

模拟扑克牌游戏的洗牌

实验编号 exp02

0.1 实验要求

结合面向对象设计原则，分析和设计模拟扑克牌游戏的洗牌过程。

1. 编程定义一个表示扑克牌的类 **Poke**，用 **suit** 来表示扑克牌的花色，用 **face** 来表示扑克牌的牌面值，每副牌为 52 张，不考虑包含大小王。
2. 假设扑克牌游戏需要 2 付 (或 2 付以上) 的牌，请编程实现模拟洗牌，以及将这些扑克牌分给参加游戏的所有人，并将参加扑克牌游戏的所有人的扑克牌输出。
3. 实现 **Poke** 类中包含的 4 个静态方法，所实现的 **Poke** 类在 **PokeTest** 的 **main** 方法中进行测试，请阅读 **main** 方法的代码，并根据其中逻辑猜测 **Poke** 类中静态方法的行为并编写方法实现代码。

Poke.java

```
1 package ouc.cs.course.java.test.poke;
2 import java.util.Random;
3
4 public class Poke {
5     public static String[] createCard(int number) {
6     }
7
8     public static void display(String[] cards) {
9     }
10
11     public static void shuffle(String[] cards) {
12     }
13
14     public static void distribute(String[] cards,int player) {
15     }
16 }
```

PokeTest.java

```

1 package ouc.cs.course.java.test.poke;

3 import java.util.Scanner;
4 import ouc.cs.course.java.test.poke.Poke;

6 public class PokeTest {

8     @SuppressWarnings("resource")
9     public static void main(String[] args) {
10         System.out.println("该扑克游戏需要几付扑克牌? ");
11         Scanner sc = new Scanner(System.in);
12         int num = sc.nextInt();
13         String[] cards = Poke.createCard(num);
14         System.out.println("该扑克游戏有几个玩家? ");
15         Scanner sp = new Scanner(System.in);
16         int per = sp.nextInt();
17         System.out.println("\n显示所有的牌: ");
18         Poke.display(cards);
19         Poke.shuffle(cards);
20         System.out.println("\n显示分配给每个人的牌: ");
21         Poke.distribute(cards, per);

23     }
24 }

```

4. 要随机产生某个范围内的整数，可以用 `java.util.Random` 类的 `nextInt(int num)`。例如，输出 0 - 51 范围内的一个整数如下：

```

1 Random rand = new Random();
2 int num = rand.nextInt(52);

```

0.2 实验过程、步骤及原始记录

实验过程和代码如下：

```

1 .
2 .
3 .
4 .
5 .
6 .
7 .
8 .
9 .
10 .

```