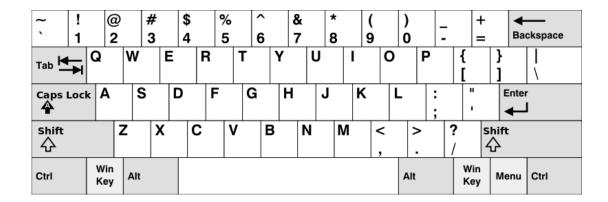
给定一个单词列表,只返回可以使用在键盘同一行的字母打印出来的单词。键盘如下图所示。



示例1:

```
输入: ["Hello", "Alaska", "Dad", "Peace"]
输出: ["Alaska", "Dad"]
分析,用3个set来记录没行的大小写字母,然后用3个状态标记字符串在那行出现过。状态为1则放入,不为1
```

分析,用3个set来记录没行的大小写字母,然后用3个状态标记字符串在那行出现过。状态为1则放入,不为1 则放弃。

```
class Solution {
 public:
     vector<string> findWords(vector<string>& words) {
          vector<string> res;
          unordered set<char> row1
 \{ \ 'q', \ 'w', \ 'e', \ 'r', \ 't', \ 'y', \ 'u', \ 'i', \ 'o', \ 'p', \ 'Q', \ 'W', \ 'E', \ 'R', \ 'T', \ 'Y', \ 'U', \ 'I', \ 'O', \ 'P' \}; 
          unordered set<char>
row2{'a','s','d','f','g','h','j','k','l','A','S','D','F','G','H','J','K','L'};
        unordered set<char>
row3{'z','x','c','v','b','n','m','Z','X','C','V','B','N','M'};
          for(string word : words) {
              int a = 0;
              int b = 0;
              int c = 0;
              for(char ch : word) {
                   if (row1.count(ch)) a = 1;
                   else if (row2.count(ch)) b = 1;
                   else if (row3.count(ch)) c = 1;
                   if (a + b + c > 1) break;
              if (a + b + c == 1) res.push back(word);
         return res;
     }
};
```