package com.iimtiaz.day\_09;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class PalindromePartitioning {  
 public static void main(String[] args) {  
 String s = "aab";  
 System.*out*.println(new Solution\_1().partition(s));  
  
 }  
}  
  
class Solution\_1 {  
 // Checking the given substring is palindrome  
 private boolean isPalindrome(String s, int start, int end) {  
 while (start <= end) {  
 if (s.charAt(start++) != s.charAt(end--)) {  
 return false;  
 }  
 }  
 return true;  
 }  
 private void solveRecBacktracking(String s, int idx, List<List<String>> res, List<String> l) {  
 //Edge or Base Case  
 if (s.length() == idx) {  
 res.add(new ArrayList<>(l));  
 return;  
 }  
  
 // backtracking  
 for (int i = idx; i < s.length(); i++) {  
 if (isPalindrome(s, idx, i)) {  
 l.add(s.substring(idx, i + 1));  
 solveRecBacktracking(s, i + 1, res, l);  
 l.remove(l.size() - 1);  
 }  
 }  
 }  
 public List<List<String>> partition(String s) {  
 List<List<String>> res = new ArrayList<>();  
 solveRecBacktracking(s, 0, res, new ArrayList<>());  
 return res;  
 }  
}