



Java 项目开发 与毕业设计指导

朱福喜 黄昊 编著



- 专家点评案例，提升技能，圆满完成学业
- 项目案例经典，难度适中，适合毕业设计



清华大学出版社



Java

项目开发与毕业设计指导

朱福喜 黄昊 编著

清华大学出版社

北 京

内 容 简 介

作者长年坚守在教学科研一线,在多年指导研究生、本科生毕业设计与项目开发方面积累了丰富的经验,本书可引导拥有 Java 书本知识、无实战经验的读者开发出满足企业应用需求的项目,使学生的理论、实践与技能得到全面提升。

书中精选 12 个案例,包括俄罗斯方块单人游戏、俄罗斯方块双人对战游戏、五子棋双人对战游戏、五子棋人机对战游戏、多媒体展示系统、野人和传教士过河问题的实现、蚁群算法的模拟实现、SchoolEbay 的实现、Ajax 技术在網上教学平台的应用、简单编译器的实现、基于 RMI 分布的计算实例、基于 Agent 分布的计算实例。

光盘中提供所有案例的源代码和数据库,案例后附有专家对此案例功能缺陷及待改进方面的点评,可以把点评和光盘文件结合起来快速高效地进行二次开发,缩短毕设时间。

本书适合高等院校计算机及相关专业的本科毕业生阅读,以帮助毕业生更好地完成毕业设计。对于正在学习 Java 课程的高校学生和自学者,也是一本极具价值的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Java 项目开发与毕业设计指导/朱福喜,黄昊编著. —北京:清华大学出版社,2008.6

ISBN 978-7-302-17647-3

I. J… II. ①朱… ②黄… III. JAVA 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2008)第 073315 号

责任编辑:夏非彼 张楠

装帧设计:图格新知

责任校对:贾淑媛

责任印制:李红英

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:清华大学印刷厂

经 销:全国新华书店

开 本:185×260 印 张:22 字 数:535 千字

附光盘 1 张

版 次:2008 年 6 月第 1 版

印 次:2008 年 6 月第 1 次印刷

印 数:1~5000

定 价:39.50 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:027311-01



前言

许多学生在即将读完大学时只掌握一些简单的Java知识，没有从事Java项目实战的经验，碰到问题时不知如何下手。因此如何掌握Java应用最广泛最频繁的知识、很快切入到一个实际应用的场合呢？碰到毕业设计的题目或自选一些题目时如何利用最快捷的方法解决它呢？这就需要有一个好的引导者，引导他们与IT界的企业应用紧密结合，在实际的环境中学习和掌握实际案例，融理论、实践与技能为一体，学以致用。这样既完成了毕业设计，又能掌握Java的深入技术和技能，还锻炼了实际问题的求解能力和软件应用的设计能力。

本书正是本着上述目的而编写的，在Java游戏、人工智能、Java EE以及Java高级应用等领域中精选出12个大小适合于大学毕业设计程度的应用实例。这些经典实例紧紧围绕Java EE的各种核心技术和人工智能中的经典求解算法，从基础知识入手层层分析和分步解析，既保证具有一定的实用性和趣味性，也确保具有一定的广度和深度，在难易程度上遵循由浅入深、循序渐进的原则。其内容主要覆盖了Java EE中比较基础但又非常流行的技术，如JavaBean、Swing、SWT、Servlet、Socket、JSP、RMI、JDBC、MVC设计模式、Ajax以及Java的多媒体技术，同时也涉及一些常用的人工智能中的求解技术、Java游戏技术以及Internet的相关技术。在应用这些技术的同时还涉及一些数据库技术、应用服务器平台以及更高级的平台技术。这些实例不仅能帮助初学者立即进入设计环境，而且许多实例稍加修改或扩充即可运用在实际项目中。这些实例是学习和掌握Java EE软件技术的最佳途径。

本书适合于初学Java、具有一定Java基础的学生和开发人员作为自学实战的参考书，尤其适合计算机相关专业的本科生作为毕业设计时的参考书。

附赠光盘中提供了光盘的使用方法、案例源代码以及光盘中文件包与正文各章节的对应关系。

在本书的编写过程中杨文涛、杨宝奎、李晓清、王立明、唐晓军、杨敏、余永宏、黄方胜、黄超给予了作者很大帮助，协助调试和验证了本书的所有应用实例，他们中有的是国际程序设计大赛的获奖者，有的现在已是国内外IT公司的开发精英，他们为本书的编写付出了辛勤的劳动，同时本书的编写也得到了清华大学出版社夏非彼老师的大力支持，在此谨向他们表示衷心感谢。

由于作者水平有限，书中难免存在疏漏和错误之处，恳请专家和广大读者批评指正。在学习过程中，遇到疑难问题，可以通过以下方式与我们联系：booksaga@126.com，也可以登录图格新知网站<http://www.booksaga.com>留言，我们将在第一时间给予答复！

2008年樱花盛开的时节于武昌珞珈山

编者

本书适合初学Java，具有一定Java基础的读者阅读。本书共分12章，第1章介绍Java的发展历史、特点、应用及安装；第2章介绍Java的语法基础；第3章介绍Java的面向对象编程；第4章介绍Java的异常处理；第5章介绍Java的集合框架；第6章介绍Java的字符串处理；第7章介绍Java的日期和时间处理；第8章介绍Java的线程和并发编程；第9章介绍Java的数据库编程；第10章介绍Java的图形用户界面编程；第11章介绍Java的网络编程；第12章介绍Java的XML编程。

目 录

第1章 俄罗斯方块单人游戏的实现	1
1.1 功能描述	1
1.2 实现机制	2
1.2.1 Java GUI编程	2
1.2.2 Java多线程机制	10
1.3 设计方案	12
1.3.1 游戏总体构思	12
1.3.2 基本类设计	16
1.4 程序实现	30
1.4.1 菜单类的实现	30
1.4.2 方格类的实现	36
1.4.3 方块类的实现	37
1.4.4 主类的实现	41
1.4.5 控制面板类的实现	46
1.4.6 游戏画布类的实现	52
1.5 点评	55
第2章 俄罗斯方块双人对战游戏的实现	56
2.1 功能描述	56
2.2 实现机制	57
2.2.1 Java与网络通信	57
2.2.2 Socket介绍	58
2.2.3 使用对象流传递数据	62
2.3 设计方案	64
2.3.1 网络通信的设计方案	64
2.3.2 Socket的数据传输	64
2.3.3 程序的类结构	65
2.4 程序实现	66
2.4.1 主类的实现	66
2.4.2 俄罗斯方块类的实现	69
2.4.3 游戏画布类的实现	70
2.4.4 菜单类的实现	72
2.4.5 对象数据传输类的实现	75



2.4.6 游戏的运行	76
2.5 点评	78
第3章 五子棋双人对战游戏的实现	79
3.1 功能描述	79
3.2 实现机制	80
3.2.1 界面设计	80
3.2.2 游戏控制	81
3.2.3 网络结构与数据传输	81
3.3 设计方案	82
3.3.1 游戏总体构思	83
3.3.2 客户端设计	83
3.3.3 服务器端设计	89
3.3.4 程序的类结构	94
3.4 程序设计	95
3.4.1 客户端程序实现	95
3.4.2 服务器端程序实现	103
3.5 运行效果	109
3.6 点评	110
第4章 多媒体展示系统——网络游鱼	111
4.1 功能描述	111
4.2 实现机制	111
4.2.1 UDP数据通信	111
4.2.2 利用Datagram方式实现通信	112
4.2.3 IP多播	113
4.2.4 动画实现	115
4.2.5 音频处理	116
4.2.6 LinkedList类	117
4.2.7 UML建模	117
4.3 设计方案	118
4.3.1 需求分析	118
4.3.2 系统建模	119
4.3.3 基本类的设计	124
4.4 程序设计	132
4.4.1 服务器端程序设计	132
4.4.2 客户端程序设计	137
4.5 运行效果	143



4.6 点评	145
第5章 野人和传教士过河问题的实现	146
5.1 功能描述	146
5.2 实现机制	147
5.2.1 状态空间法	147
5.2.2 搜索算法	148
5.3 设计方案	150
5.3.1 问题的形式化描述	150
5.3.2 主要数据结构的定义	153
5.3.3 系统的总体构思	154
5.4 程序实现	159
5.4.1 A*算法搜索模块	159
5.4.2 动画演示模块	165
5.5 点评	171
第6章 五子棋人机对战游戏的实现	172
6.1 功能描述	172
6.2 实现机制	173
6.2.1 Java GUI编程	173
6.2.2 定时更新信息	178
6.3 设计方案	179
6.3.1 游戏总体设计	179
6.3.2 博弈策略	179
6.3.3 基本类的设计	184
6.4 程序实现	188
6.4.1 窗口初始化及相应图片的绘制	188
6.4.2 棋盘的绘制及鼠标事件的监听	189
6.4.3 左侧标签的实现	194
6.4.4 右侧按钮的实现	195
6.4.5 极大极小值搜索策略的实现	197
6.4.6 计算机落子的实现	202
6.4.7 定时更新的实现	203
6.5 点评	204
第7章 蚁群算法的模拟实现	205
7.1 功能描述	205
7.2 实现机制	206



7.2.1	SWT	206
7.2.2	蚁群算法	206
7.3	设计方案	209
7.3.1	总体框架的设计	210
7.3.2	演示界面的设计	210
7.3.3	算法的设计	211
7.3.4	基本类的设计	213
7.4	程序实现	218
7.4.1	主界面的实现	218
7.4.2	算法类的实现	223
7.4.3	效果与效率	229
7.4.4	参数的设置	231
7.5	点评	231
第8章	SchoolEbay的实现	232
8.1	功能描述	232
8.2	系统分析	232
8.2.1	需求分析	233
8.2.2	实现机制	237
8.3	系统设计	244
8.3.1	数据库设计	244
8.3.2	系统总体描述	248
8.3.3	详细设计	249
8.4	系统实现	252
8.4.1	模块功能的实现	252
8.4.2	编码转换、HTML处理和加密的实现	258
8.5	系统配置与运行	260
8.6	点评	261
第9章	Ajax技术在网 上教学平台的应用	262
9.1	功能描述	262
9.2	系统分析与设计	263
9.2.1	自主学习平台	263
9.2.2	实现机制	266
9.3	实现方案	268
9.3.1	Ajax实现级联菜单	268
9.3.2	Ajax+XML+XSL实现页面数据格式化显示	270
9.3.3	Ajax实现无刷新数据搜索	272



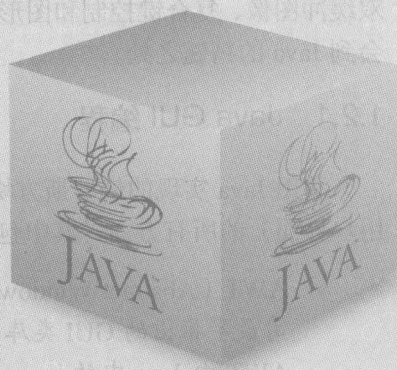
9.4 程序实现	275
9.4.1 级联菜单的实现	275
9.4.2 数据显示的实现	277
9.4.3 无刷新搜索的实现	278
9.5 点评	280
第10章 简单编译器的实现	281
10.1 功能描述	281
10.2 系统分析与设计	281
10.2.1 系统分析	282
10.2.2 系统设计	283
10.3 实现机制	283
10.3.1 StringTokenizer类	283
10.3.2 Iterator接口	285
10.3.3 栈和队列	286
10.4 程序框架	288
10.4.1 表达式的实现类	288
10.4.2 队列类和栈类	289
10.4.3 主类	291
10.5 程序设计	292
10.5.1 表达式分析用到的类	292
10.5.2 表达式转换的实现	295
10.5.3 表达式计算的完整实现	299
10.6 点评	301
第11章 基于RMI分布的计算实例	302
11.1 功能描述	302
11.2 系统分析和设计	302
11.3 实现机制	303
11.3.1 RMI的简介	303
11.3.2 RMI的使用	305
11.4 程序框架	307
11.5 程序实现	310
11.5.1 客户和服务配置的实现	311
11.5.2 远程接口和远程方法的实现	312
11.5.3 客户程序的实现	314
11.5.4 RMI服务器的实现	319
11.5.5 计算过程	320



11.6 点评	323
第12章 基于Agent分布的计算实例	324
12.1 功能描述	324
12.2 系统分析和设计	325
12.3 实现机制	326
12.3.1 Agent简介	326
12.3.2 Aglet简介	327
12.3.3 Aglet的使用方法	331
12.4 程序框架	333
12.5 程序实现	334
12.5.1 Matrix类	334
12.5.2 Calculator类	337
12.5.3 计算过程	338
12.6 点评	341

第1章

俄罗斯方块单人游戏的实现



在当前的游戏开发领域中，Java 有自己的独到之处，用 Java 编写的代码简单明了、移植性强，这些都是 Java 开发游戏的优点；另外，最新的二维三维图形技术、搜索技术、碰撞探测、BeanShell 的游戏脚本以及多方游戏引擎的生成都是 Java 开发游戏的优势。

1.1 功能描述

本章着重讨论基于 Java 技术实现俄罗斯方块游戏。俄罗斯方块是一个非常复杂但比较有趣并且耐玩的游戏，相信很多人对俄罗斯方块游戏并不陌生，如图 1-1 所示，游戏中共有 7 种方块，每种方块由 4 个方格组成。

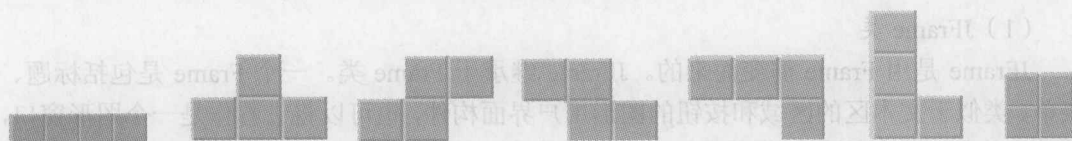


图 1-1 俄罗斯方块状态图

这 7 种方块可以旋转 90° 、 180° 、 270° ，因此每种方块有 4 种状态，7 种方块总共有 28 种状态。这 28 种状态的方块随机产生，自由落下，落下时可由玩家用上、下、左、右控制键控制翻转和移动，以便以玩家所需要的形态和位置落下。如果落下时，方块的方格能填满某一行，则这一行可消去。消去一行后，游戏可给玩家加分，若由存在空格的方块填满整个窗口，则游戏失败。



1.2 实现机制

游戏块的运行采用多线程 (Thread) 机制, 界面利用 JavaSwing 提供的图形类实现。游戏采用开源开发环境 Eclipse。

在整个游戏的开发过程中将涉及到 Java 基本知识的大部分内容, 如事件、线程、动画、双缓冲图像、复合键控制和图形界面等知识。可以说 Java 游戏编程能够让我们真正全面体会到 Java 的精髓之处。

1.2.1 Java GUI 编程

利用 Java 实现的俄罗斯方块游戏几乎能用到 Java GUI (Graphical User Interface, 图形用户界面) 的所有方面, 其中包括 Sun 公司为 Java 提供的两个图形界面类库。

- AWT (Abstract Windowing Toolkit), 中文译为抽象窗口工具包, AWT 是 Sun 公司最早推出的 GUI 类库, 其中提供了建立和设置 Java 的图形用户界面的基本工具。AWT 由 Java 中的 java.awt 包提供, 里面包含了许多可用来建立与平台无关的图形用户界面 (GUI) 的类, 这些类被称为构件 (Components)。
- Swing 包是 Java Foundation Classes (JFC) 的一部分, 与 AWT 的重量级构件不同, Swing 中大部分是轻量级构件。正是这个原因, Swing 几乎无所不能, 不但有各式各样先进的构件, 而且更为美观易用。所以一开始使用 AWT 的程序员很快就转向使用 Swing 了。

在本章俄罗斯方块游戏的实现中, 主要采用 Swing 类库。

1. 基本窗口类

图形用户界面的程序效果类似于 Windows 系统下运行的程序。在俄罗斯方块游戏中用到 JFrame、JWindow、JPanel 和 JDialog 等几种窗口类。

(1) JFrame 类

JFrame 是由 Frame 演变而来的。JFrame 继承于 Frame 类。一个 Frame 是包括标题、菜单、类似于文本区的区域和按钮的图形用户界面构件, 也可以将它看作是一个图形窗口, 其外观依赖于所使用的操作系统。Frame 包含在一个方框内, 其大小可以伸缩。它通常使用 BorderLayout 进行布局。Frame 在默认的情况下是不可见的。如果一个 Frame 不再需要时, 应调用 dispose 方法撤销它。

JFrame 类与 Frame 类略微不兼容。与其他所有 JFC/Swing 顶层容器一样, JFrame 包含一个 JRootPane 作为其惟一的子容器, 即 ContentPane, 该内容面板直接或间接地容纳其他可视构件。为了方便地使用 add 方法及 add 方法的其他版本, JFrame 类重写了 remove 和 setLayout 方法。

ContentPane 始终是非 NULL 的。若试图将其设置为 NULL 会导致 JFrame 抛出异常。默认的内容窗格上会设置有 BorderLayout 管理器。

与 Frame 不同, 当用户试图关闭窗口时 JFrame 知道如何进行响应。用户关闭窗口时默

认的行为仅仅是简单地隐藏 JFrame。若要更改默认的行为可调用方法 `setDefaultCloseOperation(int)`。若要使 JFrame 的行为与 Frame 实例相同可使用 `setDefaultCloseOperation(WindowConstants.DO_NOTHING_ON_CLOSE)` 方法。

JFrame 类的构造方法有如下几个版本。

- `JFrame()`: 构造一个初始时不可见的新窗口。
- `JFrame (GraphicsConfiguration gc)`: 以屏幕设备的指定 `GraphicsConfiguration` 和空白标题创建一个 JFrame。
- `JFrame (String title)`: 创建一个新的、初始不可见的、具有指定标题的 JFrame。
- `JFrame (String title, GraphicsConfiguration gc)`: 创建一个具有指定标题和指定屏幕设备的 `GraphicsConfiguration` 的 JFrame。

(2) JWindow 类

JWindow 类继承于 Window 类。Window 类表示一个没有边界和菜单的最高层的窗口，默认的是使用 `BorderLayout` 布局，在这个布局中可以包含 AWT 的其他构件。窗口初始化时也是不可见的。一个基本的 GUI 程序应该能够在屏幕上显示信息，而且至少要包含一个 GUI 构件使得用户能够轻松地响应用户的要求，这就需要用到事件响应类。

JWindow 是一个容器，可以显示在用户桌面上的任何位置。它没有标题栏、窗口管理按钮或者其他与 JFrame 关联的修饰。JWindow 组件包含一个 `JRootPane` 作为其仅有的子组件。`ContentPane` 为 JWindow 类的所有子窗口的父窗口。

由于一般并不直接使用 JWindow 类的构造方法创建窗口，所以在此只作简单介绍。

(3) JPanel 类

JPanel 类是一个经常使用的中间容器，它同个面板 (Panel) 一样是一个可容纳其他构件的类属容器 (Generic Container)。一个面板可以享有自己的布局管理器，并且不同的面板可以使用另一个布局管理器将它们组合在一起，并容纳在一个 JFrame、JApplet 或其他面板之中。面板类继承于 Container 类并且继承它的所有方法。其中 `setLayout` 和 `add` 是最常用的方法。

在默认情况下 JPanel 除了背景颜色外并不绘制任何东西，也没有边框，但很容易为它设置边框和绘制特性，默认的状态是透明的，默认的布局是 `FlowLayout`。

JPanel 类的构造方法有如下几个版本。

- `JPanel()`: 创建具有双缓冲和流布局的 JPanel。
- `JPanel(boolean isDoubleBuffered)`: 创建具有 `FlowLayout` 布局和指定缓冲策略的 JPanel。
- `JPanel(LayoutManager layout)`: 创建具有指定布局管理器的 JPanel。
- `JPanel(LayoutManager layout, boolean isDoubleBuffered)`: 创建具有指定布局管理器和缓冲策略的 JPanel。

(4) JDialog 类

JDialog 类是 Dialog 类的子类，是一个 Frame 或另一个对话框拥有的窗口，如果它的父窗口消失，它也将随之撤销。Dialog 继承于 Window 类，因而继承了 Window 类的 `setVisible`



和 pack 等方法。Dialog 类具有两种类型的对话框。

- 模态对话框 (Modal Dialog): 这种对话框阻塞它的父对象的输入, 并且必须在它的父对象再次获得输入之前被关闭或自动消失。
- 非模态对话框 (Non-modal Dialog): 这种对话框并不阻塞父对象的输入, 它可以与父对象并存, 除非特别声明, 否则一般的对话框是非模态的。

JDialog 类也分为模态和非模态的, 并且只能在 Dialog 和 Frame 的基础上创建。若用 JDialog 作为对话框, 则必须实现对话框中的每一个构件。可利用 JOptionPane 类显示一段文字或作一些选择, 它包括一些静态方法 showxxxDialog, 并可产生 4 种简单的对话框。

JDialog 类的构造方法有如下几个版本。

- JDialog(): 创建一个没有标题并且没有指定 Frame 拥有者的无模式对话框。
- JDialog(Dialog owner): 创建一个没有标题但将指定的 Dialog 作为其拥有者的无模式对话框。
- JDialog(Dialog owner, boolean modal): 创建一个具有指定拥有者和模式的对话框。
- JDialog(Dialog owner, String title): 创建一个具有指定标题和指定拥有者的无模式对话框。
- JDialog(Dialog owner, String title, boolean modal): 创建一个具有指定标题、模式和指定拥有者的对话框。
- JDialog(Dialog owner, String title, boolean modal, GraphicsConfiguration gc): 创建一个具有指定标题、拥有者、模式和 GraphicsConfiguration 的对话框。

JDialog 的构造方法还具有将上述方法中的 owner 对象的类型改为 JFrame 和 JWindow 的相应各个版本。

- JDialog(Frame owner): 创建一个没有标题但将指定的 Frame 作为其拥有者的无模式对话框。
- JDialog(Window owner): 创建一个具有指定的 Window 类型的拥有者和空标题的无模式对话框。

2. 原子构件

俄罗斯方块游戏中用到的原子构件很多, 这些构件的用法比较简单, 下面只对俄罗斯方块游戏中用到的原子构件进行简单介绍。

(1) 标签类 JLabel

JLabel 构件上可显示文字和图像并能指定两者的位置。标签不对输入事件作出反应。因此, 它无法获得键盘焦点。

JLabel 对象可以显示文本、图像或同时显示二者。可以通过设置垂直和水平对齐方式指定标签显示区中的标签内容在何处对齐。在默认情况下标签在其显示区内垂直居中对齐, 且只显示文本的标签是开始边对齐; 显示图像的标签则是水平居中对齐。

JLabel 还可以指定文本相对于图像的位置。在默认情况下, 文本位于图像的结尾边上,

文本和图像都垂直对齐。根据标签的 `ComponentOrientation` 属性值可确定其开始边和结尾边。目前默认的 `ComponentOrientation` 设置将开始边映射到左边, 将结尾边映射到右边。

最后, 还可以使用 `setIconTextGap` 方法指定文本和图像之间应该出现多少像素。在默认情况下为 4 个像素。

`JLabel` 类的构造方法有如下几个版本。

- `JLabel()`: 创建无图像并且其标题为空字符串的 `JLabel`。
- `JLabel(Icon image)`: 创建具有指定图像的 `JLabel` 实例。
- `JLabel(Icon image, int horizontalAlignment)`: 创建具有指定图像和水平对齐方式的 `JLabel` 实例。
- `JLabel(String text)`: 创建具有指定文本的 `JLabel` 实例。
- `JLabel(String text, Icon icon, int horizontalAlignment)`: 创建具有指定文本、图像和水平对齐方式的 `JLabel` 实例。
- `JLabel(String text, int horizontalAlignment)`: 创建具有指定文本和水平对齐方式的 `JLabel` 实例。

(2) 按钮类 `JButton`

`JButton` 类继承于 `AbstractButton` 类, `AbstractButton` 类定义了所有按钮的共同属性, 例如都可以显示文本和图片并设置文本相对于图片的位置。

`JButton` 类相对 `AbstractButton` 类来说增加了一些功能, 例如可以把一个 `JButton` 类的对象设置成默认按钮。在顶层容器中, 最多只有一个按钮被设置成默认的, 默认按钮通常是高亮显示的, 当顶层容器获得了键盘焦点且用户按下回车键时就会触发默认按钮的 `click` 事件。

`JButton` 类的构造方法具有如下 5 个版本。

- `JButton()`: 创建不带有设置文本或图标按钮。
- `JButton(Action a)`: 创建一个按钮, 其属性从所提供的 `Action` 对象中获取。
- `JButton(Icon icon)`: 创建一个带图标的按钮。
- `JButton(String text)`: 创建一个带文本的按钮。
- `JButton(String text, Icon icon)`: 创建一个带初始文本和图标的按钮。

(3) `JColorChooser` 类

`JColorChooser` 类可以让用户自己手工选择色彩, 它提供一个用于允许用户操作和选择颜色的控制面板, 并提供以下三个级别的 API:

- 显示具有模式颜色选择器的对话框并返回用户所选颜色的静态方法。
- 创建颜色选择器对话框的静态方法, 可以指定当用户按下其中一个对话框按钮时要调用的 `ActionListener`。
- 能直接在任何容器中创建 `JColorChooser` 窗格实例。可以添加 `PropertyChange` 监听器以检测当前“颜色”属性何时更改。

`JColorChooser` 类的构造方法有如下三个版本。



- `JColorChooser()`: 创建初始颜色为白色的颜色选择器窗格。
- `JColorChooser(Color initialColor)`: 创建具有指定初始颜色的颜色选择器窗格。
- `JColorChooser(ColorSelectionModel model)`: 创建具有指定 `ColorSelectionModel` 的颜色选择器窗格。

(4) 文本域 `JTextField`

`JTextField` 类可以用于输入或显示字符串,即允许编辑单行文本。`JTextField` 是 `JTextComponent` 类的一个子类,`JTextComponent` 类又是 `JComponent` 的一个子类。`JTextField` 与 `java.awt.TextField` 应具有源代码兼容性,当光标移动到文本域并且按回车键时,它触发一个 `ActionEvent` 事件。`JTextField` 类的构造方法有如下 5 个版本。

- `JTextField()`: 构造一个 `TextField` 对象。
- `JTextField(Document doc, String text, int columns)`: 构造一个 `JTextField`, 它使用给定的文本存储模型和给定的列数。
- `JTextField(int columns)`: 构造一个具有指定列数的空 `TextField`。
- `JTextField(String text)`: 构造一个用指定文本初始化的 `TextField`。
- `JTextField(String text, int columns)`: 构造一个用指定文本和列初始化的 `TextField`。

3. 菜单类

对一个大的 GUI 程序而言通常包含各种菜单,俄罗斯方块游戏中也包含多种菜单,如 `JMenu`、`JMenuItem`、`JMenuBar` 等,Java 为这些菜单构件提供了相应的类。

(1) `JMenu`

一个菜单 (`JMenu`) 由多个菜单项组成并具有一个标题,这个标题出现在菜单条上,当单击这个标题时这些菜单项立即弹出。选择一个菜单项就可以引起一个动作事件。多个菜单又可以组合成一个新的菜单或菜单条,增加在最顶层的窗口 (`Frame` 或 `JFrame`) 上。

菜单的实现是一个包含 `JMenuItem` 的弹出窗口,用户选择 `JMenuBar` 上的项时会显示该 `JMenuItem`。除 `JMenuItem` 之外,`JMenu` 还可以包含 `JSeparator`。

菜单在本质上是带有关联 `JPopupMenu` 的按钮。当单击按钮时就会显示 `JPopupMenu`。如果按钮位于 `JMenuBar` 上则该菜单为顶层窗口。如果按钮是另一个菜单项,则 `JPopupMenu` 就是右拉菜单。

菜单中的菜单项也可以是菜单,可使用它自身的 `add` 方法来增加菜单项或菜单对象。

(2) `JMenuItem`

`JMenuItem` 是菜单中的项的实现。菜单项在本质上是位于列表中的按钮。当用户单击按钮时则执行与菜单项关联的操作。

(3) `JMenuBar`

`JMenuBar` 是菜单条的实现。一个菜单条可包含多个菜单,通过 `add` 方法可增加这些菜单对象。菜单条可通过 `JFrame` 的 `setJMenuBar` 方法加入到一个 `JFrame` 中。

将 `JMenu` 对象添加到菜单条以构造菜单。当用户选择 `JMenu` 对象时就会显示其关联的 `JPopupMenu`,并允许用户选择其上的某一个 `JMenuItem` 对象。