

综合项目1【界面抢红包--普通红包与手气红包】

安徽大学 软件工程 刘金辉

邮箱：ljh.ahu@qq.com

项目Github地址：<https://github.com/ahuljh/Java-Project>

项目包括知识点：

重写(implements)、继承(extends)、接口(interface)、抽象类(abstract)、final关键字(修饰类、方法、变量)

注释、包的分层、代码替换快捷键

一、项目介绍

1、RedPacketFrame抽象类【red包下】：

实现图形界面和对应位置数据的输入

定义成员变量String类型的ownerName(群主名称)

定义成员变量OpenMode类型的openWay(红包的类型 [普通红包/手气红包])

定义两个成员变量的Set方法

在红包展示的页面中，得到不同红包mode中的divide返回的list，即各份红包的金额

2、OpenMode接口【red包下】：

包括divide函数，传入两个参数totalMoney和totalCount

总金额totalMoney分成totalCount份，并将结果以ArrayList<Integer>的方式进行返回

3、MyRed为RedPacketFrame抽象类的子类【demo08包下】：

抽象类不能被实例化，因此需要通过子类进行实例化

加入了子类的有参构造方法，即对父类RedPacketFrame的构造方法进行了重写
super(String);

4、普通红包NormalMode为implements实现的OpenMode接口【demo08包下】：

Tips:只能继承extends一个父类，但可以实现implements多个接口

@Override重写了OpenMode接口的divide函数

实现了红包金额的平均分配，群主分平均值+余数

5、手气红包RandomMode为implements实现的OpenMode接口【demo08包下】：

@Override重写了OpenMode接口的divide函数

实现了红包金额的随机分配，各份红包数量满足范围 $1 + \text{random.nextInt}(\text{leftMoney} / \text{leftCount} * 2)$

最后一个金额无需随机

6、测试实现类Bootstrap【demo08包下】：

实例化界面及红包数目显示子类的对象red

red对象调用其set成员方法

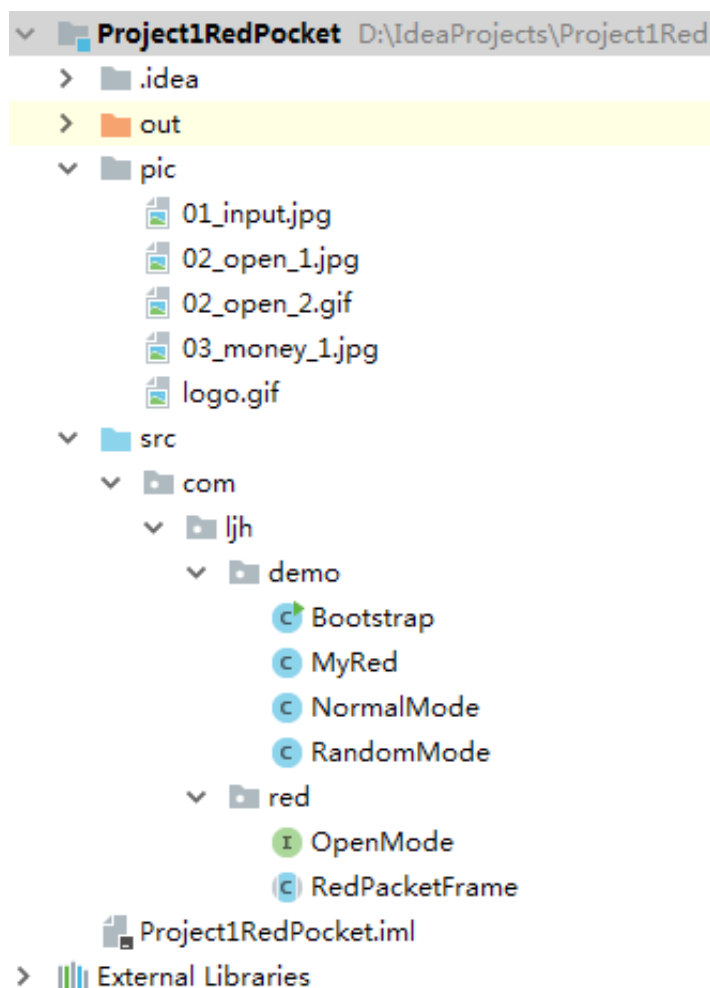
实际输入的红包数目、总金额，是在其父类RedPacketFrame中实现的

在RedPacketFrame中调用Mode重写的divide方法，得到最终的返回list

实际实现类中只需要有ownerName和openWay两个关键参数，并通过set方法传入抽象父类中

二、项目结构及运行结果

1、项目结构



2、运行结果

(1) 手气红包





(2) 普通红包





三、所需图片素材

1、红包界面图

路径：DemoName/pic/01_input.jpg

←

发红包

?

拼

总金额

元

每人抽到的金额随机，[改为普通红包](#)

红包个数

个

本群共 人

留言

塞钱进红包

2、抢红包页面图

路径：DemoName/pic/02_open_1.jpg



3、开红包感应区

路径：DemoName/pic/02_open_2.gif



4、红包详情

路径：DemoName/pic/03_money_1.jpg



5、位于导航栏的红包logo



路径：DemoName/pic/logo.gif



1.红包框架图形界面抽象父类RedPacketFrame

路径：DemoName/red/RedPacketFrame.java

```
1 package com.ljh.red;  
2  
3 //图形界面
```

```
4 import javax.swing.*;
5 import java.awt.*;
6 import java.awt.event.KeyAdapter;
7 import java.awt.event.KeyEvent;
8 import java.awt.event.MouseAdapter;
9 import java.awt.event.MouseEvent;
10 import java.util.ArrayList;
11 import java.util.HashMap;
12
13 /**
14  * 红包的框架 RedPacketFrame
15  *
16  * AWT / Swing / JavaFX
17  *
18  * @author 不是我
19  */
20 public abstract class RedPacketFrame extends JFrame {
21
22     private static final long serialVersionUID = 1L;
23
24     private static final String DIR = "pic";
25
26     private ArrayList<Integer> moneyList = null;
27
28     private static int initMoney = 0;
29     private static int totalMoney = 0; // 单位为“分”
30     private static int count = 0;
31
32     private static HashMap<JPanel, JLabel> panelLabel = new HashMap<>();
33
34     // 设置字体
35     private static Font fontYaHei = new Font("微软雅黑", Font.BOLD, 20);
36     private static Font msgFont = new Font("微软雅黑", Font.BOLD, 20);
37     private static Font totalShowFont = new Font("微软雅黑", Font.BOLD, 40);
38     private static Font nameFont = new Font("微软雅黑", Font.BOLD, 40);
39     private static Font showNameFont = new Font("微软雅黑", Font.BOLD, 20);
40     private static Font showMoneyFont = new Font("微软雅黑", Font.BOLD, 50);
41     private static Font showResultFont = new Font("微软雅黑", Font.BOLD, 15);
42
43     /**
```



```
44      * 窗体大小 WIDTH:400 HEIGHT:600
45      */
46      private static final int FRAME_WIDTH = 416; // 静态全局窗口大小
47      private static final int FRAME_HEIGHT = 650;
48      private static JLayeredPane layeredPane = null;
49
50      /// private static JPanel contentPane = null;
51
52      /**
53       * page1:输入页面 - InputPanel . 组件和初始化!
54       */
55      private static JPanel inputPanel = new JPanel();
56
57      // private static JTextField input_total = new JTextField("200"); // 测试
58      // private static JTextField input_count = new JTextField("3"); // 测试月
59      private static JTextField input_total = new JTextField();
60      private static JTextField input_count = new JTextField();
61      private static JTextField input_people = new JTextField("30");
62      private static JTextField input_msg = new JTextField("恭喜发财 , 大吉大");
63      private static JTextField input_total_show = new JTextField("$ " + input
64      private static JLabel input_inMoney = new JLabel(); // 不可见
65      private static JLabel input_bg_label = new JLabel(new ImageIcon(DIR + "\
66
67      static {
68
69          // 设置位置
70          input_total.setBounds(200, 90, 150, 50);
71          input_count.setBounds(200, 215, 150, 50);
72          input_people.setBounds(90, 275, 25, 30);
73          input_msg.setBounds(180, 340, 200, 50);
74          input_total_show.setBounds(130, 430, 200, 80);
75          input_inMoney.setBounds(10, 535, 380, 65);
76          input_bg_label.setBounds(0, 0, 400, 600); // 背景
77
78          // 设置字体
79          input_total.setFont(fontYaHei);
80          input_count.setFont(fontYaHei);
81          input_people.setFont(fontYaHei);
82          input_msg.setFont(msgFont);
83          input_msg.setForeground(new Color(255, 233, 38)); // 字体颜色 为金色
```

```
84         input_total_show.setFont(totalShowFont);
85         input_inMoney.setFont(fontYaHei);
86
87         // 透明
88         input_people.setOpaque(false);
89         input_total_show.setOpaque(false);
90         // 编辑 -- 不可编辑
91         input_people.setEditable(false);
92         input_total_show.setEditable(false);
93
94         // 边界 -- 无
95         input_total.setBorder(null);
96         input_count.setBorder(null);
97         input_people.setBorder(null);
98         input_msg.setBorder(null);
99         input_total_show.setBorder(null);
100
101     }
102
103     /**
104      * page2:打开页面 - openPanel . 组件和初始化!
105      */
106     private static JPanel openPanel = new JPanel();
107
108     private static JTextField open_ownerName = new JTextField("谁谁谁");
109     private static JLabel open_label = new JLabel(new ImageIcon(DIR + "\\02_
110     private static JLabel open_bg_label = new JLabel(new ImageIcon(DIR + "\\
111
112     static {
113
114         // 设置 位置.
115         open_ownerName.setBounds(0, 110, 400, 50);
116         open_bg_label.setBounds(0, 0, 400, 620);
117         open_label.setBounds(102, 280, 200, 200);
118         open_ownerName.setHorizontalAlignment(JTextField.CENTER);
119
120         // 设置字体
121         open_ownerName.setFont(nameFont);
122         open_ownerName.setForeground(new Color(255, 200, 163)); // 字体颜色
123
```

```
124         // 背景色
125         // open_name.setOpaque(false);
126         open_ownerName.setBackground(new Color(219, 90, 68));
127
128         // 不可编辑
129         open_ownerName.setEditable(false);
130         // 边框
131         open_ownerName.setBorder(null);
132
133     }
134
135     /**
136      * page3:展示页面 - showPanel . 组件和初始化!
137      */
138     private static JPanel showPanel = new JPanel();
139     private static JPanel showPanel2 = new JPanel();
140     private static JScrollPane show_jsp = new JScrollPane(showPanel2);
141
142     private static JLabel show_bg_label = new JLabel(new ImageIcon(DIR + "\
143
144     private static JTextField show_name = new JTextField("用户名称");
145     private static JTextField show_msg = new JTextField("祝福信息");
146     private static JTextField show_money = new JTextField("99.99");
147     private static JTextField show_result = new JTextField(count + "个红包共'
148
149     static {
150         // 分别设置水平和垂直滚动条自动出现
151         // jsp.setHorizontalScrollBarPolicy(JScrollPane.HORIZONTAL_SCROLLBAR_AS_NEEDED);
152         // jsp.setVerticalScrollBarPolicy(JScrollPane.VERTICAL_SCROLLBAR_AS_NEEDED);
153
154         /*
155          * 两部分 页面 . 1.本人获得的红包-- showPanel 2.别人获得的红包-- show_jsp
156          */
157         show_name.setBounds(125, 180, 100, 30);
158         show_name.setOpaque(false);
159         show_name.setBorder(null);
160         show_name.setFont(showNameFont);
161
162         show_msg.setBounds(0, 220, 400, 30);
163         show_msg.setOpaque(false);
```

```
164     show_msg.setBorder(null);
165     show_msg.setFont(msgFont);
166     show_msg.setHorizontalAlignment(JTextField.CENTER);
167
168     show_money.setBounds(0, 270, 250, 40);
169     show_money.setOpaque(false);
170     show_money.setBorder(null);
171     show_money.setFont(showMoneyFont);
172     show_money.setForeground(new Color(255, 233, 38)); // 字体颜色 为金色
173     show_money.setHorizontalAlignment(SwingConstants.RIGHT);
174
175     show_result.setBounds(10, 460, 400, 20);
176     show_result.setOpaque(false);
177     show_result.setBorder(null);
178     show_result.setFont(showResultFont);
179     show_result.setForeground(new Color(170, 170, 170)); // 字体颜色 为灰
180
181     // 设置 图片.
182     show_bg_label.setBounds(0, 0, 400, 500);
183
184 }
185
186 static {
187
188     // 页面和 背景的对对应关系.
189     panelLable.put(inputPanel, input_bg_label);
190     panelLable.put(openPanel, open_bg_label);
191     panelLable.put(showPanel, show_bg_label);
192 }
193
194 private void init() {
195     // 层次面板-- 用于设置背景
196     layeredPane = this.getLayeredPane();
197 //     System.out.println("层次面板||" + layeredPane);
198     // System.out.println(layeredPane);
199
200     // 初始化框架 -- logo 和基本设置
201     initFrame();
202     // 初始化 三个页面 -- 准备页面
203     initPanel();
```

```

204
205     // 2.添加 页面 --第一个页面, 输入 panel 设置到 页面上.
206     setPanel(inputPanel);
207
208     // 3.添加 监听
209     addListener();
210 }
211
212
213 /**
214  * 初始化框架 -- logo 和基本设置
215  */
216 private void initFrame() {
217     // logo
218     this.setIconImage(Toolkit.getDefaultToolkit().getImage(DIR + "\\logo
219 //      System.out.println("LOGO初始化...");
220
221     // 窗口设置
222     this.setSize(FRAME_WIDTH, FRAME_HEIGHT); // 设置界面大小
223     this.setLocation(280, 30); // 设置界面出现的位置
224     this.setDefaultCloseOperation(EXIT_ON_CLOSE);
225     this.setLayout(null);
226
227     // 测试期 注释 拖拽 , 运行放开
228     // this.setResizable(false);
229     this.setVisible(true);
230 }
231
232 /**
233  * 初始化页面-- 准备三个页面
234  */
235
236 private void initPanel() {
237 //      System.out.println("页面初始化...");
238     initInputPanel();
239     initOpenPanel();
240     initShowPanel();
241
242 }
243

```

```
244     private void initInputPanel() {
245
246         inputPanel.setLayout(null);
247         inputPanel.setBounds(0, -5, 400, 600);
248
249         // this.add(bg_label);
250         inputPanel.add(input_total);
251         inputPanel.add(input_count);
252         inputPanel.add(input_people);
253         inputPanel.add(input_msg);
254         inputPanel.add(input_total_show);
255         inputPanel.add(input_inMoney);
256
257         //      System.out.println("输入页面||" + inputPanel);
258
259     }
260
261     private void initOpenPanel() {
262         openPanel.setLayout(null);
263         openPanel.setBounds(0, 0, 400, 600);
264         // this.add(bg_label);
265         openPanel.add(open_ownerName);
266         openPanel.add(open_label);
267         //      System.out.println("打开页面||" + openPanel);
268     }
269
270     private void initShowPanel() {
271         showPanel.setLayout(null);
272         showPanel.setBounds(10, 10, 300, 600);
273
274         // =====
275         showPanel.add(show_name);
276         showPanel.add(show_msg);
277         showPanel.add(show_money);
278         showPanel.add(show_result);
279         //      System.out.println("展示页面||" + showPanel);
280         // =====
281         // showPanel2.setLayout(null);
282         // showPanel2.setBounds(0, 500, 401, 300);
283
```

```
284     showPanel2.setPreferredSize(new Dimension(300, 1000));
285     showPanel2.setBackground(Color.white);
286
287     show_jsp.setBounds(0, 500, 400, 110);
288
289 }
290
291 /**
292  * 每次打开页面，设置 panel的方法
293  */
294 private void setPanel(JPanel panel) {
295     // 移除当前页面
296     layeredPane.removeAll();
297
298     // System.out.println("重新设置:新页面");
299     // 背景lable添加到layeredPane的默认层
300     layeredPane.add(panelLable.get(panel), JLayeredPane.DEFAULT_LAYER);
301
302     // 面板panel设置为透明
303     panel.setOpaque(false);
304
305     // 面板panel 添加到 layeredPane的modal层
306     layeredPane.add(panel, JLayeredPane.MODAL_LAYER);
307 }
308
309 // private void setShowPanel(JPanel show) {
310 // setPanel(show);
311 // layeredPane.add(show_jsp, JLayeredPane.MODAL_LAYER);
312 //
313 // }
314
315 /**
316  * 设置组件的监听器
317  */
318 private void addListener() {
319
320     input_total.addKeyListener(new KeyAdapter() {
321         @Override
322         public void keyReleased(KeyEvent e) {
323             // System.out.println(e);
```

```

324         String input_total_money = input_total.getText();
325         input_total_show.setText("$ " + input_total_money);
326     }
327 });
328
329 input_count.addKeyListener(new KeyAdapter() {
330
331     @Override
332     public void keyReleased(KeyEvent e) {
333         // System.out.println(e);
334 //         System.out.println("个数:" + input_count.getText());
335     }
336 });
337 input_msg.addKeyListener(new KeyAdapter() {
338
339     @Override
340     public void keyReleased(KeyEvent e) {
341         // System.out.println(e);
342 //         System.out.println("留言:" + input_msg.getText());
343     }
344 });
345
346 input_inMoney.addMouseListener(new MouseAdapter() {
347     @Override
348     public void mouseClicked(MouseEvent e) {
349         try {
350
351             // 获取页面的值。
352             totalMoney = (int) (Double.parseDouble(input_total.getText()));
353             count = Integer.parseInt(input_count.getText());
354             if (count > 30) {
355                 JOptionPane.showMessageDialog(null, "红包个数不得超过30");
356                 return;
357             }
358
359             initMoney = totalMoney;
360
361             System.out.println("总金额: [" + totalMoney + "]分");
362             System.out.println("红包个数: [" + count + "]个");
363

```



```

364         input_inMoney.removeMouseListener(this);
365
366         //         System.out.println("跳转-->打开新页面");
367
368         // 设置群主名称
369         open_ownerName.setText(ownerName);
370         // 设置打开页面
371         setPanel(openPanel);
372
373         } catch (Exception e2) {
374             JOptionPane.showMessageDialog(null, "请输入正确【总金额】可
375
376         }
377     }
378 });
379
380 // open_ownerName ,点击 [名称],触发的方法 , 提示如何设置群主名称.
381
382 open_ownerName.addMouseListener(new MouseAdapter() {
383     @Override
384     public void mouseClicked(MouseEvent arg0) {
385         JOptionPane.showMessageDialog(null, "请通过【setOwnerName】方
386             JOptionPane.QUESTION_MESSAGE);
387     }
388 });
389
390 // open_label , 点击 [开],触发的方法,提示如何设置打开方式.
391 open_label.addMouseListener(new MouseAdapter() {
392     @Override
393     public void mouseClicked(MouseEvent e) {
394         if (openWay == null) {
395             JOptionPane.showMessageDialog(null, "请通过【setOpenWay】
396                 JOptionPane.QUESTION_MESSAGE);
397             return;
398         }
399
400 //         System.out.println("跳转-->展示页面");
401 //点 开 之后, 会调用openWay接口的divide函数, 将各成员红包列表返回,
402 moneyList = openWay.divide(totalMoney, count);
403

```

```
404 //          System.out.println(moneyList);
405          /*
406          * showPanel 添加数据
407          *
408          */
409          show_name.setText(ownerName);
410          show_msg.setText(input_msg.getText());
411          if (moneyList.size() > 0) {
412              show_money.setText(moneyList.get(moneyList.size() - 1) ,
413              }
414          show_result.setText(count + "个红包共" + (initMoney / 100.0)
415
416          open_label.removeMouseListener(this);
417
418          setPanel(showPanel);
419
420          // 添加数据
421          for (int i = 0; i < moneyList.size(); i++) {
422
423              JTextField tf = new JTextField();
424              tf.setBorder(null);
425              tf.setFont(showNameFont);
426              tf.setHorizontalAlignment(JTextField.LEFT);
427              if (i == moneyList.size() - 1) {
428                  tf.setText(ownerName + ": \t" + moneyList.get(i) / 1
429
430              } else {
431
432                  tf.setText("群成员-" + i + ": \t" + moneyList.get(i)
433              }
434              showPanel2.add(tf);
435          }
436
437          layeredPane.add(show_jsp, JLayeredPane.MODAL_LAYER);
438      }
439
440  });
441
442  }
443
```

```

444  /* =====
445  * *****
446  * *   以上代码均为页面部分处理,包括布局/互动/跳转/显示等,大家
447  * *
448  * *
449  * *****
450  * =====
451  */
452
453  /**
454  //成员变量
455  * ownerName : 群主名称
456  */
457  private String ownerName = "谁谁谁"; // 群主名称
458  /**
459  * openWay : 红包的类型 [普通红包/手气红包]
460  */
461  private OpenMode openWay = null;
462
463
464  /**
465  * 构造方法: 生成红包界面。
466  *
467  * @param title 界面的标题
468  */
469  //只有一个有参数的构造方法
470  public RedPacketFrame(String title) {
471      super(title);
472
473      // 页面相关的初始化
474      init();
475  }
476
477  public void setOwnerName(String ownerName) {
478      this.ownerName = ownerName;
479  }
480
481  public void setOpenWay(OpenMode openWay) {
482      this.openWay = openWay;
483  }

```

2、分红包策略接口OpenMode

路径：DemoName/red/OpenMode.java

```

1 package com.ljh.red;
2
3 import java.util.ArrayList;
4
5 //接口---除法
6 public interface OpenMode {
7     /**
8      * 请将totalMoney分成count份，保存到ArrayList<Integer>中，返回即可。
9      *
10     * @param totalMoney      总金额为方便计算，已经转换为整数，单位为分。【
11     * @param totalCount      红包个数
12     * @return ArrayList<Integer> 元素为各个红包的金额值，所有元素的值累和等于总
13     */
14     //除法，将总金额分成totalCount份【分发】
15     ArrayList<Integer> divide(int totalMoney, int totalCount);
16 }

```

3、红包框架图形界面抽象类RedPacketFrame的子类MyRed

路径：DemoName/src/cn/itcast/day11/demo08/MyRed.java

原因：抽象类不能被实例化，只能通过子类

备注：子类可以使用父类的成员方法

```

MyRed red = new MyRed("微信红包");
// 设置发红包的人
red.setOwnerName("王思聪");

//设置红包的打开方式
red.setOpenWay(random);

```

```

1 package com.ljh.demo;
2
3 import com.ljh.red.RedPacketFrame;

```

```

4
5
6 public class MyRed extends RedPacketFrame {
7     /**
8      * 构造方法的目的：生成红包界面。
9      *
10     * @param title 界面的标题
11     */
12     public MyRed(String title) {
13         super(title);
14     }
15 }

```

4、具体发红包的主类Bootstrap

路径：DemoName/src/cn/itcast/day11/demo08/Bootstrap.java

```

1 package com.ljh.demo;
2
3 import com.ljh.red.OpenMode;
4 import com.ljh.red.RedPacketFrame;
5
6 /*
7 场景说明：
8     红包发出去之后，所有人都有红包，大家抢完了之后，最后一个红包给群主自己。
9 大多数代码都是现成的，我们需要做的就是填空题。
10 我们自己要做的事情有：
11     1. 设置一下程序的标题，通过构造方法的字符串参数
12     2. 设置群主名称
13     3. 设置分发策略：平均，还是随机？
14
15 红包分发的策略：
16     1. 普通红包（平均）： $totalMoney / totalCount$ ，余数放在最后一个红包当中。
17     2. 手气红包（随机）：最少1分钱，最多不超过平均数的2倍。应该越发越少。
18 */
19 public class Bootstrap {
20
21     public static void main(String[] args) {
22         //RedPacketFrame frame=new RedPacketFrame();抽象类报错，不能被实例化

```

```

23 //报错原因：只有一个构造方法，没有无参构造方法
24 /*
25     public RedPacketFrame(String title) {
26         super(title);
27     // 页面相关的初始化
28     init();
29     }
30     */
31 //所以需要新建一个实现类/子类---MyRed
32 //所以子类中也要调用其有参数的构造方法
33 //目的：生成红包界面
34 //只有这一行就能出现界面
35 MyRed red = new MyRed("微信红包");
36 // 设置发红包的人
37 red.setOwnerName("王思聪");
38 //有安全监测
39 // 普通红包
40 //     OpenMode normal = new NormalMode();
41 //     red.setOpenWay(normal);
42
43 // 手气红包
44 OpenMode random = new RandomMode();
45 //设置打开方式，没有就不能打开
46 red.setOpenWay(random);
47 }
48 }

```

5、接口实现方法1平均分发---NormalMode

路径：DemoName/src/cn/itcast/day11/demo08/NormalMode.java

```

1 package com.ljh.demo;
2
3 import com.ljh.red.OpenMode;
4
5 import java.util.ArrayList;
6 public class NormalMode implements OpenMode {
7     @Override
8     //重载接口的方法

```

```

9      //参数中的红包总数和个数不可变，定义为final变量
10     public ArrayList<Integer> divide(final int totalMoney, final int totalCo
11         ArrayList<Integer> list = new ArrayList<>();
12         int avg = totalMoney / totalCount; // 平均值
13         int mod = totalMoney % totalCount; // 余数，模，零头
14         //把各份红包金额放入ArrayList中
15         // 注意totalCount - 1代表，最后一个先留着
16         for (int i = 0; i < totalCount - 1; i++) {
17             list.add(avg);
18         }
19         // 有零头，需要放在最后一个红包当中
20         list.add(avg + mod);
21         return list;
22     }
23 }

```

6、接口实现方法2随机分发---RandomMode

路径：DemoName/src/cn/itcast/day11/demo08/RandomMode.java

```

1 package com.ljh.demo;
2
3 import com.ljh.red.OpenMode;
4
5 import java.util.ArrayList;
6 import java.util.Random;
7 public class RandomMode implements OpenMode {
8     @Override
9     public ArrayList<Integer> divide(final int totalMoney, final int totalCo
10         ArrayList<Integer> list = new ArrayList<>();
11         // 随机分配，有可能多，有可能少。
12         // 最少1分钱，最多不超过“剩下金额平均数的2倍”
13         // 第一次发红包，随机范围是0.01元~6.66元
14         // 第一次发完之后，剩下的至少是3.34元。
15         // 此时还需要再发2个红包
16         // 此时的再发范围应该是0.01元~3.34元（取不到右边，剩下0.01）
17         // 总结一下，范围的【公式】是：1 + random.nextInt(leftMoney / leftCount
18         Random r = new Random(); // 首先创建一个随机数生成器
19         // totalMoney是总金额，totalCount是总份数，不变

```

```
20 // 额外定义两个变量，分别代表剩下多少钱，剩下多少份
21 int leftMoney = totalMoney;
22 int leftCount = totalCount;
23
24 // 随机发前n-1个，最后一个不需要随机
25 for (int i = 0; i < totalCount - 1; i++) {
26     // 按照公式生成随机金额
27     int money = r.nextInt(leftMoney / leftCount * 2) + 1;
28     list.add(money); // 将一个随机红包放入集合
29     leftMoney -= money; // 剩下的金额越发越少
30     leftCount--; // 剩下还应该再发的红包个数，递减
31 }
32
33 // 最后一个红包不需要随机，直接放进去就得了
34 list.add(leftMoney);
35
36 return list;
37 }
38 }
```