1、键盘输入任意字符串，打乱里面的内容   
  
1) 字符串是不可变的  
2) 如果要变化需要变成字符数组  
3) 调整完毕之后再变回字符串  
4) 使用随机数类Random   
  
代码如下：  
```*java*package hello;  
  
import java.util.Scanner;  
import java.util.Random;  
public class Test {  
 public static void main(String[] args) {  
 Scanner input = new Scanner(System.in);  
 Random random = new Random();  
 System.out.print("请输入字符串：");  
 String str = input.nextLine();  
  
 char[] c = str.toCharArray();   
  
 for(int i = c.length-1;i > 0; i--) {  
 int j = random.nextInt(i+1);  
 char temp = c[i];  
 c[i] = c[j];  
 c[j] = temp;   
 }  
  
 String NewStr = new String(c); // 字符数组创建字符串  
 System.out.println("打乱顺序后的字符串：" + NewStr);  
 input.close();  
 }  
}  
```  
  
结果展示：  
  
  
2、生成一个随机验证码  
  
规则：  
长度为5  
内容中是四位字母，1位数字。  
其中数字只有1位，但是可以出现在任意的位置。

代码如下：  
```*java*package hello;  
  
import java.util.Random;  
  
public class Test {  
 public static void main(String[] args) {  
 String str = TestKey();  
 System.out.println("随机生成的验证码"+str);  
 }  
  
 public static String TestKey() {  
 Random random = new Random();  
 StringBuilder s = new StringBuilder(5);  
 char[] letters = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz".toCharArray();  
  
 for(int i = 0; i < 4; i++) {  
 s.append(letters[random.nextInt(letters.length)]);  
 }  
  
 s.append(random.nextInt(10));  
 return shuffleString(s.toString());  
 }  
  
 private static String shuffleString(String str) {  
 char[] array = str.toCharArray();  
 Random random = new Random();  
 for (int i = array.length - 1; i > 0; i--) {  
 int j = random.nextInt(i + 1);   
   
 char temp = array[i];  
 array[i] = array[j];  
 array[j] = temp;  
 }  
 return new String(array);  
 }  
}  
```

结果展示：

文本

中度可信度描述已自动生成

  
  
3、请编写程序，由键盘录入一个字符串，统计字符串中英文字母和数字分别有多少个。比如：Hello12345World中字母：10个，数字：5个。

代码如下：  
```*java*package hello;  
  
import java.util.Scanner;  
public class Test {  
 public static void main(String[] args) {  
 Scanner input = new Scanner(System.in);  
   
 System.out.println("请输入一个字符串：");  
 String str = input.nextLine();  
  
 char[] c = str.toCharArray();  
  
 int letter = 0,num = 0;  
 for(int i = 0; i < c.length; i++) {  
 if ((c[i] >= 'a' && c[i] <= 'z') || (c[i] >= 'A' && c[i] <= 'Z')) {  
 letter++;  
 } else if (c[i] >= '0' && c[i] <= '9') {  
 num++;  
 }  
 }  
  
 System.out.println("字母：" + letter+ "个," + "数字:" + num +"个");  
 input.close();  
 }  
}  
```

结果展示：

图示, 示意图

描述已自动生成

4、请定义一个方法用于判断一个字符串是否是对称的字符串，并在主方法中测试方法。例如："abcba"、"上海自来水来自海上"均为对称字符串。  
  
提示：将字符串转换为StringBuilder类型，调用StringBuilder的reverse()方法将字符串反转。

代码如下：  
```*java*package hello;  
  
import java.util.Scanner;  
public class Test {  
 public static void main(String[] args) {  
 Scanner input = new Scanner(System.in);  
   
 System.out.println("请输入一个字符串：");  
 String str = input.nextLine();  
  
 StringBuilder sb = new StringBuilder(str);  
 sb.reverse();  
 String strp = sb.toString();  
 if(str.equals(strp)) {  
 System.out.println("是对称字符串。");   
 } else {  
 System.out.println("不是对称字符串。");  
 }  
 }  
}  
```

结果展示：

图示, 示意图

描述已自动生成

手机屏幕的截图

描述已自动生成  
5、我国的居民身份证号码，由由十七位数字本体码和一位数字校验码组成。请定义方法判断用户输入的身份证号码是否合法，并在主方法中调用方法测试结果。规则为：号码为18位，不能以数字0开头，前17位只可以是数字，最后一位可以是数字或者大写字母X。  
  
代码如下：

```*java*package hello;  
  
import java.util.Scanner;  
public class Test {  
 public static void main(String[] args) {  
 Scanner input = new Scanner(System.in);  
   
 System.out.println("请输入一个身份证号：");  
 String Id = input.nextLine();  
 if(IsId(Id)) {  
 System.out.println("该身份证号码是合法的。");  
 } else {  
 System.out.println("该身份证号码是不合法的。");  
 }  
 input.close();  
 }  
  
 public static boolean IsId(String Id) {  
 if(Id == null || Id.length() != 18) {  
 return false;  
 } else {  
 char[] IdC = Id.toCharArray();  
 char[] test = "1234567890X".toCharArray();  
 boolean temp = false;  
   
 if (IdC[0] == '0') {  
 temp = false;  
 } else {  
 for(char c : test) {  
 if(IdC[17] == c) {  
 temp = true;  
 break;  
 }  
 }  
 }  
   
 return temp;  
 }  
 }   
  
}  
```

结果展示：

文本

中度可信度描述已自动生成

文本

描述已自动生成

文本

描述已自动生成  
6、定义一个方法，把 int 数组中的数据按照指定的格式拼接成一个字符串返回，调用该方法，并在控制台输出结果。

代码如下：  
```*java*package hello;  
  
public class Test {  
 public static void main(String[] args) {  
 int[] arr = {1,2,3};  
 String str = ArrayToString(arr);  
 System.out.println("拼接后的字符串："+str);  
 }  
  
 public static String ArrayToString(int[] arr) {  
 if(arr == null) {  
 return "null";  
 } else if(arr.length == 0) {  
 return "[]";  
 } else {  
 String str = "";  
 str += '[';  
  
 for(int i = 0;i < arr.length; i++) {  
 str += arr[i];  
  
 if(i != arr.length-1) {  
 str += ",";  
 }  
   
 }   
 str += "]";  
 return str;  
 }  
 }  
}  
```

结果展示：

  
  
7、定义一个方法，把 int 数组中的数据按照指定的格式拼接成一个字符串返回，调用该方法，并在控制台输出结果。（至少使用两种方法实现）  
  
要求：  
  
1、如果传递的参数为空，返回null  
  
2、如果传递的数组元素个数为0，返回[]  
  
3、如果数组为int[] arr = {1,2,3}; ，执行方法后的输出结果为：[1, 2, 3]

代码如下：  
```*java*package hello;  
  
public class Test {  
 public static void main(String[] args) {  
 int[] arr = {1,2,3};  
 String str = ArrayToString(arr);  
 System.out.println("拼接后的字符串："+str);  
 }  
  
 public static String ArrayToString1(int[] arr) {  
 if(arr == null) {  
 return "null";  
 } else if(arr.length == 0) {  
 return "[]";  
 } else {  
 StringBuilder str = new StringBuilder("");  
 str.append("[");  
  
 for(int i = 0;i < arr.length; i++) {  
 str.append(arr[i]);  
  
 if(i != arr.length-1) {  
 str.append(",");  
 }  
   
 }   
 str.append(']');  
 return str.toString();  
 }  
 }  
   
 public static String ArrayToString2(int[] arr) {  
 if(arr == null) {  
 return "null";  
 } else if(arr.length == 0) {  
 return "[]";  
 } else {  
 String str = "";  
 str += '[';  
  
 for(int i = 0;i < arr.length; i++) {  
 str += arr[i];  
  
 if(i != arr.length-1) {  
 str += ",";  
 }  
   
 }   
 str += "]";  
 return str;  
 }  
 }  
}  
  
```

结果展示：

