



赠光盘



● 宛延闯
等著

Java

编程技巧



机械工业出版社
CHINA MACHINE PRESS

Java 编程技巧

宛延闯 等著



机械工业出版社

Java 是面向对象通用的程序设计语言。Java 不但适用于网络通信编程, 而且适用于一般大规模软件工程信息处理系统。

为使读者尽快学会 Java 编程, 本书精选了 18 个典型实例, 展示了如何将 Java 编程技巧应用于实践。这 18 个典型实例, 既有 Java 小应用, 又有 Java 应用程序和网络编程。这些编程技巧实用、易于掌握, 实例设计构思独到、动画优美流畅、音乐美妙动听, 可与用户对话, 具有很强的实用性。

本书最后还附有一些练习题, 读者可模仿练习。所有 Java 小应用和应用程序的源代码、HTML 文件、图像和声音文件等均收录在本书的光盘中。本书适用于广大 Java 编程爱好者、大专院校学生和专业人士学习和参考。

版权声明

本书所有典型实例的源程序代码著作权属作者所有, 未经作者书面许可, 不得以任何形式摘抄、复制本书的任何部分。

图书在版编目 (CIP) 数据

Java 编程技巧/宛延闯等著. —北京: 机械工业出版社, 2005.1
ISBN 7-111-15298-0

I. J... II. 宛... III. Java 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2004) 第 096774 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策划编辑: 胡毓坚 责任编辑: 董欣 版式设计: 张世琴

责任校对: 刘志文 封面设计: 孔德华 责任印制: 李妍

北京蓝海印刷有限公司印刷·新华书店北京发行所发行

2005 年 1 月第 1 版·第 1 次印刷

787mm×1092mm 1/16·25.75 印张·635 千字

0001-5000 册

定价: 41.00 元 (含 1CD)

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

本社购书热线电话 (010) 68993821、88379646

68326294、68320718

封面无防伪标均为盗版

前 言

Java 诞生于 20 世纪 90 年代中期,在过去的几年里,Java 程序设计语言在计算机软件领域里引起的革命,使它一跃成为在万维网 (WWW) 上开发软件的主要工具,并正在迅速取代 Pascal、C 和 C++ 在大学讲堂的地位。Java 本身也在不断地发展和完善。当前 Java 的版本是 JDK1.3 和 JDK1.4。Java 从 C++ 而来,是有史以来惟一被工业界和学术界同时认可的语言。所以,Java 应该是初学者的第一语言。在我国,因特网 (Internet) 正在迅速普及,信息高速公路很快就会像电话一样进入每个家庭,随之而来的就是 Java。

WWW 和 Java 是因特网上两个最具革命性的事件。在 Java 之前,万维网上的网页基本上是静态的,一个网点不能执行另一个网点上的程序,因为这两个网点可能用的是不同的计算机、不同的操作系统,或者只是不同版本的软件。即使这些全一样,也还有个关键的安全性问题。Java 的出现迅速改变了这种状况。Java 的小应用 (Applet) 可以被下载到任何用户机器上安全运行,这使得用户与网页的互动对话、动画等成为现实。用 Java 做成的动画优美流畅,音乐美妙动听。Java 为万维网带来了生命和一场真正意义上的技术革命。

人们预言 Java 还将带来更深刻的革命。它将改变我们使用软件的方式。你不用再去购买专用软件,然后没完没了地更新。你只要在需要时访问专门网页,交使用费,就能使用最新版本的软件。这样的网络将面向用户,给你的计算机“减肥”(减少 fatware);而不是像现在的微机那样面向一个大公司,经常要你去购买新版本软件。

Java 是一种通用的面向对象程序设计语言。Java 不但适用于网上程序设计,也适用于一般大规模软件工程项目。

为什么要学习 Java?

因为

- Java 是彻底的面向对象语言;
- Java 是 Web 上的面向对象语言;
- Java 是 Web 上的并行语言;
- Java 可以设计出先进的数据结构。

好的程序设计风格,才有可能造就好的程序员。Java 可以!

Java 不仅仅是一门语言,而且已经形成了一门技术。它包括 Java 软件技术、Java 虚拟机技术、Java 嵌入技术、Java 信息综合处理技术和 Java 应用技术等。

如何学习 Java?

学习 Java 程序设计语言,就一定要学会编程。如果你不编程,就永远学不会 Java。为此,本书为广大学习 Java 的读者提供了一个极好的机会,即从我们自主设计开发的上千个 Java 小应用和应用程序中,精选出 18 个典型实例,展示了如何将 Java 的编程技巧应用到实际生活之中,并循序渐进地教会读者如何进行 Java 程序设计,如何用面向对象方法设计构思,如何确定具体的开发步骤,如何运用编程技巧加以实现等。与此同时,我们还提供了相应的 Java 小应用和应用程序的程序流程图(或用 UML 对象图来解释实例)、带注释的 Java

源代码、HTML 文件以及相应的图像和声音文件，帮助广大读者和学生尽快地掌握 Java，学会 Java 编程。希望读者从本书中领悟 Java 编程的真谛，实践——理解——再实践，从中深入理解 Java，汲取 Java 编程的知识和技巧。子曰：“知之者不如好之者，好之者不如乐之者。”读者必须输入、修改、编译、运行每一个实例程序，多做练习题。每运行一个程序，都可能给你一些新的启示。我们相信广大读者很快就会发现编制 Java 程序是一件很有趣的事情。

如何获得 JDK 开发套件？

本书的实例都是在 JDK 开发套件和 Windows 9x 操作系统环境中进行开发和严格测试的，所以读者必须拥有 JDK 开发套件。我们知道，JDK 是由 Sun 公司开发提供的，可以免费下载。在此，我们提供相应的网址，最新的 JDK 开发套件由读者自己下载获得。

目前，JDK 开发套件的版本是 JDK1.3 和 JDK1.4，都隶属于 Java2 的版本。下载 Java 软件应包含两个部分：一是 JDK 开发套件本身；另一是 Java 插件。

本书所用的 Java 软件为 JDK1.3.1（或 JDK1.4.1）及 JRE1.3。

(1) 下载 JDK 开发套件的网址为

<http://java.sun.com/products/jdk/1.3/index.html>

<http://www.javasoft.com/products/jdk/1.3/index.html>

(2) 下载 Java 插件（plug-in）的网址为

<http://java.sun.com/j2se/1.3/jre/download-windows.html>

Java 插件是 Sun 公司开发的基于 Java2 的 Applet 和 JavaBean 的运行环境 JRE（Java Runtime Environment）的软件产品。Java 插件可以确保得到 Sun 公司的最新的 JDK 开发套件的支持，而不像以前那样要等待较长时间 IE 和 Navigator 才会支持这些最新功能，使 JDK 向最新版本升级变得简单。读者若要运行 Java2 的 Applet 和 Application，就需要安装 Java 插件。

当然，我们也可以从国内的一些 ftp 站点中获得 Java 插件。这比从国外站点下载快一点，比如从 <ftp.pku.edu.cn> 站点的 pub \ javasoft \ pub \ jdk1.3.1 目录下载。

如何安装 JDK 开发套件？

以在 Windows 9x 下安装 JDK1.3.1 为例说明。

(1) 将已下载的 JDK1.3.1 和 JRE1.3 安装在 C:\ 目录下。因为 JDK 是自解压的软件，所以只要按 JDK 提示的步骤安装即可。注意，在安装 JRE 时，由用户选择所安装的浏览器，通常我们选择 IE。

(2) 安装完毕，就要在 MS-DOS 下改写批处理文件 Autoexec.bat，把需要的路径加进去（edit autoexec.bat）：

```
PATH = C:\windows;C:\windows\command;C:\dos;C:\jdk1.3.1\bin;
```

```
set CLASSPATH = . ;C:\jdk1.3.1\lib\tools.jar
```

(3) 重新启动 Windows 9x，Windows ME 或 Windows NT，以使环境变量 PATH 和 CLASSPATH 生效。

JDK 自带的浏览器是 AppletViewer。常用的命令有 Javac，Java 等。

利用 Windows 内建立的“记事本”，或者用 Ultra Editor 编辑器编写或修改 Java 程序。当然，用 MS-DOS 的 edit 编辑器也可以编制 Java 程序。

Java 的软件在 MS-DOS，Windows，Unix，Macintosh 等各种不同平台上都可以运行，但

必须下载安装不同平台的 JDK 和 JRE。

本书的特点

本书的所有典型实例都经过了严格的测试和筛选。在这些实例中，既有 Java 小应用，又有 Java 应用程序和网络编程。实例内容丰富、设计构思独到、动画优美流畅、可与用户互动对话，实用性强。本书最后还附有有趣的练习题，读者可参照相应章的典型实例进行模仿练习，编制 Java 小应用、Java 应用程序，以及 Java 网络编程，可起到事半功倍的效果。我们相信，这些丰富多彩的 Java 程序能够帮助读者迅速地掌握 Java 和面向对象程序设计的精髓。

本书著者

本书的著者还有石良秀、宛霞、甄炜、李保林、乔立琴、郭应中、王浩枫、宛辉、马岩、张雪囡、王子滨、马同智、石新路、李俊、宛宁、王心颖、胡骏、潘京、高嵩华、代宁、张纪敏、甄玉、李永麟、季英洁、王玥、鲁日乔和高晓丽。参加各章开发设计实现的还有：第 1 章潘晖；第 2 章沈瑶青和定海；第 3 章徐兴和鲁日乔；第 4 章梁春燕和苏文；第 5 章李斌和鲁玛勒；第 6 章李速、方彩红和李蔚；第 7 章陶薇和英小平；第 8 章魏本富和崔可爱；第 9 章张雄飞和蔡凤奇；第 10 章马维国和王令；第 11 章陈果和王静；第 12 章李征宇、崔柯和苏文。第 13 章贾杨和李一林；第 14 章武明辉和关灵艳；第 15 章王雪飞、苏斌和定海；第 16 章黄庆妮和国华；第 17 章王平和李石乔；第 18 章肖艳文和储春梅等。

本书光盘的内容

光盘中收录了本书 18 个典型实例的类文件、HTML 文件、Java 源代码以及相应的图像和声音文件。其目录已制作成动态超链接模式，读者可轻松点击要查看的章节，使用起来非常方便。读者可把它复制到 JDK 所指示的路径上运行、学习和参考。

推荐的参考书

与本书配套的教材是《实用 Java 程序设计教程》和《实用 Java 程序设计教程习题解答及上机实践》，都已由机械工业出版社出版。

宛延闯

目 录

前言

第 1 章 名字旋转..... 1

- 1.1 问题陈述 1
- 1.2 实现原理及程序流程图 1
- 1.3 使用本小应用的方法 2
- 1.4 对上述程序的改进 2
- 1.5 Java 源程序 3
- 1.6 运行结果 11

第 2 章 庆祝党的生日 12

- 2.1 问题陈述 12
- 2.2 开发环境 12
- 2.3 使用方法 12
- 2.4 文件清单 12
- 2.5 编程思想及程序框图 13
- 2.6 Java 源程序 13
- 2.7 HTML 源程序 23
- 2.8 程序运行结果 24

第 3 章 世界杯 25

- 3.1 问题陈述 25
- 3.2 开发环境 25
- 3.3 运行描述 26
- 3.4 Java 源程序 (Example1 和
Example2) 26
- 3.5 运行结果 32

第 4 章 咪咪闹钟 33

- 4.1 问题陈述 33
- 4.2 开发环境 33
- 4.3 使用方法和文件说明 33
- 4.4 编程思想和程序流程图 34
- 4.5 Java 和 HTML 源程序 35
- 4.6 咪咪闹钟的用户界面显示 44

第 5 章 十二生肖大摆钟 46

- 5.1 问题陈述 46
- 5.2 开发环境 46
- 5.3 使用方法和文件说明 46

- 5.4 编程思想和程序流程图 47

- 5.5 Java 源程序 48

- 5.6 HTML 程序清单 59

- 5.7 运行结果 59

第 6 章 音乐时钟万年历 61

- 6.1 问题陈述 61
- 6.2 开发环境 61
- 6.3 程序框图及设计思想 61
- 6.4 使用说明 62
- 6.5 Java 源程序 63
- 6.6 运行结果 72

第 7 章 购房子、汽车、飞机 73

- 7.1 问题陈述 73
- 7.2 设计要求 73
- 7.3 开发步骤 73
- 7.4 开发环境 74
- 7.5 使用说明 74
- 7.6 Java 源程序 74
- 7.7 运行结果 83

第 8 章 购买电影票 86

- 8.1 问题陈述 86
- 8.2 开发环境 86
- 8.3 程序功能及框图 86
- 8.4 该小应用所使用方法 86
- 8.5 Java 源程序和 HTML 源程序 86
- 8.6 运行结果 94

第 9 章 网上售书 95

- 9.1 问题陈述 95
- 9.2 开发环境 95
- 9.3 使用方法和文件说明 95
- 9.4 编程思想和程序框图 96
- 9.5 Java 源程序 96
- 9.6 运行结果 110
- 9.7 使用说明 111

第 10 章 青蛙吃小虫 112

10.1 问题陈述	112	14.7 Java 源程序	194
10.2 设计说明及框图	112	14.8 运行结果	210
10.3 文件说明	112	第 15 章 潜艇打商船	212
10.4 Java 源程序	113	15.1 问题陈述	212
10.5 运行结果	119	15.2 开发环境	212
第 11 章 Java 电子琴	120	15.3 游戏规则	212
11.1 问题陈述	120	15.4 程序设计分析	212
11.2 设计思想	120	15.5 程序框图	213
11.3 开发步骤	120	15.6 文件清单	214
11.4 开发环境	121	15.7 Java 源程序	214
11.5 程序框图	121	15.8 运行结果	240
11.6 使用说明和文件清单	122	第 16 章 三层电梯模拟	242
11.7 Java 源程序和 HTML 源程序	122	16.1 问题陈述	242
11.8 运行结果	135	16.2 设计思想	242
第 12 章 打电话 1	137	16.3 开发环境	242
12.1 问题陈述	137	16.4 调试运行环境	243
12.2 设计思想	138	16.5 如何使用电梯模拟的小应用	243
12.3 开发步骤	141	16.6 程序框图及文件说明	243
12.4 软件开发环境	144	16.7 电梯模拟演示操作说明	244
12.5 如何使用该小应用	144	16.8 Java 源程序	247
12.6 类、方法、属性说明	144	16.9 HTML 源代码	324
12.7 Java 源程序	144	第 17 章 Java 聊天室	325
第 13 章 打电话 2	174	17.1 问题陈述	325
13.1 问题陈述	174	17.2 设计与分析	325
13.2 设计分析	174	17.3 开发步骤	325
13.3 开发步骤	175	17.4 软件开发环境	326
13.4 开发环境	175	17.5 程序框图及文件说明	326
13.5 程序框图	176	17.6 如何使用该小应用	327
13.6 详细说明	176	17.7 Java 源程序	328
13.7 使用方法	177	17.8 运行结果	339
13.8 Java 源程序	178	第 18 章 跳水计分系统	342
13.9 运行结果	187	18.1 问题陈述	342
第 14 章 城铁和地铁运行调度	192	18.2 设计与分析	342
14.1 问题陈述	192	18.3 开发步骤	343
14.2 设计与分析	192	18.4 开发环境	343
14.3 开发步骤	192	18.5 如何使用该应用程序	343
14.4 软件开发环境	193	18.6 Java 源程序 (应用程序)	346
14.5 程序框图及文件说明	193	18.7 运行结果	368
14.6 使用方法	194	练习题	402

第1章 名字旋转

读者通过本章的实例将学会：

- 把你自己的名字画在自己的网页上
- 把自己的名字作为字符串，围绕既定的中心点翻转和环绕旋转

本章所用到的编程技巧为：

- (1) 名字旋转处理：为实现立体感效果添加了缩放系数；
- (2) 用 Runnable 接口创建多线程；
- (3) 名字画法及处理(用 drawline() 方法等)；
- (4) 坐标变换公式的应用；
- (5) 背景颜色设置；
- (6) 防止名字闪烁的双缓冲技术。

1.1 问题陈述

- 在自己网页上，画出自己的名字
- 设置网页背景颜色
- 按照不同的坐标变换公式，进行名字旋转

1.2 实现原理及程序流程图

1. 类说明

本程序以类 name 实现“名字旋转”题目的要求。

```
import java.awt.*;
import java.applet.*;
public class name extends java.applet.Applet implements Runnable{
    int width;      //以下是变量说明
    int height;
    int r;
    int d;
    double alfa;
    Thread runner;
    Image offscreenImage;
    Graphics offscreenG;
    public void init(){} //初始化,以下是方法说明
    public void start(){}
    public void stop(){}
    public void run(){}
    public void update(Graphics g){ }
```

```
public void paint(Graphics g){ }
```

程序首先调用 `init()` 设置初值等, 然后调用 `start()` 方法, 在此方法中创建一个新的线程 `runner`, 并调用线程的 `start()` 方法, 使新产生线程从新生态转换到可运行态。并重写线程的 `run()` 方法。调用 `start()` 方法之后, `paint()` 方法被调用。此方法用来画图, 这时将名字按一定角度写在画面上(实际上是先写入缓冲区, 再由缓冲区画在屏幕上, 即采用双缓冲技术, 以消除闪烁)。调用 `stop()` 方法的作用是当离开含有本小应用的网页时, 挂起所有线程, 停止小应用。另外, `repaint()` 方法用于重画名字, 以达到动画效果。

2. 关于名字旋转的处理

根据题意, 我们设置了一个虚轴: 它是与屏幕相距 r 距离的一个垂直轴, 从用户来看是在屏幕内部的一个竖直轴, 名字将环绕它旋转, 旋转的轨迹是以 r 为半径的圆, 如图 1-1 所示。在实际画图时, 将计算每旋转 0.5° 的投影(即计算名字每一笔划的始点与终点的投影坐标)。

当围绕轴转到 α 角时, $d = r * \sin\alpha$

保持姓名的中点与虚轴始终保持 r 的距离, 并且姓名平面始终与半径垂直, 设姓名上任一点坐标为 (x_0, y_0) , 姓名中心的坐标为 $(midx, midy)$ 。旋转 α 角后, 原来的 (x_0, y_0) 点的坐标变为 (x', y') , 则

$$\begin{cases} x' = x_0 \cos\alpha + d + midx \\ y' = y_0 \end{cases}$$

3. 程序流程图

程序流程图见图 1-2。

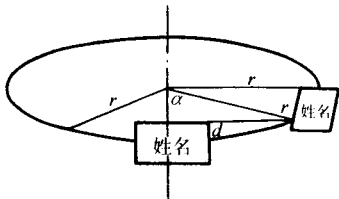


图 1-1 名字旋转示意图

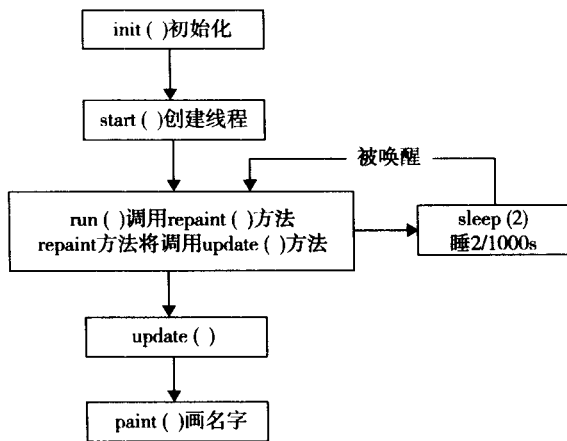


图 1-2 名字旋转框图

1.3 使用本小应用的方法

在 Netscape 浏览器中单击“文件”→“打开”选择目录及文件 `A:\name.html`, 打开后即出现名字旋转。在 IE 浏览器中, 直接双击图标即可。程序清单:

```
name.java    name.class    name.html
```

本小应用是在 JDK1.1.7 下编译的, 对 JDK1.2 和 JDK1.3.1 完全适用。

1.4 对上述程序的改进

根据上述方法实现的小应用看起来稍微缺少一点立体感。因而对旋转过程的算法做了一些修改, 即增加了一个缩小系数 $ratio$, $ratio = 1 - \frac{1}{2} \sin \frac{2}{\alpha}$, 因为在观察时离人眼越远的物体显得越小, 所以让物体乘上一个缩小系数(见下面的坐标变换公式), 从而增强了小应用的立体效果。程序清单见 `name2.java`。

相应的程序清单为:

name2.java name2.class name2.html

坐标变换公式改为:
$$\begin{cases} x' = ratio x_0 \cos \alpha + d + midx \\ y' = y_0 * ratio \end{cases}$$

1.5 Java 源程序

1. 功能初步实现的小应用 name.java

```
import java.awt.*;
import java.applet.*;

public class name extends java.applet.Applet implements Runnable{
    int width=400;           //小应用的窗口宽度
    int height=200;          //小应用的窗口高度
    int r=60;                //名字旋转轨迹圆的半径
    int d;                   //名字旋转时的偏移量
    double alfa = -0.5;      //旋转步长
    Thread runner;
    Image offscreenImage;
    Graphics offscreenG;

    public void init() {
        offscreenImage = createImage(width,height); //建立一个图像对象
        offscreenG = offscreenImage.getGraphics(); //建立缓冲区
    }

    public void start() {
        if(runner==null){
            runner = new Thread(this);
            runner.start();
        }
    }

    public void stop() {
        if(runner!=null){
            runner.stop();
            runner = null;
        }
    }

    public void run() {
        while(true){
            alfa = alfa + 0.5; //计算旋转角度
            d = (int) Math.ceil(r * Math.sin(alfa * Math.PI/180)); //计算偏移量
            repaint();
            try{
                Thread.sleep(3);
            }
        }
    }
}
```

```

        }catch (InterruptedException e){}
        if (alfa > 360) alfa = 0;    //每当旋转角度达到一周即 360°时,则清零
    }

    public void update(Graphics g) {
        paint(g);
    }

    public void paint (Graphics g) {

        offscreenG.setColor (Color.green);        //设置背景颜色
        offscreenG.fillRect (0,0,width,height);    //画背景
        offscreenG.setColor (Color.black);        //设置线条颜色

```

//下面画“潘”字

```

        offscreenG.drawLine(
            (int)Math.ceil(-80 * Math.cos(alfa * Math.PI/180) + d + width/2),52,
            (int)Math.ceil(-70 * Math.cos(alfa * Math.PI/180) + d + width/2),72);
        offscreenG.drawLine(
            (int)Math.ceil(-80 * Math.cos(alfa * Math.PI/180) + d + width/2),80,
            (int)Math.ceil(-70 * Math.cos(alfa * Math.PI/180) + d + width/2),100);
        offscreenG.drawLine(
            (int)Math.ceil(-80 * Math.cos(alfa * Math.PI/180) + d + width/2),145,
            (int)Math.ceil(-65 * Math.cos(alfa * Math.PI/180) + d + width/2),110);
        offscreenG.drawLine(
            (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) + d + width/2),52,
            (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) + d + width/2),62);
        offscreenG.drawLine(
            (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) + d + width/2),65,
            (int)Math.ceil(-40 * Math.cos(alfa * Math.PI/180) + d + width/2),70);
        offscreenG.drawLine(
            (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) + d + width/2),65,
            (int)Math.ceil(-30 * Math.cos(alfa * Math.PI/180) + d + width/2),70);
        offscreenG.drawLine(
            (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) + d + width/2),75,
            (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) + d + width/2),75);
        offscreenG.drawLine(
            (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) + d + width/2),90,
            (int)Math.ceil(-40 * Math.cos(alfa * Math.PI/180) + d + width/2),80);
        offscreenG.drawLine(
            (int)Math.ceil(-30 * Math.cos(alfa * Math.PI/180) + d + width/2),80,
            (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) + d + width/2),90);
        offscreenG.drawLine(
            (int)Math.ceil(-35 * Math.cos(alfa * Math.PI/180) + d + width/2),65,

```

```

        (int)Math.ceil(-35 * Math.cos(alfa * Math.PI/180) + d + width/2), 95);
    offscreenG.drawLine(
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) + d + width/2), 100,
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) + d + width/2), 100);
    offscreenG.drawLine(
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) + d + width/2), 120,
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) + d + width/2), 120);
    offscreenG.drawLine(
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) + d + width/2), 140,
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) + d + width/2), 140);
    offscreenG.drawLine(
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) + d + width/2), 100,
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) + d + width/2), 140);
    offscreenG.drawLine(
        (int)Math.ceil(-35 * Math.cos(alfa * Math.PI/180) + d + width/2), 100,
        (int)Math.ceil(-35 * Math.cos(alfa * Math.PI/180) + d + width/2), 140);
    offscreenG.drawLine(
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) + d + width/2), 100,
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) + d + width/2), 140);
//以上是画“潘”字
//以下是画“晖”字
    offscreenG.drawLine(
        (int)Math.ceil(10 * Math.cos(alfa * Math.PI/180) + d + width/2), 80,
        (int)Math.ceil(25 * Math.cos(alfa * Math.PI/180) + d + width/2), 80);
    offscreenG.drawLine(
        (int)Math.ceil(10 * Math.cos(alfa * Math.PI/180) + d + width/2), 100,
        (int)Math.ceil(25 * Math.cos(alfa * Math.PI/180) + d + width/2), 100);
    offscreenG.drawLine(
        (int)Math.ceil(10 * Math.cos(alfa * Math.PI/180) + d + width/2), 120,
        (int)Math.ceil(25 * Math.cos(alfa * Math.PI/180) + d + width/2), 120);
    offscreenG.drawLine(
        (int)Math.ceil(10 * Math.cos(alfa * Math.PI/180) + d + width/2), 80,
        (int)Math.ceil(10 * Math.cos(alfa * Math.PI/180) + d + width/2), 120);
    offscreenG.drawLine(
        (int)Math.ceil(25 * Math.cos(alfa * Math.PI/180) + d + width/2), 80,
        (int)Math.ceil(25 * Math.cos(alfa * Math.PI/180) + d + width/2), 120);
    offscreenG.drawLine(
        (int)Math.ceil(35 * Math.cos(alfa * Math.PI/180) + d + width/2), 55,
        (int)Math.ceil(35 * Math.cos(alfa * Math.PI/180) + d + width/2), 65);
    offscreenG.drawLine(
        (int)Math.ceil(35 * Math.cos(alfa * Math.PI/180) + d + width/2), 55,
        (int)Math.ceil(75 * Math.cos(alfa * Math.PI/180) + d + width/2), 55);
    offscreenG.drawLine(

```

```

        (int)Math.ceil(75 * Math.cos(alfa * Math.PI/180) + d + width/2),55,
        (int)Math.ceil(75 * Math.cos(alfa * Math.PI/180) + d + width/2),65);
    offscreenG.drawLine(
        (int)Math.ceil(37 * Math.cos(alfa * Math.PI/180) + d + width/2),70,
        (int)Math.ceil(73 * Math.cos(alfa * Math.PI/180) + d + width/2),70);
    offscreenG.drawLine(
        (int)Math.ceil(39 * Math.cos(alfa * Math.PI/180) + d + width/2),95,
        (int)Math.ceil(71 * Math.cos(alfa * Math.PI/180) + d + width/2),95);
    offscreenG.drawLine(
        (int)Math.ceil(33 * Math.cos(alfa * Math.PI/180) + d + width/2),120,
        (int)Math.ceil(77 * Math.cos(alfa * Math.PI/180) + d + width/2),120);
    offscreenG.drawLine(
        (int)Math.ceil(55 * Math.cos(alfa * Math.PI/180) + d + width/2),70,
        (int)Math.ceil(55 * Math.cos(alfa * Math.PI/180) + d + width/2),145);
    offscreenG.drawLine(
        (int)Math.ceil(55 * Math.cos(alfa * Math.PI/180) + d + width/2),60,
        (int)Math.ceil(39 * Math.cos(alfa * Math.PI/180) + d + width/2),95);
//以上是画“晖”字
    g.drawImage(offscreenImage,0,0,this);    //将缓冲区里的内容画到屏幕上
}
}

```

2. 改进后的 Java 小应用 name2.java

```

import java.awt.*;
import java.applet.*;

public class name2 extends java.applet.Applet implements Runnable{
    int width=400;
    int height=200;
    int r=60;           //名字旋转轨迹圆的半径
    int d;              //名字旋转时的偏移量
    double alfa = -0.5; //旋转步长
    double ratio;       //缩短系数
    Thread runner;
    Image offscreenImage;
    Graphics offscreenG;

    public void init(){
        offscreenImage = createImage(width,height);
        offscreenG = offscreenImage.getGraphics();
    }

    public void start(){
        if(runner == null){
            runner = new Thread(this);

```

```

        runner.start();
    }
}

public void stop() {
    if (runner != null) {
        runner.stop();
        runner = null;
    }
}

public void run() {
    while (true) {
        alfa = alfa + 0.5;
        // 计算每旋转  $\Delta\alpha$  即  $0.5^\circ$  之后的缩短系数
        ratio = 1 - 0.5 * Math.sin(0.5 * alfa * Math.PI / 180);
        d = (int) Math.ceil(r * Math.sin(alfa * Math.PI / 180));
        repaint();
        try {
            Thread.sleep(3);
        } catch (InterruptedException e) {}
        if (alfa > 360) alfa = 0;
    }
}

public void update(Graphics g) {
    paint(g);
}

public void paint(Graphics g) {

    offscreenG.setColor(Color.green);
    offscreenG.fillRect(0, 0, width, height);
    offscreenG.setColor(Color.black);

    //  $\begin{cases} x' = ratio * x_0 \cos\alpha + d + midx \\ y' = y_0 * ratio \end{cases}$ 
    // 以下是画“潘”字
    offscreenG.drawLine(
        (int) Math.ceil(-80 * Math.cos(alfa * Math.PI / 180) * ratio + d + width / 2),
        (int) Math.ceil(52 * ratio),
        (int) Math.ceil(-70 * Math.cos(alfa * Math.PI / 180) * ratio + d + width / 2),
        (int) Math.ceil(72 * ratio));
    offscreenG.drawLine(

```



```

        (int)Math.ceil(-80 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(80 * ratio),
        (int)Math.ceil(-70 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(100 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-80 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(145 * ratio),
        (int)Math.ceil(-65 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(110 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(52 * ratio),
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(62 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(65 * ratio),
        (int)Math.ceil(-40 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(70 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(65 * ratio),
        (int)Math.ceil(-30 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(70 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(75 * ratio),
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(75 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(90 * ratio),
        (int)Math.ceil(-40 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(80 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-30 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(80 * ratio),
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(90 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-35 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(65 * ratio),
        (int)Math.ceil(-35 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),

```

```

        (int)Math.ceil(95 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(100 * ratio),
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(100 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(120 * ratio),
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(120 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(140 * ratio),
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(140 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(100 * ratio),
        (int)Math.ceil(-60 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(140 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-35 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(100 * ratio),
        (int)Math.ceil(-35 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(140 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(100 * ratio),
        (int)Math.ceil(-10 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(140 * ratio));
    //以下是画“晖”字
    offscreenG.drawLine(
        (int)Math.ceil(10 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(80 * ratio),
        (int)Math.ceil(25 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(80 * ratio));
    offscreenG.drawLine(
        (int)Math.ceil(10 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(100 * ratio),
        (int)Math.ceil(25 * Math.cos(alfa * Math.PI/180) * ratio + d + width/2),
        (int)Math.ceil(100 * ratio));
    offscreenG.drawLine(

```