

目 录

第一章	顺序表	1	1.4	查找	1
1.1	创建顺序表	1	1.5	获取/修改元素	2
1.2	往顺序表中添加元素	1	1.6	迭代器: <code>Iterator</code>	2
1.3	删除元素	1	1.7	练习: 扑克牌游戏	2

第一章 顺序表

1.1 创建顺序表

```
1 //1. 创建ArrayList实例
2 ArrayList<String> arrayList = new ArrayList<>();
3 //向上转型
4 List<String> arrayList = new ArrayList<>();
```

1.2 往顺序表中添加元素

add 一个参数

```
1 //2. 添加元素
2 //add一个参数版本的方法是把元素添加到顺序表的末尾
3 arrayList.add("C");
4 arrayList.add("C++");
5 arrayList.add("Java");
6 arrayList.add("Python");
```

add 两个参数

```
1 //2. 添加元素
2 //add两个参数版本的方法是把元素添加到顺序表的指定位置上
3 arrayList.add(2, "JavaScript");
4 //直接打印对象，会触发对象的toString方法
5 System.out.println(arrayList);
```

1.3 删除元素

```
1 //3. 删除元素
2 //按位置删除
3 arrayList.remove(2);
4 //按照值删除，如果有多个相同的值，只会删除第一个
5 arrayList.remove("JavaScript");
6 //特殊情况，删除整型的
7 List<Integer> arrayList2 = new ArrayList<>();
8 for (int i = 0; i < 5; i++) {
9     arrayList2.add(i + 1);
10 }
11 arrayList2.remove(2); //删除的是下标为2的元素而不是值为2的
12 arrayList2.remove(Integer.valueOf(2)); //这个才是删除值为2的元素
```

1.4 查找

```

1 //4. 查找
2
3 //查找是否存在
4 boolean ret = arrayList.contains("Java");
5 System.out.println("查找Java的结果: "+ ret);
6 //查找具体的位置
7 int index = arrayList.indexOf("Java");
8 System.out.println("查找Java的位置: "+ index);

```

1.5 获取/修改元素

```

1 //5. 获取元素/修改元素
2 //获取元素
3 arrayList.get(0); //获取下标为0的元素
4 //修改元素
5 arrayList.set(0, "C#"); //把下标为0的元素修改为C#

```

1.6 迭代器: Iterator

迭代器是用来遍历集合的一种重要手段, 在非线性结构迭代器会大显身手

```

1 //6. 遍历操作
2 //通过下标遍历
3 for (int i = 0; i < arrayList.size(); i++) {
4     System.out.println(arrayList.get(i));
5 }
6
7 //通过迭代器来遍历
8 //先通过iterator方法获取到迭代器对象
9 Iterator<String> iterator = arrayList.iterator();
10 //再通过while循环来进行遍历
11 while (iterator.hasNext()){
12     String elem = iterator.next();
13     System.out.println(elem);
14 }
15
16 //使用for-each来遍历
17 for (String str:arrayList) {
18     System.out.println(str);
19 }

```

1.7 练习: 扑克牌游戏

```

1 //一张扑克牌
2 public class Card {

```

```

3    //花色
4    protected String suit;
5    //点数
6    protected String rank;
7
8    public Card(String suit, String rank) {
9        this.suit = suit;
10       this.rank = rank;
11    }
12    @Override
13    public String toString(){
14        return "(" + this.suit + this.rank + ")";
15    }
16 }
17
18
19 import java.util.ArrayList;
20 import java.util.Collections;
21 import java.util.List;
22 import java.util.Random;
23
24 public class PokerName {
25     //存放4种花色
26     public static final String[] suits = {" ", " ", " ", " "};
27     //存放点数
28     public static final String[] ranks = {"A", "2", "3", "4", "5", "6", "7", "8", "9", "10",
29         "J", "Q", "K"};
30
31     private static List<Card> buyPoker(){
32         List<Card> poker = new ArrayList<>();
33         for (int i = 0; i < suits.length; i++) {
34             for (int j = 0; j < ranks.length ; j++) {
35                 poker.add(new Card(suits[i], ranks[j]));
36             }
37         }
38         poker.add(new Card("red", "Joker"));
39         poker.add(new Card("black", "Joker"));
40         return poker;
41     }
42
43     private static void shuffle(List<Card> poker){
44         Random random = new Random();
45         for (int i = poker.size()-1; i > 0 ; i--) {
46             int pos = random.nextInt(i);
47             swap(poker, i, pos);
48         }
49     }
50
51     private static void swap(List<Card> poker, int i, int pos) {
52         Card tmp = poker.get(i);
53         poker.set(i, poker.get(pos));

```

```
53     poker.set(pos,tmp);
54 }
55
56 public static void main(String[] args) {
57     //1.创建出一副扑克牌
58     List<Card> poker = buyPoker();
59     System.out.println(poker);
60     System.out.println("-----");
61     //2.洗牌
62     shuffle(poker);
63     System.out.println(poker);
64     System.out.println("-----");
65     shuffle(poker);
66     System.out.println(poker);
67     System.out.println("-----");
68     //标准库自带的洗牌方法
69     Collections.shuffle(poker);
70     System.out.println(poker);
71     //3.发牌：每个玩家发5张牌，这五张牌用另一个ArrayList表示
72     List<List<Card>> players = new ArrayList<>();
73     Scanner scanner = new Scanner(System.in);
74     System.out.println("请输入玩家的个数：");
75     int num = scanner.nextInt();
76     for (int i = 0; i < num; i++) {
77         players.add(new ArrayList<Card>());
78     }
79     for (int i = 0; i < 5; i++) {
80         for (int j = 0; j < num; j++) {
81             Card topCard = poker.remove(0);
82             List<Card> player = players.get(j);
83             player.add(topCard);
84         }
85     }
86     System.out.println(players);
87     System.out.println("-----");
88     //4.展示手牌：
89     for (int i = 0; i < players.size(); i++) {
90         System.out.println("第" + (i + 1) + "位玩家的牌是：" + players.get(i))
91         ;
92     }
93 }
```