

<<老兵新传>>

图书基本信息

书名：<<老兵新传>>

13位ISBN编号：9787302283102

10位ISBN编号：7302283109

出版时间：2012-8

出版时间：清华大学出版社

作者：张宁

页数：733

字数：1350000

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<老兵新传>>

前言

<<老兵新传>>

内容概要

visual basic作为一种windows软件开发工具，以简单易用和胜任快速开发著称，但visual basic系统本身提供的功能比较少，很多重要的软件功能不能直接实现。

《老兵新传：visual basic核心编程及通用模块开发》正是介绍如何弥补它的这种不足、增强vb程序功能的书籍，但增强功能的代码仍不失简单易用的特点，使visual basic成为既强大又好用的编程工具。

书中介绍了使用visual basic语言进行win32核心编程和高级编程的技术，并面向visual basic的开发实践和针对编程中的常见问题，编写了调用简单、运行独立、能胜任大型软件开发的可重用的代码模块，不仅大大增强visual basic的功能，而且减轻编程的工作量，并使程序代码更为简洁。

《老兵新传：visual basic核心编程及通用模块开发》属于visual basic高级编程的书籍，适合有一定visual basic基础并想进一步提高vb开发技能的读者，可作为使用visual basic语言编程的软件开发人员、科研人员和广大vb爱好者的提高技能的读物。由于书中介绍了大量有关windows运行机理和win32编程的技术，因此对使用其他语言编程的windows程序设计人员也有一定参考价值。

<<老兵新传>>

作者简介

张宁，南开大学生物信息学博士和博士后，现任职于天津大学精密仪器与光电子工程学院，多年来一直从事生物医学信息相关的数据分析、数据库、软件开发等工作。

精通Visual

Basic、C语言、Windows核心编程等，有16年编程和软件开发的实践经验，积累了大量程序设计方法和编程技巧，在国内外高水平刊物发表论文28篇，多次在各类编程竞赛中获奖。

<<老兵新传>>

书籍目录

- 《老兵新传：visual basic核心编程及通用模块开发》
- 第一篇 高级技术篇
- 第1章 万丈高楼平地起，一劳永逸 打地基——知识准备
- 1.1 使用windows api函数
 - 1.1.1 api函数的声明
 - 1.1.2 使用api函数的注意事项
 - 1.2 模块化编程
 - 1.2.1 我能在一天之内做出一个软件吗
 - 1.2.2 用标准模块还是类模块
 - 1.2.3 怎样创建类模块
 - 1.3 编程风格
 - 1.3.1 变量在使用前一定要定义
 - 1.3.2 尽量不要用variant数据类型
 - 1.3.3 代码的缩进和换行
 - 1.3.4 避免重复用对象名称做一串调用
 - 1.3.5 boolean型条件表达式的写法
 - 1.3.6 字符串为空串的判断
 - 1.3.7 能用常量就不要用函数求值
 - 1.3.8 数组下标从1开始
 - 1.3.9 用optional选项定义函数的参数
 - 1.3.10 不能滥用goto语句不代表不能用goto语句
 - 1.3.11 尽量少用activex控件
 - 1.4 条件编译
 - 1.4.1 什么是条件编译
 - 1.4.2 条件编译有什么用
 - 1.5 获取资料
 - 1.5.1 msdn
 - 1.5.2 借助visual c++的头文件
 - 1.5.3 其他资料
- 第2章 这个可以有——在vb中使用指针
- 2.1 在vb中使用指针变量
 - 2.1.1 什么是指针
 - 2.1.2 如何实现指针变量
 - 2.2 varptr, strptr, objptr函数和addressof运算符
 - 2.2.1 内联类型和指针类型
 - 2.2.2 varptr, strptr和objptr函数
 - 2.2.3 addressof运算符
 - 2.3 copymemory函数
 - 2.3.1 copymemory函数的声明和功能
 - 2.3.2 copymemory函数用于内联类型的数据传递
 - 2.3.3 copymemory函数用于字符串类型的数据传递
 - 2.3.4 对象的弱引用
 - 2.4 在vb中使用指针程序举例
 - 2.4.1 使用指针实现不同类型数据的交换
 - 2.4.2 使用指针实现不同窗体模块的数据交换

<<老兵新传>>

- 2.4.3 不使用数组类型的参数向函数传递数组
- 2.4.4 用一个long型参数向函数传递任意类型任意大小的数据
- 2.5 部分字节运算的问题和解决办法
 - 2.5.1 or按位或运算
 - 2.5.2 取长整数的高位和低位
- 第3章 不做数据的奴隶——常用数据结构
 - 3.1 何谓数据结构
 - 3.2 堆栈
 - 3.2.1 什么是堆栈
 - 3.2.2 堆栈的基本算法
 - 3.2.3 堆栈的通用模块实现
 - 3.2.4 堆栈通用模块的使用
 - 3.3 哈希表
 - 3.3.1 什么是哈希表
 - 3.3.2 哈希表的基本算法
 - 3.3.3 哈希表的通用模块实现
 - 3.3.4 哈希表通用模块的使用
 - 3.3.5 哈希表类模块与vb的collection对象的效率比较
 - 3.4 带“回收站”的数组
 - 3.4.1 基本思想
 - 3.4.2 程序实现
- 第4章 青出于蓝胜于蓝——子类技术
 - 4.1 windows窗口和消息驱动机制
 - 4.1.1 窗口
 - 4.1.2 窗口的消息驱动机制
 - 4.1.3 消息的组织 and 发送
 - 4.1.4 visual basic事件与消息
 - 4.2 子类技术及其常规实现
 - 4.2.1 子类技术简介
 - 4.2.2 子类技术的常规实现
 - 4.2.3 子类技术常规实现的问题
 - 4.3 开发实现子类技术的通用模块
 - 4.3.1 通用模块的目标
 - 4.3.2 封装子类信息的类模块
 - 4.3.3 实现子类技术的标准模块
 - 4.3.4 子类技术通用模块小结
 - 4.4 子类技术通用模块的用法举例
 - 4.4.1 标准模块module1的设计
 - 4.4.2 窗体form1的设计
 - 4.4.3 程序的运行效果
- 第5章 想解雇activex通用控件吗——使用api函数创建控件
 - 5.1 控件的深入理解
 - 5.1.1 activex控件
 - 5.1.2 dll通用控件库
 - 5.1.3 控件的运行机制
 - 5.1.4 简单实例——用api函数创建animation控件
 - 5.2 用api函数创建控件的模块划分

<<老兵新传>>

- 5.2.1 实现控件的类模块
- 5.2.2 支持的标准模块
- 5.2.3 其他“黑匣子”模块
- 5.2.4 模块划分小结
- 5.3 mdiapicontrolssupport标准模块
- 5.3.1 公有常量定义
- 5.3.2 控件类型的枚举类型定义
- 5.3.3 管理已创建控件
- 5.3.4 子类处理的自定义窗口程序
- 5.3.5 标准模块的其他内容
- 5.4 clscontrolbase基类模块
- 5.4.1 api函数声明和常量、类型定义
- 5.4.2 创建控件
- 5.4.3 实现控件的共有属性
- 5.4.4 实现控件的共有方法
- 5.4.5 类模块的initialize和terminate事件过程
- 5.5 实现跳格表控件的控件类模块clsapitabs
- 5.5.1 api函数声明和常量、类型定义
- 5.5.2 继承的“变通”实现
- 5.5.3 创建跳格表控件
- 5.5.4 实现跳格表控件的属性
- 5.5.5 实现跳格表控件的方法
- 5.5.6 实现跳格表控件的事件
- 5.5.7 clsapitabs控件类模块的应用实例
- 5.6 实现状态栏控件的控件类模块clsapistabar
- 5.6.1 api函数声明和常量、类型定义
- 5.6.2 继承的“变通”实现
- 5.6.3 创建状态栏控件
- 5.6.4 状态栏分栏和分栏属性
- 5.6.5 状态栏的其他功能
- 5.6.6 实现状态栏的事件
- 5.6.7 clsapistabar控件类模块的应用实例
- 5.7 实现丰富格式文本框控件的控件类模块clsapirichtextbox
- 5.7.1 richtextbox控件加载和继承的“变通”实现
- 5.7.2 创建richtextbox控件
- 5.7.3 管理文本和rtf代码
- 5.7.4 选择区域
- 5.7.5 文本的剪切、复制、粘贴
- 5.7.6 设置文本格式
- 5.7.7 实现richtextbox的事件
- 5.7.8 实现richtextbox的其他功能
- 5.7.9 clsapirichtextbox控件类模块的应用实例
- 第6章 路见不平一声吼，强大功能我也有——常见vb系统功能问题和解决
- 6.1 长字符串处理
- 6.1.1 vb系统中的字符串连接运算及问题
- 6.1.2 解决方案——clslongstring类模块
- 6.1.3 改进字符串连接效率测试

<<老兵新传>>

- 6.2 多种分行格式的纯文本文件的按行读取
 - 6.2.1 vb按行读取文本文件的常规方法及问题
 - 6.2.2 二进制读取文本文件
 - 6.2.3 设计类模块支持多种分行格式
 - 6.2.4 类模块的使用
- 6.3 部分字符串函数的增强
 - 6.3.1 split函数的增强
 - 6.3.2 trim函数的增强
 - 6.3.3 判断collection对象的键是否存在
- 6.4 解决控件不支持鼠标滚轮的问题
 - 6.4.1 鼠标滚轮消息wm_mousewheel
 - 6.4.2 窗口滚动消息wm_hscroll和wm_vscroll
 - 6.4.3 解决鼠标滚轮问题的通用模块
- 6.5 增强vb的滚动条控件
 - 6.5.1 基本思路
 - 6.5.2 实现技术
 - 6.5.3 类模块的完整代码
 - 6.5.4 增强滚动条通用模块使用举例
- 第7章 让你的程序在哪都不out——使程序支持多国语言版
 - 7.1 多国语言支持的主要问题
 - 7.2 字符串资源和资源加载
 - 7.2.1 字符串资源
 - 7.2.2 vb的资源编辑器
 - 7.2.3 解决不同语言语法元素顺序问题
 - 7.3 编写支持多国语言版程序的通用模块
 - 7.3.1 公有函数
 - 7.3.2 条件编译的全局元素
 - 7.3.3 开发多国语言版的程序小结
 - 7.4 支持多国语言版的程序实例
 - 7.4.1 简单实例
 - 7.4.2 改造clsreadlines类模块
- 第二篇 功能模块篇
- 第8章 程序骨感没人爱——windows绘图和图形界面增强
 - 8.1 gdi的基本原理
 - 8.1.1 色彩和坐标
 - 8.1.2 设备环境简介
 - 8.1.3 gdi对象
 - 8.2 高级文本描绘的类模块clsdrawtext
 - 8.2.1 drawtext函数
 - 8.2.2 clsdrawtext类模块的开发
 - 8.2.3 clsdrawtext使用的实例程序
 - 8.3 绘制windows图形元素
 - 8.3.1 绘制蚀刻线
 - 8.3.2 绘制3d效果的边框
 - 8.3.3 绘制windows按钮元素
 - 8.3.4 绘制首尾移动样式的按钮
 - 8.4 绘制透明位图

<<老兵新传>>

- 8.4.1 位图简介
- 8.4.2 光栅操作
- 8.4.3 绘制透明位图
- 8.4.4 程序实例
- 8.5 为静态图形添加图形热区
 - 8.5.1 图形热区实现的基本思路
 - 8.5.2 开发图形热区管理类模块clsgraphcoords
 - 8.5.3 图形热区编程实例
- 8.6 改变窗体透明度
 - 8.6.1 设置窗体透明度的api函数
 - 8.6.2 编写类模块封装设置窗体透明度的api函数
 - 8.6.3 程序实例
- 8.7 使控件具有windows xp风格的外观
 - 8.7.1 xp风格外观的控件探秘
 - 8.7.2 使应用程序中的控件具有xp风格
 - 8.7.3 程序实例
- 第9章 搞定这个怕什么？
你懂我的—— 菜单高级控制和菜单选择助手
 - 9.1 编写菜单控制标准模块mdlmenufuncs
 - 9.1.1 菜单的句柄
 - 9.1.2 获取菜单项信息
 - 9.1.3 设置菜单项信息
 - 9.1.4 设置菜单项高亮状态
 - 9.1.5 返回或设置子菜单默认项
 - 9.1.6 获得菜单中的菜单项总数
 - 9.1.7 系统菜单控制
 - 9.2 菜单选择助手
 - 9.2.1 基本思路
 - 9.2.2 在mdlmenufuncs中添加的完整代码
 - 9.2.3 clsmenuselassist的完整代码
 - 9.3 程序实例
 - 9.3.1 菜单选择助手功能演示
 - 9.3.2 标准模块mdlmenufuncs部分功能演示
- 第10章 玩转注册表，换汤不换药——简化ini文件和注册表编程
 - 10.1 ini文件和使用ini文件保存信息
 - 10.1.1 ini文件的结构
 - 10.1.2 ini文件读写的主要api函数
 - 10.1.3 封装ini文件的读写操作
 - 10.1.4 ini文件读写的实例程序
 - 10.2 注册表结构简介
 - 10.2.1 控制项
 - 10.2.2 子项
 - 10.2.3 键值
 - 10.3 编写通用模块封装常用注册表读写操作
 - 10.3.1 项的句柄和项的打开、关闭
 - 10.3.2 固定子项位置
 - 10.3.3 创建或删除子项

<<老兵新传>>

- 10.3.4 读取子项键值
- 10.3.5 设置子项键值
- 10.3.6 删除子项键值
- 10.3.7 读取注册表任意位置数据
- 10.3.8 设置文件关联
- 10.4 注册表编程实例
 - 10.4.1 记忆窗体位置
 - 10.4.2 在固定子项下读写数据
 - 10.4.3 建立和删除文件关联
- 第11章 真的很给力——文件系统编程
 - 11.1 文件系统操作
 - 11.1.1 列文件目录
 - 11.1.2 判断文件(夹)存在
 - 11.1.3 文件名操作函数
 - 11.1.4 创建文件夹
 - 11.1.5 文件的复制、移动和删除
 - 11.1.6 获得驱动器信息
 - 11.1.7 获得临时文件
 - 11.2 获得文件属性
 - 11.2.1 文件名属性
 - 11.2.2 文件大小和文件属性
 - 11.2.3 文件时间属性
 - 11.2.4 文件属性对话框
 - 11.2.5 文件类型字符串和文件图标
 - 11.3 打开/保存文件对话框的通用模块
 - 11.3.1 打开/保存文件对话框的相关api函数
 - 11.3.2 编写通用模块
 - 11.3.3 通用模块使用实例
 - 11.4 浏览文件夹对话框的通用模块
 - 11.4.1 浏览文件夹对话框的相关api函数
 - 11.4.2 编写通用模块
 - 11.4.3 通用模块使用实例
 - 11.5 制作自己的简易资源管理器
- 第12章 非“程”勿扰——其他常见系统功能
 - 12.1 运行应用程序
 - 12.1.1 运行程序或打开文档
 - 12.1.2 运行程序并等待程序结束
 - 12.1.3 程序实例
 - 12.2 监视windows剪贴板
 - 12.2.1 剪贴板查看器和查看器链
 - 12.2.2 使vb程序具有剪贴板监视功能的通用模块
 - 12.2.3 剪贴板监视程序举例
 - 12.3 设置系统热键
 - 12.3.1 设置系统热键的有关知识
 - 12.3.2 实现系统热键通用模块的完整代码
 - 12.3.3 程序实例
 - 12.4 添加系统托盘图标

<<老兵新传>>

- 12.4.1 相关api函数
- 12.4.2 实现系统托盘图标通用模块的完整代码
- 12.4.3 程序实例
- 第三篇 应用模块篇
- 第13章 我是勤奋的“猪”，我很乖——智能组合框
- 13.1 智能列表项维护
- 13.1.1 类模块对象与组合框控件的关联
- 13.1.2 不重复地添加列表项
- 13.1.3 列表项的保存和加载
- 13.2 实现按汉字拼音字头的中文自动输入
- 13.2.1 实现原理
- 13.2.2 获得汉字的拼音字头
- 13.2.3 查找拼音字头
- 13.2.4 实现按拼音字头自动输入
- 13.2.5 自动提示
- 13.3 智能组合框的其他功能
- 13.3.1 返回或设置下拉列表是否被拉下
- 13.3.2 设置下拉列表的高度
- 13.3.3 获取或设置下拉列表的最小宽度
- 13.4 智能组合框应用实例
- 第14章 “时”上编程——用api函数实现定时器
- 14.1 用api函数创建定时器的通用模块
- 14.1.1 定时器简介
- 14.1.2 使用api函数创建定时器
- 14.1.3 定时器支持模块mdltimersupport
- 14.1.4 定时器类模块clstimer
- 14.2 定时器通用模块应用实例
- 14.2.1 类模块内的定时器使用
- 14.2.2 标准模块内的定时器使用
- 14.3 通过定时器变通实现多线程
- 第15章 谁说这事不能说太细——进度指示
- 15.1 编写长时间运行程序需注意的主要问题
- 15.2 使程序在运行途中响应用户按下【取消】按钮
- 15.3 开发工作助手类模块clsoperassistant
- 15.3.1 数据定义
- 15.3.2 更新进度指示
- 15.3.3 显示简短提示文本
- 15.3.4 响应用户中途取消
- 15.3.5 出错提示
- 15.3.6 程序开始和结束
- 15.3.7 清除方法clear
- 15.3.8 类模块的initialize和terminate事件过程
- 15.4 工作助手类模块的应用实例
- 15.5 开发指示进度窗体
- 15.5.1 指示进度窗体的运行效果和调用
- 15.5.2 在工作助手类模块中添加驱动窗体的代码
- 15.5.3 开发frmworkingavi窗体

<<老兵新传>>

第16章 边界条约用户定——实现运行时控件的边框调整

16.1 解决边框调整问题的基本思路

16.1.1 通用模块的使用

16.1.2 frame 控件的作用

16.1.3 控件大小位置的重新调整

16.1.4 上下型和左右型风格

16.2 开发边框调整的通用模块

16.2.1 控件关联

16.2.2 有关的数据定义

16.2.3 重新安排控件大小和位置

16.2.4 处理分隔线拖动事件

16.2.5 类模块的initialize和terminate事件过程

16.3 边框调整通用模块应用实例

第17章 “泡泡”几时有，自己编程瞅——自己编程实现气泡提示框

17.1 制作气泡型窗体

17.1.1 制作气泡型窗体的基本思路

17.1.2 制作气泡型窗体实例

17.2 气泡提示框的通用模块

17.2.1 窗体的界面设计

17.2.2 气泡样式和图标类型的枚举类型

17.2.3 属性和常量定义

17.2.4 显示气泡提示

17.2.5 指向控件的气泡提示

17.2.6 气泡提示的关闭

17.2.7 窗体的加载和卸载

17.3 气泡提示框程序实例

第18章 乱“式”英雄——具有高级功能的表达式计算

18.1 表达式计算的算法简介

18.2 clscacuexp类模块的数据定义

18.2.1 数据类型

18.2.2 表达式“解析”与表达式元素

18.2.3 表达式中的常量元素

18.2.4 表达式中的变量元素

18.2.5 运算符和函数的常量定义

18.2.6 表达式字符串属性

18.2.7 计算结果属性

18.2.8 错误信息属性

18.3 clscacuexp类模块的功能代码

18.3.1 表达式解析

18.3.2 表达式计算

18.4 clscacuexp类模块的使用实例

索引

参考文献

<<老兵新传>>

章节摘录

3.3 哈希表 3.3.1 什么是哈希表 3.3.1.1 提出问题 在程序设计中，经常需要维护一个列表，比如学生列表、货品列表、窗体列表等。

对这些列表最频繁的一种操作是能够通过给定的一个编号快速找到项目。

例如给定一个学生的学号，就能快速找到该学生的数据；给定一个窗体的编号，就能快速定位到窗体以便对窗体进一步操作（如将窗体显示出来）等。

也就是说这类列表中的每项数据是由两部分组成的：一是可以惟一标识本条数据的一个编号，二是数据本身。

我们把前者这种可以惟一标识一条数据的编号称为键，后者称为数据或值。

因此问题可以概括为：维护一个列表，给定一个键后，能快速地找到对应的数据。

这可以通过数组来实现，比如把学生、货品、窗体等信息存放到数组中，数组的下标就是键，即把数据存储到与编号相同下标的空间中。

如果给定了学号、货品号、窗体编号，就是给定了数组的下标，通过数组的下标可以直接找到数组元素。

然而，这里有一个问题是：学号、货品号、窗体编号等可能是一个很大的数字，例如学号可能是8位数字，那么定义的数组就至少应该有108个元素，而实际学生人数远没有108个。

也就是说实际数据只占用了数组的很小一部分，在“数据结构”中称这个数组是稀疏的（sparse）。

显然采用上述这种数组的存储方法将大量浪费存储空间。

VB中有一种Collection对象也可以用于处理这类数据，Collection对象中的每个元素可以有一个键和一个值。

通过Item方法给定一个键作为参数就能快速地返回数据。

然而Collection对象的效率很低，处理少量数据尚可，当处理的数据量很大时，会严重影响程序的效率，稍后还要讨论这个问题。

.....

<<老兵新传>>

编辑推荐

《老兵新传:Visual Basic核心编程及通用模块开发》面向Visual Basic编程和开发实践，介绍实现软件的实际功能，直击编程实际问题，帮助程序员学习开发实践知识！
编程“高手”的功力体现在对语言的彻底掌握和丰富的实践经验，书稿内容由作者原创编写，直接面向实际应用。

《老兵新传:Visual Basic核心编程及通用模块开发》程序代码凝聚了作者十余年的编程经验，书中所有通用模块可不加修改直接调用。

<<老兵新传>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>