

day14_String_王世杰

1、

String的API, 自己练习一下,敲一敲.

判断功能

用来比较字符串的内容, 注意区分大小写

`boolean equals(Object obj)`

忽略字符串大小写比较字符串内容, 常见用于比较网址URL

`boolean equalsIgnoreCase(String str)`

判断当前字符串对象是否包含, 目标字符串的字符序列

`boolean contains(String str)`

判断当前字符串对象, 是否以目标字符串的字符序列开头

`boolean startswith(String str)`

判断当前字符串, 是否以目标字符串对象的字符序列结尾, 常用于确定文件后缀名格式

`boolean endswith(String str)`

判断一个字符串, 是不是空字符串 "" null

`boolean isEmpty()`

获取功能

获取当前字符串对象中, 包含的字符个数

`int length()`

获取字符串对象代表字符序列中, 指定位置的字符

`char charAt(int index)`

在当前字符串对象中查找指定的字符, 如果找到就返回字符, 首次出现的位置, 如果没找到返回-1

也可以填字符

`int indexOf(int ch)`

指定从当前字符串对象的指定位置开始, 查找首次出现的指定字符的位置, (如果没找到返回-1)

可以填入字符

`int indexOf(int ch, int fromIndex)`

查找当前字符串中, 目标字符串首次出现的位置(如果包含), 找不到, 返回-1

这里的位置是指目标字符串的第一个字符, 在当前字符串对象中的位置

`int indexOf(String str)`

指定, 从当前字符串对象的指定位置开始, 查找首次出现的指定字符串的位置(如果没找到返回-1)

这里的位置是指目标字符串的第一个字符, 在当前字符串对象中的位置

`int indexOf(String str, int fromIndex)` ,

返回字符串, 该字符串只包含当前字符串中, 从指定位置开始(包含指定位置字符)到结束的那部分字符串

`String substring(int start)`

返回字符串, 只包含当前字符串中, 从start位置开始(包含), 到end(不包含)指定的位置的字符串

`String substring(int start, int end)`

转换功能

获取一个用来表示字符串对象字符序列的，字节数组

```
byte[] getBytes()
```

获取的是用来表示字符串对象字符序列的，字符数组

```
char[] toCharArray()
```

把字符数组转换成字符串

```
static String valueOf(char[] chs)
```

把各种基本数据类型和对象转换成字符串 1 → "1"

```
static String valueOf(int i/double...)
```

把字符串全部转化为小写

```
String toLowerCase()
```

把字符串全部转换为大写

```
String toUpperCase()
```

字符串拼接，作用等价于 + 实现的字符串拼接

```
String concat(String str)
```

String类的替换功能

在新的字符串中，用新(new)字符，替换旧(old)字符

```
String replace(char old,char new)
```

在新的字符串中，用新的字符串(new)，替换旧(old)字符串

```
String replace(String old, String new)
```

截取功能

返回一个新的String，开头到结束

```
public String substring(int start)
```

返回一个新的String，指定区间

```
public String substring(int start,int end)
```

在新的字符串中，去掉开头和结尾的空格字符 " abc "

```
String trim()
```

分隔功能

将字符串按照符号分隔成字符串数组 "a,b,c,d"

```
String[] split(String re)
```

String类的比较功能

```
int compareTo(String str)
```

```
int compareToIgnoreCase(String str)
```

```
package com.cskaoyan.homework.day14;
```

```
import java.util.Arrays;
```

```
public class StringApiTest {
```

```

public static void main(String[] args) {
    String str1 = "王世杰666";
    String str2 = "卫屹柔666";
    String str4 = "wangshijie666";
    String str5 = "WANGSHIJIE666";
    String str6 = "666";
    String str7 = "aaa";
    String str8 = "王世杰";

    System.out.println(str1.equals(str2)); //false
    System.out.println(str4.equalsIgnoreCase(str5)); //true
    System.out.println(str1.contains(str6)); //true
    System.out.println(str4.startsWith("w")); //true
    System.out.println(str1.endsWith(str6)); //true
    System.out.println(str6.isEmpty()); //false
    System.out.println(str5.length()); //13
    System.out.println(str1.charAt(0)); //'王'
    System.out.println(str4.indexOf('i')); //6
    System.out.println(str4.indexOf('i', 7)); //8
    System.out.println(str4.indexOf(str6)); //10
    System.out.println(str5.indexOf(str6, 11)); //-1
    System.out.println(str1.substring(3)); //666
    System.out.println(str1.substring(3,5)); // 66
    System.out.println(Arrays.toString(str1.toCharArray())); //
    '王','世','杰','6','6','6'
    System.out.println(Arrays.toString(str7.getBytes())); // '97' '97' '97'

    char[] chars = {'王','世','杰','6','6','6'};
    System.out.println(String.valueOf(chars)); //"王世杰666"
    System.out.println(String.valueOf(96)); //96

    System.out.println(str5.toLowerCase().equals(str4)); //true
    System.out.println(str4.toUpperCase().equals(str5)); //true
    System.out.println(str8.concat(str6)); //"王世杰666"
    System.out.println(str1.replace('6', '7')); //"王世杰777"
    System.out.println(str1.replace("666", "777")); //"王世杰777"

    System.out.println(" abc ".trim()); //"abc"
    System.out.println(Arrays.toString("i am very good".split(" ")));
    //["i","am","very","good"]
    System.out.println(str1.compareTo(str2)); //3
    System.out.println(str5.compareToIgnoreCase(str4)); //13

}

}

```

2、服务器PPT那个文件夹中有个Data日期与格式化输出的资料,大家下载下来,练习一下,敲一敲

```

package com.cskaoyan.homework.day14;

import java.util.Date;
import java.util.Random;

```

```

public class DateTest {

    public static void main(String[] args) {
        Date date = new Date(); //Thu Apr 07 16:39:55 CST 2022
        System.out.println(date);

        Date date1 = new Date(0); //Thu Jan 01 08:00:00 CST 1970
        System.out.println(date1);

        long t1 = date.getTime();
        System.out.println(t1);
        //      int num = febo(45);
        //      System.out.println(num);
        long t2 = date.getTime();
        System.out.println(t2);
        System.out.println(t2 - t1);

        date1.setTime(1);
        System.out.println(date1);
        int a = 100;
        String name = "黄鹤";
        String s = "好消息,好消息,江南皮革厂倒闭了,老板%s欠了%d万,带着小姨子跑了%n";
        System.out.format(s,name,a);
        String format = String.format(s,name, a);
        System.out.println(format);
        int year = 2022;
        double num = 3.1415;
        // 直接输数字
        System.out.format("%d%n",year);
        // 总长度是8,默认右对齐
        System.out.format("%8d%n",year);
        // 左对齐
        System.out.format("%-8d%n",year);
        // 总长度为8,不够补0
        System.out.format("%08d%n",year);
        System.out.println(num);
        //小数点后几位
        System.out.format("%.2f%n",num);

    }

    static int febo(int n) {
        if (n == 0) {
            return 1;
        } else if (n == 1) {
            return 1;
        } else {
            return febo(n - 1) + febo(n - 2);
        }
    }

}

```

```
Run: DateTest x
D:\eve\windows\jdk-8u311\bin\java.exe ...
Thu Apr 07 16:59:13 CST 2022
Thu Jan 01 08:00:00 CST 1970
1649321953045
1649321953045
0
Thu Jan 01 08:00:00 CST 1970
好消息,好消息,江南皮革厂倒闭了,老板黄鹤欠了100万,带着小姨子跑了
好消息,好消息,江南皮革厂倒闭了,老板黄鹤欠了100万,带着小姨子跑了

2022
    2022
2022
00002022
3.1415
3.14

Process finished with exit code 0
```

3、

- 1: 遍历获取字符串中的每一个字符 "abc001DEF"
- 2: 统计一个字符串中大写字母字符, 小写字母字符, 数字字符出现的次数。(不考虑其他字符)

```
package com.cskaoyan.homework.day14;
/*
    1: 遍历获取字符串中的每一个字符 "abc001DEF"
    2: 统计一个字符串中大写字母字符, 小写字母字符, 数字字符出现的次数。(不考虑其他字符)
*/

public class CharInString {
    public static void main(String[] args) {
        String str = "abc001DEF";
        System.out.println(str);
        for (int i = 0; i < str.length(); i++) {
            System.out.print(str.charAt(i)+", ");
        }
        System.out.println("\n"+"\"abc001DEF\"中小写字母
有: "+statisticalUpperCharacter(str)+"个");
        System.out.println("\"abc001DEF\"中大写字母
有: "+statisticalLowerCharacter(str)+"个");
        System.out.println("\"abc001DEF\"中数字字符
有: "+statisticalNumberCharacter(str)+"个");
    }

    //    2: 统计一个字符串中大写字母字符, 小写字母字符, 数字字符出现的次数。(不考虑其他字符)
    static int statisticalUpperCharacter(String str){
        int res = 0;
        for (int i = 0; i < str.length(); i++) {
            if (str.charAt(i) >= 97 && str.charAt(i) <= 112){
                res++;
            }
        }
    }
}
```

```

    }
    return res;
}

static int statisticalLowerCharacter(String str){
    int res = 0;
    for (int i = 0; i < str.length(); i++) {
        if (str.charAt(i) >= 65 && str.charAt(i) <= 90){
            res++;
        }
    }
    return res;
}

static int statisticalNumberCharacter(String str){
    int res = 0;
    for (int i = 0; i < str.length(); i++) {
        if (str.charAt(i) >= 48 && str.charAt(i) <= 57){
            res++;
        }
    }
    return res;
}
}

```

Run: CharInString x

```

D:\eve\windows\jdk-8u311\bin\java.exe ...
abc001DEF
a, b, c, 0, 0, 1, D, E, F,
"abc001DEF"中小写字符有: 3个
"abc001DEF"中大写字符有: 3个
"abc001DEF"中数字字符有: 3个

Process finished with exit code 0

```

4、穷举匹配密码

1. 生成一个长度是3的随机数字字符串，把这个字符串作为当做密码
2. 使用穷举法生成长度是3的字符串(000,001,002...999)，匹配上述生成的密码,将列举到的可能的密码打印,直到匹配成功结束.

```

package com.cskaoyan.homework.day14;

import java.util.Random;

public class ExhaustiveMatchPassword {

    public static void main(String[] args) {
        Random random = new Random();
        int randomNum1 = random.nextInt(10);
        int randomNum2 = random.nextInt(10);
        int randomNum3 = random.nextInt(10);
    }
}

```








```

        String randomPassword = String.valueOf(randomNum1) +
String.valueOf(randomNum2) + String.valueOf(randomNum3);
        System.out.println("随机密码为: "+randomPassword);
        exhaustiveMatch(randomPassword);
    }

    public static void exhaustiveMatch(String password){
        int count = 1;
        for (int i = 0; i < 10; i++) {
            for (int j = 0; j < 10; j++) {
                for (int k = 0; k < 10; k++) {
                    String guess = String.valueOf(i) + String.valueOf(j) +
String.valueOf(k);
                    System.out.print(guess+" ");
                    if((count++) % 10 == 0 ) {
                        System.out.println();
                    }
                    if (guess.equals(password)){
                        return;
                    }
                }
            }
        }
    }
}

```

Run: ExhaustiveMatchPassword x

D:\eve\windows\jdk-8u311\bin\java.exe ...

随机密码为: 149
 000, 001, 002, 003, 004, 005, 006, 007, 008, 009,
 010, 011, 012, 013, 014, 015, 016, 017, 018, 019,
 020, 021, 022, 023, 024, 025, 026, 027, 028, 029,
 030, 031, 032, 033, 034, 035, 036, 037, 038, 039,
 040, 041, 042, 043, 044, 045, 046, 047, 048, 049,
 050, 051, 052, 053, 054, 055, 056, 057, 058, 059,
 060, 061, 062, 063, 064, 065, 066, 067, 068, 069,
 070, 071, 072, 073, 074, 075, 076, 077, 078, 079,
 080, 081, 082, 083, 084, 085, 086, 087, 088, 089,
 090, 091, 092, 093, 094, 095, 096, 097, 098, 099,
 100, 101, 102, 103, 104, 105, 106, 107, 108, 109,
 110, 111, 112, 113, 114, 115, 116, 117, 118, 119,
 120, 121, 122, 123, 124, 125, 126, 127, 128, 129,
 130, 131, 132, 133, 134, 135, 136, 137, 138, 139,
 140, 141, 142, 143, 144, 145, 146, 147, 148, 149,

Process finished with exit code 0

5、

(选做)创建一个长度是8的的字符串数组

使用8个长度是5的随机字符串初始化这个数组

对这个数组进行排序，按照每个字符串的首字母排序(无视大小写)

注1: 不能使用Arrays.sort() 要自己写 可以使用冒泡排序

注2: 无视大小写, 即 Axxxx 和 axxxxx 没有先后顺序

举例: 排序前 [UxoEl, If3js, 9SSwC, czHuA, lZLBF, lhynX, VgjrY, dJAIW]

排序后 [9SSwC, czHuA, dJAIW, If3js, lhynX, lZLBF, UxoEl, VgjrY]

```
package com.cskaoyan.homework.day14;

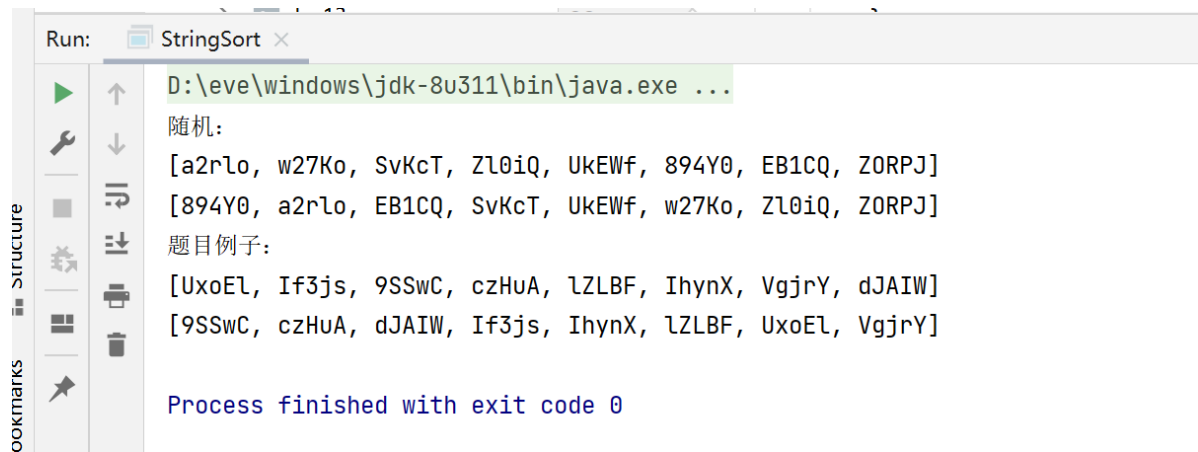
import java.util.Arrays;
import java.util.Random;

public class StringSort {

    public static void main(String[] args) {
        System.out.println("随机: ");
        String[] strings = getStrings();
        stringSort(strings);
        System.out.println("题目例子: ");
        String[] str = {"UxoEl", "If3js", "9SSwC", "czHuA", "lZLBF", "lhynX",
            "VgjrY", "dJAIW"};
        stringSort(str);
    }

    private static void stringSort(String[] str) {
        System.out.println(Arrays.toString(str));
        String temp;
        for (int i = 0; i < str.length; i++) {
            for (int j = 1; j < str.length - i; j++) {
                if (str[j].toUpperCase().charAt(0) < str[j - 1].toUpperCase().charAt(0)) {
                    temp = str[j];
                    str[j] = str[j - 1];
                    str[j - 1] = temp;
                }
            }
        }
        System.out.println(Arrays.toString(str));
    }

    private static String[] getStrings() {
        String str =
            "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789";
        Random random = new Random();
        String[] res = new String[8];
        for (int i = 0; i < 8; i++) {
            StringBuilder st = new StringBuilder();
            for (int j = 0; j < 5; j++) {
                int number = random.nextInt(62);
                st.append(str.charAt(number));
            }
            res[i] = st.toString();
        }
        return res;
    }
}
```

```
Run: StringSort x
D:\eve\windows\jdk-8u311\bin\java.exe ...
随机:
[a2rlo, w27Ko, SvKcT, Zl0iQ, UKEWf, 894Y0, EB1CQ, ZORPJ]
[894Y0, a2rlo, EB1CQ, SvKcT, UKEWf, w27Ko, Zl0iQ, ZORPJ]
题目例子:
[UxoEl, If3js, 9SSwC, czHuA, lZLBF, IhynX, VgjrY, dJAIW]
[9SSwC, czHuA, dJAIW, If3js, IhynX, lZLBF, UxoEl, VgjrY]

Process finished with exit code 0
```

6、

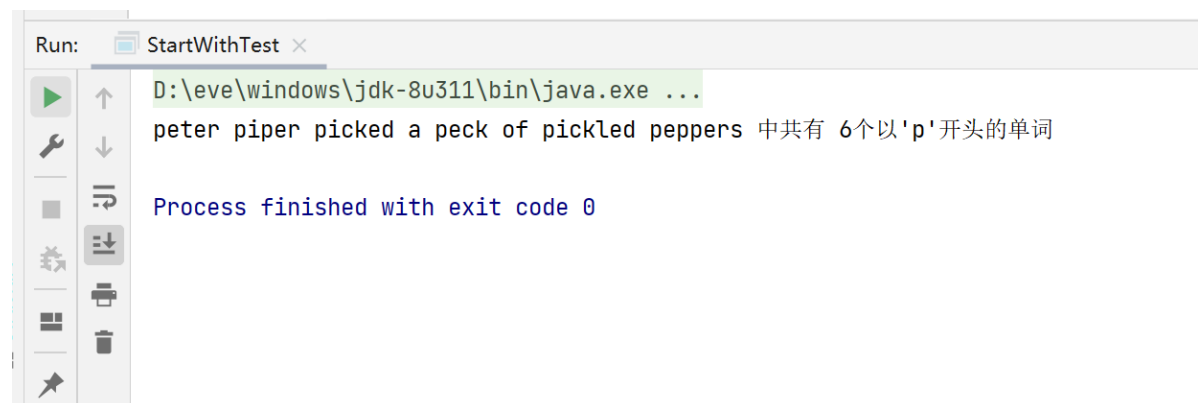
"peter piper picked a peck of pickled peppers"

统计这段绕口令有多少个以p开头的单词

提示:借助split(" ")方法分隔成字符串数组, startsWith("p")判断是否是p开头

```
package com.cskaoyan.homework.day14;

public class StartWithTest {
    public static void main(String[] args) {
        String str = "peter piper picked a peck of pickled peppers";
        String[] strings = str.split(" ");
        int count = 0;
        for (String astring : strings) {
            if (astring.startsWith("p")){
                count++;
            }
        }
        System.out.println(str+" 中共有 "+count+"个以'p'开头的单词");
    }
}
```



```
Run: StartWithTest x
D:\eve\windows\jdk-8u311\bin\java.exe ...
peter piper picked a peck of pickled peppers 中共有 6个以'p'开头的单词

Process finished with exit code 0
```

7、

把字符串"legendary"中字符 改成间隔大写小写模式,

即 "LeNgEnDaRy"

提示:

借助StringAPI

toLowerCase() 把字符串全部转化为小写

String toUpperCase() 把字符串全部转换为大写



```
package com.cskaoyan.homework.day14;




public class UpperLowerTest {

    public static void main(String[] args) {
        String res = upperLower("lengendary");
        System.out.println(res);
    }

    public static String upperLower(String str){
        StringBuilder res = new StringBuilder();
        for (int i = 0; i < str.length(); i++) {
            if (i%2 == 0){
                res.append(String.valueOf(str.charAt(i)).toUpperCase());
            }else {
                res.append(String.valueOf(str.charAt(i)).toLowerCase());
            }
        }
        return res.toString();
    }
}
```

Run: UpperLowerTest x

 D:\eve\windows\jdk-8u311\bin\java.exe ...
LeNgEnDaRy

 Process finished with exit code 0

