

天津商业大学学生实验报告

开课实验室：现代信息交流中心 403

开课时间 2014 年 12 月 10 日

实验报告 2014 年 12 月 10 日

学院名称	信息工程学院	年级、专业、班	软件工程 1201	学号	20125041	姓名	王靖伟	同组姓名	
课程名称	软件体系结构	实验项目名称	实验二 设计模式上机实验二			指导教师	高珊		
实验类型	验证 <input type="checkbox"/> 综合 <input type="checkbox"/> 设计 <input type="checkbox"/> 创新 <input type="checkbox"/>						成绩		
教师评语	<div style="text-align: right;">教师签名：_____</div> <div style="text-align: right;">_____ 年 月 日</div>								
<p>实验报告内容一般包括以下几个内容：1、目的要求 2、仪器用具及材料（仪器名称及主要规格、用具名称） 3、实验内容及原理（简单但要抓住要点，写出依据原理） 4、操作方法与实验步骤 5、数据图表格（照片） 6、实验过程原始记录 7 数据处理及结果（按实验要求处理数据、结论） 8、作业题 9、讨论（对实验中存在的问题、进一步的想法等进行讨论）</p>									
<p>实验报告内容：</p> <p>一、实验目的</p> <p>通过本实验的学习，使学生了解软件体系结构与软件体系设计的基本思想，掌握各类设计模式的软件体系结构的设计与实现方法，训练学生在分析各种软件体系静态结构图基础上，利用面向对象的程序设计语言模拟实现各类设计模式实例的基本方法，为今后软件开发与设计的学习奠定基础。</p> <p>二、实验内容</p> <p>根据设计模式的具体实例，在完成对系统的需求建模，得到用例模型后，应针对每个用例进行业务分析，说明其具体的业务流程，现系统分析部指派您完成该项任务。要求：</p> <p>使用任意一种面向对象编程语言实现各类设计模式的实例，要求代码运行正确无误。</p> <p>三、实验要求</p> <p>根据本实验的特点，要求和具体条件，采用集中授课形式组织教学。</p> <p>四、实验准备</p> <p>预习实验目的和实验要求，熟悉各类设计模式的基本思想、设计思路 and 主要特征，为各设计模式的设计与实现做好预习工作。</p>									

注 1. 每个实验项目一份实验报告。2. 实验报告第一页学生必须使用规定的实验报告纸书写，附页用实验报告附页纸或 A4 纸书写，字迹工整，曲线要画在坐标纸上，线路图要整齐、清楚（不得徒手画）。3. 实验教师必须对每份实验报告进行批改，用红笔指出实验报告中的错、漏之处，并给出评语、成绩，签全名、注明日期。4. 待实验课程结束以后，要求学生把实验报告整理好，交给实验指导教师，加上实验课学生考勤及成绩登记表（见附件 2）、目录和学院统一的封面（见附件 3）后，统一装订成册存档。

天津商业大学学生实验报告附页

开课实验室：现代信息交流中心 403

开课时间 2014 年 12 月 10 日

实验报告 2014 年 12 月 10 日

五、实验原理、方法和手段

本实验属于综合型实验，其实验原理遵循面向对象的软件体系设计与实现的基本原理与实施步骤。实验方法为通过对各种设计模式静态结构类图、模式结构图的分析理清问题解决思路，并通过运用面向对象的程序设计语言对设计模式进行编码实现。具体实现方法由学生自行设计实验方案并加以实现。

六、实验器材

1. 计算机一台。
2. 面向对象程序设计语言，如Java开发环境。

七、实验步骤

1. 分析设计模式实例的各类静态结构图。
2. 运用面向对象的程序设计语言实现该模式。

八、实验结果

1. 原型模式实例的程序资源包结构图，如图1所示。

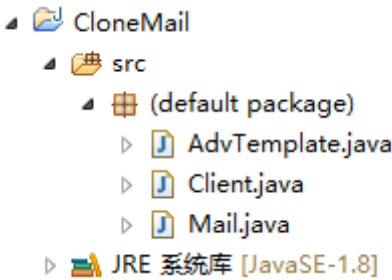


图1. 原型模式实例的程序资源包结构图

2. 源代码

行数	AdvTemplate. java
1	<code>public class AdvTemplate {</code>
2	<code> // 广告信名称</code>
3	<code> private String advSubject = "XX 银行国庆信用卡抽奖活动";</code>
4	<code> // 广告信内容</code>
5	<code> private String advContext = "国庆抽奖活动通知：只要刷卡就送你 1 百万!";</code>
6	
7	<code> // 取得广告信的名称</code>
8	<code> public String getAdvSubject() {</code>
9	<code> return this.advSubject;</code>
10	<code> }</code>
11	
12	<code> // 取得广告信的内容</code>
13	<code> public String getAdvContext() {</code>

14	<code>return this.advContext;</code>
15	<code>}</code>
16	<code>}</code>

行数	Mail.java
1	<code>public class Mail implements Cloneable {</code>
2	<code>// 收件人</code>
3	<code>private String receiver;</code>
4	<code>// 邮件名称</code>
5	<code>private String subject;</code>
6	<code>// 称谓</code>
7	<code>private String appellation;</code>
8	<code>// 邮件内容</code>
9	<code>private String contxt;</code>
10	<code>// 邮件的尾部，一般都是加上“xxx 版权所有”等信息</code>
11	<code>private String tail;</code>
12	
13	<code>// 构造函数</code>
14	<code>public Mail(AdvTemplate advTemplate) {</code>
15	<code> this.contxt = advTemplate.getAdvContext();</code>
16	<code> this.subject = advTemplate.getAdvSubject();</code>
17	<code>}</code>
18	
19	
20	<code>public Mail clone() {</code>
21	<code> Mail mail = null;</code>
22	<code> try {</code>
23	<code> mail = (Mail) super.clone();</code>
24	<code> } catch (CloneNotSupportedException e) {</code>
25	<code> // TODO Auto-generated catch block</code>
26	<code> e.printStackTrace();</code>
27	<code> }</code>
28	<code> return mail;</code>
29	<code>}</code>
30	
31	<code>// 以下为getter/setter方法</code>
32	<code>public String getReceiver() {</code>
33	<code> return receiver;</code>
34	<code>}</code>
35	
36	<code>public void setReceiver(String receiver) {</code>
37	<code> this.receiver = receiver;</code>
38	<code>}</code>
39	
40	<code>public String getSubject() {</code>
41	<code> return subject;</code>
42	<code>}</code>

43	
44	<code>public void setSubject(String subject) {</code>
45	<code> this.subject = subject;</code>
46	<code>}</code>
47	
48	<code>public String getAppellation() {</code>
49	<code> return appellation;</code>
50	<code>}</code>
51	
52	<code>public void setAppellation(String appellation) {</code>
53	<code> this.appellation = appellation;</code>
54	<code>}</code>
55	
56	<code>public String getContxt() {</code>
57	<code> return contxt;</code>
58	<code>}</code>
59	
60	<code>public void setContxt(String contxt) {</code>
61	<code> this.contxt = contxt;</code>
62	<code>}</code>
63	
64	<code>public String getTail() {</code>
65	<code> return tail;</code>
66	<code>}</code>
67	
68	<code>public void setTail(String tail) {</code>
69	<code> this.tail = tail;</code>
70	<code>}</code>
71	<code>}</code>

行数	Client.java
1	<code>import java.util.Random;</code>
2	
3	<code>public class Client {</code>
4	<code> // 发送账单的数量，这个值是从数据库中获得</code>
5	<code> private static int MAX_COUNT = 6;</code>
6	
7	<code> public static void main(String[] args) {</code>
8	<code> // 模拟发送邮件</code>
9	<code> int i = 0;</code>
10	<code> // 把模板定义出来，这个是从数据中获得</code>
11	<code> Mail mail = new Mail(new AdvTemplate());</code>
12	<code> mail.setTail("XX 银行版权所有");</code>
13	<code> while (i < MAX_COUNT) {</code>
14	<code> // 以下是每封邮件不同的地方</code>
15	<code> Mail cloneMail = mail.clone();</code>
16	<code> cloneMail.setAppellation(getRandString(5) + " 先生（女士）");</code>

```

17         cloneMail.setReceiver(getRandString(5) + "@" + getRandString(8)
18                                + ".com");
19         // 然后发送邮件
20         sendMail(cloneMail);
21         i++;
22     }
23 }
24
25 // 发送邮件
26 public static void sendMail(Mail mail) {
27     System.out.println("标题: " + mail.getSubject() + "\t 收件人: "
28                        + mail.getReceiver() + "\t...发送成功!");
29 }
30
31 // 获得指定长度的随机字符串
32 public static String getRandString(int maxLength) {
33     String source =
34 "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ";
35     StringBuffer sb = new StringBuffer();
36     Random rand = new Random();
37     for (int i = 0; i < maxLength; i++) {
38         sb.append(source.charAt(rand.nextInt(source.length())));
39     }
40     return sb.toString();
41 }
42 }

```

3. 最终结果，如图2所示。

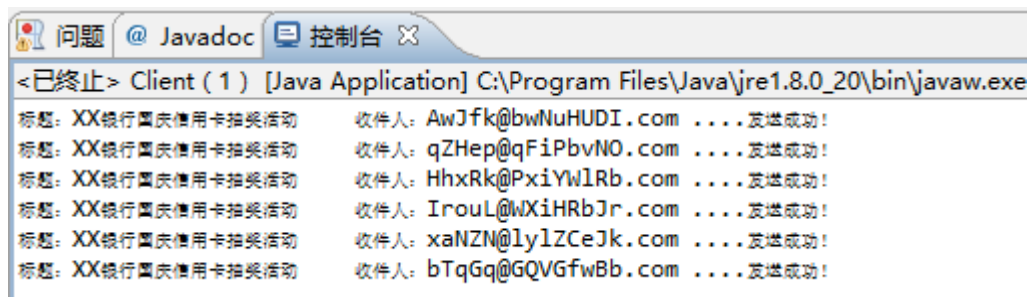


图2. 本原型模式实例的最终运行结果