187. Repeated DNA Sequences

link

最开始, 用Hashtable必须import, 不确定substring的复杂度.

```
import java.util.Hashtable;
class Solution {
    public List<String> findRepeatedDnaSequences(String s) {
        List<String> res = new ArrayList();
        int len = s.length();
        if(len < 11) return res;
        StringBuilder sb = new StringBuilder();
        Hashtable<String, Integer> ht = new Hashtable<String, Integer>();
        for(int i = 0; i < 10; ++i){
            sb.append(s.charAt(i));
        }
        ht.put(sb.toString(), 1);
        for(int i = 10; i < len; ++i){
//这里注意每次删第一个就好了.
            sb.deleteCharAt(0);
            sb.append(s.charAt(i));
            String current = sb.toString();
            if(ht.containsKey(current)){
                int times = ht.get(current);
                if(times == 1)
                    res.add(current);
                ht.put(current, times + 1);
            }else
                ht.put(current, 1);
        }
        return res;
    }
}
```

可以用两个HashSet来判断.

```
   public List<String> findRepeatedDnaSequences(String s) {
        Set seen = new HashSet(), repeated = new HashSet();
        for (int i = 0; i + 9 < s.length(); i++) {
            String ten = s.substring(i, i + 10);
            if (!seen.add(ten))
                repeated.add(ten);
        }
        return new ArrayList(repeated);
    }
```