

配合《Java 从入门到精通》第5版使用

课后实战训练手册

挑战你的编程技能

电子版

第 2 章 熟悉 Eclipse 开发工具

1. 输出长春地铁 1 号线运行图

图 1 为长春市地铁 1 号线的运行路线，请编写一个程序，输出该运行路线图，实现效果如图 2 所示。



图 1 参考图片

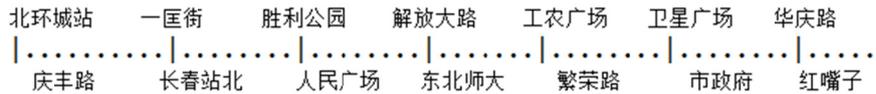


图 2 实现效果图

2. 输出秘密电文

假设为了防止敌方获取我方机密，需要以数字代替我方文件中的字母，即用字母的 Unicode 码值替换字母。编写一个程序，输入一个字母后，输出该字母对应的 Unicode 码值，实现效果如图 3 所示。

```
输文件中的字符：Z  
Z替换为：90
```

图 3 实现效果图

3. 输出轨道交通充值信息

编写一个程序，输出长春轨道交通充值信息，实现效果如图 4 所示。

7. 界上最好的六个医生

编写程序换行输出世界上最好的六个医生：

- 1.阳光
- 2.休息
- 3.锻炼
- 4.饮食
- 5.自信
- 6.朋友

实现效果如图 8 所示。



```
~~~~~  
*1.阳光  
*2.休息  
*3.锻炼  
*4.饮食  
*5.自信  
*6.朋友  
_____
```

图 8 实现效果图

8. 输出对联

请用输出语句输出对联：横批：福星高照，上联：万事如意步步高。下联：一帆风顺年年好。实现效果如图 9 所示。



```
福星高照  
万  
事  
如  
意  
步  
步  
高  
一  
帆  
风  
顺  
年  
年  
好
```

图 9 实现效果图

第 3 章 Java 语言基础

1. 计算牛奶中蛋白质的总量

已知每盒牛奶含有蛋白质 6.4g。编写一个程序，帮助用户根据购买牛奶的盒数计算蛋白质的含量。并输出如图 1 所示的运行效果。

```
请输入购买牛奶的盒数：  
24  
24袋牛奶含有的蛋白质：153.6
```

图 1 实现效果图

2. 输出肯德基一天售出汉堡包的数量及金额

肯德基是人们非常喜欢去的一个场所，因为在那里环境干净，食物快捷……情人节这一天，肯德基某连锁店光是汉堡就销售了 5532 个，假设每个汉堡的金额为 15.5 元，那么这些汉堡一共卖了多少钱呢？请编写一个程序，帮助店员计算每天销售汉堡包的数量及金额，输出效果如图 2 所示的运行效果。

```
请输入售出汉堡的数量：  
5532  
一天总出售的汉堡数量为：5532个  
全天售出的总金额为：85746.0元
```

图 2 实现效果图

3. 计算德邦物流车承载数

已知德邦物流车的车厢长 4.2 米，宽 1.9 米，高 1.9 米，快递的箱子长是 0.5 米，宽是 0.5 米，高是 0.3 米，那么这个物流车能装多少个这样规格的箱子。计算的公式是：箱子总数（取整数）=（物流车厢宽/快递箱宽）*（物流车厢长/快递箱长）*（物流车厢高/快递箱高）、

4. 计算小组成员平均分

某班成立学习小组，每小组有 5 个成员。最近考试成绩出来了，如下表所示，要求编写一个程序，要求可以分别输入 5 位成员的分数，然后将小组平均分数输出出来。

| 姓名 | 分数 |
|----|----|
|----|----|

| | |
|-----|----|
| 郭帆 | 95 |
| 郭京 | 88 |
| 冯小刚 | 86 |
| 王晶 | 90 |
| 黄渤 | 92 |

5. 将高铁速度 km/h 转换成 m/s

世界上第一条正式的高速铁路系统是 1964 年建成通车的日本新干线,设计速度 200km/h,所以高速铁路的初期速度标准就是 200km/h。后来随着技术进步,高铁的速度越来越快。目前我国运行中的高铁速度已经达到了 350 公里/小时。请编写一个程序,将用户输入的高铁速度 km/h 转换成 m/s,输出结果如图 3 所示。(保留整数部分)

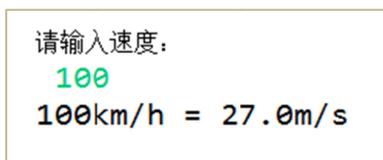


图 3 实现效果图

6. 计算身体质量指数 (BMI)

BMI 指数,又称身体质量指数,英文为 Body Mass Index,是简称 BMI 目前国际上常用的衡量人体胖瘦程度以及是否健康的一个标准,给出了身体胖瘦的指标,如图 4 所示。BMI 指数的计算公式为 $BMI = \text{体重(kg)} / \text{身高(m)}^2$,编写一个程序,用户输入体重和身高,算出用户的 BMI 指数,如图 5 所示。

| BMI (kg/m ²) | 肥胖参考 |
|--------------------------|---------|
| 低於 16 | 第二級營養不良 |
| 16-17.9 | 第一級營養不良 |
| 18-19.9 | 瘦 |
| 20-25 | 正常 |
| 25.1-26.9 | 過重 |
| 27-29.9 | 第一級肥胖 |
| 30-40 | 第二級肥胖 |
| 超過 40 | 第三級肥胖 |

图 4 参考图片



图 5 实现效果图

7. 京东商城支付成功界面

图 6 所示为京东商城支付成功界面。编写一个程序,首先提示用户输入支付金额(输入 80-200 之间的数字),然后输出包含刚输入金额支付成功页面,支付成功页面实现效果如图 7 所示。



图 6 参考图片

请输入支付金额: 89

支付成功
京东商城
89.0元

优惠金额 10.00元
支付方式 工商银行储蓄卡(5009)
交易时间 2018-11-27 00:34:02
订单编号 893412929

图 7 实现效果图

8. 地铁购票金额计算

图 8 所示为某城市地铁 2 号线自动售票机售票的界面，到某站票价为 2 元，用户输入购买票数后，可以看到应付金额和已付金额。编写一个程序，模拟实现地铁购票金额的计算，输入效果如图 9 所示，输入票数后的程序输出结果如图 10 所示。



图 8 参考图片

购票信息

目的车站: 人民广场
票价: 2.00
购票数量: |

图 9 输入购票数量

购票信息

目的车站: 人民广场
票价: 2.00
购票数量: 3
应付金额: 6
已付金额: 10
找零: 4

图 10 计算应付金额

第 4 章 流程控制

1. 年会大抽奖

某公司年会抽奖规则如下：

- ① 1”代表“一等奖”，奖品是“42 寸彩电”；
- ② 2”代表“二等奖”，奖品是“光波炉”；
- ③ 3”代表“三等奖”，奖品是“加湿器”；
- ④ 4”代表“安慰奖”，奖品是“16G-U 盘”。

根据控制台输入的奖号，输出与该奖号对应的奖品，实现效果如图 1 所示。

```
请输入您的奖号：1
恭喜获得一等奖：42寸彩电
```

图 1 实现效果图

2. 查询高考录取分数线

使用多分支语句实现查询高考录取分数线的功能，其中，民办本科：350 分；艺术类本科：290 分；体育类本科：280 分；二本：445 分，一本：555 分。实现效果如图 2 所示。

```
请输入要查询的录取分数线(比如民办本科、艺术类本科、体育类本科、二本、一本)：
二本
二本录取分数线：445分
```

图 2 实现效果图

3. 查看星座

米米想编写一个程序，实现用户输入一个数字 0-9 或”a”，输出一个对应的星座，对应关系分别为：a, b, 1, 2, 3, 4, 5, 6, 7, 8, 9 对应的白羊座，金牛座，双子座，巨蟹座，狮子座，处女座，天平座，射手座，魔蝎座，水瓶座，双鱼座。实现效果如图 3 所示。

```
请输入要查询的星座编号：
7
您查询的是：魔蝎座
```

图 3 实现效果图

4. 百度一下

请编写一个程序模拟百度一下搜索功能,要求输出公司的名称输出公司简介,公司如下,明日科技,京东集团,阿里巴巴。明日科技介绍:主要出版专业的互联网编程语言书籍,针对零基础,本着快速上手,易学的原则。出版的书籍有 python, c 语言, java。阿里巴巴介绍:是国内最大的互联网购物平台之一,旗下有支付宝,淘宝等业务。京东集团介绍:主要经营网上购物商城,几乎包含了所有人们需要的东西,包括书籍,日常生活用品,交通工具,电子产品系列等。实现效果如图 4 所示。

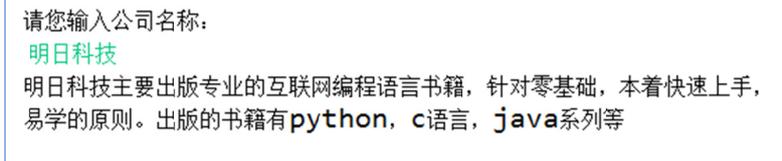


图 4 实现效果图

5. 模拟谷歌浏览器搜索明日学院

模拟如图 5 所示 Google 浏览器搜索界面,并在搜索“明日学院”后,输出明日学院的网址 <http://www.mingrisoft.com/>。实现效果如图 6 所示。



图 5 参考图片

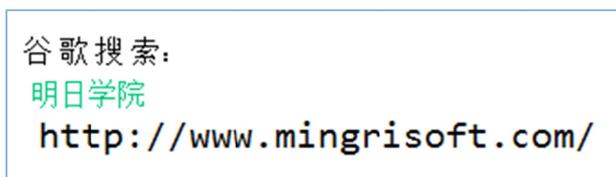


图 6 实现效果图

6. 模拟字母验证码

QQ 在异地登录时,会让我们输入验证号如图 7 所示,以确保我们的 QQ 密码不被他人盗取,编写一个程序,要求用户输入验证码,输入完成后,输出统一为大写字母输出。实现效果如图 8 所示。



图 7 参考图片

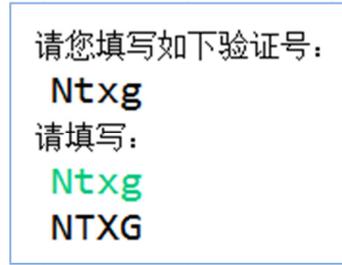


图 8 实现效果图

7. 模拟缴纳电费

编写一个程序，先模拟输出电费账单如图 9 所示，然后要求用户输入缴费金额（大于 574 元），输出当前账户电费余额。实现效果如图 10 所示。



图 9 参考图片

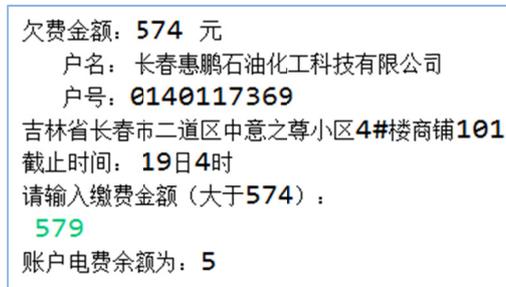


图 10 实现效果图

8. 模拟 58 同城买楼房

现在有一个需求，让我们模拟 58 同城买房子如图 11 所示，输入房子面积筛选出一部分，输入楼层和总价格进行筛选，张女士，想买 90 平米的，6 楼以上，总价格 60 万的。实现效果如图 12 所示。



图 11 参考图片

请输入您要购买的楼房的面积：
90
以上购房面积符合你的要求
请输入您的楼层数：
8
以上的楼层数符合您的要求
请输入您能承担总价格：
60
以上楼房总价格再能的承受范围内

图 12 实现效果图

第 5 章 字符串

1. 明日科技年会找座位程序

过年时，公司会组织年会，有的人会忘记自己是哪个座位的，现在有一个需求，要求编写一个程序输入员工的姓名就可以找到自己的桌号，假设公司有张龙，赵虎，王潮。实现效果如图 1 所示。

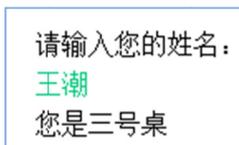


图 1 实现效果图

2. 屏蔽手机号的中间数字

屏蔽手机号中间的四位数字，例如“137****5569”，实现效果如图 2 所示。

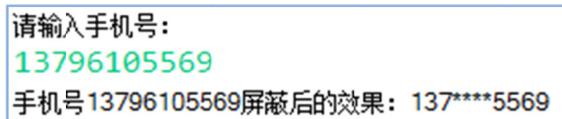


图 2 实现效果图

3. 数字位置查询功能

下面是一张火车票如图 3 所示，请你编写一个程序计算出火车出发日期中第一次出现 6 的位置索引是多少。实现效果如图 4 所示。



图 3 参考图片

6第一次出现的位置为：6

图 4 实现效果图

4. 设置百度地图常用地点

在使用百度地图时，会弹出设置常用地点的对话框，如图 5 所示。请定义家庭住址和单位地址的变量，保存输入的家庭地址和单位地址，输入和输出效果如图 6 所示。

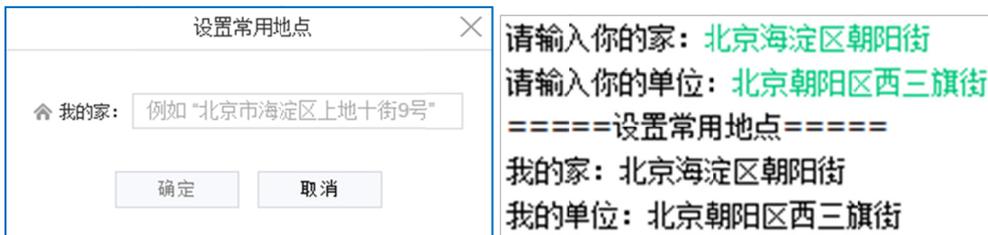


图 5 参考图片

```
请输入你的家: 北京海淀区朝阳街
请输入你的单位: 北京朝阳区西三旗街
=====设置常用地点=====
我的家: 北京海淀区朝阳街
我的单位: 北京朝阳区西三旗街
```

图 6 实现效果图

5. 实时更新导航菜单

编写一个程序，根据用户输入的数据，实时更新导航菜单的名称，原导航菜单如图 7 所示。首先要求用户输入第一个要替换的导航菜单名称，如把第一个导航菜单“天猫”替换成“明日”。实现效果如图 8 所示。



图 7 参考图片

```
天猫 聚划算 天猫超市 | 淘抢购 司法拍卖 淘宝心选 兴农扶贫 | 飞猪旅行
请您选择要被替换的导航菜单:
天猫
请您输入新的导航菜单:
明日
被替换后的导航菜单:
明日 聚划算 明日超市 | 淘抢购 司法拍卖 淘宝心选 兴农扶贫 | 飞猪旅行
```

图 8 实现效果图

6. 模拟商品入库功能

商品入库管理模块是进销存类软件必备的功能，如图 9 所示图片是一个简单的商品入库功能模块。编写一个程序，模拟实现简单的商品入库。首先输出入库界面（不带商品信息等内容），然后要求用户分别输入商品编号、商品名称、商品规格、商品价格和入库数量，输入完成后，输出带数据的商品入库模块。实现效果如图 10 和 11 所示。



图 9 参考图片



图 10 实现效果图



图 11 实现效果图

7. 模拟输出中国联通流量提醒

定义两个浮点型变量，分别表示已用流量（3.59）和剩余流量（3.40），定义一个字符串型的变量，用来表示网址（<http://u.10010.cn/tAE3v>），编写一个程序，输出中国联通流量提醒。实现效果如图 12 所示。

中国联通流量提示：
截止10月21号24时：
您的共享通用流量已用3.59GB剩余流量为3.4GB
如需查询流量详情，请您登陆：<http://u.10010.cn/tAE3v>进行查询

图 12 实现效果图

8. 保存搜索热词

网上购物时，在搜索栏，如图 13 所示，大家输入较多的词会被作为搜索热词显示在搜索栏下面，以备用户快速搜索。编写一个程序，模拟搜索热词的功能。首先提示用户输入搜索词，如输入 "Java"。"java" 会自动添加到搜索栏上面。效果如图 14 所示。



图 13 参考图片

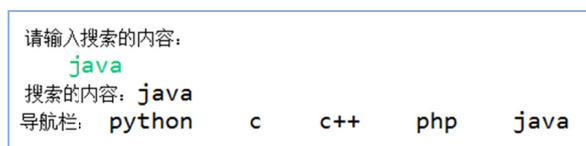


图 14 实现效果图

第 6 章 数组

1. 输出九宫格

一个 3*3 的网格，将从 1 到 9 的数字放入方格，达到能够使得每行每列以及每个对角线的值相加都相同（提示：矩阵中心的元素为 5），实现效果如图 1 所示。

| |
|-------|
| 九宫格： |
| 4 9 2 |
| 3 5 7 |
| 8 1 6 |

图 1 输出九宫格

2. 交换二维数组

交换二维数组 `int[][] array = {{ 91,25,8 }, { 56,14,2 }, { 47,3,67 }}` 的行、列数据，实现效果如图 2 所示。

| | | |
|---------------|----|----|
| ——原始数组—— | | |
| 91 | 25 | 8 |
| 56 | 14 | 2 |
| 47 | 3 | 67 |
| ——调换位置之后的数组—— | | |
| 91 | 56 | 47 |
| 25 | 14 | 3 |
| 8 | 2 | 67 |

图 2 交换二维数组

3. 杨辉三角算法

使用二维数组实现杨辉三角算法，运行结果如图 3 所示。

| | | | | | | | | | | |
|---|---|----|----|-----|-----|----|----|---|---|--|
| 1 | | | | | | | | | | |
| 1 | 1 | | | | | | | | | |
| 1 | 2 | 1 | | | | | | | | |
| 1 | 3 | 3 | 1 | | | | | | | |
| 1 | 4 | 6 | 4 | 1 | | | | | | |
| 1 | 5 | 10 | 10 | 5 | 1 | | | | | |
| 1 | 6 | 15 | 20 | 15 | 6 | 1 | | | | |
| 1 | 7 | 21 | 35 | 35 | 21 | 7 | 1 | | | |
| 1 | 8 | 28 | 56 | 70 | 56 | 28 | 8 | 1 | | |
| 1 | 9 | 36 | 84 | 126 | 126 | 84 | 36 | 9 | 1 | |

图 3 杨辉三角算法

4. 推箱子小游戏

编写一个简易的推箱子游戏，使用 10*8 的二维字符数据表示游戏画面，H 表示墙壁；& 表示玩家角色；o 表示箱子，* 表示目的地。玩家可以通过输入 a、d、w、s 字符控制角色移动，当箱子推到目的地时显示游戏结束，运行结果如图 4 所示。

```

-----
H H H H H H H H H
H & H H H H
H o H H H
H H H H H H
H H H H H
H H H H H
H H H H H
H H H H H H H
-----
A左移，D右移，W上移，S下移，请输入你的指令：
a

```

图 4 推箱子游戏

5. 简易五子棋游戏

编写一个简易五子棋，棋盘在控制台中绘制，棋盘每一个点都有对应的坐标，下棋者输入对应坐标落棋子，运行效果如图 5 所示。两位数字表示的坐标中，第一个数字表示横坐标，第二个数字表示纵坐标，运行结果如图 5 所示。

```

-----
 0 1 2 3 4 5 6 7 8 9
0 - - - - - - - - -
1 - - - - - - - - -
2 - - - - - - - - -
3 - - - - o - - - -
4 - - - * * - - - -
5 - - - o * - - - -
6 - - - - o - - - -
7 - - - - - - - - -
8 - - - - - - - - -
9 - - - - - - - - -
-----
请*输入棋子坐标：
46

```

图 5 简易五子棋游戏

6. 统计成绩

统计学生成绩，输入学生的学号及语文、数学、英语成绩，输出学生各科成绩信息、平均成绩和总成绩，运行结果如图 6 所示。



图 6 统计成绩

7. 客车售票

编写一个程序，一辆大巴有 9 排 4 列的座位，模拟这辆客车的售票过程（1 代表“有票”，0 代表“无票”），运行结果如图 7 所示。



图 7 客车售票

8. 查询成绩

现有学号为 1~8 的 8 名学生和 10 道题目（标准答案为 { "B", "A", "D", "C", "C", "B", "C", "A", "D", "B" }），将学生的答案存储在一个二维数组中，通过学号找到并输出该学生的答案以及回答正确的题目总数，运行结果如图 8 所示。

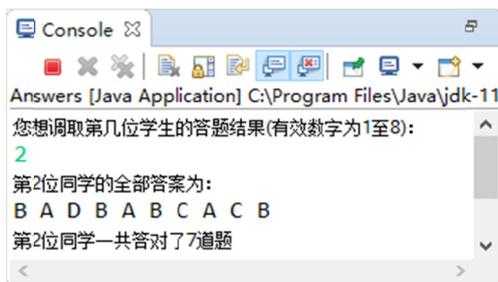
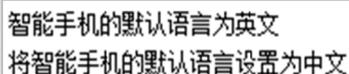


图 8 查询成绩

第 7 章 类和对象

1. 修改手机默认语言

智能手机的默认语言为英文，但制造手机时可以将默认语言设置为中文。编写手机类，无参构造方法使用默认语言设计，利用有参构造方法修改手机的默认语言，实现效果如图 1 所示。

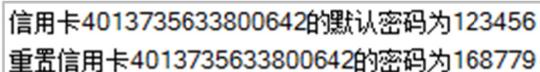


```
智能手机的默认语言为英文
将智能手机的默认语言设置为中文
```

图 1 实现效果图

2. 设置信用卡密码

创建信用卡类，有两个成员变量分别是卡号和密码，如果用户开户时没有设置初始密码，则使用“123456”作为默认密码。设计两个不同的构造方法，分别用于用户设置密码和用户未设置密码两种构造场景，实现效果如图 2 所示。

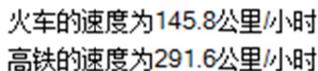


```
信用卡4013735633800642的默认密码为123456
重置信用卡4013735633800642的密码为168779
```

图 2 实现效果图

3. 飞速的高铁

设计火车类和高铁类，高铁类继承火车类，不管火车类的行进速度是多少，高铁的行进速度永远是火车的二倍，实现效果如图 3 所示。



```
火车的速度为145.8公里/小时
高铁的速度为291.6公里/小时
```

图 3 实现效果图

4. 输出圆形和矩形的面积

创建 Shape（图形）类，该类中有一个计算面积的方法。圆形和矩形都继承自图形类，输出圆形和矩形的面积，实现效果如图 4 所示。

```
圆形面积: 7.0685834705770345  
矩形面积: 11.0
```

图 4 实现效果图

5. 灿烂的烟花

创建抽象类烟花（Fireworks）类，该类中有一个抽象的爆炸方法 `boom()`，使用匿名内部类实现点燃红色烟花并爆炸的效果。这样的需求该如何实现？注意：每个爆竹只能调用一次 `boom()` 方法，实现效果如图 5 所示。

```
红色的烟花照亮天空
```

图 5 实现效果图

6. 计算机械中和石英手表的时间

设计钟表类，钟表有一个静态的获取时间方法，此方法与钟表的结构、样式、价格无关，运行结果如图 6 所示。

```
机械钟的价格为189.99元RMB  
当前时间: 10点10分  
石英手表的价格为69.0元RMB  
当前时间: 10点10分
```

图 6 实现效果图

7. 设计自动加载数据库的驱动

设计数据库连接类，在首次创建类对象时，会自动加载数据库的驱动程序，之后创建的所有类对象都不需要重复加载驱动了，实现效果如图 7 所示。

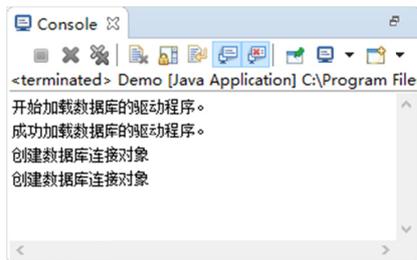


图 7 实现效果图

8. 定义人类的介绍方式

设计人类，定义年龄属性，重写 `toString()` 方法，在方法中判断此人类对象是否大于或

等于 18 岁，如果大于或等于 18 岁，则输出“我 XX 岁，我是成年人。”；否则输出“我 XX 岁，我是未成年人。”，实现效果如图 8 所示。

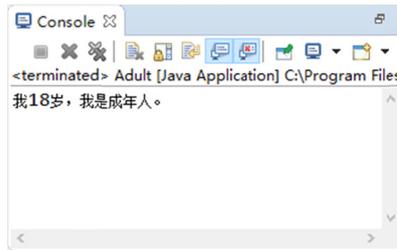


图 8 实现效果图

第 8 章 包装类

1. 100 元纸钞

张三、李四去饭店吃饭，饭后结账时，张三、李四各拿出 100 元人民币，几经推搡，最后收款员接过了李四的 100 元人民币。判断张三和李四的 100 元人民币是不是同一张钞票？李四和收款员手中的 100 元 RMB 是不是同一张钞票？张三和收款员手中的 100 元人民币是不是同一张钞票？，运行结果如下图所示。

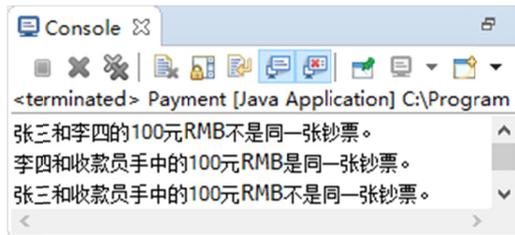


图 1 实现效果图

2. 解析条形码

使用 Integer 类的常用方法，指出条形码“6936983800013”中的“商品的国家代码”、“商品的生产厂商代码”、“商品的厂内商品代码”和“校验码”，运行结果如下图所示。

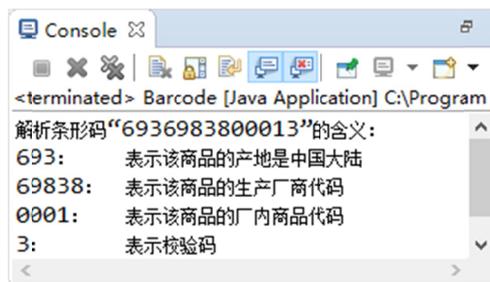


图 2 实现效果图

3. 将字符串转换成整数

如果用户输入的是一串数字，而程序又需要使用这些数字进行运算，则可以将字符串转换成整型或浮点型。本实例利用用户输入的整数来计算其平方数，效果图如下图所示。

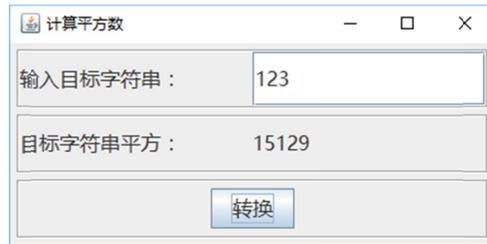


图 3 实现效果图

4. 整数进制转换器

由于计算机的特殊结构，其内部使用二进制数据。为了节约空间，又定义了八进制和十六进制格式来表示二进制数据。一个八进制数可以表示 3 位二进制数，一个十六进制数可以表示 4 位二进制数。而对于普通人而言，使用十进制更加容易阅读。本实例将实现一个简单的进制转换器，效果图如下图所示。



图 4 实现效果图

5. ASCII 编码查看器

ASCII 是 American Standard Code Information Interchange 的缩写，是基于拉丁字母的一套电脑编码系统，主要用于显示英语字符，是目前世界上最通用的单字节编码。基本的 ASCII 编码包括了 128 个字符。本实例将编写一个 ASCII 编码查看器，可以将字符转换成数字，也可以反向转换，效果图如下图所示。

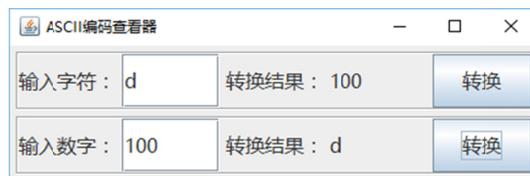


图 5 实现效果图

6. Double 类型的比较

对于 double 类型（基本类型）的数据，可以直接使用普通的运算符来进行比较，如“=”。然而，对于 Double 类型（引用类型）却不行。引用类型如果使用“=”来进行比较则判断内存地址是否相同，答案通常是是否定的。本实例演示如何使用 Double 类中定义的方法来进行对象间比较，运行结果如下图所示。

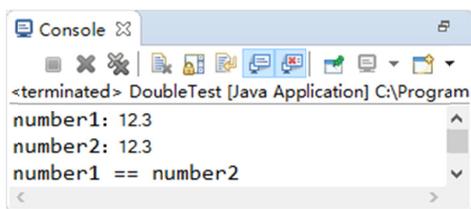


图 6 实现效果图

第 9 章 数字处理类

1. 飞机的加速度

飞机在跑道上加速滑行起飞，滑行时间为 15.5 秒，滑行距离为 1050 米。在起飞的整个过程中，飞机的加速度保持不变，控制台输出这个加速度。（因为飞机的初始速度为 0，所以飞机的加速度等于滑行距离的 2 倍除以滑行时间的平方）。

2. 卖苹果

一苹果商卖苹果有个“不找零钱（四舍五入）”习惯，苹果售价为每 500 克 2.49 元，苹果商输入顾客购买苹果的数量后，输出这些苹果的未经四舍五入的总价格与顾客的应付金额。



图 1 实现效果图

3. 更近的地点

把 A 地设为坐标原点，B 地的坐标为 (3.8, 4.2)，C 地的坐标为 (3.2, 4.5)，在不计算出结果的前提下，使用 `Math.min()` 方法输出 B、C 哪一个地点距 A 地更近。

4. 抢红包

使用 `Random` 类模拟微信的抢红包功能，根据用户输入的红包额度和个数随机生成每个红包的金额，效果图如下图所示。

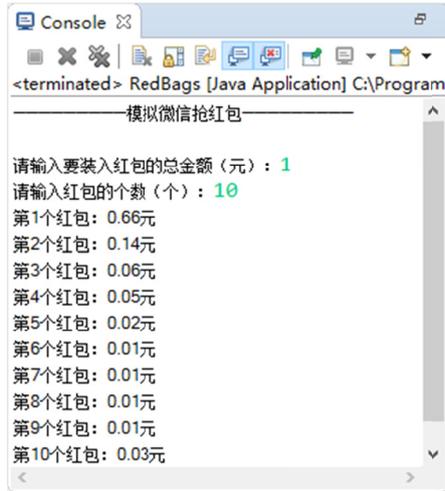


图 2 实现效果图

5. 七星彩

七星彩是指从 0000000-9999999 中选择任意 7 位自然号码进行投注，每注 2 元 RMB。使用 Random 类模拟根据用户控制台输入的“购买的彩票数”，随机为用户生成相应数目的七星彩号码，并与当期的七星彩结果比较，最后输出中奖情况，效果图如下图所示。

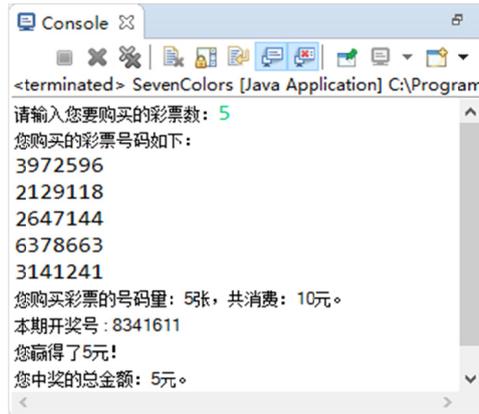


图 3 实现效果图

6. 精确的小数运算

浮点运算的典型实例是货币运算，在商品金额计算中，经常会涉及小数运算，如某个商品的价格是 1.10 元，而顾客现有金额是 2 元整。在计算机中所有数字都是使用二进制进行存储的，而二进制无法精确地表示所有的小数，所以使用基本数据类型进行小数运算会有一些误差，本实例将通过 BigDecimal 类实现精确的小数运算，效果图如下图所示。

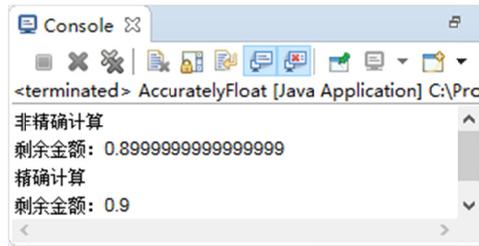


图 4 实现效果图

第 10 章 接口、继承与多态

1. 编写简易计算器

将加减乘除四个算数计算封装成四个方法（只计算两个数字），当用户输入对应计算符号时调用对应方法并返回计算结果，实现效果如图 1 所示。

```
请输入两个数字，用空格隔开：  
12 3  
请输入运算符{+-* /}：  
-  
计算结果为：9.0  
请输入两个数字，用空格隔开：
```

图 1 实现效果图

2. 多功能参数

编写一个方法，如果输入一个参数，则以该参数作为圆的半径返回圆的面积；如果输入两个参数，则以这两个参数作为矩形的宽和高返回矩形的面积；如果不输入任何参数，则返回 π 的值，实现效果如图 2 所示。

```
3.141592653589793  
50.26548245743669  
12.0
```

图 2 实现效果图

3. 编写登录方法

编写一个登录方法，参数为用户的账号和密码，把所有数据校验的逻辑都封装到方法中，返回校验结果，实现效果如图 3 所示。

```
请输入用户名：  
张三  
请输入密码：  
123456  
-----  
登录成功
```

图 3 实现效果图

4. 看看数字的升序效果

编写数字升序排序方法，排序完整之后会打印排序结果。方法参数为不定长参数，方法支持三种类型的参数，分别为：{6, 9, 1, 3, 7, 2}、{'6', '9', '1', '3', '7', '2'}、{"6", "9", "1", "3", "7", "2"}，实现效果如图 4 所示。

```
1 2 3 6 7 9
1 2 3 6 7 9
1 2 3 6 7 9
```

图 4 实现效果图

5. 求出斐波那契额数列

使用递归方法求出斐波那契额数列中任意一位的数字是多少。斐波那契额数列的特征为数列中任何一个数（不包含第一位和第二位数字）等于该数字前两位的数字之和，前几位数字为：1、1、2、3、5、8、13、21、34、……，实现结果如图 5 所示。

图 5 实现效果图

6. 模拟上课场景

创建老师类和学生类，两个类都实现了问候接口和工作接口，模拟上课的场景，运行效果如图 6 所示。

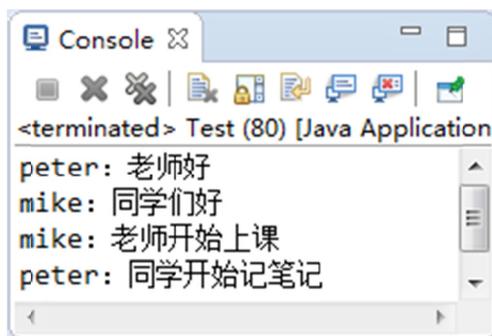
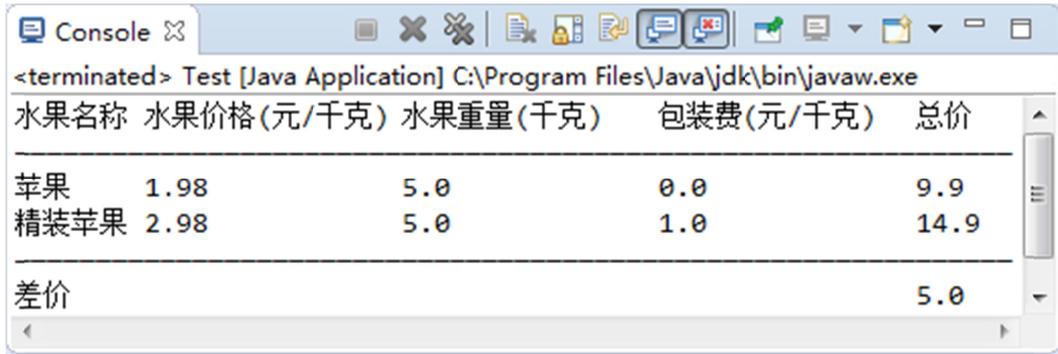


图 6 模拟上课场景的效果图

7. 人工包装的水果与普通水果的价格

使用继承说明经过人工包装的水果与普通水果在价格上的区别。运行效果如图 7 所示。

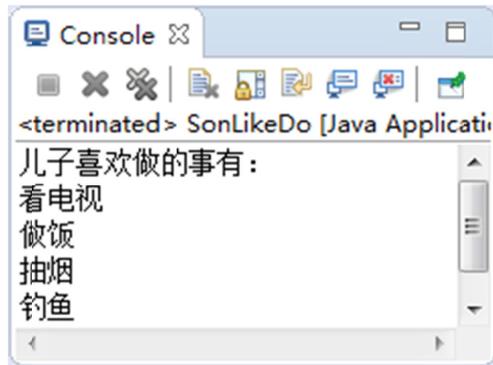


| 水果名称 | 水果价格(元/千克) | 水果重量(千克) | 包装费(元/千克) | 总价 |
|------|------------|----------|-----------|------|
| 苹果 | 1.98 | 5.0 | 0.0 | 9.9 |
| 精装苹果 | 2.98 | 5.0 | 1.0 | 14.9 |
| 差价 | | | | 5.0 |

图 7 使用继承说明人工包装的水果与普通水果在价格上的区别

8. 儿子喜欢做的事

爸爸喜欢做的事有抽烟和钓鱼，妈妈喜欢做的事有看电视和做饭，儿子完全继承了爸爸妈妈的爱好，使用多重继承输出儿子喜欢做的事。运行结果如图 8 所示。



```
<terminated> SonLikeDo [Java Applicati  
儿子喜欢做的事有：  
看电视  
做饭  
抽烟  
钓鱼
```

图 8 使用多重继承输出儿子喜欢做的事

第 11 章 类的高级特性

1. 五星红旗

使用 `final` 类编写一个程序，实现在控制台输出“五星红旗是由红色的旗面和 5 颗黄色的五角星组成的”。

2. 红灯停、绿灯行

编写交通规则类，将遵守交通规则的方法设为 `final` 方法，不管是行人、非机动车辆，还是机动车辆，要遵守的交通规则都是一样的（红灯停、绿灯行）。

3. 滞留人数

动车组每节车厢只有 108 个座位，共有 10 节车厢，现有旅客 1189 人，控制台输出滞留旅客的人数。

4. 为面板设置不同颜色

使用内部类编写一个程序：定义 3 个按钮，用户通过单击不同的按钮，可以为面板设置不同的颜色，效果图如下图所示。

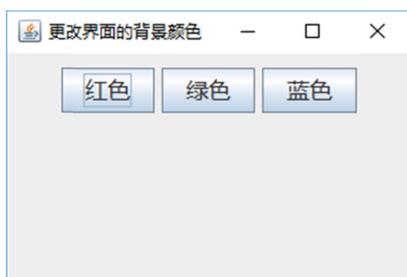


图 1 初始时

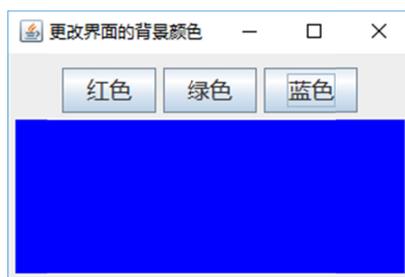


图 2 单击“蓝色”按钮

5. 图片展示器

使用匿名内部类编写一个图片展示器，它包含 6 张图片。通过单击不同的按钮即可展示不同的图片，效果图如下图所示。



图 3 实现效果图

6. 一次遍历求最大和最小值

通常需要通过两步来求出一个数组中元素的最大值和最小值。现使用静态内部类来实现通过一次遍历，求出一个数组中元素的最大值和最小值。

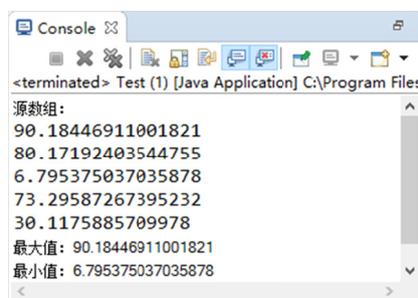


图 4 实现效果图

第 12 章 异常处理

1. 模拟银行取款

银行账号中现有余额 1023.79 元。模拟取款，当在控制台上输入的取款金额不是整数时，会引起数字格式转换异常，实现效果如图 1 所示。

```
请输入取款金额：  
1000  
您账号上的余额：23.79元
```

图 1 实现效果图

2. 模拟老师上课点名

模拟老师上课前的点名过程，并将旷课的学生作为异常抛出：张三、李四、王五（老师在点名册上记下了“王五旷课”），实现效果如图 2 所示。

```
老师：同学们早上好！现在我们开始点名...  
班长：大家保持安静！准备上课...  
老师：张三！  
同学张三：到！  
老师：李四！  
同学李四：Here!  
老师：下次说中文。下一个！  
王五！  
同学王五：....  
老师：王五！  
同学王五：....  
抛出异常：老师在点名册上记下了“王五旷课”!!!  
老师：今天点名到此结束。现在开始上课！
```

图 2 实现效果图

3. 超市特价商品限购

超市经常会对定价较市场价低的产品实施限购：超市里的鲜鸡蛋每 500 克 3.98 元，每人限购 1500 克。现将超过 1500 克的作为异常抛出，而对于满足条件的，计算出应付款，实现效果如图 3 所示。

```
请输入鲜鸡蛋的重重(斤): 10  
异常提示: 这份鲜鸡蛋的重重为10.0斤, 超过3斤了, 超重了!!!
```

图 3 实现效果图

4. 车钥匙哪里去了

有位车主想打开车门，不巧的是，他发现自己没带车钥匙，由此引发了空指针异常（NullPointerException），这种场景该如何设计？实现效果如图 4 所示。

```
车钥匙忘带了! 车暂时是启动不了了.....  
去取车钥匙吧T_T
```

图 4 实现效果图

5. 统计学校人数

编写自定义异常 CountIsNotIntegerException，当统计的学校的人数为不是整数时，则抛出该异常，实现效果如图 5 所示。

```
CountIsNotIntegerException: 人数不能为小数: 456214.2  
at School.setCount(CountIsNotIntegerException.java:27)  
at CountIsNotIntegerException.main(CountIsNotIntegerException.java:11)
```

图 5 实现效果图

6. 西红柿单价不得超过 7 元

当某种商品的价格过高时，国家会对这种商品采取宏观调控，进而使得这种商品的价格趋于稳定。编写一个程序，规定西红柿单价不得超过 7 元，超过 7 元的情况作为异常抛出，运行结果如图 6 所示。

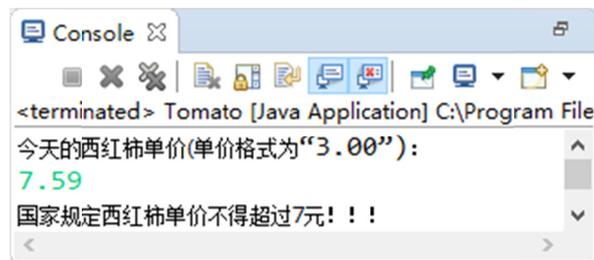


图 6 实现效果图

第 13 章 Swing 程序设计

1. 实现背景图片自动适应窗体的大小

开发人员在开发桌面应用程序时，有时为一个窗体设置了背景图片，但是由于图片的大小与窗体的大小并不一定相同，所以就可能导致图片显示不全，那么如何来避免这种情况的发生呢？本实例将通过自定义面板控件来实现背景图片自动适应窗体的大小的功能，效果图如下图所示。



图 1 实现效果图

2. 颜色选择框

颜色也是系统资源之一，它和文件同样重要。在设置颜色值时，不像文件路径那样可以通过字符串来表示，颜色值大多使用对话框进行选择，总的来说颜色选择对话框就是让用户通过视觉来确定颜色值，而不是通过文本来确定。本实例实现了颜色选择框的应用，效果图如下图所示。

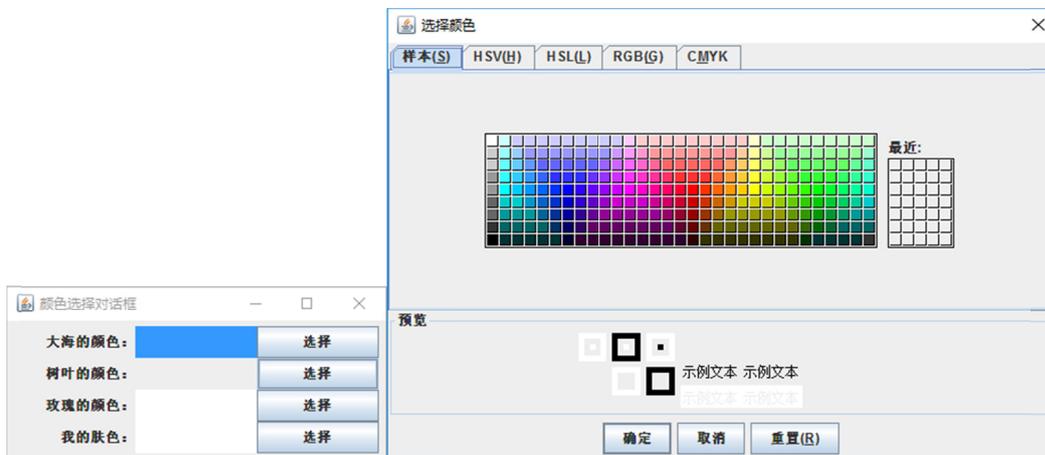


图 2 实现效果图

3. 抖动窗体

编写一个程序，模拟 QQ 的窗体抖动效果，在 Java 语言的窗体中加入抖动效果，效果图如下图所示。



图 3 实现效果图

4. 自定义最大化、最小化和关闭按钮

在制作应用程序时，为了使用户界面更加美观，一般都自己设计窗体的外观，以及窗体的最大化、最小化和关闭按钮。本实例实现设计窗体的外观及最大化、最小化和关闭按钮，再通过鼠标来实现窗体移动效果，效果图如下图所示。



图 4 实现效果图

5. 公司简介

对于一些功能比较复杂的软件，可以在软件启动时弹出一个对话框来显示一些提示信息，如软件的快捷键、软件的使用技巧、软件公司的简介等。本实例使用 `JDialog` 实现了一个公司简介的提示对话框，效果图如下图所示。

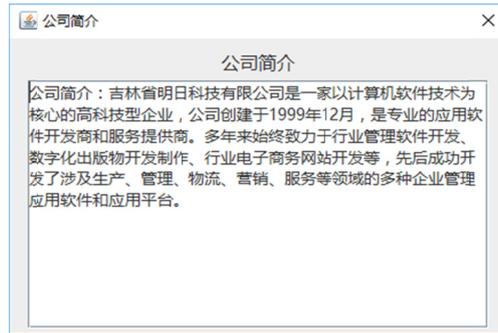


图 5 实现效果图

6. 计算器

编写一个实例，使用网格布局，模拟一个计算器，效果图如下图所示。



图 6 实现效果图

7. 字符统计工具

在使用文本编辑软件，如 Word 时，会在软件界面中提示总共字符等信息，方便用户掌握文档编写的进度。本实例将模拟 Word 的功能并进行增强，可以实时显示光标所在的位置和用户选择的文本所包含的字符数量，效果图如下图所示。

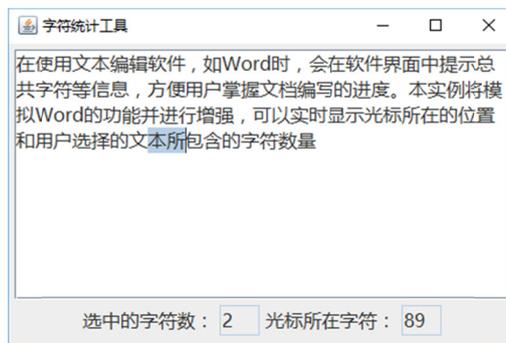


图 7 实现效果图

8. 使用滑块来选择日期

当可以选择的选项很多时，使用单选按钮并不理想，因为需要创建大量的按钮。此时可以考虑使用滑块。滑块可以让用户在一组离散值中进行选择。本实例将使用滑块来选择日期，效果图如下图所示。

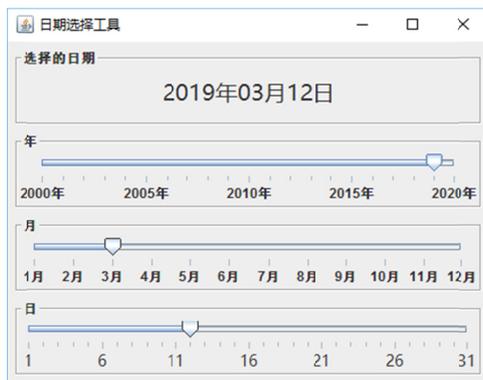


图 8 实现效果图

第 14 章 集合类

1. 管理学生姓名

ArrayList 集合可以看作动态数组。它突破普通数组固定长度的限制，可以随时向数组中添加和移除元素，这将使数组更加灵活，如果要获取普通数组，还可以通过该类的 `toArray()` 方法获得。本实例通过这个 ArrayList 集合类实现向程序动态添加与删除学生姓名的功能，其中所有数据都是保存在 ArrayList 集合的实例对象中，效果图如下图所示。

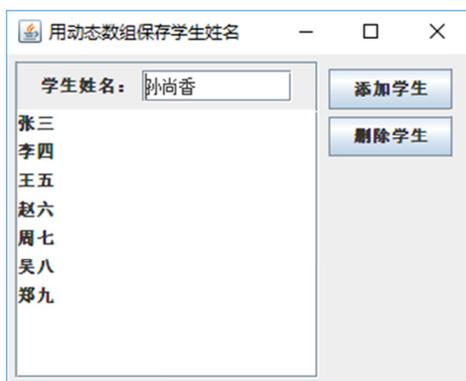


图 1 实现效果图

2. 传递学生信息

集合在程序开发中经常用到，例如，在业务方法中将学生信息、商品信息等存储到集合中，然后作为方法的返回值返回给调用者，以此传递大量的有序数据。本实例将使用 List 集合在方法之间传递学生信息，效果图如下图所示。

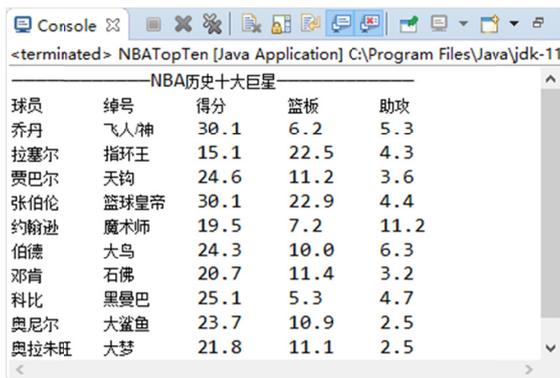


| 姓名 | 性别 | 出生日期 |
|----|----|-----------|
| 李哥 | 男 | 2001-1-1 |
| 小陈 | 女 | 2001-3-7 |
| 小刘 | 男 | 2001-4-5 |
| 小张 | 男 | 2001-5-12 |
| 小董 | 男 | 2002-1-10 |
| 小吕 | 男 | 2002-1-22 |

图 2 实现效果图

3. NBA 历史十大巨星

编写一个程序，使用 List 集合，在控制台输出 NBA 历史十大巨星的姓名、绰号、得分、篮板和助攻等内容，效果图如下图所示。

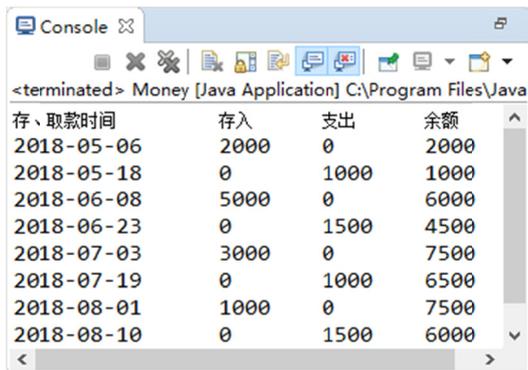


| 球员 | 绰号 | 得分 | 篮板 | 助攻 |
|------|------|------|------|------|
| 乔丹 | 飞人神 | 30.1 | 6.2 | 5.3 |
| 拉塞尔 | 指环王 | 15.1 | 22.5 | 4.3 |
| 贾巴尔 | 天钩 | 24.6 | 11.2 | 3.6 |
| 张伯伦 | 篮球皇帝 | 30.1 | 22.9 | 4.4 |
| 约翰逊 | 魔术师 | 19.5 | 7.2 | 11.2 |
| 伯德 | 大鸟 | 24.3 | 10.0 | 6.3 |
| 邓肯 | 石佛 | 20.7 | 11.4 | 3.2 |
| 科比 | 黑曼巴 | 25.1 | 5.3 | 4.7 |
| 奥尼尔 | 大鲨鱼 | 23.7 | 10.9 | 2.5 |
| 奥拉朱旺 | 大梦 | 21.8 | 11.1 | 2.5 |

图 3 实现效果图

4. 账户存取款明细

编写一个程序，模拟某一账户的存取款明细，效果图如下图所示。



| 存、取款时间 | 存入 | 支出 | 余额 |
|------------|------|------|------|
| 2018-05-06 | 2000 | 0 | 2000 |
| 2018-05-18 | 0 | 1000 | 1000 |
| 2018-06-08 | 5000 | 0 | 6000 |
| 2018-06-23 | 0 | 1500 | 4500 |
| 2018-07-03 | 3000 | 0 | 7500 |
| 2018-07-19 | 0 | 1000 | 6500 |
| 2018-08-01 | 1000 | 0 | 7500 |
| 2018-08-10 | 0 | 1500 | 6000 |

图 4 实现效果图

5. 模拟购物车列表

有位买家看中了这三本书：《Java 从入门到精通（第 3 版）》→明日科技编著→59.8 元；《Java 从入门到精通（实例版）》→明日科技编著→69.8 元；《Java Web 从入门到精通》→明日科技编著→69.8 元。他把这三本书放进了购物车里打算结账：使用封装和 HashSet，把这三本书的信息输出出来，并求出三本书的价格总和，效果图如下图所示。

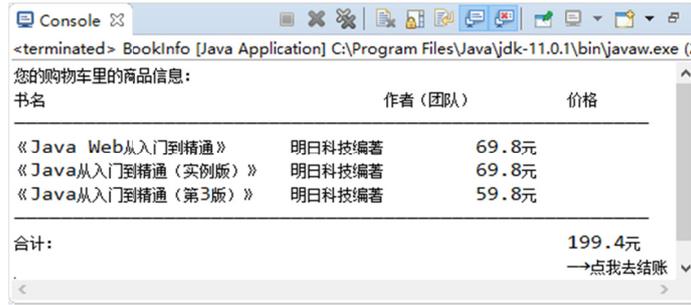


图 5 实现效果图

6. 利用 TreeSet 集合实现不重复的数列

随机数组就是在指定长度的数组中用随机数字为每个元素赋值,这常用于需要不确定数值的环境,如拼图游戏需要随机数组来打乱图片排序。可是同时也存在问题,就是随机数的重复问题,这个问题也常常被忽略,本实例将利用 TreeSet 集合实现不重复的数列,并自动完成元素的排序然后生成数组,效果图如下图所示。

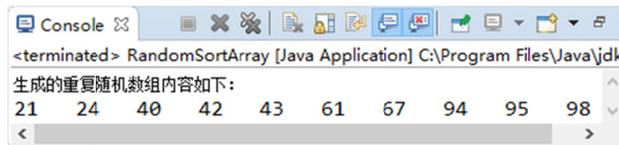


图 6 实现效果图

7. NBA 扣篮大赛评分

模拟 2016 年 NBA 扣篮大赛评分:请五位评委(冰人格文、穆大叔、魔术师约翰逊、大鲨鱼奥尼尔以及麦蒂)打分,控制台输入五个 0-10 内的整数,中间用逗号隔开(例如:10,9,9,8,10),最后计算五位评委给出的分数之和,效果图如下图所示。

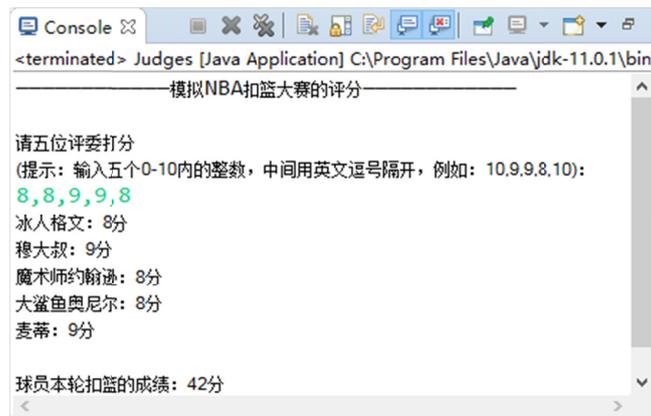


图 7 实现效果图

8. Map 映射集合实现省市级联选择框

Map 集合可以保存键值映射关系，这非常适合本实例所需要的数据结构，所有省份信息可以保存为 Map 集合的键，而每个键可以保存对应的城市信息。本实例就利用这个 Map 集合实现了省市级联选择框，当选择省份信息时，将改变城市下拉列表框对应的内容。实例运行效果如图 8 所示。



图 8 省市级联选择框

第 15 章 I/O (输入/输出)

1. 获取文件属性

在操作系统平台中文件是数据的存储单位，每个文件都有不同的属性。在 Windows 操作系统中可以通过文件的属性对话框查看其对应的属性信息。本实例通过程序编码也实现了该功能，即获取文件大小、创建时间、路径等属性，效果图如下图所示。

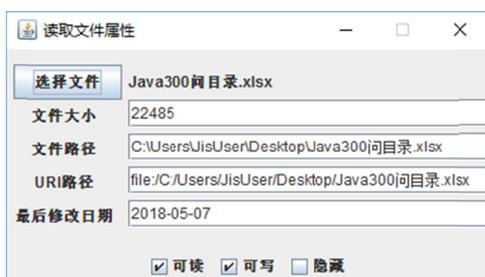


图 1 实现效果图

2. 显示指定类型的文件

文件作为存储数据的单元，会根据数据类型产生很多分类，也就是所谓的文件类型。在对数据文件进行操作时，常常需要根据不同的文件类型来做不同的处理。本实例实现的是读取文件夹指定类型的文件并显示到表格控件中，效果图如下图所示。



图 2 实现效果图

3. 以树结构显示文件路径

Java 的 JTree 树控件用于显示多层次结构的数据，这类似于文件夹的上下层关系，而且各个操作系统也都因为这个相似之处，采用树控件来显示文件夹的层次结构。本实例也结合了文件夹数据与树控件来显示计算机中的文件与文件夹信息，效果图如下图所示。



图 3 实现效果图

4. 支持图片预览的文件选择对话框

为了保证面向对象的封装特性，通常会将域设置成私有的，然后提供对应的 get 和 set 方法。对于非内部类而言，只能使用 get 和 set 方法来操作该域。然而利用反射机制，就可以在运行时修改类的私有域。本实例通过简单的 Student 类来演示反射的这种用法，效果图如下图所示。

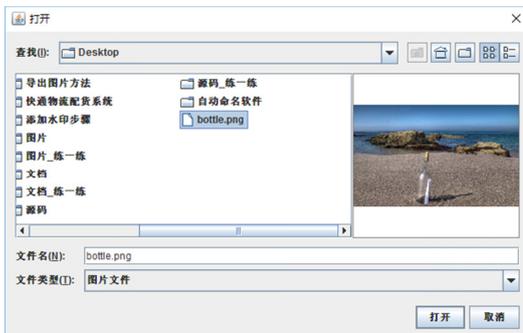


图 4 选择并预览图片

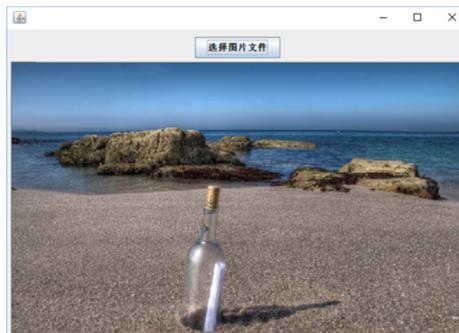


图 5 展示图片

5. 动态列出文件夹下的所有文件

在使用图形界面操作系统时，当打开一个文件夹系统时会自动列出该文件夹下的所有文件及子文件夹。本实例实现了类似的功能：首先让用户选择一个文件夹，程序会动态列出该文件夹下的所有文件；如果该文件是隐藏文件，就在属性栏中显示“隐藏文件”，效果图如下图所示。



图 6 实现效果图

6. 合并 txt 文件

使用窗体实现将多个 txt 文件合并为 1 个 txt 文件，效果图如下图所示。

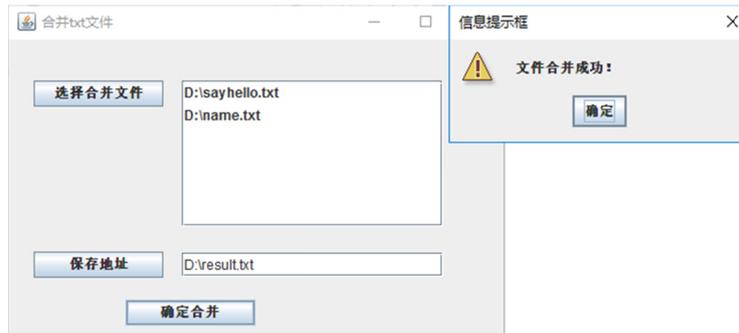


图 7 实现效果图

7. 批量添加文件夹

使用 JfileChooser 实现在指定文件夹下批量添加根据“数字型样式”或“非数字型样式”命名的文件夹，效果图如下图所示。

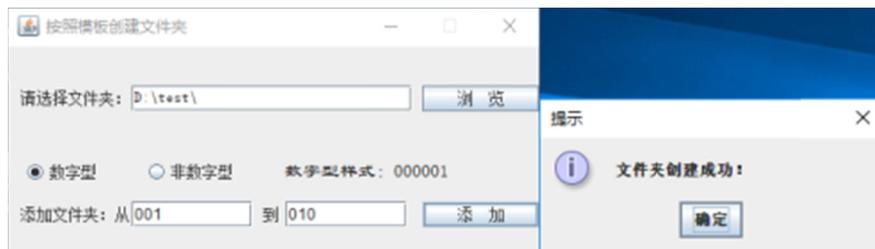


图 8 实现效果图

8. 文件批量重命名

用户可以将一个文件夹内同一类型的文件按照一定的规则批量重命名, 在给出重命名模板后, 程序将根据模板对相应的文件进行重命名, 而且还可以在重命名模板中添加特殊符号, 程序会将这些特殊符号替换成重命名后的文件编号, 效果图如下图所示。

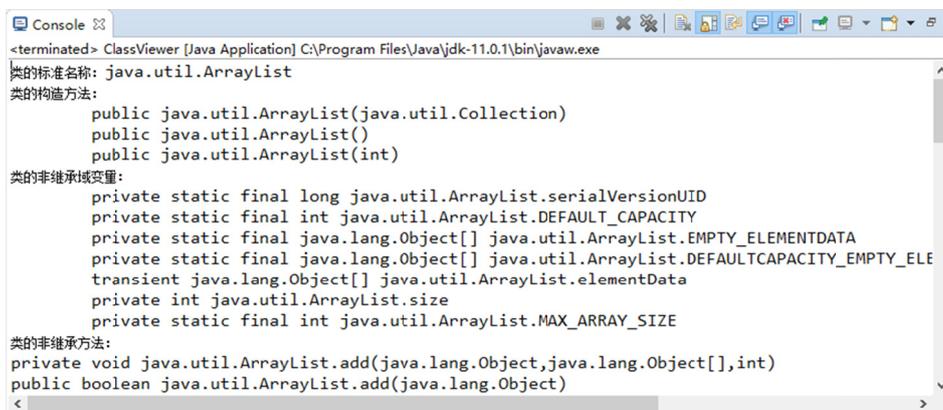


图 9 实现效果图

第 16 章 反射

1. 查看类的成员

在一个类的内部，一般包括域、构造方法、普通方法和内部类等成员。使用发射机制可以在无法查看源代码的情况下查看类的成员。本实例将使用反射机制查看 `ArrayList` 类中定义的域、构造方法和普通方法，效果图如下图所示。

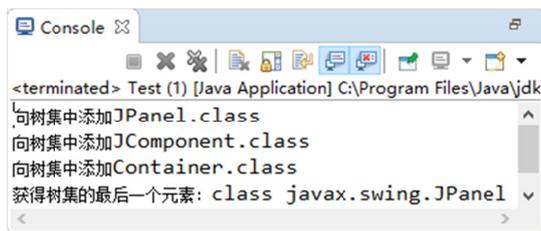


```
<terminated> ClassViewer [Java Application] C:\Program Files\Java\jdk-11.0.1\bin\javaw.exe
类的标准名称: java.util.ArrayList
类的构造方法:
    public java.util.ArrayList(java.util.Collection)
    public java.util.ArrayList()
    public java.util.ArrayList(int)
类的非继承域变量:
    private static final long java.util.ArrayList.serialVersionUID
    private static final int java.util.ArrayList.DEFAULT_CAPACITY
    private static final java.lang.Object[] java.util.ArrayList.EMPTY_ELEMENTDATA
    private static final java.lang.Object[] java.util.ArrayList.DEFAULTCAPACITY_EMPTY_ELEMENTDATA
    transient java.lang.Object[] java.util.ArrayList.elementData
    private int java.util.ArrayList.size
    private static final int java.util.ArrayList.MAX_ARRAY_SIZE
类的非继承方法:
    private void java.util.ArrayList.add(java.lang.Object,java.lang.Object[],int)
    public boolean java.util.ArrayList.add(java.lang.Object)
```

图 1 实现效果图

2. 按继承层次对类排序

Java 提供了 `instanceof` 运算符来比较两个类（或接口）之间是否存在继承关系。但是如果对多个类按照继承关系排序，使用这种方式会非常麻烦。本实例利用反射来对存在继承关系的类进行排序，效果图如下图所示。



```
<terminated> Test (1) [Java Application] C:\Program Files\Java\jdk
向树集中添加JPanel.class
向树集中添加JComponent.class
向树集中添加Container.class
获得树集的最后一个元素: class javax.swing.JPanel
```

图 2 实现效果图

3. 查看内部类信息

Java 中支持在类的内部定义类，这种类称为内部类。内部类有些像 Java 中的方法，可

以使用访问权限限定符修饰，可以使用 `static` 修饰等。本实例将利用 Java 的反射机制来查看内部类的信息，效果图如下图所示。

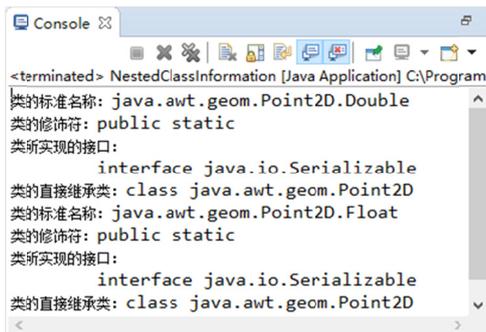


图 3 实现效果图

4. 动态设置类的私有域

为了保证面向对象的封装特性，通常会将域设置成私有的，然后提供对应的 `get` 和 `set` 方法。对于非内部类而言，只能使用 `get` 和 `set` 方法来操作该域。然而利用反射机制，就可以在运行时修改类的私有域。本实例通过简单的 `Student` 类来演示反射的这种用法，效果图如下图所示。

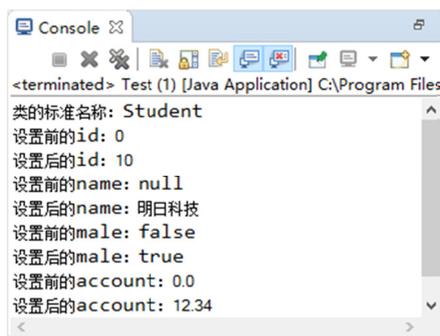


图 4 实现效果图

5. 动态调用类中的方法

Java 中，调用类的方法有两种方式：对于静态方法可以直接使用类名调用，对于非静态方法必须使用类的对象调用。反射机制提供了比较另类的调用方式，即可以根据需要指定要调用的方法，而不必在编程时确定。调用的方法不仅限于 `public` 的，还可以是 `private` 的。本实例将演示如何使用反射机制调用方法，效果图如下图所示。

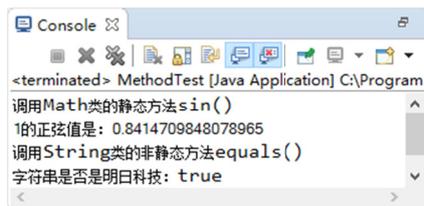


图 5 实现效果图

6. 创建长度可变的数组

Java 中对于数组的支持并不强大。程序员必须时刻注意数组中元素的个数，否则会出现数组下标越界异常。因此才在 API 中定义了 `ArrayList` 帮助开发，但这意味着需要学习新的方法。本实例将使用反射机制实现一个工具方法，每当调用该方法时数组的长度就会增加 5，效果图如下图所示。

```
<terminated> UsefulArray [Java Application] C:\Program Files\Java\jdk-11
整型数组原始长度是: 10
整型数组的内容:
[8, 8, 8, 8, 8, 8, 8, 8, 8, 8]
整型数组扩展后长度是: 15
整型数组的内容:
[8, 8, 8, 8, 8, 8, 8, 8, 8, 0, 0, 0, 0, 0]
```

图 6 实现效果图

7. 利用反射重写 `toString()`方法

为了方便输出对象，`Object` 类提供了 `toString()`方法。但是该方法的默认值是由类名和哈希码组成的，实用性并不强。通常需要重写该方法以提供更多的信息。本实例使用反射输出类的包、类的名字、类的公共构造方法、类的公共域和类的公共方法。在重写不同类的 `toString()`方法时调用该方法就可以避免多次重写 `toString()`方法，效果图如下图所示。

```
<terminated> StringUtils [Java Application] C:\Program Files\Java\jdk-11.0.1\bin\javaw.exe
包名: java.lang 类名: Object
公共构造方法:
public java.lang.Object()
公共域:
公共方法:
public final native void java.lang.Object.wait(long) throws java.lang.InterruptedException
public final void java.lang.Object.wait(long,int) throws java.lang.InterruptedException
public final void java.lang.Object.wait() throws java.lang.InterruptedException
public boolean java.lang.Object.equals(java.lang.Object)
public java.lang.String java.lang.Object.toString()
public native int java.lang.Object.hashCode()
public final native java.lang.Class<?> java.lang.Object.getClass()
public final native void java.lang.Object.notify()
public final native void java.lang.Object.notifyAll()
```

图 7 实现效果图

8. 反射与动态代理

代理是 Java SE 1.3 版新增的特性。使用代理可以在程序运行时创建一个实现指定接口的新的类。通常只有在编译时无法确定需要使用哪个接口时才需要使用代理，这对于应用程序员很少见。对于系统程序员而言，代理可以为工具类提供更加灵活的特性。本实例模拟一个简单的房屋销售场景，效果图如下图所示。

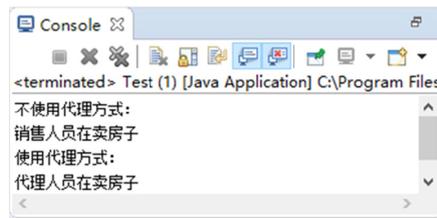


图 8 实现效果图

第 17 章 枚举类型与泛型

1. 根据枚举元素匹配值

在本实例中，将使用枚举类型保存 JDBC 参数，并自定义方法以根据不同的枚举元素获得相应的参数值，效果图如下图所示。

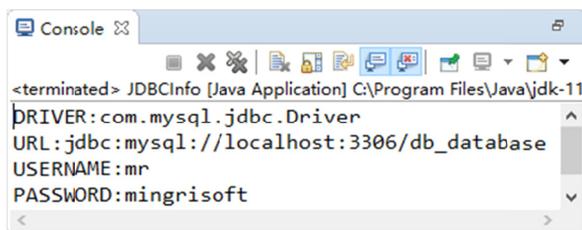


图 1 实现效果图

2. 高效的枚举元素集合

Set 是 Java 集合类的重要组成部分，它用来存储不能重复的对象。枚举类型也要求其枚举元素各不相同。看起来枚举类型和集合是很相似的。然而枚举类型中的元素不能随意地增加、删除，作为集合而言，枚举类型非常不实用。EnumSet 是专门为 enum 实现的集合类，本实例将演示其用法，效果图如下图所示。

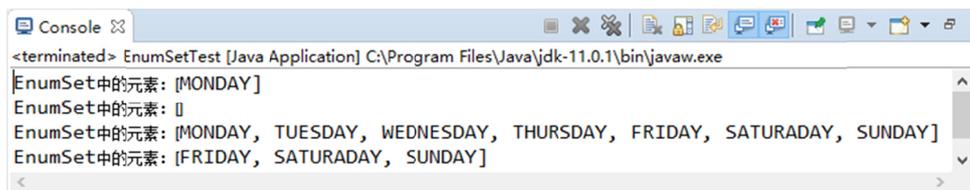


图 2 实现效果图

3. 高效的枚举元素映射

Map 是 Java 集合类的重要组成部分，其用途是利用键值对来保存对象。当需要使用值时，可以根据键获得。这要求 Map 的键必须唯一，而枚举类型的元素都是唯一的，因此可以用来做 Map 的键。EnumMap 类就是 Java 专门为枚举类型提供的 Map 实现类。本实例将演示其用法，效果图如下图所示。

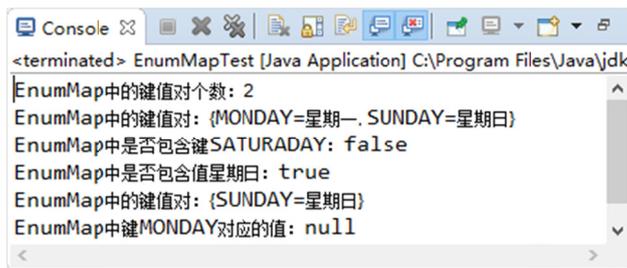


图 3 实现效果图

4. 遍历枚举接口的元素

集合类可以用来管理一组相关的对象。当需要查看、使用集合中的所有对象时可以使用枚举接口对其进行遍历。枚举接口中定义了两个方法，它通常和向量一起使用。本实例将演示其用法，效果图如下图所示。

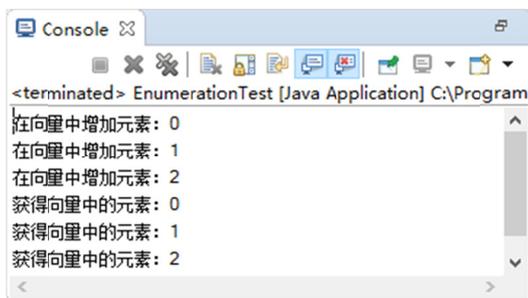


图 4 实现效果图

5. 自定义泛型化数组类

Java 虚拟机中并没有泛型类型的对象，所有有关泛型的信息都被擦除了。这虽然可以避免 C++ 语言的模板代码膨胀问题，但是也引起了其他问题。例如，不能直接创建泛型数组。为了弥补这个不足，本实例将利用反射机制创建一个泛型化数组，效果图如下图所示。



图 5 实现效果图

6. 使用泛型实现栈结构

在使用泛型时要指明泛型的具体类型，这样就避免了类型转换。本实例将使用泛型来实

现一个栈结构，并对其进行测试，效果图如下图所示。

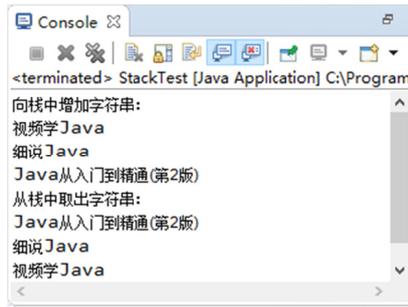


图 6 实现效果图

7. 泛型化方法与最小值

任何实现了 `Comparable` 接口的类的实例，都可以比较大小。本实例将实现一个泛型版本的比较大小方法，效果图如下图所示。

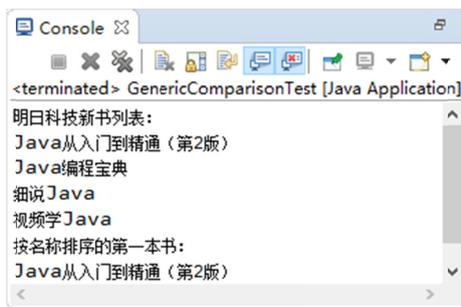


图 7 实现效果图

8. 泛型化方法与最大值

泛型接口的作用与普通接口是相同的，只是其实用性更强。本实例定义了一个泛型接口，用于获得给定数组 `array` 中的最大元素，效果图如下图所示。



图 8 实现效果图

第 18 章 多线程

1. 球员入场

足球比赛开赛前，A、B 两队的上场球员（每队个 11 名球员）会依次出现在两个半场处。受球场条件限制，球员通道口每次只能通过一名球员，使用 Thread 类，模拟球员入场场景：A 队的 11 名球员先入场，B 队的 11 名球员后入场。

2. 模拟下载进度条

通过实现 Runnable 接口模拟下载进度条：单击“开始下载”按钮后，“开始下载”按钮失效且进度条从 0 不断加 5，直至加至 100。进度条达到 100 后，失效的“开始下载”按钮变为被启用的“下载完成”按钮，单击“下载完成”按钮后，销毁当前窗体，效果图如下所示。

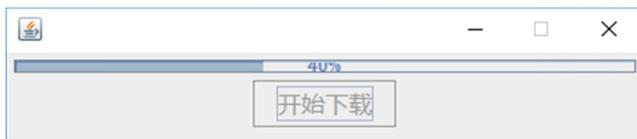


图 1 实现效果图

3. 子弹数

射击游戏中，许多玩家都喜欢狙击步枪与手枪的组合：远距离狙杀，静距离博弈。现使用 Thread 类表示狙击步枪（10 发子弹）、实现 Runnable 接口表示手枪（7 发子弹），在控制台输出它们射击后的剩余子弹数，直至把所有子弹打光为止，效果图如下所示。



图 2 实现效果图

4. 红绿灯

模拟红绿灯变化场景：红灯亮 8 秒，绿灯亮 5 秒，黄灯亮 2 秒，效果图如下所示。



图 3 实现效果图

5. 霓虹灯

霓虹灯之“流·浪·地·球”（改变字体样式与颜色以及面板背景色，变化的时间间隔为 3 秒），效果图如下所示。

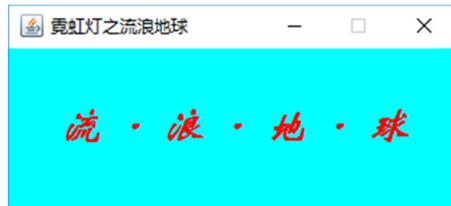


图 4 实现效果图

6. 龟兔赛跑

使用线程的加入模拟龟兔赛跑：兔子跑到 70 米的时候，开始睡觉；乌龟爬至终点时，兔子醒了跑至终点，效果图如下所示。



图 5 实现效果图

7. 电梯超载

电梯最大载重量为 1000 千克，使用线程的中断模拟当乘客的总重量超过 1000 千克时，电梯发出预警提示音。假设乘客的平均体重为 80 千克，使用 Thread 类输出电梯发出预警提示音前可容纳的人数，效果图如下所示。

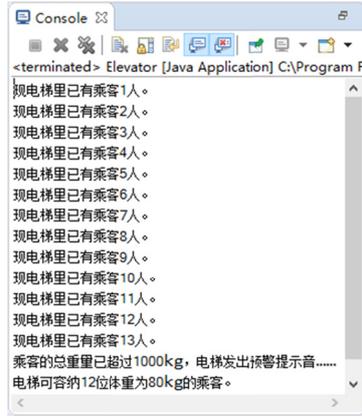


图 6 实现效果图

8. 无规则运动的“●”和“★”

使用 Swing 和线程实现“●”和“★”在窗体中做无规则运动，效果图如下所示。

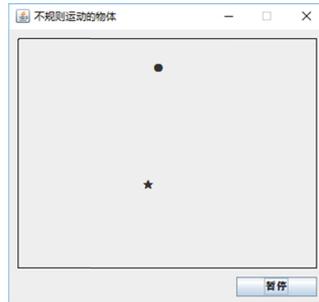


图 7 实现效果图

第 19 章 网络通信

1. 获取局域网的所有 IP 地址

在进行网络编程时，有时需要对局域网内的所有主机进行遍历，为此需要获取内网的所有 IP 地址。本实例将演示如何在 Java 应用程序中，获取内网的所有 IP 地址，效果图如下图所示。



图 1 获取内网的所有 IP 地址

2. 使用 Socket 传输图片

使用套接字进行网络编程，有时需要通过 Socket 传输图片，本实例讲解如何通过套接字传输图片，效果图如下图所示。



图 2 服务器端选择图片并发送



图 3 客户端显示接收到的图片

3. 使用 Socket 传输音频

使用套接字进行网络编程，有时需要通过 Socket 传输音频文件，本实例讲解如何通过套接字传输音频文件，效果图如下所示。



图 4 服务器端选择音频并发送



图 5 客户端显示文件接收完毕

4. 一个服务器与一个客户端通信

在使用套接字进行网络编程时，需要在服务器和客户端之间进行通信，本实例讲解如何实现一个服务器与一个客户端进行通信，效果图如下所示。

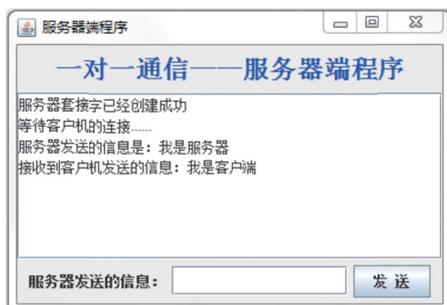


图 6 服务器端发送和接收信息的效果

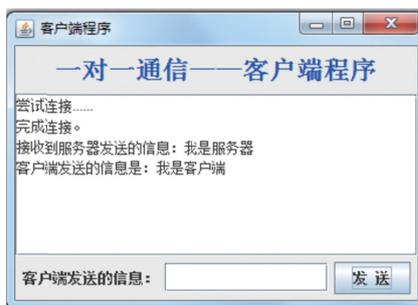


图 7 客户端接收和发送信息的效果

5. 一个服务器与多个客户端通信

在使用套接字进行网络编程时，需要在服务器和客户端之间进行通信，本实例讲解如何实现一个服务器与多个客户端进行通信，效果图如下所示。

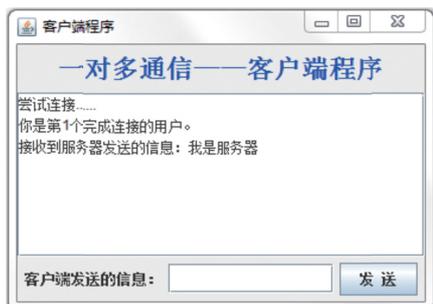


图 8 第一个客户端接收到的服务器信息



图 9 第二个客户端接收到的服务器信息

6. 客户端一对多通信

使用套接字进行网络编程，有时需要在不同的客户端之间进行通信，其中有一种通信方式就是一个客户端与其他多个客户端进行通信。本实例讲解如何实现一个客户端与其他多个客户端进行通信，效果图如下所示。

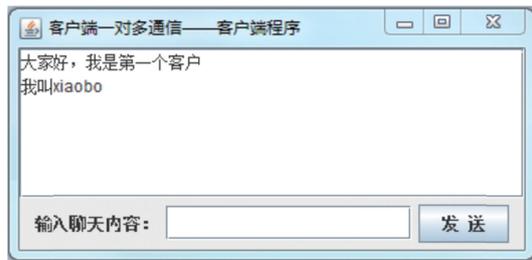


图 10 第二个客户端接收到第一个客户端的信息

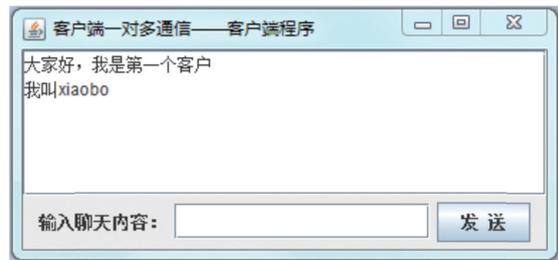


图 11 第三个客户端接收到第一个客户端的信息

7. 客户端一对多通信

使用套接字进行网络编程，有时需要在不同的客户端之间进行通信，其中有一种通信方式就是一个客户端与其他多个客户端进行通信。本实例讲解如何实现一个客户端与其他多个客户端进行通信，效果图如下所示。

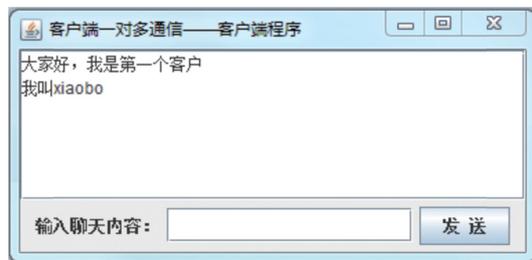


图 12 第二个客户端接收到第一个客户端的信息

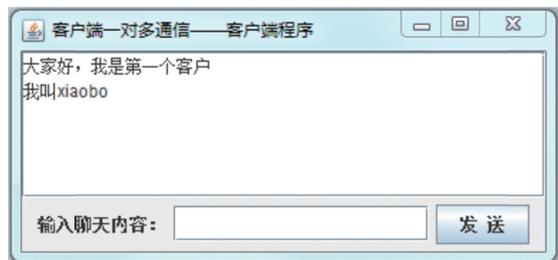


图 13 第三个客户端接收到第一个客户端的信息

8. 客户端一对一通信

使用套接字进行网络编程，有时需要在不同的客户端之间进行通信，其中有一种通信方式就是一个客户端与另一个指定的客户端进行通信。本实例讲解如何实现一个客户端与另一个指定的客户端进行通信，效果图如下所示。

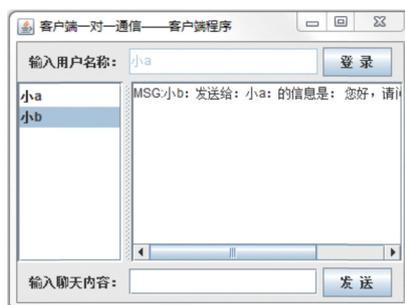


图 14 客户端 aaa 接收到 bbb 的信息



图 15 客户端 bbb 接收到 aaa 的信息

第 20 章 数据库操作

1. 连接 MySQL 数据库

使用 `final` 变量、常量和 JDBC 技术连接 MySQL 数据库。

2. 连接 JavaDB 数据库

JDBC 连接 JavaDB 数据库。

3. 编程词典 6 月的销量

查询 MySQL 数据库 `test` 中的数据表 `tb_book` 中编程词典 6 月的销量后将其输出在控制台上。

4. 销量占前 50%的所有图书信息

查询 MySQL 数据库 `test` 中的数据表 `tb_ware` 中销量占前 50%的所有图书信息，并将这部分信息显示在窗体中的表格中。

5. 动态查询

动态查询 MySQL 数据库 `test` 中的数据表 `tb_book` 中编程词典 6 月的销量后将其输出在控制台上。

6. 显示用户 ID 和头像

首先通过没有通配符“?”的查询语句将 MySQL 数据库 `test` 中的数据表 `tb_picture` 中的用户 ID 和用户头像这两项数据显示在表格中。然后对表格的每一项设置单击事件，即单击表格的某行数据时，在相应的位置上显示用户 ID 和用户头像（在这个显示过程中，需要使用动态查询），效果图如下所示。



图 2 实现效果图

7. 键入员工信息

使用预处理语句将窗体相应文本框中键入的员工信息（包括“姓名”、“性别”、“年龄”、“部门”、“电话”以及“备注”）添加到 MySQL 数据库 db_employeeInfo 中的数据表 tb_employer。（编写代码之前要先将 database 文件夹下的 db_employeeinfo.sql 文件通过“source 命令”导入到 MySQL 数据库中）。

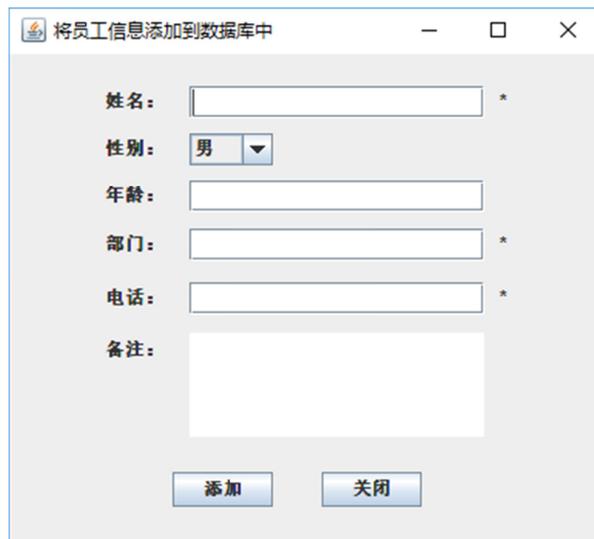


图 3 实现效果图

8. 批量删除

使用预处理语句和事务根据单据编号批量删除 MySQL 数据库 test 中数据表 tb_listInfo 和 tb_productInfo 中与指定单据编号对应的商品信息。

第 21 章 Swing 表格组件

1. 表格的选择方式

对于一个二维的表格而言，其选择模式有很多种。以行为例，可以选择一行、连续几行、任意几行，对于列同理。此外，对于单元格也可以设置成选择一个单元格、选择一个连续区域的单元格或选择一个不连续区域的单元格等。本实例将演示它们的用法，效果图如下所示。

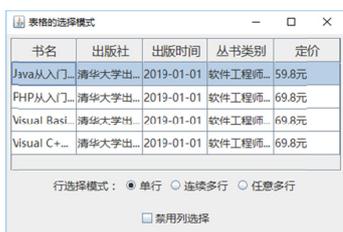


图 1 单选

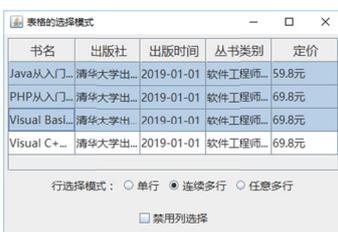


图 2 连续多行

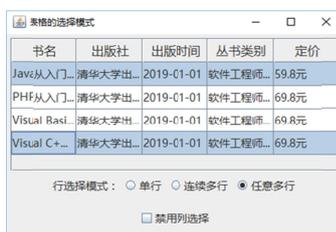


图 3 任意多行

2. 表头显示提示信息

作为对表头含义的补充说明，可以为表头添加提示信息。本实例将实现让不同的表头显示不同的提示信息，效果图如下所示。

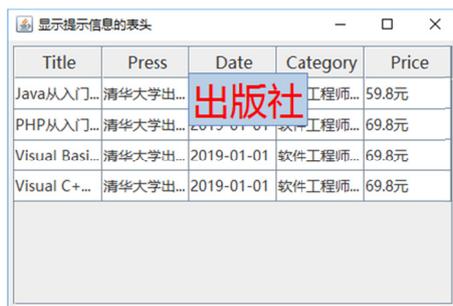
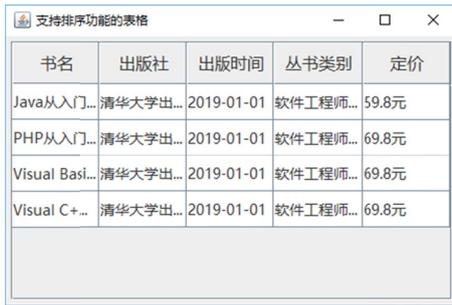


图 4 实现效果图

3. 单元格排序

在使用表格时，会出现根据不同的列来对表格进行排序的情况。例如，可以根据价格的不同来对各种书籍进行排序。本实例将演示如何实现简单的排序功能，效果图如下所示。



| 书名 | 出版社 | 出版时间 | 丛书类别 | 定价 |
|----------------|----------|------------|----------|-------|
| Java从入门... | 清华大学出... | 2019-01-01 | 软件工程师... | 59.8元 |
| PHP从入门... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |
| Visual Basi... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |
| Visual C+... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |

图 5 排序前

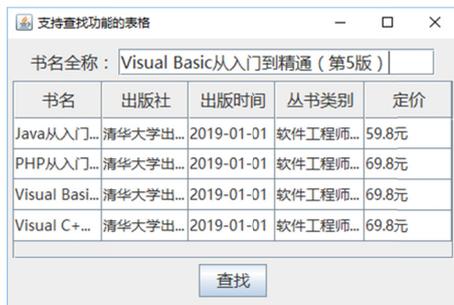


| 书名 | 出版社 | 出版时间 | 丛书类别 | 定价 |
|----------------|----------|------------|----------|-------|
| PHP从入门... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |
| Visual Basi... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |
| Visual C+... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |
| Java从入门... | 清华大学出... | 2019-01-01 | 软件工程师... | 59.8元 |

图 6 排序后

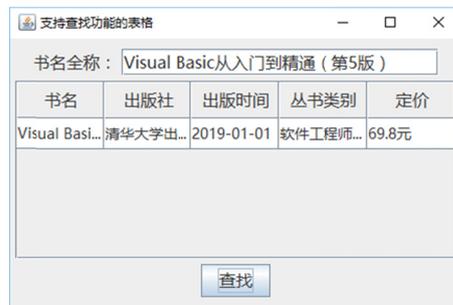
4. 表格的查找功能

对于数据量较大的表格，通常会为用户提供查找功能。这可以让用户快速找到定位到需要的行。本实例将演示如何在表格中实现这个功能，效果图如下所示。



| 书名 | 出版社 | 出版时间 | 丛书类别 | 定价 |
|----------------|----------|------------|----------|-------|
| Java从入门... | 清华大学出... | 2019-01-01 | 软件工程师... | 59.8元 |
| PHP从入门... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |
| Visual Basi... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |
| Visual C+... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |

图 7 查找前

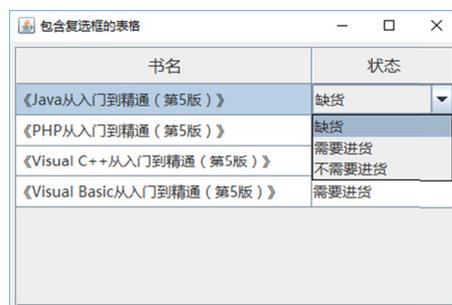


| 书名 | 出版社 | 出版时间 | 丛书类别 | 定价 |
|----------------|----------|------------|----------|-------|
| Visual Basi... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |

图 8 查找后

5. 表格中的组合框

当使用表格统计商品信息时，在一个单元格中使用组合框来设置商品的销售状态比提供代表不同状态的列让用户填写增加方便。本实例将演示如何在表格中实现这个功能，效果图如下所示。



| 书名 | 状态 |
|---------------------------|-------|
| 《Java从入门到精通（第5版）》 | 缺货 |
| 《PHP从入门到精通（第5版）》 | 缺货 |
| 《Visual C++从入门到精通（第5版）》 | 需要进货 |
| 《Visual Basic从入门到精通（第5版）》 | 不需要进货 |

图 9 实现效果图

6. 删除行数据

通常情况下，表格中的数据是经常变化的。本实例将演示如何在表格中删除行数据，效果图如下所示。



Figure 10 shows a window titled "删除表格中的行" (Delete rows in the table). It contains a table with 5 columns: 书名 (Book Name), 出版社 (Publisher), 出版时间 (Publication Date), 丛书类别 (Series Category), and 定价 (Price). The table has 5 rows of data. The third row, "Visual Basi...", is highlighted in blue. Below the table is a "删除" (Delete) button.

| 书名 | 出版社 | 出版时间 | 丛书类别 | 定价 |
|----------------|----------|------------|----------|-------|
| Java从入门... | 清华大学出... | 2019-01-01 | 软件工程师... | 59.8元 |
| PHP从入门... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |
| Visual Basi... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |
| Visual C+... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |

图 10 删除前

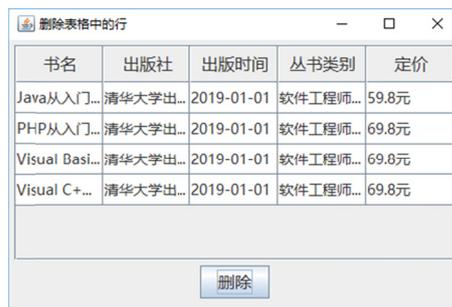


Figure 11 shows the same window after the deletion of the third row. The table now has 4 rows of data. The "删除" button is still present.

| 书名 | 出版社 | 出版时间 | 丛书类别 | 定价 |
|--------------|----------|------------|----------|-------|
| Java从入门... | 清华大学出... | 2019-01-01 | 软件工程师... | 59.8元 |
| PHP从入门... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |
| Visual C+... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |

图 11 删除后

7. 为表格绘制背景色

如果程序的界面中只有黑白两种颜色，则会让用户感觉界面很不好看。通常可以使用各种颜色的搭配来美化界面。本实例使用表格单元格渲染工具来为不同的单元格设置不同的颜色，效果图如下所示。

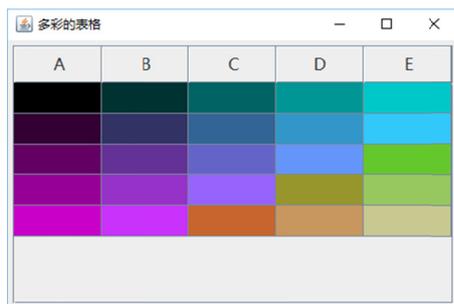


Figure 12 shows a window titled "多彩的表格" (Colorful Table). It displays a 5x5 grid of colored cells. The columns are labeled A, B, C, D, and E. The colors transition from dark purple/black on the left to bright cyan/green on the right, with a gradient effect across both rows and columns.

| A | B | C | D | E |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |
| | | | | |

图 12 实现效果图

8. 表格的栅栏效果

为了让用户获得更舒适的体验，通常将表格设置成栅栏效果，即奇数行和偶数行的背景颜色是不同的。本实例将演示如何实现该效果，效果图如下所示。

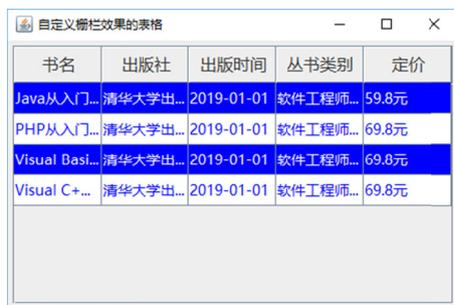


Figure 13 shows a window titled "自定义栅栏效果的表格" (Table with Custom Striped Effect). It displays a table with 5 columns: 书名, 出版社, 出版时间, 丛书类别, and 定价. The table has 5 rows of data. The rows alternate between a light blue background and a white background, creating a striped effect.

| 书名 | 出版社 | 出版时间 | 丛书类别 | 定价 |
|----------------|----------|------------|----------|-------|
| Java从入门... | 清华大学出... | 2019-01-01 | 软件工程师... | 59.8元 |
| PHP从入门... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |
| Visual Basi... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |
| Visual C+... | 清华大学出... | 2019-01-01 | 软件工程师... | 69.8元 |

图 13 实现效果图

第 22 章 Swing 树组件

1. 编写中国省市信息树

对于具有层次关系的结构，使用树控件描述是非常方便的，如文件夹及其子文件夹之间的关系、国家的行政结构关系等。本实例将使用树控件来表示中国的各个行政区域，效果图如下所示。

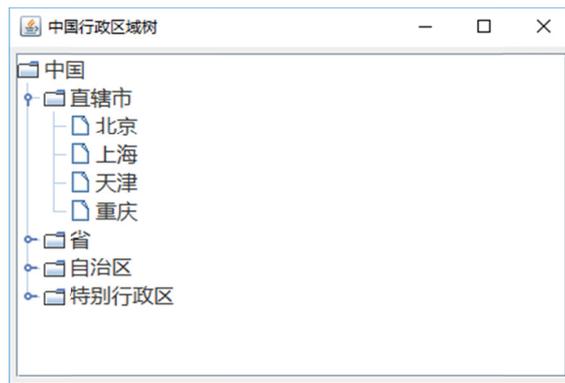


图 1 实现效果图

2. 自定义树节点的图标

树默认节点图标并不好看，为了美化界面，通常需要修改图标。本实例将使用 `UIManager` 类来改变树的图标，效果图如下所示。



图 2 实现效果图

3. 监听节点

编写一个程序，对节点的选择事件进行监听，当用户选择不同的节点时，会在右侧的文本区中显示该节点的一些信息，效果图如下所示。



图 3 选择第 1 个节点



图 4 选择第 2 个节点

4. 树组件的选择模式

树组件的选择模式有 3 种：只能选择一个节点、可以选择连续若干节点和可以选择若干不连续的节点。本实例将演示树组件的各个选择模式，效果图如下所示。

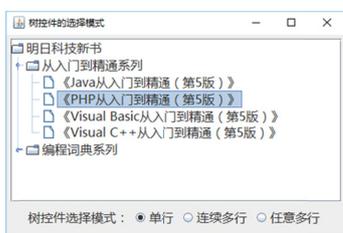


图 5 单选

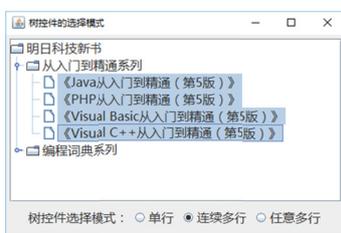


图 6 连续多行

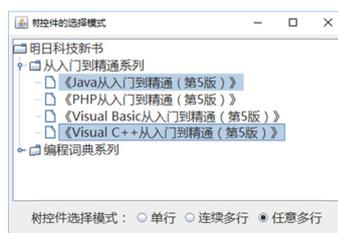


图 7 任意多行

5. 查看节点的状态值

对于一棵树而言，其各个节点的状态是不同的。例如，是否是根节点、是否是叶子节点、具有子节点的个数等。这些状态值对于树而言是非常重要的。本实例将实现查看树节点状态的功能，效果图如下所示。

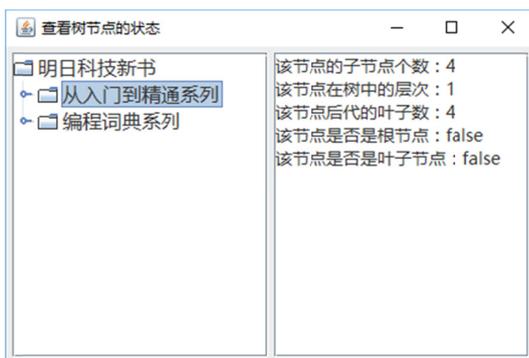


图 8 实现效果图

6. 添加子节点

除了让用户在给定的节点中进行选择外，还可以让用户在需要的位置添加自定义节点。这样程序会有更好的交互效果。本实例将实现让用户添加可以编辑的节点的功能，还可以为新的节点添加子节点，效果图如下所示。

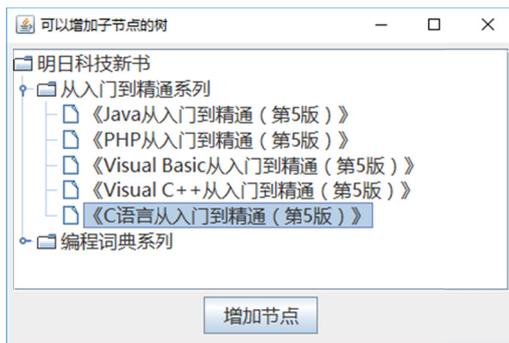


图 9 实现效果图

7. 删除子节点

编写一个程序，实现让用户删除子节点的功能，如果该子节点包括其他子节点，则一并删除，效果图如下所示。

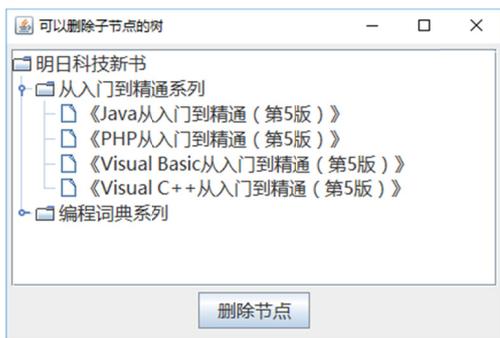


图 10 删除前

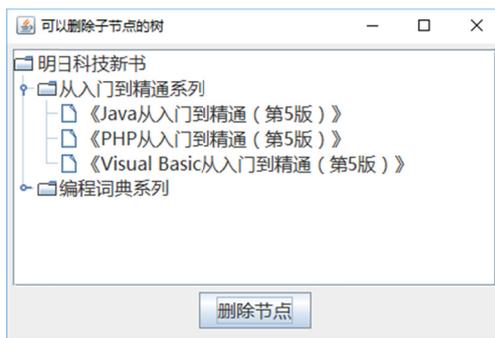


图 11 删除后

8. 查找子节点

让用户展开每个节点来查找信息，是非常麻烦的。因此，要为用户提供查找功能。本实例实现了树节点的查找功能，效果图如下所示。

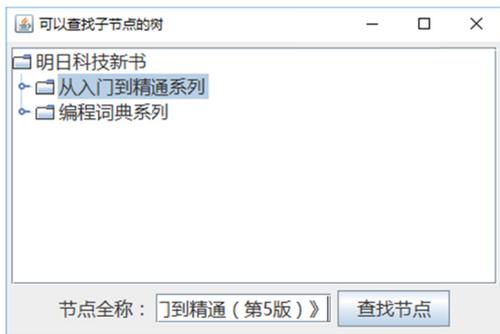


图 12 查找前

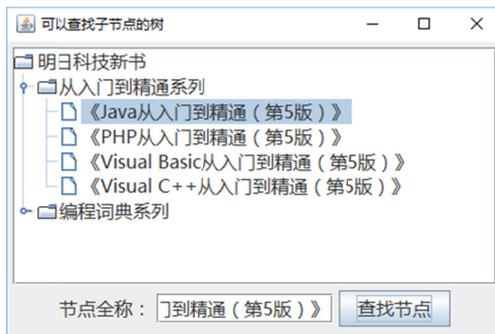


图 13 查找后

第 23 章 Swing 其他高级组件

1. 图片展示板

使用分割面板，编写一个程序：在窗体中，使用了两个分割面板来组织列表和标签，用来显示图片的名称、内容和描述信息，效果图如下所示。

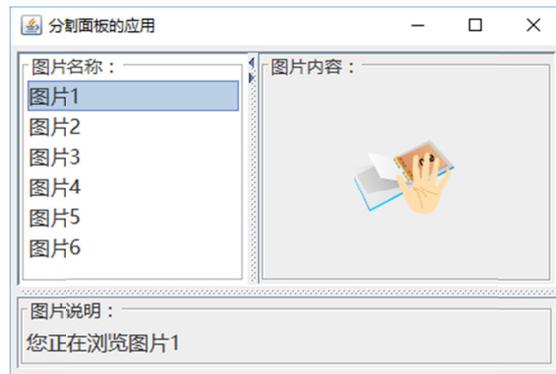


图 1 实现效果图

2. 为选项卡增加快捷键

对于一个复杂的窗体而言，可以使用选项卡面板来组织相关的控件。如果为不同的选项卡增加快捷键，可以让用户快速定位到自己需要的选项卡。本实例将通过 **Alt** 键和选项卡中有下划线的字符组合使用（例如 <Alt+C>），切换不同的选项卡，效果图如下所示。



图 2 实现效果图

3. 平铺子窗体

如果一个窗体中有多个子窗体被打开，那么界面会显得非常混乱。本实例将让这些子窗体平铺在窗体中，使其看起来更加有序，效果图如下所示。

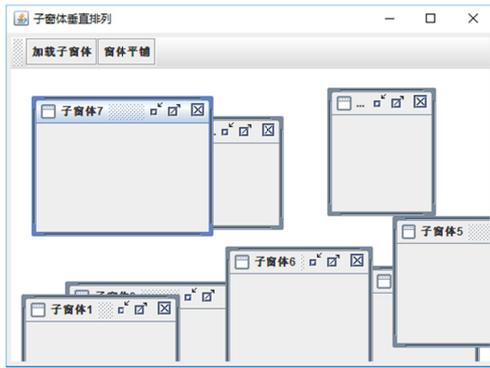


图 3 多个子窗体被打开

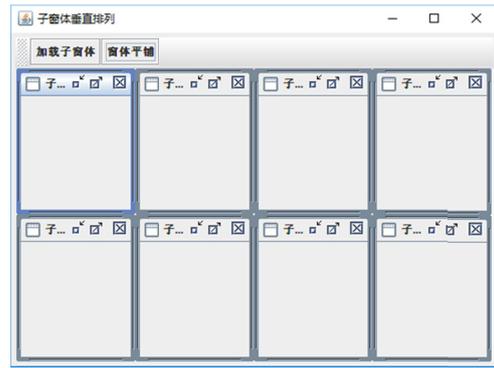


图 4 平铺子窗体

4. 模拟记事本的菜单栏

在 Windows 操作系统中，自带了一款简单的文本编辑工具——记事本。记事本主要由菜单栏和文本区两部分组成。菜单栏实现了各种常用的功能，文本区用于让用户输入文本。本实例将实现一个类似记事本的菜单栏，效果图如下所示。

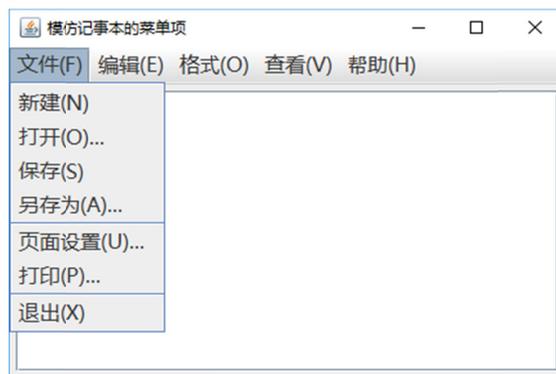


图 5 实现效果图

5. 可以拖拽的图标工具栏

对于一些功能非常复杂的软件，可以将一些常用的操作放置在一个工具栏中方便用户使用。本实例将演示如何实现一个可以拖拽的图标工具栏，效果图如下所示。

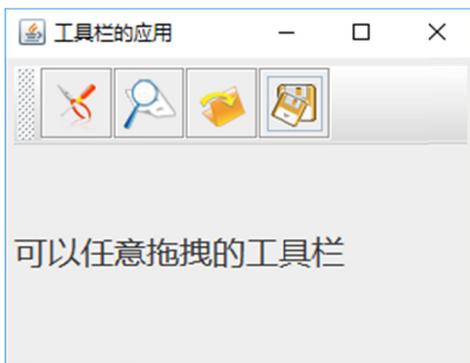


图 6 横置工具栏

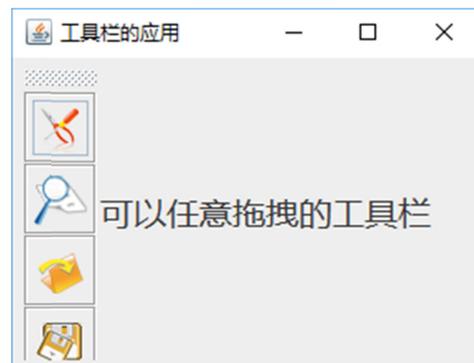


图 7 纵置工具栏

6. 文件的加密与解密

加密文件能够提高文件的安全性，为了读懂加密后的文件，还要提供文件的解密方法。本实例将借助 Java 中的流技术，实现文件的加密与解密，加密的效果图如下所示。

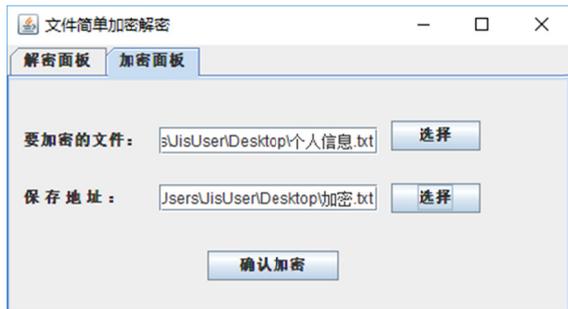


图 8 加密面板

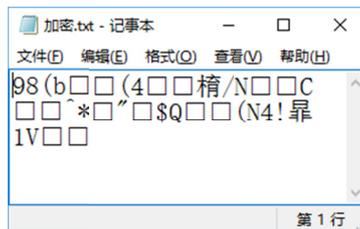


图 9 加密后的文件

7. 进度指示器

当程序执行一些耗时操作时，使用进度指示器提示用户程序正在运行是非常有用的。本实例将演示在文本区中输出 500 以内的全部素数，并使用进度监视器提示用户输出的进度，效果图如下所示。

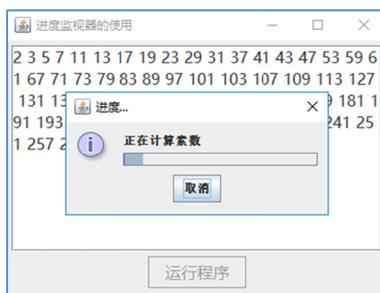


图 10 实现效果图

8. 自定义系统托盘的图标

编写一个程序，自定义系统托盘的图标，并在系统托盘图标上增加弹出式菜单来对程序进行一些快捷操作，效果图如下所示。



图 11 实现效果图

第 24 章 高级事件处理

1. 虚拟键盘

键盘是最常用也是最主要的输入设备，通过键盘可以将英文字母、数字、标点符号等输入到计算机中，从而向计算机发出命令、输入数据等。使用键盘事件，模拟一个虚拟键盘，效果图如下所示。



图 1 实现效果图

2. 按下回车键确认登录

使用键盘事件，编写一个程序：在窗体的文本框中分别输入用户名和密码后，按下回车键，实现 QQ 登录验证，效果图如下所示。

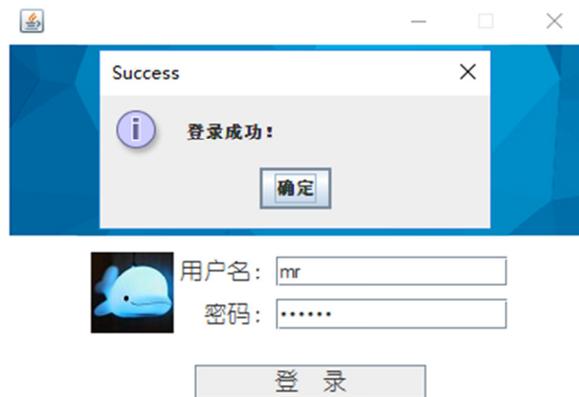


图 2 实现效果图

3. 永远拆不了的红包

微信红包风靡一时，使用鼠标事件，编写一个程序：当鼠标移入“红包”时，“红包”就会移动至另一个位置，效果图如下所示。



图 3 “红包”初始位置

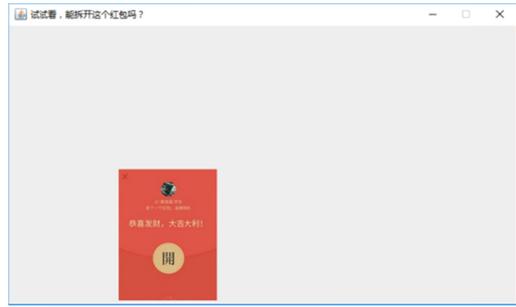


图 4 “红包”移动后的位置

4. 真的要退出本窗口么？

当一个 APP 使用结束后退出时，会弹出一个确认是否退出的对话框。使用窗体事件，编写一个程序：关闭窗体时，弹出“确认”或“取消”对话框，效果图如下所示。

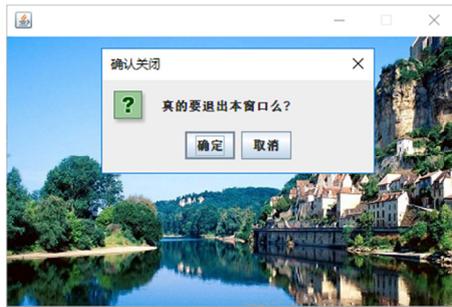


图 5 实现效果图

5. 东三省的省、市联动

随着互联网的快速发展，越来越多的网页开始使用省、市联动。为此，使用选项事件，在窗体中，模拟东三省的省、市联动，效果图如下所示。



图 6 实现效果图

6. 列表项的单选、连选和多选

设置列表项的选择模式：MULTIPLE_INTERVAL_SELECTION（一次选择一个或多个连续的索引范围）、SINGLE_INTERVAL_SELECTION（一次选择一个连续的索引范围）以及 SINGLE_SELECTION（一次选择一个索引），效果图如下所示。

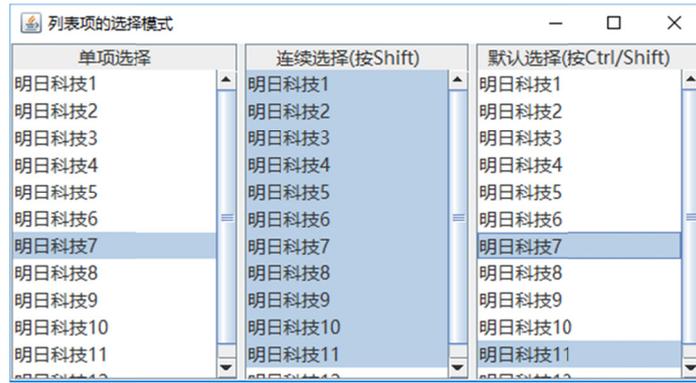


图 7 实现效果图

7. 2019 编程语言前三甲

近 20 年来，java、c 和 c++ 一直排在前三名。但是 Python 越来越受欢迎。随着人工智能的发展，Python 成为了机器学习的首选语言。这样，Python 成功地挤掉了 c++，排在了第 3 的位置上。自定义一个表格，列出 2019 编程语言前三甲，效果图如下所示。

| 排名 | 语言 |
|----|--------|
| 1 | Java |
| 2 | C |
| 3 | Python |

图 8 实现效果图

8. 表格的分页

当表格显示的数据量比较庞大时，为了用户浏览数据方便，会使用分页技术。在表格模型的基础上，实现表格的分页，效果图如下所示。

| 自然数 | 平方数 |
|-----|-----|
| 0 | 0 |
| 1 | 1 |
| 2 | 4 |
| 3 | 9 |
| 4 | 16 |

图 9 实现效果图

第 25 章 AWT 绘图

1. 2019 年必看影片

2019 年将有许多好看的影片陆续上映，在这么多的电影中，影迷们肯定不会错过重头戏。使用 AWT 的相关技术，在窗体中绘制“2019”必看影片，效果图如下所示。



图 1 实现效果图

2. 绘制五环图案

五环由蓝、黄、黑、绿、红 5 个奥林匹克环套接组成。这 5 个不同颜色的圆环象征着 5 大洲。使用 AWT 的相关技术，在窗体中绘制五环图案，效果图如下所示。



图 2 实现效果图

3. 绘制公章

公章是指机关、团体、企事业单位使用的印章。公章的印文自左而右环行，使用简化的宋体字。使用 AWT 的相关技术，在窗体中绘制一个公章，效果图如下所示。



图 3 实现效果图

4. 裁剪图片

裁剪是 Windows 的画图工具常用的功能之一，用户按住鼠标左键拉动，框选一部分想要截取的图片，单击工具栏上的“剪裁”，即可得到裁剪后的图片。使用 AWT 的相关技术，模拟上述过程，效果图如下所示。



图 4 实现效果图

5. 调整图片亮度

使用 AWT 的相关技术，编写一个程序：用户能够根据喜好，既可以使得图片变亮或者变暗，又可以使得图片恢复为原来的样式，效果图如下所示。



图 5 调高亮度



图 6 调低亮度



图 7 恢复原样

6. 水印文字特效

网络上的图文资源丰富，为了维护正版，各个网站采用为这些图文资源加水印文字特效的方式方法。使用 AWT 的相关技术，为图片加上“盗版必究”的水印效果，效果图如下所示。



图 8 实现效果图

7. 带背景图片的验证码

随着互联网的不断发展，验证码的出现频率越来越高，验证码的样式也愈加复杂多样。使用 AWT 的相关技术，实现一个带背景图片的验证码的登录界面，效果图如下所示。

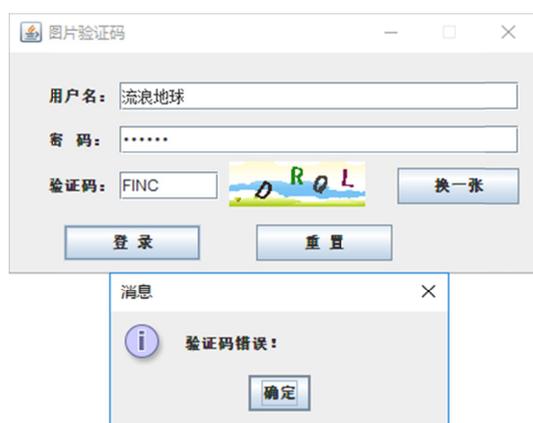


图 9 实现效果图

8. 绘制石英钟

在日常生活中，时钟准到 1 秒，就已经足够了。而石英钟就是这样的一种计时器具。使用 AWT 的相关技术，绘制一个石英钟，效果图如下所示。



图 10 实现效果图