The assignment is to build an API which identifies **SINGLE** word anagrams. There are 5 requirements. Even if you can not complete all of them you are still encouraged to submit the assignment.

1. A working **Javascript** app which can run locally.
2. A Github repository which must include:
    a. The code - multiple commits and comments are encouraged.
    b. A **detailed** readme describing how the project can be run locally and how the endpoints can be tested with Postman.
3. A POST endpoint (endpoint A) which must:
    a. Receive two strings in the body of the request (keys not important). Each string will be a **single** word.
    b. Return a JSON with the key **"outcome"** whose value is a boolean.
    c. **Outcome** must tell us whether or not the two words sent in the body are anagrams.
    d. Example:
        i. Input:
            1. String 1: "Elvis"
            2. String 2: "Lives"
        ii. Output: { outcome: true }
4. A POST endpoint (endpoint B) which must:
    a. Receive two strings in the body of the request (keys not important). The two strings will be a **single** word (string1) and a **sentence** (string2).
    b. Return a JSON with the key **"outcome"** whose value is an array of strings.
    c. **Outcome** must identify all **unique single** word anagrams of string1 in string2.
    d. Example:
        i. Input:
            1. String 1: "Elvis"
            2. String 2: "Elvis lives in a house"
        ii. Output: { outcome: ["elvis", "lives"] }
5. A POST endpoint (endpoint C) which must:
    a. Receive one string in the body of the request (keys not important). The string will be a **sentence**.
    b. Return a JSON with the key **"outcome"** whose value is an array of arrays of strings.
    c. **Outcome** must identify all anagram **groups** present in the string. Every anagram must be a **single unique** word.
    d. Example:
        i. Input:
            1. String 1: "Elvis lives in a house. His cat can act. He has a study that's dusty."
        ii. Output: { outcome: [["elvis", "lives"], ["cat", "act"], ["study", "dusty"]] }

General constraints:

1. Ignore all punctuation when parsing strings.
2. A word is defined as all characters delimited by a single space.

Bonus points

1. The functionality of endpoint C must be of O(n) time and space complexity.
2. Deploy the app on heroku so when we want to test it we don't have to run it locally.