290. Word Pattern

Given a `pattern` and a string `s`, find if `s` follows the same pattern.

Here **follow** means a full match, such that there is a bijection between a letter in `pattern` and a **non-empty** word in `s`.

Example 1:

```
Input: pattern = "abba", s = "dog cat cat dog"
Output: true
```

Example 2:

```
Input: pattern = "abba", s = "dog cat cat fish"
Output: false
```

Example 3:

```
Input: pattern = "aaaa", s = "dog cat cat dog"
Output: false
```

Example 4:

```
Input: pattern = "abba", s = "dog dog dog dog"
Output: false
```

Constraints:

- `1 <= pattern.length <= 300`
- `pattern` contains only lower-case English letters.
- `1 <= s.length <= 3000`
- `s` contains only lower-case English letters and spaces `' '`.
- `s` **does not contain** any leading or trailing spaces.
- All the words in `s` are separated by a **single space**.

```python
class Solution:
    def wordPattern(self, pattern: str, s: str) -> bool:
        pattern = list(pattern)
```

```
        s = s.split(' ')
        return self.patternGeneraation(pattern) ==
self.patternGeneraation(s)




    def patternGeneraation(self,arr):
        ans = ''
        freq = {}
        for i,ele in enumerate(arr):
            if ele not in freq:
                freq[ele] = i
        return ''.join(str(freq[x]) for x in arr)
```