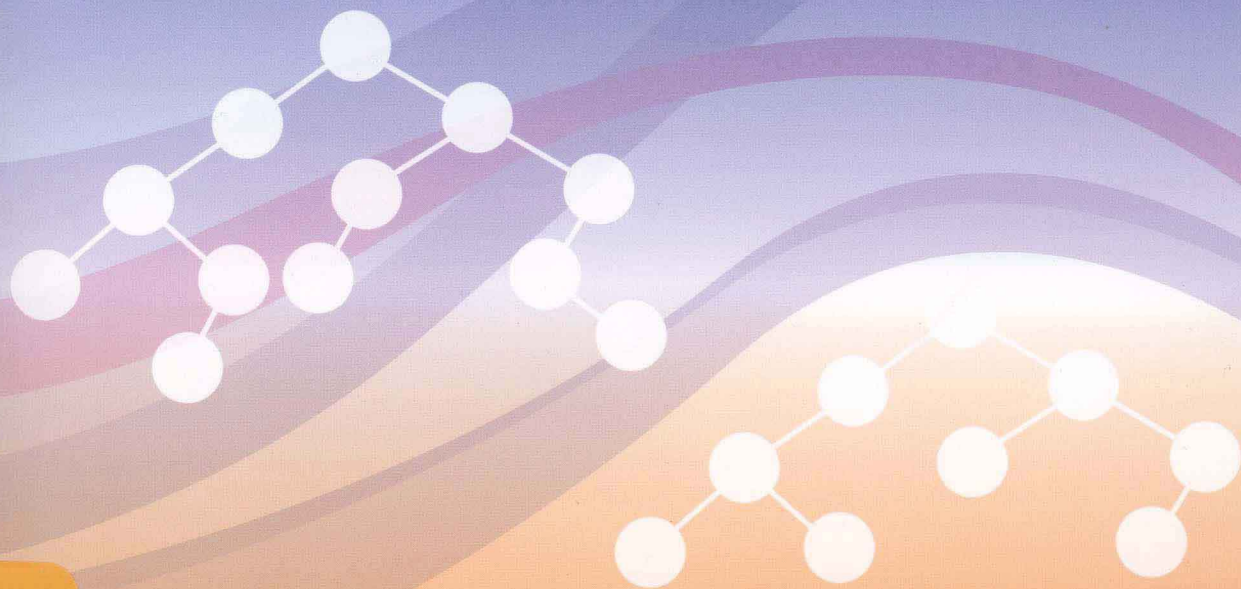


高等学校计算机专业规划教材

Visual Basic程序设计 (2012版)

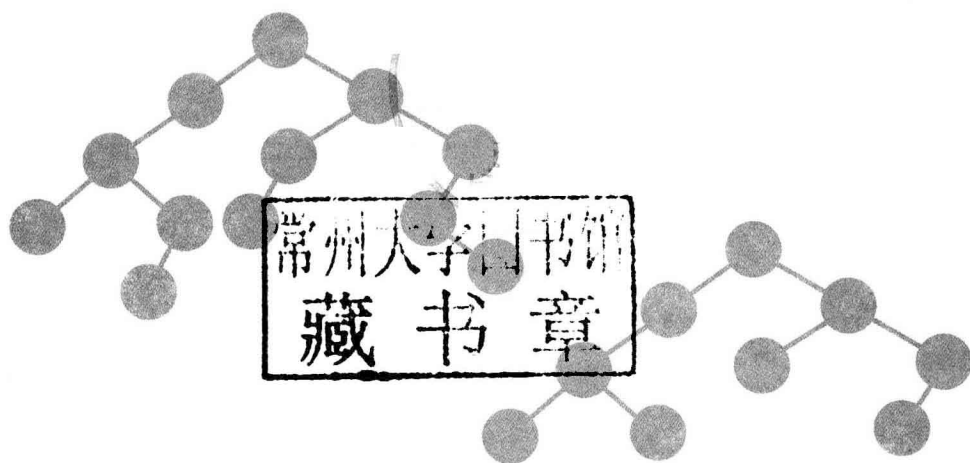


邱仲潘 王帅 孙溢泽 周丹丹 编著

清华大学出版社

高等学校计算机专业规划教材

Visual Basic程序设计 (2012版)



邱仲潘 王帅 孙溢泽 周丹丹 编著

清华大学出版社

内 容 简 介

本书全面介绍 Visual Basic 2012 的编程思想及特色,全书共 11 章,主要内容包括使用控件创建简单的应用程序、编写条件分支结构和循环结构、调试程序以及更正程序中的错误、设计用户界面以及实现数据库和 Web 编程等。本书从最基础的知识讲起,结合丰富的示例,由浅入深,内容翔实,非常适合初学者迅速入门并获得提高,也适用于有经验的编程人员快速掌握 Visual Basic 编程的核心技术。

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

Visual Basic 程序设计(2012 版)/邱仲潘,王帅等编著. —北京:清华大学出版社,2013
(高等学校计算机专业规划教材)

ISBN 978-7-302-32703-5

I. ①V… II. ①邱… ②王… III. ①BASIC 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2013)第 125548 号

责任编辑:龙启铭 战晓雷

封面设计:何凤霞

责任校对:白 蕾

责任印制:王静怡

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 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

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:21.5 字 数:496 千字

版 次:2013 年 7 月第 1 版 印 次:2013 年 7 月第 1 次印刷

印 数:1~2000

定 价:39.00 元

产品编号:050831-01



微软公司在 Visual Studio 6 之后发布了 .NET 平台,其后的 Visual Studio 都是基于 .NET 平台的,尽管无论是开发环境的改善,还是新的开发语言 C# 的引入,都没有使得 Visual Studio 能够再创辉煌,再续经典。每次新版本的发布,程序员们只看到安装程序越来越大,整个开发环境越来越臃肿,却没有看到多少革命性的变化,这多少有点让我们这些天天使用 Visual Studio 的程序员们有些失望。随着 Visual Studio 2012 的发布,微软公司宣称会将它打造成跟 Visual Studio 6 一样的经典产品。新版本什么样?有什么新功能?真的会成为下一个经典吗?关心 Visual Studio 的人都在问,迫切地想先睹为快。

根据微软公司发布的一份官方文档宣称,Visual Studio 2012 和 .NET Framework 4.5 在下面几个方面有所创新。

1. 全新的外观和感受

一打开 IDE,就会发现不同之处。整个界面经过了重新设计,简化了工作流程,并且提供了访问常用工具的捷径。工具栏经过了简化,减少了选项卡的混乱性,现在可以使用全新快速的方式找到代码。所有这些改变都可以更轻松地导航应用程序,以用户喜爱的方式工作。

2. 为 Windows 8 做好准备

Visual Studio 2012 提供了新的模板、设计工具以及测试和调试工具——在尽可能短的时间内构建具有强大吸引力的应用程序所需要的一切。同时,Blend for Visual Studio 还提供了一款可视化工具集,可以充分利用 Windows 8 全新而美观的界面。

不过,最有价值的地方在于创建应用程序之后。以前,要想将一款客户需要的产品展现在客户面前并不总是一件容易的事情。但是现在,通过 Windows Store 这一广泛的分布式渠道,可以接触数百万的用户。Windows Store 的条款是透明且易于理解的。所以,可以轻松编写代码和销售软件,而且说不定接下来几年可以悠闲地在海滩度假。

3. Web 开发升级

对于 Web 开发,Visual Studio 2012 也提供了新的模板、更优秀的发布工具和对新标准(如 HTML5 和 CSS3)的全面支持,以及 ASP.NET 中的最新特性。此外,还可以利用 Page Inspector 在 IDE 中与正在编码的页面进行

交互,从而更轻松地进行调试。那么对于移动设备又如何呢?有了 ASP.NET,现在便可以使用优化的控件,为手机、平板电脑以及其他小屏幕创建应用程序。

4. 云功能

以前,每个人都需要维护一台服务器。光是扩展容量便占用了基础架构投资的一大半。而现在,可以利用云环境中动态增加存储空间和计算能力的功能快速访问无数虚拟服务器。Visual Studio 提供了新的工具将应用程序发布到 Windows Azure(包括新模板和发布选项),并且支持分布式缓存,维护时间更少。

5. 为重要业务做好准备

在 SharePoint 开发中,也会发现很多重要的改进,包括新设计工具、模板以及部署选项。可以利用为 SharePoint 升级的应用生命周期管理功能,如性能分析、单元测试和 IntelliTrace。但是最令人惊讶的还是 LightSwitch,有了它,用户只需编写少量代码就可以创建业务级应用程序。

6. 灵活敏捷的流程和可靠的应用生命周期管理

随着应用程序变得越来越复杂,个人还需要能帮助团队更快、更智能地工作的工具。这就是微软公司在其产品中要加入一种灵活的敏捷方法的原因。利用 Visual Studio 和 Team Foundation Server,可以根据自己的步调采用效率更高的方法,同时还不会影响现有工作流程。微软公司还提供了让整个组织来参与整个开发测试过程的方法,通过新的方法让利益相关方、客户和业务团队成员跟踪项目进度并提出新的需求和反馈。

上面这段引自微软公司官方文档的文字可以翻译成程序员更好理解的表述:“Visual Studio 2012 将致力于引领下一代平台技术的发展,提高开发人员的工作效率和热情,创建突破性的应用程序,推动云计算等新兴技术趋势,合理化应用程序生命周期管理(ALM)。另外,Visual Studio 2012 将支持统一建模语言(UML)和特定域语言(DSL),它将可以为各类开发人员提供合适的工具,而这也是微软公司更大的模型平台的一部分。”

这些介绍都高度地概括了 Visual Studio 2012 的新特性。实际上,用户希望能够看到一些更加具体、更加实际的内容。所以,本书会以 Visual Studio 2012 为基础,带领大家一同来看看 Visual Studio 2012 中的新特性,从现在开始体验下一代 Visual Studio。根据微软公司提供的一份指导文档为蓝本,本书将依次介绍其全新的基于 WPF 创建的 IDE、增强的代码编辑器以及 Visual Basic 中所带来的一些令人激动不已的新特性等等。

本书坚持以语言为基础和应用为主导的编写原则,循序渐进地展示如何使用 Visual Basic 2012 开发 Windows 窗体应用程序和 ASP.NET 应用程序。为了更好地帮助读者在短时间内掌握使用 Visual Basic 语言开发各种应用程序中的知识点和编程技巧,全书的基础知识介绍清晰,理论联系实际,具有很强的操作性。本书还提供了大量的通过测试可运行的完整实例,这些实例都有设计步骤、代码详解和程序运行结果等,不但复习了前面所学的内容,而且还增加了一定量的创作技巧。对于容易出现问题的地方,则以“注”的方式介绍常用的技巧和注意事项。



本书由邱仲潘、王帅、孙溢泽、周丹丹编写，郑朦、郑旭也参加了部分章节的编写及校正，此外感谢宋智军、张显威、于洋、甘星超、姜夕凯等给予的帮助和支持，也非常感谢清华大学出版社为本书的选题策划、编辑加工和出版发行所付出的努力。由于编者水平有限，加之时间仓促，书中难免有疏漏和不足之处，恳请专家和广大读者指正。

编 者

2013 年 5 月



第 1 章 初识 Visual Basic 2012 /1

1.1	Visual Basic 2012	1
1.1.1	Visual Basic 的发展	1
1.1.2	Visual Basic 2012 新特点	2
1.2	.NET Framework	5
1.2.1	Visual Basic 与 .NET Framework	5
1.2.2	.NET Framework 简介	5
1.2.3	.NET Framework 4.5	6
1.3	Visual Studio 2012	9
1.3.1	Visual Studio 2012 的新特点	9
1.3.2	Visual Studio 2012 的安装环境与安装过程	11
1.3.3	认识 Visual Studio 2012 集成开发环境	16
1.4	创建第一个 Visual Basic 2012 程序	20
1.4.1	创建工程文件	20
1.4.2	添加控件并设置属性	20
1.4.3	添加代码	21
1.4.4	运行程序	21
1.4.5	保存项目	22

第 2 章 Visual Basic 2012 编程基础 /23

2.1	字符集	23
2.2	标识符	23
2.3	关键字	24
2.4	数据类型	24
2.4.1	基本数据类型	24
2.4.2	用户自定义数据类型	27
2.4.3	枚举类型	28
2.5	常量	30
2.5.1	直接常量	30
2.5.2	符号常量	31



2.6	变量	31
2.6.1	变量的声明	32
2.6.2	变量的命名规则	33
2.6.3	变量的作用域	33
2.6.4	类型转换	35
2.7	运算符与表达式	39
2.7.1	算术运算符与算术表达式	39
2.7.2	关系运算符与关系表达式	40
2.7.3	逻辑/位运算符和逻辑/位运算表达式	40
2.7.4	连接运算符	42
2.7.5	其他运算符	42
2.7.6	运算符的优先级	44
2.7.7	结合性	45
2.8	代码编写规则	45

第 3 章 程序控制结构 /48

3.1	顺序结构程序设计	48
3.1.1	赋值语句	48
3.1.2	复合赋值语句	49
3.1.3	数据输入	49
3.1.4	数据输出	53
3.2	选择结构程序设计	56
3.2.1	If 语句	56
3.2.2	Select Case 语句	60
3.2.3	条件函数	63
3.3	循环结构程序设计	64
3.3.1	For 语句	64
3.3.2	While 语句	67
3.3.3	Do 语句	69
3.3.4	嵌套语句	73
3.4	辅助控制语句	74
3.4.1	GoTo 语句	74
3.4.2	Continue 语句	76
3.4.3	Stop 语句与 End 语句	77
3.4.4	Exit 语句	78

第 4 章 数组 /80

4.1	数组概述	80
-----	------	----



4.2	数组的声明	80
4.3	数组的结构	82
4.4	默认数组与嵌套数组	85
4.4.1	默认数组	85
4.4.2	嵌套数组	86
4.5	数组初始化	87
4.5.1	数组的初始化方式	88
4.5.2	一维数组初始化	89
4.5.3	多维数组初始化	90
4.5.4	嵌套数组的初始化	91
4.6	数组的基本操作	92
4.6.1	数组元素的引用	92
4.6.2	数组元素的赋值	93
4.6.3	数组元素的输出	94
4.6.4	数组元素的复制	97
4.7	数组的常用属性和方法	98
4.8	动态数组	102
4.9	结构数组	105

第5章 过程 /108

5.1	Sub 过程	109
5.1.1	Sub 过程的创建	109
5.1.2	Sub 过程的调用	111
5.1.3	事件过程	112
5.2	Function 过程	114
5.2.1	Function 过程的创建	114
5.2.2	Function 过程的调用	117
5.2.3	常用内部函数	117
5.3	参数传递	124
5.3.1	形参和实参	124
5.3.2	按值传递	125
5.3.3	按引用传递	127
5.3.4	数组参数传递	128
5.4	可选参数与可变参数	131
5.4.1	可选参数	131
5.4.2	可变参数	132
5.5	Main 过程	134
5.6	递归调用	134

**第 6 章 类和对象 /136**

6.1	面向对象程序设计概述	136
6.2	类	137
6.2.1	类的定义	137
6.2.2	类的建立	139
6.2.3	类的字段和属性	141
6.2.4	类的方法	145
6.2.5	类的事件	145
6.2.6	类的构造函数	151
6.3	对象	153
6.3.1	对象的创建	154
6.3.2	对象的释放	154
6.3.3	对象之间的关系	155
6.4	类与命名空间	155
6.5	重载	157
6.5.1	方法签名	157
6.5.2	方法重载	158
6.6	共享成员	159

第 7 章 继承 /162

7.1	继承的基础知识	162
7.2	继承的实现	163
7.3	重载、覆盖和隐藏	164
7.3.1	重载	165
7.3.2	覆盖	166
7.3.3	隐藏	169
7.3.4	隐藏和访问级别	172
7.3.5	隐藏和覆盖	174
7.4	Me、MyBase 和 MyClass	174
7.4.1	Me 关键字	175
7.4.2	MyBase 关键字	175
7.4.3	MyClass 关键字	176
7.5	构造函数的继承	177
7.6	基于继承的多态	179
7.7	抽象类	180

第 8 章 接口、结构和委托 /184

8.1 接口	184
8.1.1 接口的定义	185
8.1.2 接口的实现	187
8.1.3 创建和实现接口	188
8.1.4 基于接口的多态	190
8.2 联编与多态	192
8.3 结构	194
8.3.1 结构的声明	194
8.3.2 结构和类	196
8.4 委托	198
8.4.1 委托的声明	198
8.4.2 委托调用方法	199
8.4.3 委托与事件	200

第 9 章 窗体和常用控件 /205

9.1 窗体	205
9.1.1 窗体的基本元素	205
9.1.2 窗体的类别	206
9.1.3 窗体的属性	207
9.1.4 窗体的方法	210
9.1.5 窗体的事件	211
9.1.6 窗体设计器	213
9.2 常用控件	215
9.2.1 绘制窗体控件	215
9.2.2 控件的命名	217
9.2.3 显示类控件	217
9.2.4 Button 控件和选择类控件	226
9.2.5 组合类控件	231
9.2.6 其他常用控件	234

第 10 章 文件系统 /242

10.1 文件概述	242
10.1.1 文件结构	242
10.1.2 文件类型	243
10.2 传统的文件访问方法	244
10.2.1 顺序访问模式	246



10.2.2	随机访问模式	254
10.2.3	二进制访问模式	259
10.3	文件系统对象	259
10.3.1	FileSystemObject 对象	260
10.3.2	Driver 对象	262
10.3.3	Folder 对象	263
10.3.4	File 对象	264
10.3.5	TextStream 对象	267
10.4	System.IO 模型	268
10.4.1	File 类和 FileInfo 类	268
10.4.2	Directory 类和 DirectoryInfo 类	271
10.4.3	FileStream 类	273
10.4.4	StreamReader 类和 StreamWriter 类	274
10.4.5	BinaryReader 类和 BinaryWriter 类	279
 第 11 章 Visual Basic 2012 数据库技术 /285		
11.1	数据库的基本概念	285
11.1.1	信息、数据、数据处理及数据库	285
11.1.2	数据库系统	287
11.1.3	数据库技术的发展	288
11.1.4	三个世界的转化	289
11.1.5	概念模型和数据模型	290
11.2	SQL 语言	293
11.2.1	数据定义	294
11.2.2	数据操作	295
11.2.3	数据查询	296
11.2.4	数据控制	297
11.3	ADO.NET 概述	297
11.3.1	ADO.NET 的发展	298
11.3.2	ADO.NET 的结构	298
11.4	使用 ADO.NET 访问数据库	301
11.4.1	Connection 对象	302
11.4.2	Command 对象	307
11.4.3	DataReader 对象	312
11.4.4	DataAdapter 对象	315
11.4.5	DataSet 对象	318

第 1 章

初识 Visual Basic 2012

1.1 Visual Basic 2012

Visual Basic 2012 是从 BASIC 发展而来的,后者是 20 世纪 60 年代中期由 Dartmouth 大学的 John Kemeny 和 Thomas Kurtz 教授所开发,作为一种能够快速方便地编写简单程序的语言。Visual Basic 2012 是由 Microsoft 公司开发的包含协助开发环境的事件驱动编程语言,它拥有图形用户界面(GUI)和快速应用程序开发(RAD)系统,可以使用 DAO(Database Access Object,数据访问对象)、RDO(Remote Data Objects,远程数据对象)和 ADO(ActiveX Data Objects,ActiveX 数据对象)方便地连接数据库,或者轻松地创建 ActiveX 控件。程序员可以轻松地使用 Visual Basic 2012 提供的组件快速建立一个应用程序。

1.1.1 Visual Basic 的发展

在 20 世纪 80 年代末,Microsoft 公司开发了 Microsoft Windows 图形用户界面(GUI),即操作系统的可视化部分,用户可以通过 GUI 与操作系统进行交互。随着 Windows GUI 的创建,BASIC 语言也要求可视化,因此 BASIC 自然而然地演变为 Visual Basic,它是 Microsoft 公司于 1991 年推出的。在 Visual Basic 问世之前,开发基于 Microsoft Windows 的应用程序是个艰苦而又笨拙的过程,自 Visual Basic 问世之后,Windows 应用程序的开发变得极为轻松。目前 Visual Basic 的最新版本是 2012 版,它是一种面向对象、事件驱动的可视化编程语言。程序员可以通过一种称为集成开发环境(IDE)的工具,极为快速方便地编写、运行、测试和调试 Visual Basic 程序。

- Visual Basic 1.0: 1991 年 Microsoft 公司发布了 Visual Basic 1.0 的 Windows 版和 DOS 版,使编程语言和用户界面连接了起来,这个进步在当时引起了很大的轰动。许多专家把 Visual Basic 1.0 的出现当做软件开发史上一个具有划时代意义的事件。
- Visual Basic 2.0: 1992 年 11 月,Microsoft 公司发布 Visual Basic 2.0,对上一版本的界面和速度进行了改善。
- Visual Basic 3.0: 1993 年,Visual Basic 3.0 发布了标准版和专业版两个版本。Visual Basic 3.0 提高了数据库编程的能力,其中增加了一个数据引擎,可以直接读取 Access 数据库。

- Visual Basic 4.0: 1995 年, Visual Basic 4.0 发布了 32 位版本和 64 位版本。引入了面向对象的程序设计思想,增加了对类的支持;同时又引入了“控件”的概念,使得许多事先编好的程序可以被拿来直接使用。
- Visual Basic 5.0: 1997 年, Visual Basic 5.0 发布,其中 32 位版本可以兼容 Visual Basic 4.0 的 16 位程序,并且增加了对用户自定义控件的支持。
- Visual Basic 6.0: 1998 年, Visual Basic 6.0 发布,它包括 3 个版本:学习版、专业版和企业版。此版本是经典的版本,至今也无法完全取代。
- Visual Studio .NET 7.0: 2001 年 Visual Basic .NET 和 .NET Framework 发布。2002 年 Visual Basic .NET 2002(v7.0)问世,它被包含在 Visual Studio 套装中,由于引入了 .NET Framework,其中使用了新的核心和特性,所以许多 Visual Basic 程序员都要改写程序。2003 年 Visual Basic .NET 2003(v7.1)问世。
- Visual Basic .NET 8.0: 2005 年 Visual Basic 2005(v8.0)问世,Microsoft 公司为初学者和学生提供了 Visual Basic 2005 免费简化版本 Express Edition。Visual Basic 2005 的显著优点是可以直接编写 Windows XP 风格的控件,缺点是其编写的小程序也需要近 10MB 的内存空间。
- Visual Basic .NET 9.0: 2008 年 Visual Basic 2008(v9.0)发布。经过几年的发展,此时的 Visual Basic 2008 已经成为一种专业化的开发语言和环境,用户使用它可以编写企业水平的客户/服务器程序以及强大的数据库应用程序。
- Visual Basic 2010: 2010 年 Visual Basic 2010 随着 Visual Studio 2010 一起发布。此版本的 Visual Basic 增加了自动实现的属性、集合初始值设定项、隐式行断续、突出显示引用等诸多新功能。
- Visual Basic 2012: 2012 年 9 月 12 日微软公司在西雅图发布 Visual Studio 2012。Visual Studio 2012 是目前为止最卓越的版本。它有着全新的外观和感受,随着 Windows 8 的发布,界面已经发生了显著的变化。对于 Web 开发,Visual Studio 2012 也提供了新的模板、更优秀的发布工具和对新标准(如 HTML5 和 CSS3)的全面支持,以及 ASP.NET 中的最新特性。同时,Visual Studio 2012 还提供了一些可以增进团队生产力的功能和云功能等。

Visual Basic 1.0~4.0 都必须将源代码编译成 Visual Basic 伪代码后解释执行。自 Visual Basic 5.0 后,Visual Basic 支持将源代码编译成本机代码,但必须在 Visual Basic 运行库的支持下工作。

1.1.2 Visual Basic 2012 新特点

从 Visual Basic 1.0 到现在的 Visual Basic 2012,Visual Basic 在不断地完善自己的功能与性能,每一个新的版本都在前一个版本的基础上有所改善与提高,Visual Basic 2012 也有着自己的新特点或增强的性能。

1.1.2.1 异步功能

新的异步功能是一种简单优雅的编写异步代码的技术,使异步编程几乎变得和同步

编程一样简单。

当编写的用户界面反应比较迟钝,或提供服务的服务器能力不够时,可以考虑使用异步编程技术。在传统的异步编程中,需要在代码中设置一个回调(callback)来表达异步操作完成之后的逻辑。这一点是异步代码相比同步代码所多出的复杂结构。在使用 Visual Basic 2012 的异步功能之后,用户在调用异步函数时将不再需要回调,也不再需要将代码分割成众多的小方法或是 lambda 表达式。

在异步功能中,Async 和 Await 两个关键词将被用于异步编程技术中。被 Async 关键字修饰的方法被称作异步方法,此方法作为一个任务。当一个方法使用 Await 关键字来等待一个任务时,此方法将立即退出,直至这个被等待的任务完成后才恢复执行。同时,在 Visual Basic 2012 中,调用一个异步方法,不分配任何额外的线程。它可能简单地使用现有的 I/O 完成线程。

1.1.2.2 迭代器

迭代器被用来在列表或数组等集合上执行定制好了的迭代。在 Visual Basic 2010 之前,在 Visual Basic 中是不能使用迭代器的。迭代器被执行时,用户可以使用 Yield 语句每次返回集合中的一个元素。当代码执行到 Yield 语句时,当前的代码位置将被记录,下次再次执行迭代时将从记录的位置继续执行。用户可以使用 For Each...Next 语句来使用迭代器。

1.1.2.3 调用层次结构

调用层次结构使用户可以通过代码导航显示如下信息:

- 被选中的方法、属性或构造器调用和被调用的信息。

- 实现了一个接口的所有成员。
- 重载一个虚拟或抽象成员的所有信息。

调用层次结构可以使用户更好地理解代码的流程,并能更准确地评估代码变化所带来的影响。“调用层次结构”视图可以通过选择“视图”→“调用层次结构”来打开,如图 1.1 所示。图 1.2 是“调用层次结构”视图。

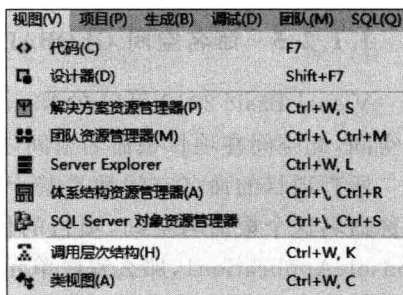


图 1.1 选择“调用层次结构”选项

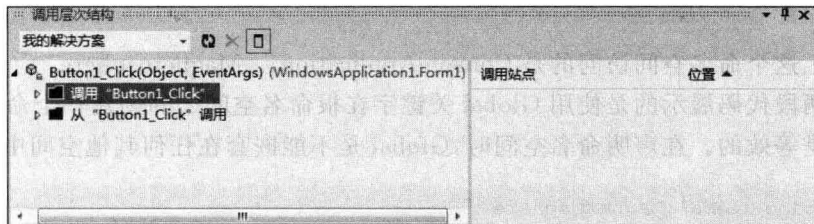


图 1.2 “调用层次结构”视图

1.1.2.4 调用者信息

Visual Basic 2012 引入了调用者信息的功能,使用此功能可以很容易地获取调用一个方法的调用者的信息。用户可以通过调用者信息来获取源代码文件的路径、源代码的行号和调用者的成员名称。这些信息对程序员跟踪、调试代码以及创建诊断工具都很有帮助。

程序在编译时会识别 3 个特别的属性: <CallerMemberName>、<CallerLineNumber>、<CallerFilePath>。下面的代码显示了这 3 个属性的使用方法。程序执行结果将显示调用者名称、文件路径和源文件中调用者所在位置的行号。

```
Private Sub DoProcessing()  
    TraceMessage()  
End Sub  
  
Public Sub TraceMessage(  
    <CallerMemberName>Optional memberName As String=Nothing,  
    <CallerFilePath>Optional sourcefilePath As String=Nothing,  
    <CallerLineNumber()>Optional sourceLineNumber As Integer=0)  
  
    Trace.WriteLine("member name: " & memberName)  
    Trace.WriteLine("source file path: " & sourcefilePath)  
    Trace.WriteLine("source line number: " & sourceLineNumber)  
End Sub
```

1.1.2.5 命名空间 GLOBAL

Visual Basic 2012 可以在 Namespace 语句中使用 Global 关键字定义一个项目的命名空间,此空间在项目根命名空间之外。

所有项目的命名空间都是基于项目的根命名空间的。Visual Studio 默认用项目名称命名这个根命名空间,项目的代码就放在这个命名空间中。例如,如果项目名称为 ConsoleApplication1,那么程序单元将隶属于 ConsoleApplication1 命名空间。如果在此项目中声明了一个空间,如:

```
Namespace Magnetosphere
```

实际上这个命名空间访问的是 ConsoleApplication1. Magnetosphere。

下面两段代码展示的是使用 Global 关键字在根命名空间之外创建一个命名空间,这两段代码是等效的。在声明命名空间时,Global 是不能嵌套在任何其他空间中的。

```
Namespace Global.Magnetosphere
```

```
End Namespace
```



```
Namespace Global
    Namespace Magnetosphere
        End Namespace
    End Namespace
```

1.2 .NET Framework

Microsoft 公司发布的 .NET Framework 简称为 .NET, 是支持生成和运行下一代应用程序和 Web 服务的 Windows 内部组件, 它提供了托管执行环境和简化的开发和部署, 是一种采用系统虚拟机运行的编程平台, 以通用语言运行库 (Common Language Runtime, CLR) 为基础, 支持多种语言 (Visual Basic、C# 和 C++ 等) 的开发。

1.2.1 Visual Basic 与 .NET Framework

由计算机编程语言编写的程序, 若要在计算机上运行, 必须先要将源程序代码转换成计算机能够理解和运行的机器码。有两种比较常见的执行方式能够完成这种转变, 一种是编译执行, 一种是解释执行。编译执行是把源程序代码编译成机器码后, 再执行机器码。例如, C 语言程序必须给出完整程序, 编译器通过几次扫描、翻译、编排和链接, 变成 exe 文件执行。采用编译执行的方式, 程序执行时间短, 效率高。解释执行并不把整个程序变成机器码, 而是按顺序, 读一句, 解释一句, 执行一句, 所以不必给出完整程序就可以执行。例如, 浏览器处理网页, 每读取一句, 就可以解释执行一句, 不用等整个网页读取完毕后再处理。

编译执行和解释执行是一种比较传统的程序代码执行方式, 另一种比较新型的执行方式是“中间码+虚拟机”的形式。在这种执行方式中, 程序源代码被编译成一种特殊的中间码, 而这种中间码并不能像机器码那样被直接执行, 因为计算机并不能理解这种中间码。理解中间码的是一种叫做“虚拟机”(Virtual Machine)的“装置”(虚拟机指通过软件模拟的具有完整硬件系统功能的、运行在一个完全隔离环境中的完整计算机系统)。虚拟机负责管理和执行中间码, 可以是编译执行, 也可以是解释执行。 .NET Framework 类似于管理和执行中间码的虚拟机, Visual Basic 则可以编写源代码。

1.2.2 .NET Framework 简介

.NET Framework 具有两个主要组件: 公共语言运行库和 .NET Framework 类库。公共语言运行库是 .NET Framework 的基础。可以将运行库看作一个在执行时管理代码的代理, 它提供内存管理、线程管理和远程处理等核心服务, 并且还强制实施严格的类型安全以及可提高安全性和可靠性的其他形式的代码准则。事实上, 实施代码管理是运行库的基本原则。以运行库为目标的代码称为托管代码, 而不以运行库为目标的代码称为非托管代码。 .NET Framework 的另一个主要组件是类库, 它是一个综合性的面向对象的可重用类型集合, 可以使用它开发多种应用程序, 这些应用程序包括传统的命令行或