题目：

Given a string **s**, find the longest palindromic substring in **s**. You may assume that the maximum length of **s** is 1000.

**Example:**

**Input:** "babad"

**Output:** "bab"

**Note:** "aba" is also a valid answer.

**Example:**

**Input:** "cbbd"

**Output:** "bb"

题目的意思时找到字符串中最长的回文。什么是回文呢？abcdedcba是回文，以e为中间轴。abcddcba是回文，没有中间轴。

首先我来聊聊我的解题过程把。哈哈哈哈哈哈

审题错了，审题成找字符串中头尾重复的最长的字符串了，abcade—>adca,并不是找的回文。

我怎么做的呢？利用上次找最长的无重复的子字符串的方法，到判断为重复的时候，记录从第一个重复字符到第二个重复字符的长短，当大于所记录最长的时候就保存在一个字符串中。用set判断的是否重复，但是用最长用的hashset来存储的话输出的数据是随机的，treeset输出是做了排序的，linkedhashset是按照输入顺序保存的set，所以我把之前的hashset改成了linkedhashset这样能保证输入的输出顺序。

此次算法我学到的是如何遍历set,用迭代器遍历

Set<Character> set = new linkedhashset<>();

Iterator<Character> it = set.iterator();

While(it.hasNext){

String str = it.next();

System.out.println(str);

}

如何将字符串制空，string res = “”；

聊了这么多审题错的思路，下面回归正题，如何找出字符串中的回文呢？

找出字符串中回文函数

private int expandAroundCenter(String s, int left, int right) {

int L = left, R = right;

while (L >= 0 && R < s.length() && s.charAt(L) == s.charAt(R)) {

L--;

R++;

}

return R - L - 1;

}

回文分为两种情况：有中轴和无中轴。如何判断呢，有中轴的话从中轴开始，左=右

中轴为i的话，left=right=I,string.charAt（i）=string.charAt（i）开始。无中轴的话，找到开始折半的点Abcddcba，找到d为I ，string.charAt（i）=string.charAt（i+1）。

从中间开始，逐个判断左是否等于右，若不等于跳出循环，此时的位置R-L-1，因为需要判断到最长回文的两端，每次都是多数了一个字符，位置长度在R-L上需要减去1

例如：abad，从b开始，（1）L=R=1，： b=b🡪符合条件🡪L=L-1=0，R=R+1=2（2）L=0,R=2, s.charAt(L) = s.charAt(R)：a=a🡪符合条件🡪 L=L-1=-1，R=R+1=3. L<0,不符合条件推出。最终得到的R=3,L=-1。回文长度为R-L-1=4-1=3.

返回了回文长度，因为是从I点为中轴的长度，所以回文开始start = i - (len - 1) / 2，结束end = i + len / 2。

例如：abad，i=1，len=3，start=0=1-（3-1）/2，end=1+3/2（取整）=2

public String longestPalindrome(String s) {

int start = 0, end = 0;

for (int i = 0; i < s.length(); i++) {

int len1 = expandAroundCenter(s, i, i); //有中心轴

int len2 = expandAroundCenter(s, i, i + 1); //无中心轴

int len = Math.max(len1, len2);

if (len > end - start) {

start = i - (len - 1) / 2;

end = i + len / 2;

}

}

return s.substring(start, end + 1);

}

当有中心轴的回文字符串用了无中心轴的len返回0，相反同返回。

最后调用s.substring（）方法返回字符串的一部分。