

<<VBScript程序员参考手册>>

图书基本信息

书名：<<VBScript程序员参考手册>>

13位ISBN编号：9787302194644

10位ISBN编号：7302194645

出版时间：2009-2

出版时间：清华大学出版社

作者：（美）金斯利·休斯（Kingsley-Hughes, A）等

页数：696

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>

<<VBScript程序员参考手册>>

前言

我们希望，能够在—个文本编辑器中快速而简便地编写代码，而不用考虑任何复杂的开发环境。我们希望，不用牵绊于编译代码或发布复杂的安装程序的繁杂工作。

我们希望，能够以多种方式部署代码。

我们还希望，只要学习—种语言就可以完成服务器端Internet、客户端Internet和桌面程序的编码。

实际上这不仅仅是希望，使用VBScript能够实现的不仅是这些。

VBScript是—种绝对高级的语言，甚至可以用它来“说话”。

它易于学习，功能强大，灵活性强，而且很便宜。

这使得VBScript无论对于经验丰富还是初出茅庐的程序员来说，都成为其首选语言。

如果您是经验丰富的程序员，就会享受到由于不需要复杂的开发环境和编译的需要而带来的编码乐趣。

。

如果您是初学者，那么只要懂—点文本编辑器的知识，就可以开始学习编程。

VBScript的知识和经验还将开启很多技术的大门。

如果有很好的VBScript编程基础，就可以涉足Internet开发、网络管理、服务器端编码甚至其他编程语言的使用（VisualBasic是—条最流行的路线，因为它的语法与VBScript非常相似）。

使用VBScript，还可以创建看上去像使用C++这样复杂的编程语言编写的程序。

还有一点值得注意的是，对脚本的支持已经嵌入到所有新版本的Windows操作系统中——这对于帮助了解VBScript的工作机制来说，无疑是一种便利条件。

通过在文本编辑器中编写—些简单的脚本，就可以完成很多工作，如复制和移动文件、创建文件夹和文件、修改Windows注册表，还有很多很多。

—种易于使用的脚本语言居然无所不能。

我们相信，了解如何编写VBScript应用程序对很多人来说都将是一种有用且有益的技能，无论他们是IT行业用户、SOHOPC用户、学生还是家庭用户。

了解和使用VBScript可以节省时间，更重要的是，节省金钱。

<<VBScript程序员参考手册>>

内容概要

《VBScript程序员参考手册(第3版)》是一本针对最新版本VBScript的编程参考手册，内容涵盖了与VBScript相关的各个方面。

《VBScript程序员参考手册(第3版)》首先介绍了VBScript的基本概念和语法，然后通过具体的例子逐步地向读者介绍了如何在各种不同的脚本宿主上编写、调试和运行VBScript，涉及到的脚本宿主包括Windows脚本宿主、IIS服务器、Internet Explorer浏览器等。

同时还介绍了如何用VBScript开发WindowsVista操作系统的桌面小工具、如何对脚本进行加密以保护开发人员的工作成果，以及如何在VB或.NET程序中提供对VBScript的支持。

在《VBScript程序员参考手册(第3版)》的网站上提供了所有示例程序的代码，学习起来非常方便。

无论您是否有其他语言的编程经验，《VBScript程序员参考手册(第3版)》都是您开始学习VBScript不可或缺的好书。

对于经验丰富的VBScript程序员来说，《VBScript程序员参考手册(第3版)》也是一本值得常备的VBScript参考手册。

书籍目录

第1章 编程快速入门1.1 变量及数据类型1.1.1 变量的使用1.1.2 注释的使用1.1.3 VBScript内建函数的使用1.1.4 理解语法问题1.2 流程控制1.2.1 分支1.2.2 循环1.3 操作符与操作符优先级1.4 代码的组织结构和重用1.4.1 模块化、黑盒、过程和子过程1.4.2 编写函数1.4.3 使用过程的好处1.5 自顶向下与事件驱动1.5.1 理解自顶向下程序设计1.5.2 理解事件驱动程序设计1.5.3 自顶向下和事件驱动的结合1.5.4 事件驱动实例1.6 关于编程的一些忠告1.6.1 要考虑到意外情况1.6.2 保持代码的可读性1.6.3 将代码模块化成模块、类和组件1.6.4 匈牙利变量命名法1.6.5 不要在多个任务中使用同一个变量1.6.6 时刻都要合理地布局代码1.6.7 用注释使您的代码清晰可读，但不要滥用1.7 小结第2章 VBScript是什么？不是什么？

2.1 Windows脚本2.2 版本信息2.3 VBScript是VB的子集2.4 VBScript是一种脚本语言2.5 VBScript在运行时被解释2.5.1 运行时编译的缺点2.5.2 运行时编译的优点2.6 使用VBScript的优势2.7 VBScript适合您吗2.8 VBScript在VisualBasic家族中的地位2.8.1 VisualBasic2.8.2 VisualBasicforApplications2.8.3 VBScript2.8.4 VBScript是一门“真正”的编程语言吗2.9 VBScript能做什么2.9.1 PowerShell2.9.2 Windows脚本宿主2.9.3 桌面小工具2.9.4 Windows脚本组件2.9.5 客户端Web脚本2.9.6 服务器端Web脚本2.9.7 远程脚本2.9.8 HTML应用程序2.9.9 在程序中添加VBScript2.10 VBScript的开发工具2.10.1小结第3章 数据类型3.1 弱类型脚本语言3.2 为什么数据类型是如此重要3.3 Variant——VBScript中唯一的数据类型3.3.1 子类型的检查和类型转换3.3.2 隐式类型转换3.3.3 隐式类型转换实战3.3.4 Empty与Null3.3.5 对象(Object)子类型3.3.6 Error子类型3.4 复杂数据类型——数组3.4.1 什么是数组3.4.2 数组的维度3.4.3 数组边界和声明数组3.4.4 用下标访问数组3.4.5 遍历数组3.4.6 清除数组3.4.7 在数组中使用VarType()3.5 小结第4章 变量与过程4.1 OptionExplicit4.2 变量命名4.3 过程和函数4.3.1 过程的语法4.3.2 函数的语法4.3.3 调用过程和函数4.3.4 可选参数4.3.5 退出过程或函数4.4 变量作用域、声明及生存期4.4.1 理解变量作用域4.4.2 理解变量声明4.4.3 变量生存期4.5 脚本和过程的设计策略4.5.1 限制代码读取和修改变量4.5.2 将代码分割成过程和函数4.5.3 关于脚本设计的忠告4.6 ByRef和ByVal4.7 文字常量和具名常量4.7.1 什么是文字常量4.7.2 什么是具名常量4.7.3 使用具名常量的好处4.7.4 具名常量的使用原则4.7.5 VBScript的内建常量4.8 小结第5章 流程控制5.1 分支结构5.1.1 If分支5.1.2 SelectCase分支5.2 循环结构5.2.1 For...Next5.2.2 ForEach...Next5.2.3 DoLoop5.2.4 While...Wend5.3 小结第6章 错误处理和调试6.1 错误的类型6.1.1 语法错误6.1.2 运行时错误6.1.3 逻辑错误6.2 错误的可见性和环境6.2.1 Windows脚本宿主错误6.2.2 服务器端ASP错误6.2.3 InternetExplorer中的客户端VBScript错误6.3 错误处理6.3.1 Err对象的使用6.3.2 OnError语句的使用6.3.3 呈现和记录错误6.3.4 显示服务器端ASP错误6.4 生成自定义错误6.4.1 使用Err.Raise6.4.2 什么时候不该用Err.Raise6.4.3 什么时候该生成自定义错误6.5 调试6.5.1 什么是调试器6.5.2 VBScript调试概论6.5.3 用微软脚本调试器调试WSH脚本6.5.4 用微软脚本调试器调试客户端网页脚本6.5.5 用微软脚本调试器调试ASP6.5.6 在没有调试器的情况下调试代码6.5.7 微软脚本调试器的使用6.6 小结第7章 脚本运行时对象7.1 什么是运行时对象7.2 对象基础7.2.1 创建对象7.2.2 属性和方法7.2.3 With关键字7.2.4 对象可以有多个引用7.2.5 对象生存期和销毁对象7.3 字典对象7.3.1 概论7.3.2 三种不同的添加方法7.3.3 CompareMode属性7.3.4 Item属性7.3.5 Exists方法7.4 FileSystemObject库7.4.1 为什么要用FileSystemObject？7.4.2 集合的使用7.4.3 理解FileSystemObject7.4.4 创建文件夹7.4.5 复制文件7.4.6 复制文件夹7.4.7 读取文本文件7.4.8 写文本文件7.5 小结第8章 VBScript中的类(编写COM对象)8.1 对象、类和组件8.2 类语句8.3 定义属性8.3.1 私有属性变量8.3.2 PropertyLet8.3.3 PropertyGet8.3.4 PropertySet8.3.5 创建只读属性8.3.6 创建只写属性8.3.7 没有属性过程的公共属性8.4 定义方法8.5 类事件8.5.1 Class_Initialize事件8.5.2 Class_Terminate事件8.6 类常量8.7 构建和使用VBScript类8.8 小结第9章 正则表达式9.1 正则表达式简介9.1.1 实战正则表达式9.1.2 从简单的开始9.2 RegExp对象9.2.1 Global属性9.2.2 IgnoreCase属性9.2.3 Pattern属性9.2.4 正则表达式字符9.2.5 Execute方法9.2.6 Replace方法9.2.7 Backreferencing9.2.8 Test方法9.3 Matches集合9.3.1 Matches的属性9.3.2 Match对象9.4 一些例子9.4.1 验证电话号码输入9.4.2 分解URI9.4.3 检查HTML元素9.4.4 匹配空白9.4.5 匹配HTML注释标签9.5 小结第10章 客户端Web脚本10.1 专用工具10.2 脚本编程的发展历程10.3 各种类型的脚本语言10.3.1 JavaScript、JScript和ECMAScript10.3.2 VBScript10.4 响应浏览器事件10.4.1 添加事件句柄10.4.2 添加一个带参数的事件句柄10.4.3 取消事件10.4.4

事件的顺序10.4.5 表单验证10.4.6 验证输入框中的数值10.4.7 验证单选按钮10.4.8 验证选择控件和日期10.5 实战文档对象模型10.5.1 窗口对象10.5.2 集合10.6 小结第11章 Windows边栏和小工具11.1 小工具基础11.1.1 小工具中的文件11.1.2 Manifest文件11.1.3 图标11.2 构建小工具11.3 自动刷新小工具11.4 打包小工具11.5 小结第12章 TaskScheduler脚本编写12.1 使用TaskScheduler12.1.1 使用MMC插件12.1.2 在TaskScheduler中定义并创建任务12.1.3 TaskSchedulerXML模式12.2 TaskScheduler2.0脚本对象12.2.1 Action12.2.2 ActionCollection12.2.3 BootTrigger12.2.4 ComHandlerAction12.2.5 DailyTrigger12.2.6 EmailAction12.2.7 EventTrigger12.2.8 ExecAction12.2.9 IdleSettings12.2.1 0IdleTrigger12.2.1 1LogonTrigger12.2.1 2MonthlyDOWTrigger12.2.1 3MonthlyTrigger12.2.1 4Networksettings12.2.1 5Principal12.2.1 6RegisteredTask12.2.1 7RegisteredTaskCollection12.2.1 8RegistrationInfo12.2.1 9RegistrationTrigger12.2.2 0RepetitionPattern12.2.2 1RunningTask12.2.2 2RunningTaskCollection12.2.2 3SessionStateChangeTrigger12.2.2 4ShowMessageAction12.2.2 5TaskDefinition12.2.2 6TaskFolder12.2.2 7TaskFolderCollection12.2.2 8TaskNamedValuePair12.2.2 9TaskNamedValueCollection12.2.3 0TaskService12.2.3 1TaskSettings12.2.3 2TaskVariables12.2.3 3TimeTrigger12.2.3 4Trigger12.2.3 5TriggerCollection12.2.3 6WeeklyTrigger12.3 TaskScheduler范例脚本12.4 小结第13章 PowerShell13.1 安装需求13.2 功能特性13.3 为什么说PowerShell是一种新的脚本语言13.4 开始使用PowerShell13.4.1 使用PowerShell13.4.2 进一步深入PowerShell13.5 在PowerShell中使用脚本13.5.1 修改PowerShell的执行策略13.5.2 命名脚本13.6 创建并调用第一个PowerShellcmdlet脚本13.7 VBScript与PowerShell之间的联系13.7.1 操作符13.7.2 函数13.7.3 语句13.8 小结第14章 增强的客户端脚本14.1 使用需求与浏览器安全性14.2 Scriptlet——行为机制的前身14.2.1 什么是Scriptlet14.2.2 使用public_前缀暴露scriptlet成员14.2.3 在Scriptlet中封装代码以便重用14.3 事件管理14.3.1 与事件处理程序之间的关系14.3.2 Scriptlet模型扩展14.4 IE5中不支持Scriptlet14.4.1 行为机制14.4.2 行为机制的实现技术14.4.3 将行为应用于HTML元素14.5 HTML组件(HTC)14.6 小结第15章 Windows脚本宿主15.1 相关工具15.2 WSH的概念15.3 脚本文件的类型15.4 使用Windows脚本宿主运行脚本15.4.1 命令行执行15.4.2 在Windows环境中执行WSH15.5 使用.WSH文件运行脚本15.6 Windows脚本宿主的内建对象15.6.1 WScript对象15.6.2 WshArguments对象15.6.3 WshShell对象15.6.4 WshNamed对象15.6.5 WshUnnamed对象15.6.6 WshNetwork对象15.6.7 WshEnvironment对象15.6.8 WshSpecialFolders对象15.6.9 WshShortcut对象15.6.1 0WshUrlShortcut对象15.7 小结第16章 Windows脚本组件16.1 什么是Windows脚本组件16.2 需要的工具16.3 脚本组件运行时16.4 脚本组件文件和向导16.5 暴露属性、方法和事件16.5.1 属性16.5.2 方法16.5.3 事件16.6 注册信息16.7 创建脚本组件类型库16.8 如何引用其他组件16.9 ASP的脚本组件16.1 0编译时错误检查16.1 1在Script组件中使用VBScript类16.1.1 VBScript类的使用限制16.1.2 使用内部类16.1.3 包含外部源文件16.1.2小结第17章 脚本编码17.1 脚本编码的使用限制17.2 已编码脚本——什么该做什么不该做17.3 使用Microsoft脚本编码器进行编码17.3.1 获取方式和安装过程17.3.2 使用Microsoft脚本编码器17.3.3 语法17.3.4 可编码的文件类型17.4 脚本的解码17.5 脚本模糊化的其他方法17.6 小结第18章 远程脚本18.1 远程脚本的工作原理18.1.1 安全性18.1.2 远程脚本需要的文件18.2 使用VBScript实现远程脚本18.2.1 在服务器上安装远程脚本18.2.2 在服务端启用远程脚本18.2.3 在客户端启用远程脚本18.2.4 调用一个远程方法18.2.5 将ASP页面转换成VBScript对象18.3 小结第19章 HTML应用程序19.1 HTML应用程序的优点19.2 如何创建一个基本的HTA19.2.1 HTML文件范例19.2.2 将一个HTML文件变为一个HTML应用程序19.3 HTA: APPLICATION元素19.3.1 修改应用程序的外观19.3.2 从命令行中更改参数19.3.3 访问其他的HTA: APPLICATION属性19.4 HTA及其安全性19.4.1 关于框架的安全性问题19.4.2 使用APPLICATION属性19.4.3 使用嵌套框架19.5 HTA部署模型19.5.1 Web模型19.5.2 包模型19.5.3 混合模型19.6 HTA不支持的特性19.6.1 windows对象19.6.2 默认行为19.7 小结第20章 服务器端Web脚本20.1 理解HTTP协议的内部机制20.1.1 HTTP服务器20.1.2 协议基础20.2 ASP简介20.2.1 服务器如何识别ASP20.2.2 ASP基础20.2.3 ASP的标签20.3 使用ASP对象模型20.3.1 集合20.3.2 Request对象的集合20.3.3 Response对象的集合20.3.4 Response对象的属性20.3.5 理解Application和Session对象20.3.6 Application对象20.3.7 Session对象20.3.8 Server对象20.3.9ObjectContext对象20.4 高效地使用ASP20.4.1 站点设计20.4.2 创建global.asa文件20.4.3 创建主页面20.4.4 ASPVBScript部分20.4.5 HTML部分20.5 小结第21章 在VB和.NET应用程序中添加VBScript代码21.1 为什么要在应用程序中添加脚本21.2 宏和脚本的概念21.2.1 使用Scriptlet21.2.2 使用脚本21.2.3 选择最佳的使用范围21.3 在VB和.NET应用程序

<<VBScript程序员参考手册>>

中
添加脚本控件
21.4 脚本控件参考
21.4.1 对象模型
21.4.2 对象与集合
21.4.3 常量
21.5 脚本控件错误处理
21.6 调试
21.7 使用已编码脚本
21.8.NET工程范例
21.9 VisualBasic6工程范例
21.10 小结
附录A VBScript函数和关键字
附录B 变量命名约定
附录C 编码规范
附录D VBScript中支持的VisualBasic常量
附录E VBScript错误代码和Err对象
附录F 脚本运行时库对象参考
附录G Windows脚本宿主对象模型
附录H 正则表达式
附录 Variant的子类型
附录 ActiveX数据对象

<<VBScript程序员参考手册>>

章节摘录

鉴于VBScript这类语言以及作者所接触到的正在转向这类语言的用户特殊性，用一个涵盖VBScript所有基本知识的章节作为本书的开始是非常合适的。

在本章中，您将简要地学习一些编程的基本知识。

如果是从其他编程语言（VisualBasic、VisualBasic.NET、C、C++、Delphi、C#）转向VBScript并熟悉各种专业术语，您可能就不需要阅读本章的内容。

若确实如此，请放心地跳过本章直接阅读后面的章节。

当然，若您没有任何编程经验，本章将会为您提供牢固的基本知识，使您信心百倍地开始VBScript的学习。

若您继续阅读本章，您应该是属于以下三种人：
· 网络管理员或系统管理员，想用VBScript和Windows脚本宿主或PowerShell编写登录脚本或是自动完成管理工作。

· 想提升和拓展自己技术能力的网页设计师，以便能从事某些关于ASP的工作。

· 对编程很感兴趣（可能是VisualBasic或VisualBasic.NET），想在深入学习之前先体验一下。

编程涉及到很多方面。

多年以来，在书本中和Internet上有无数关于编程的书籍和文章。

在本章中，我们将用单独一章的篇幅来介绍一些您之前不熟悉的概念。

我们的学习进度会非常快，若您想仔细地阅读这些内容，可以自己试试其中的那些例子，这样可以更好地掌握这些概念。

此外，务必要注意的是，还有很多编程的基础知识是我们在这里没有提及的，例如：
· 软件架构
· 系统设计
· 数据库设计
· 文档化代码
· 高级测试、调试和beta版本测试

· 软件部署和支持
本章简要地介绍了构成整个程序设计领域的一些基本概念。

显然，我们不可能使您在一夜之间成为编程专家，但是我们希望本章能让您知道如何学习本书中的其他内容。

<<VBScript程序员参考手册>>

编辑推荐

本书经验丰富的作者团队再次与读者分享脚本实例、教程和便捷的疑难解答，以帮助读者提升脚本编写技能。

本书这一版针对上一版发布以来VBScript的各种变化新增了一些内容，涉及Windows Vista、Windows Server 2003、Internet Explorer、Microsoft Powershell，以及VBScript最新版本的改进。与前一版本的风格相同，本书首先还是对vBScript进行简要介绍，阐述其作为一种编程语言的工作原理。

接下来讨论如何在其他技术中使用vBScript，并提供了很多改进的vBScript代码示例。

读者可以从中学学习很多问题的解决方法，如复制和移动文件、创建文件夹和文件、修改windOWS操作系统等。

此外，本书还提供了翔实的附录，这对于读者来说是快速了解vBScript工作原理的珍贵资料，使读者能够立即开始使用VBScript。

本书主要内容 如何使用变量、注释和内建的vBScript函数 VBScript是什么，不是什么
使用程序的好处 将代码转换成函数的方法 如何将代码模块化为子程序、模块、
类和组件 自顶向下和事件驱动编程模型的区别 代码组织与重用的一些提示 使代码
更清晰、可读的相关技术 本书读者对象： 本书面向的读者包括程序员、客户端和服务
端Web开发人员以及想通过学习VBScript来提高工作效率的系统管理员。

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>