

**课程设计报告**

( 2019--2020 年度第二学期)

# 名 称：面向对象程序设计(Java)课程设计题 目： 小型绘图软件 院 系： 控制与计算机工程学院 班 级： 软件 1802

学 号： 120181080701

学生姓名： 邵博深 指导教师： 周长玉 设计周数： 1 周

成 绩：

日期： 2019 年 12 月 30 日

# 目录

1. [课程设计(综合实验)的目的与要求 1](#_bookmark0)
2. [设计（实验）正文 1](#_bookmark1)
   1. [需求分析 1](#_bookmark2)
      1. [功能需求 1](#_bookmark3)
   2. [系统总体设计 1](#_bookmark4)
      1. [类的设计 1](#_bookmark5)
      2. [文件设计 2](#_bookmark6)
      3. [界面设计 2](#_bookmark7)
   3. [系统实现 2](#_bookmark8)
      1. [开发技术 2](#_bookmark9)
      2. [系统主界面 2](#_bookmark10)
      3. [画图功能的详细设计与实现 3](#_bookmark11)
3. [课程设计总结 6](#_bookmark12)
   1. [总结及体会 6](#_bookmark13)
   2. [难题及解决方案 6](#_bookmark14)

[参考文献 6](#_bookmark15)

## 课程设计(综合实验)的目的与要求

通过课程设计的实践训练，加强学生对 Java 语言基础知识、面向对象的概念、面向对象程序设计方法的理解和掌握，提高学生综合运用所学知识分析问题、解决问题的能力。

具体如下：

1．根据题目要求进行需求分析，确定系统的功能需求、数据需求、性能需求等；

2．根据分析的结果进行软件设计，包括类的设计、数据设计、界面设计等；

3．采用Java 语言实现系统的主要功能；

4．撰写课程设计报告，要求内容完整、格式规范。

## 设计（实验）正文

* 1. 需求分析
     1. 功能需求

系统要实现的功能包括：

（1）可绘制直线、圆形、椭圆、矩形、任意多边形等

（2）可以进行图形的缩放、平移、旋转等操作

* 1. 系统总体设计
     1. 类的设计
        1. AbstractShape 类：抽象父类,所有图形类均要继承该类。包含的主要成员变量和成员方法如表 1

所示。

表 1 AbstractShape 类的规格说明

|  |  |  |
| --- | --- | --- |
| 序号 | 成员变量/成员方法 | 说明 |
| 1 | public abstract void draw(Graphics2D g) | 供所有图形类继承 |
| 2 | public int x1, y1, x2, y2 | 图形的坐标 |
| 3 | public Color color | 画笔颜色 |

* + - 1. ColorPanel 类：表示颜色面板，包含的主要成员变量和成员方法如表 2 所示。

表 2 ColorPanel 类的规格说明

|  |  |  |
| --- | --- | --- |
| 序号 | 成员变量/成员方法 | 说明 |
| 1 | public ColorPanel() | 设计颜色面板 |
| 3 | public static void chooseColor() | 选择当前颜色 |

* + - 1. MyFrame 类：界面类。包含的主要成员变量和成员方法如表 1 所示。

表 3 MyFrame 类的规格说明

|  |  |  |
| --- | --- | --- |
| 序号 | 成员变量/成员方法 | 说明 |
| 1 | MyFrame(String s) | 组装Frame 面板 |
| 2 | DrawPanel() | 画图面板，用来实现画图功能 |
| 3 | MyMenu() | 设置菜单 |
| 4 | MyToolbar() | 设置工具栏 |

* + 1. 文件设计

文件说明：

（1）文件名：image

（2）文件格式：支持.jpg、.png、.gif、.bmp 格式

（3）文件包含的数据项：用于加载到工具栏的所有图片

* + 1. 界面设计

界面采用 java GUI 技术制作，由菜单栏、工具栏、颜色面板和绘图面板共同构成，用户从工具栏里点击相应图片按钮后，可在绘图面板上绘制对应图形。

* 1. 系统实现
     1. 开发技术

主要运用了 Java 绘图技术中的Graphics 类、Graphics2D 类等 。

* + 1. 系统主界面

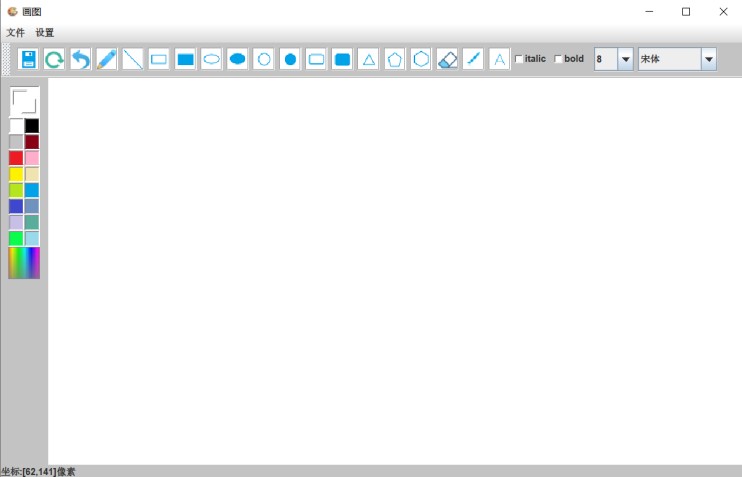
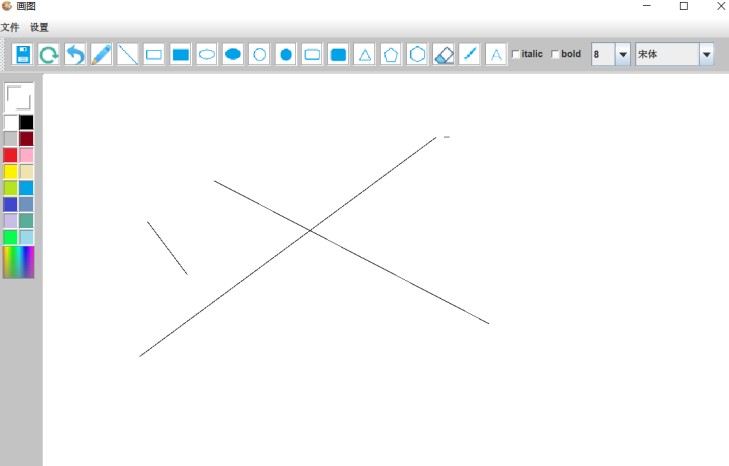
界面如图 1 所示

图 1 系统主界面

组成：菜单栏、工具栏、颜色面板、绘图面板操作方式：鼠标点击、拖拽等方式

* + 1. 画图功能的详细设计与实现

界面如图 2 所示。

（1）画图功能的实现

图 2 画图功能界面

用户点击工具栏的图标后，即可在绘图面板里绘制相应的图形。默认情况下，画笔是任意曲线。

核心代码如下：

1）重写 paintComponent 方法，使得画板每次刷新时可将之前的所有图形重新画出来。

@Override

public void paintComponent(Graphics g) { super.paintComponent(g);

Graphics2D g2d = (Graphics2D) g; // 定义画板

int j = 0;

while (j <= index) { //画笔使用次数 draw(g2d, itemList[j]);

// itemList [j]是 Abstractshape 类的数组，用来存储记录每一次操作所触发的图形

j++;

}

}

2）将画笔传入到各个子类中，用来完成各自的绘图 void draw(Graphics2D g2d, AbstractShape abstractShape) {

abstractShape.draw(g2d);

}

3）根据被选中工具栏创建对应的图形类 void createNewGraphics() {

……

//currentChoice 标记被选中的图形的序号

switch (currentChoice) { case 0:

itemList[index] = new Images(); break;

……

case 16:

itemList[index] = new Rubber(); break;

case 17:

itemList[index] = new Brush(); break;

case 18:

itemList[index] = new Text(); String input;

input = JOptionPane.showInputDialog("请输入文字");

itemList[index].s = input; itemList[index].fontSize = fSize; itemList[index].fontName = fontName; itemList[index].italic = italic; itemList[index].blodtype = blodtype; break;

default:

}

itemList[index].color = color; itemList[index].width = stroke;

}

（2）撤销功能的实现

单击撤销工具后，标记图形发生次数的 index 递减，然后通过drawingArea.repaint()重新画，从而实现撤销功能

void undo() {

index--;

if (index >= 0) {

if (currentChoice == 3 || currentChoice == 16 || currentChoice == 17) { index -= itemList[index].length;

} else {

index--;

}

drawingArea.repaint();

}

index++; drawingArea.createNewGraphics();

（4）颜色面板的设计界面如图三

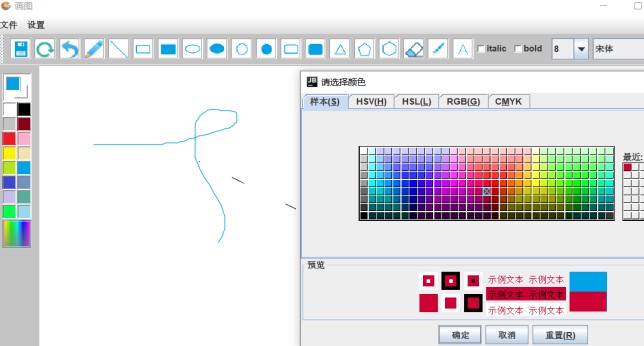


图 3

用户既可以从左边的颜色图标里直接选取画笔的颜色，也可以从最下方的调色板里挑选更为丰富的色彩。

核心代码如下

* + - 1. 循环添加颜色块

for (int i = 0; i < 24; i++) {

JButton bt3 = new JButton(); bt3.setOpaque(true);

bt3.setBackground(colors[i]); //colors[i]里存储在每个颜色块的 RGB 数值

bt3.setPreferredSize(new Dimension(20, 20)); bt3.setBorder(bb);

bt3.addActionListener(e -> {

// 拿到被选中按钮的对象

JButton jbt = (JButton) e.getSource();

// 拿到被选中按钮的背景颜色

Color c = jbt.getBackground();

// 把背景颜色复制给 WIndowStart 中的颜色属性 MyFrame.color = c;

// 把左面板中的按钮颜色设置成选中按钮的背景颜色

bt.setBackground(c); MyFrame.itemList[MyFrame.index].color = c;

});

right.add(bt3);

}

* + - 1. 加载调色板

public static void chooseColor() {

MyFrame.color = JColorChooser.showDialog(null, "请选择颜色", MyFrame.color);

// 把左面板中的按钮颜色设置成选中按钮的背景颜色 ColorPanel.bt.setBackground(MyFrame.color); MyFrame.itemList[MyFrame.index].color = MyFrame.color;

## 课程设计总结

* 1. 总结及体会

总的来说，通过绘图软件的设计，我对 java 的 GUI 有了更深刻的理解，熟练地使用了 Graphics 类、

Graphics2D 等绘图类，对 java 的监听器应用起来也更加的娴熟。这对用 java 写其他的图形界面设计很有帮助。

但是在这次 Java 课设中深刻地意识到了自己的水平很不足，尤其是在处理图像的平移、旋转等像素问题的时候，发现自己很难完成这些功能。对 Java 的学习不能懈怠，还要再接再厉！

* 1. 难题及解决方案

遇到的难题主要可归为“两变”，第一变是图形类的变换，第二变是像素坐标的变换

1.各种图形类之间繁杂的转换，使得程序经常报错

为了解决这一问题，总结了所有图形的共性后，先写了一个图形抽象类，然后另所有图形类都继承它，从而使得图形变更操作可以以数组的形式进行各自的实例化，也便于进行撤销操作，降低了设计的复杂度

2.图形的平移、放缩、旋转等像素坐标变换问题

查阅了很多书籍，发现书上的例程都是对图片的平移、放缩、旋转，而不是对刚画好的图形的像素坐标的操作，没能够迁移过来。

然后又去网上查阅相关问题，看到说 maven 里的 imageio 类可以实现对图形像素坐标的操作，就在

IDEA 里加载 maven，然而配置失败，调用不了 maven 里的 imageio 类。

## 参考文献

* + 1. 王素琴.Java 语言程序设计实用教程[M].北京：中国电力出版社，2017.
    2. 李源.Java 语言程序设计经典 300 例[M].北京：电子工业出版社，2013.
    3. 陈丹丹.Java 开发宝典[M].北京：机械工业出版社，2012.
    4. 李兴华.Java 核心技术精讲[M].北京：清华大学出版社，2013.

**意见与建议**

教材：

1、 结构上编排的并不合理，第五章 Java GUI 部分长篇累牍,然而其他几章却又显得短小精悍。

2、 内容上样例很好，比较全面，但所涉及的知识体系比较单薄，可以更加扩充

教学：

1、 进度过快，有点囫囵吞枣了，吃不消

2、 作业布置的比较多有的时候跟不上节奏