

一、填空题

1: 假设

```
String s1 = "Welcome to Java";
String s2 = s1;
String s3 = new String("Welcome to Java");
```

那么下面表达式的结果是什么?

(1) s1 == s2	_____ true _____
(2) s1 == s3	_____ flase _____
(3) s1.equals(s2)	_____ true _____
(4) s2.equals(s3)	_____ true _____
(5) s1.compareTo(s2);	_____ 0 _____
(6) s2.compareTo(s3);	_____ 0 _____
(7) s1.charAt(0);	_____ 'w' _____
(8) s1.indexOf('j');	_____ -1 _____
(9) s1.indexOf("to");	_____ 8 _____
(10) s1.lastIndexOf("o",15)	_____ 4 _____
(11) s1.substring(3, 11);	_____ come to _____
(12) s1.endsWith("Java")	_____ true _____
(13) s1.startsWith("wel");	_____ false _____
(14) " We come ".trim();	_____ "We come" _____
(15) s1.toUpperCase();	_____ "WELCOME TO JAVA" _____
(16) s1.replace('o', 'T');	_____ "WelTome tT Java" _____

2. 如果

```
StringBuffer s1 = new StringBuffer("Java");
StringBuffer s2 = new StringBuffer("HTML");
```

假设下列每个语句是独立的, 每条语句结束后, 写出相应结果

(1) s1.append(" is fun");	s1 为_ "Java_is fun" _
(2) s1.append(s2);	s1 为_ "JavaHTML" _
(3) s1.insert(2, "is fun");	s1 为_ "Jais funva" _
(4) s1.insert(1,s2);	s1 为_ "JHTMLava" _
(5) char c = s1.charAt(2);	c 为_ 'v' _
(6) int i = s1.length();	i 为_ 4 _
(7) s1.deleteCharAt(3);	s1 为_ "Jav" _
(8) s1.delete(1,3);	s1 为_ "Ja" _
(9) s1.reverse();	s1 为_ "avaJ" _
(10) s1.replace(1,3, "Computer");	s1 为_ "JComputera" _
(11) String s3 = s1.substring(1,3);	s3 为_ "av" _ , s1 为_ "Java" _
(12) String s4 = s1.substring(2);	s4 为_ "va" _ , s1 为_ "Java" _

3. 假设 `StringBuffer s = new StringBuffer("Welcome to JAVA");`
将 `s` 的内容清空的语句是 `___s.delete(0,s.length())_____`。

4. 如果

```
String s1 = "Welcome";  
String s2 = new String("Welcome");  
String s3 = s2.intern();  
String s4 = "Wel" + "come";  
String s5 = "Wel";  
String s6 = "come";  
String s7 = s5 + s6;  
String s8 = "Wel" + new String("come");
```

那么下面表达式的结果为:

- (1) `s1 == s2` `___false___`
- (2) `s1 == s3` `___true___`
- (3) `s1 == s4` `___true___`
- (4) `s1 == s7` `___false___`
- (5) `s1 == s8` `___false___`
- (6) `s1.equals(s2)` `___true___`
- (7) `s1.equals(s3)` `___true___`
- (8) `s1.equals(s4)` `___true___`
- (9) `s1.equals(s7)` `___true___`
- (10) `s1.equals(s8)` `___true___`

二、单项选择题

1. 可以获取字符串 `s` 的最后一个字符的表达式是 `___C___`。

- (A) `s.length()`
- (B) `s[s.length() - 1]`
- (C) `s.charAt(s.length() - 1)`
- (D) `charAt(s, length(s))`

2. 下面程序

```
class C {  
    public static void main(String[] args) {  
        String s = "null";  
        if(s == null)  
            System.out.print("a");  
    }  
}
```

```

        else if(s.length() == 0)
            System.out.print("b");
        else
            System.out.print("c");
    }
}

```

的输出为____C____。

(A) a

(B) b

(C) c

(D) null

3. 下面的程序

```

class C {
    public static void main(String[] args) {
        String s = "Welcome to ";
        concat(s);
        System.out.print(s);
    }
    public static void concat(String s) {
        s += "Java";
    }
}

```

的输出为__A_____。

(A) Welcome to

(B) Welcome to Java

(C) 编译错误

(D) 运行时异常

三、编程题

1: 编写程序, 从控制台或对话框任意输入一个英文字符串, 统计字符串中每个英文字母出现的次数并输出到控制台 (大小写不敏感)。

2: 假设一个车牌号码由三个大写字母和后面的四个数字组成。编写一个程序, 生成 5 个不重复的车牌号码。

1. 答:

```
countChars.java
2  import java.util.Scanner;
3  import java.util.TreeMap;
4
5  public class countChars {
6      public static void main(String[] args) {
7          Scanner in = new Scanner(System.in);
8          String input = in.nextLine();
9          input = input.trim();
10         input = input.toLowerCase();
11         Map<Character, Integer> map = new TreeMap<>();
12         for(int i = 0; i < input.length(); i++) {
13             if((input.charAt(i) >= 'a' && input.charAt(i) <= 'z') || (input.charAt(i) >= 'A' && input.charAt(i) <= 'Z')) {
14                 if(map.containsKey(input.charAt(i))) {
15                     map.replace(input.charAt(i), map.get(input.charAt(i)) + 1);
16                 } else {
17                     map.put(input.charAt(i), 1);
18                 }
19             }
20         }
21         System.out.println("统计结果如下: \n");
22         for(Map.Entry<Character, Integer> entry : map.entrySet()) {
23             System.out.println(" " + entry.getKey() + "出现次数为: " + entry.getValue());
24         }
25     }
26 }
27
```

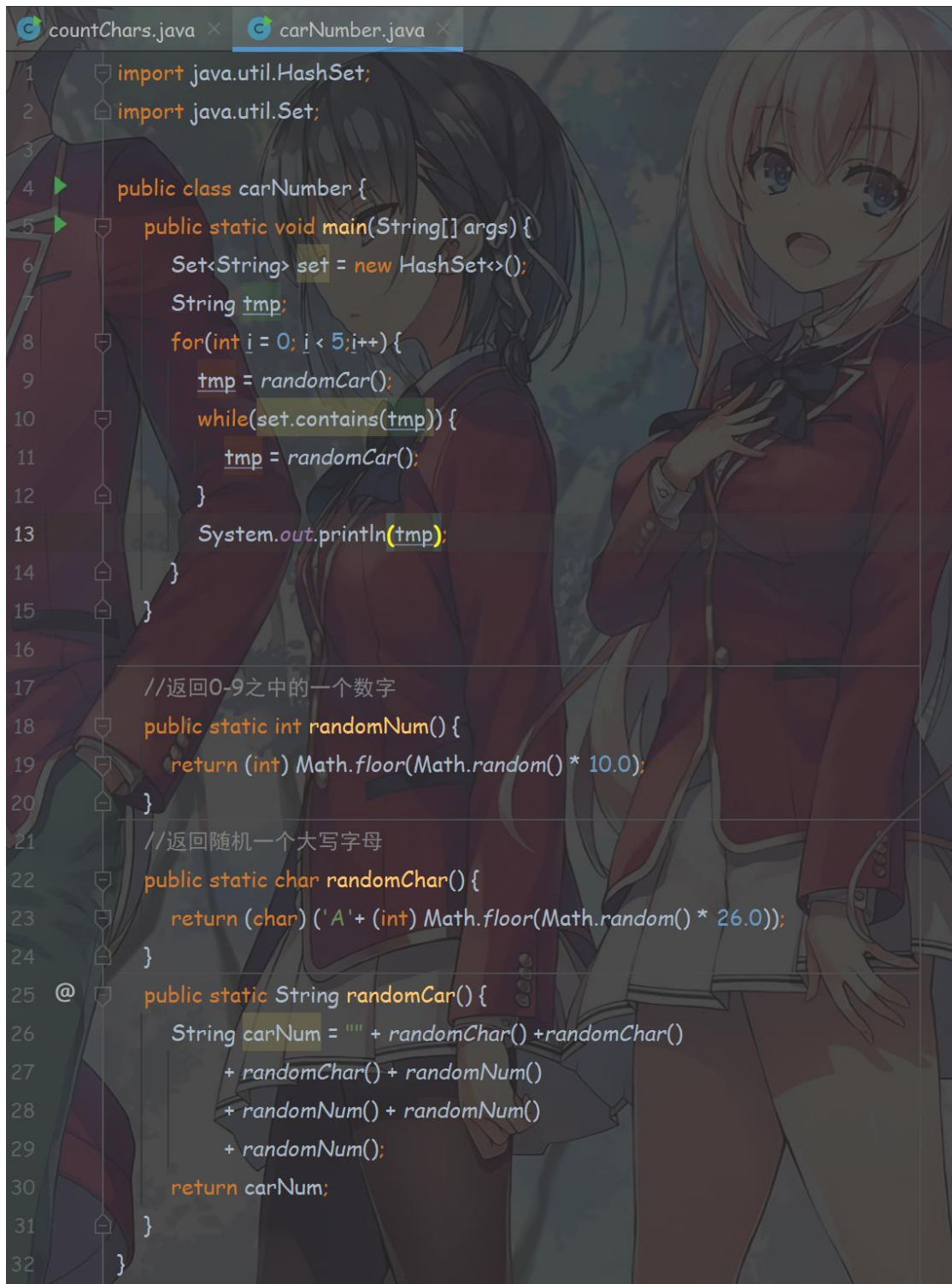
```
"D:\Program Files (x86)\jdk11\bin\java.exe" "-javaagent:D:\Program Files (x86)\IntelliJ IDEA 2020.2\lib\idea_rt.jar" 124125195khsdskjkl-fka
统计结果如下:

a出现次数为: 5
d出现次数为: 4
f出现次数为: 3
h出现次数为: 6
j出现次数为: 4
k出现次数为: 8
s出现次数为: 3

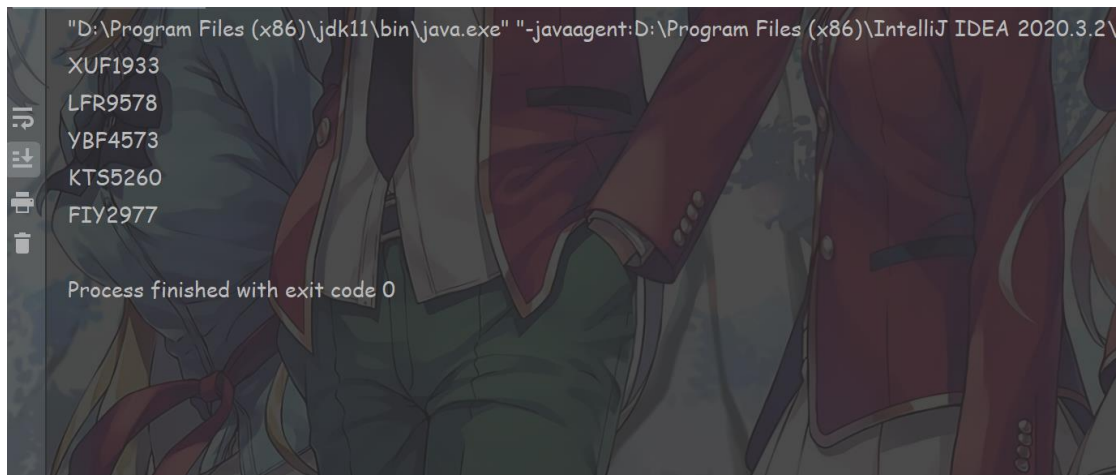
Process finished with exit code 0
```

使用 trim 处理了首尾空格, 过滤了字符串中不是英文字符的字符, 将所有英文字符都转为小写。用 map 来存储, 最后遍历得到结果。

2. 答:



```
countChars.java x carNumber.java x
1  import java.util.HashSet;
2  import java.util.Set;
3
4  public class carNumber {
5      public static void main(String[] args) {
6          Set<String> set = new HashSet<>();
7          String tmp;
8          for(int i = 0; i < 5; i++) {
9              tmp = randomCar();
10             while(set.contains(tmp)) {
11                 tmp = randomCar();
12             }
13             System.out.println(tmp);
14         }
15     }
16
17     //返回0-9之中的一个数字
18     public static int randomNum() {
19         return (int) Math.floor(Math.random() * 10.0);
20     }
21
22     //返回随机一个大写字母
23     public static char randomChar() {
24         return (char) ('A' + (int) Math.floor(Math.random() * 26.0));
25     }
26
27     @ public static String randomCar() {
28         String carNum = "" + randomChar() + randomChar()
29             + randomChar() + randomNum()
30             + randomNum() + randomNum()
31             + randomNum();
32         return carNum;
33     }
34 }
```



```
"D:\Program Files (x86)\jdk11\bin\java.exe" "-javaagent:D:\Program Files (x86)\IntelliJ IDEA 2020.3.2\
XUF1933
LFR9578
YBF4573
KT55260
FIY2977

Process finished with exit code 0
```

使用 `Math` 工具类生成 0-9 的随机数和 0-25 的随机数，从而得到 A-Z 的随机字符。
用 `Set` 集合类去重。