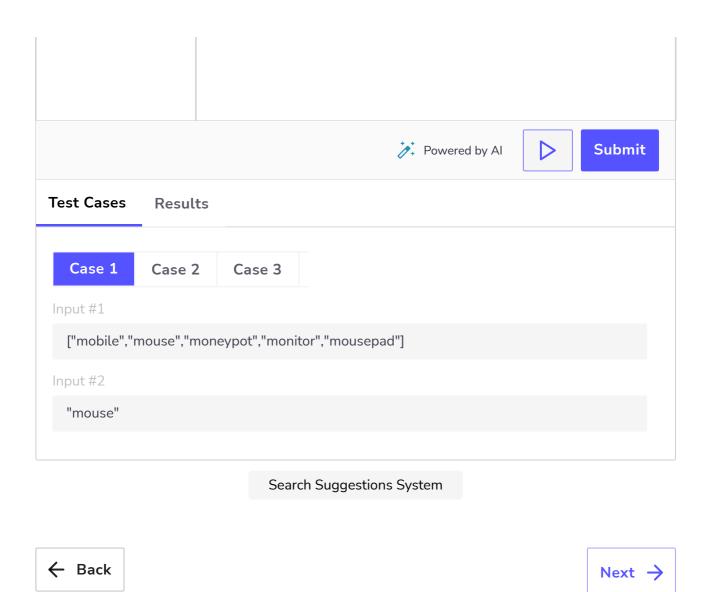
Drag and drop the cards to rearrange them in the correct sequence.

?

Tт

☾

Insert all product names in a trie, creating a node for each new character in a word.

As each new letter of the searched word is received, retrieve all words in the trie whose initial characters match. For example, if "gam" has been typed in, it may match "game", "games", "gamify" and "gamma".

If there are more than three matched strings, return the ones that appear first in the lexicographic order. In our example, we would return "game", "games" and "gamify".

Reset   Show Solution   Submit

Try it yourself

coding playground. You'll need the provided supporting code to implement your solution.

Java

```java
1  import java.util.*;
2  class SearchSuggestion {
3      public List<List<String>> suggestedProducts(String[
4          // Write your code here
5
6          return null;
7      }
8  }
```

SearchSuggestio...

Node.java

Powered by AI

▶ **Submit**

Test Cases   Results

| Case 1 | Case 2 | Case 3 |

Input #1

["mobile","mouse","moneypot","monitor","mousepad"]

Input #2

"mouse"

Search Suggestions System

← **Back**

Solution: Implement T...

Next →

Solution: Search Sugg...

☑ Mark as Completed

?

T_T

☾