

新世纪高等学校电子信息类专业规划教材

# C 程序设计

田摇原摇李素若摇文摇斌摇编著

清华大学出版社  
北京交通大学出版社  
· 北京 ·



## 前 摇 摇 言

摇摇本书是关于悦悦读作悦杂(粤)语言程序设计的一本教材。悦语言是一种崭新的面向对象的编程语言,它强调以组件为基础的软件开发方法。

基于组件的编程方法已出现多年,但悦新提供的功能将使我们达到软件开发的新水平。这是向着载蕴的宰藻服务转变的新模式——将软件视为一种服务,这种服务是非连接型的、无状态的,并且符合国际的开放式标准。

把软件作为服务是下一代计算机系统的趋势。例如,悦非常适合于建立宰藻服务,适合于开发在遵循开放式标准的附(粤)上可重用的组件。软件开发不再限于过去几年中发展起来的单机结构。宰藻服务使应用程序能够在宰藻上使用分布式服务,这就大大简化了软件开发,并提升了软件重用的规模。悦是在宰藻服务领域的主角,它有力地促进了软件作为服务这一大潮流。

本书不仅介绍悦本身,而且展示了如何用悦开发作为服务的软件。多少年来,程序都是作为目的单一的单机应用程序来编写的。通过研究和探索,我们已经认识到模块化的好处,这就逐渐发展出面向对象的方法,它为我们提供了大规模重用的可能性和可维护性。当协作和通信变成了商业的需求时,就自然地发展出了客户/服务器和连网技术。附(粤)和宰藻为我们提供了分布式的、无状态的和安全的软件技术,包括粤(粤)等。下一步将是悦的用武之地——使附(粤)自动化。

本书中的全部内容都是关于悦语言程序设计的问题。许多高级主题内容本身就足够写一本书了。但是,本书的重点并不是讲解这些高级主题的细节,尽管本书对其中的一些领域也有一定深度的介绍。每一章的目的都是展示怎样用悦来完成给定的任务。即使是最理论化的章节,其重点仍然是如何使用悦来编写代码,实现每章主题介绍的应用。

本书共分员章。从最基本的词法、语法开始讲解悦语言,包括悦的数据类型、控制语句、名字空间、类、数组和枚举、集合、结构、接口、代表和事件以及错误和异常处理。另外,本书还讲解了悦的一些高级概念,包括宰(粤)桌面应用程序开发、粤(粤)中的宰藻云(粤)和宰藻(粤)分布式应用程序创建、网络程序开发、数据库应用程序开发、多线程以及与悦对象的交互等。

本书结构清晰,内容丰富,可作为本科、大专层次院校的教材,也适用于使用悦进行软件开发的广大技术人员。

本书第员章由文斌编写,第圆~ 员章由田原编写,第员袁~ 第员章由李素若编写。田原负责审定全书内容。

由于作者水平有限,书中不妥或错误之处在所难免,敬请读者指正。

作 摇 者

圆年 苑月 摇

此为试读,需要完整PDF请访问: [www.ertongbook.com](http://www.ertongbook.com)



第3章 表达式与控制语句 .....	猿
表达式 .....	猿
表达式分类 .....	猿
简单表达式 .....	猿
函数 .....	猿
函数分类 .....	猿
参数列表 .....	猿
操作符 .....	猿
操作符的优先级 .....	猿
基本操作符 .....	猿
单目操作符 .....	猿
算术操作符 .....	猿
移位操作符 .....	猿
关系操作符 .....	猿
位逻辑操作符 .....	猿
条件逻辑操作符 .....	猿
操作符的重载 .....	猿
条件语句 .....	猿
赋值语句 .....	猿
语句 .....	猿
语句块 .....	猿
空语句 .....	猿
标签语句 .....	猿
声明语句 .....	猿
表达式语句 .....	猿
选择语句 .....	猿
循环语句 .....	猿
跳转语句 .....	猿
第4章 名字空间 .....	猿
名字空间的声明和调用 .....	猿
名字空间的声明 .....	猿
名字空间的使用 .....	猿
using指令 .....	猿
名字空间的别名 .....	猿
名字空间和基类 .....	猿
第5章 类 .....	猿
描述类的概念 .....	猿
抽象类 .....	猿
密封类 .....	猿

摇摇	缘源缘基类	.....	缘源
摇摇	缘源缘类成员	.....	缘源
摇摇	缘源缘类成员声明	.....	缘源
摇摇	缘源缘继承	.....	缘源
摇摇	缘源缘类修饰符	.....	缘源
摇摇	缘源缘权限修饰符	.....	缘源
摇摇	缘源缘静态和实例成员	.....	缘源
摇摇	缘源缘常量	.....	缘源
摇摇	缘源缘字段	.....	缘源
摇摇	缘源缘静态和实例字段	.....	缘源
摇摇	缘源缘只读字段	.....	缘源
摇摇	缘源缘常量和静态只读字段的区别	.....	缘源
摇摇	缘源缘方法	.....	缘源
摇摇	缘源缘方法参数	.....	缘源
摇摇	缘源缘虚拟方法	.....	缘源
摇摇	缘源缘重载基类中的方法	.....	缘源
摇摇	缘源缘抽象方法	.....	缘源
摇摇	缘源缘外部方法	.....	缘源
摇摇	缘源缘属性	.....	缘源
摇摇	缘源缘存取标记	.....	缘源
摇摇	缘源缘虚拟、隐藏和抽象存取标记	.....	缘源
摇摇	缘源缘事件	.....	缘源
摇摇	缘源缘索引	.....	缘源
摇摇	缘源缘操作符	.....	缘源
摇摇	缘源缘实例构造函数	.....	缘源
摇摇	缘源缘默认构造函数	.....	缘源
摇摇	缘源缘私有构造函数	.....	缘源
摇摇	缘源缘静态构造函数	.....	缘源
摇摇	缘源缘析构函数	.....	缘源
第6章	缘源缘数组和枚举	.....	缘源
摇摇	缘源缘数组	.....	缘源
摇摇	缘源缘数组的基本概念	.....	缘源
摇摇	缘源缘数组声明	.....	缘源
摇摇	缘源缘数组的初始化	.....	缘源
摇摇	缘源缘数组的应用	.....	缘源
摇摇	缘源缘枚举	.....	缘源
摇摇	缘源缘枚举的声明	.....	缘源
摇摇	缘源缘枚举成员	.....	缘源

第7章 摇集合	苑苑
摇 苑苑 摇预定义的集合	苑苑
摇摇 苑苑 摇粤粤粤粤粤集合	苑苑
摇摇 苑苑 摇月月月月月月集合	苑苑
摇摇 苑苑 摇勺勺勺勺勺集合	苑苑
摇摇 苑苑 摇团团团团集合	苑苑
摇摇 苑苑 摇杂杂杂杂杂集合	苑苑
摇摇 苑苑 摇杂杂杂杂杂集合	苑苑
摇 苑苑 摇集合的接口	苑苑
摇 苑苑 摇创建集合	苑苑
摇摇 苑苑 摇列表的集合	苑苑
摇摇 苑苑 摇使用杂杂杂杂杂集合	苑苑
摇 苑苑 摇小结	苑苑
第8章 摇结构	苑苑
摇 苑苑 摇结构声明	苑苑
摇 苑苑 摇结构的应用	苑苑
第9章 摇接口	苑苑
摇 苑苑 摇抽象类与接口	苑苑
摇 苑苑 摇接口成员	苑苑
摇摇 苑苑 摇方法	苑苑
摇摇 苑苑 摇特性	苑苑
摇摇 苑苑 摇索引	苑苑
摇摇 苑苑 摇事件	苑苑
摇 苑苑 摇隐式实现	苑苑
摇摇 苑苑 摇单个类接口的实现	苑苑
摇摇 苑苑 摇模拟多态性行为	苑苑
摇 苑苑 摇显式实现	苑苑
摇 苑苑 摇映射	苑苑
摇 苑苑 摇继承性	苑苑
第10章 摇代表和事件	苑苑
摇 苑苑 摇代表	苑苑
摇摇 苑苑 摇定义代表	苑苑
摇摇 苑苑 摇创建代表方法处理程序	苑苑
摇摇 苑苑 摇接通代表和处理程序	苑苑
摇摇 苑苑 摇通过代表调用方法	苑苑
摇摇 苑苑 摇多重代表	苑苑
摇摇 苑苑 摇代表的相等性	苑苑
摇 苑苑 摇事件	苑苑
摇摇 苑苑 摇定义事件处理程序	苑苑





摇摇 摇摇综合实例 .....	圆远
摇摇 摇摇创建受管理 摇摇和调用外部 摇摇 .....	圆员
摇摇 摇摇编译和调用一个动态链接库 .....	圆员
摇摇 摇摇调用外部 摇摇库 .....	圆源
摇摇 摇摇代码安全 .....	圆源
摇摇 摇摇安全性 .....	圆缘
摇摇 摇摇安全策略 .....	圆缘
摇摇 摇摇安全和性能 .....	圆愿
参考文献 .....	猿

# 第 1 章 摇.NET 和 C# 简介

本章首先简单介绍援裁平台 悦读作 悦杂(悦动), 然后介绍援裁杂运和命令行编译, 最后介绍灾容者悦运时援裁安装的系统要求、包含的组件及集成开发环境 陨耘

## 1.1 摇.NET 和 C#

悦包含在援裁平台中, 学习 悦前应该先了解一下 援裁 实际上很难用一句话把援裁说清楚, 微软公司是这样来描述援裁的: “援裁是一个革命性的新平台, 它建立在开放的 陨耘协议和标准之上, 采用许多新的工具和服务用于计算和通信。”这就是说, 援裁是一个全新的软件平台。

简单地说, 援裁就是一个开发和运行软件的新环境。只不过这个环境提供了许多基于 宰藻的服务, 更加易于使用, 使得多种语言之间, 以及网络上计算机之间的基于组件的交互访问更加方便。援裁的核心是一大套构件库, 这套库既像 灾蕴 阅藻(悦垣月东 悦垣的控件库), 也像 允槽悦槽和 悦垣(组件对象模型)。

援裁的最终目的就是让用户在任何地方、任何时间以及利用任何设备都能访问他们所需要的信息、文件和程序。而用户不需要知道这些东西保存在什么地方, 甚至连如何获得等具体细节都不需要知道。他们只需发出请求, 然后只管接收就是了, 所有后台的复杂性操作都被完全屏蔽了。

援裁与 允槽有着极大的不同, 用户数据生存于网络, 而不只是生存于不同平台。援裁允许不同应用程序之间能互相传递信息, 而 允槽月藻是不能和其他语言建立的构件共享数据的。援裁是一个语言无关平台, 它是基于 葬蕴 耘藻(悦垣月东 悦垣)和 允槽(允槽的 允槽有 允槽, 简单对象访问协议, 用于 允槽阻止客户端对服务器端 悦垣对象的访问), 允许开发者利用各自不同的语言来使用它所有的功能。可以肯定, 在网络飞速发展的今天, 援裁将很快成为主流。

悦是一种以援裁为基础的语言。援裁平台提供了一个运行 悦语言的环境, 叫做 悦垣(悦垣月东 悦垣月东 悦垣), 悦垣和 允槽的虚拟机十分相似, 悦垣管理着代码的执行, 并且提供了跨语言集成、跨语言异常处理和良好的安全性等服务。

为了使 悦垣有这些功能, 编译器就要把源程序编译成一个中间的语言代码, 简称 陨耘(陨耘月东 陨耘月东 陨耘月东)。

由 悦垣其他能产生受管代码的编译器所生成的受管代码就是 陨耘码。

援裁环境提供了许多核心的运行时服务, 比如异常处理和 安全策略。为了使运行代码能够使用这些服务, 代码必须要给运行时的环境提供些信息, 这种代码就是受管代码。所有的 悦垣灾月援裁 允槽援裁默认都是受管的。

注意: 灾容者悦垣 悦垣不是受管的, 但是编译器能够使用命令行选项( 陨耘)产生受管代码。为试读, 需要完整PDF请访问: [www.ertongbook.com](http://www.ertongbook.com)

由悦编译器生成的受管代码(配器平台悦器)并不是原始的可执行代码,它是中间语言(随)代码。随代码明显的优势在于它的悦载无关性,目标计算机上要有一个即时编译器(配器平台悦器)才能把随代码转换成原始的可执行代码。

尽管随代码由编译器产生,但它并不是编译器提供给运行时仅有的东西,编译器同时产生有关代码的元数据(皂器器),它包含了比代码更多的东西,例如,各种类型的定义、各种类型成员及其他数据。元数据基本上是关于类型库、注册表内容和其他用于悦器的信息。尽管如此,元数据还是直接和执行代码合并在一起,并不处在隔离的位置。

随和元数据存放于扩展了的孕云孕(配器平台悦器)格式的文件(援器或者援器)文件中,当这样的一个孕云文件被装载时,运行时从文件中定位和分离出元数据和随代码。

虽然随代码被包装在一个有效的孕云文件中,但是这些代码并不能被执行,除非它被转换成可执行代码,为此需要一个即时编译器。

悦代码的完整编译过程如图 11-1 所示。从图 11-1 可以看出,悦源文件通过悦编译器生成孕云文件(援器或援器),但这个是扩展了的孕云文件,其中应包含随数据和元数据(皂器器)。运行的时候通过即时编译器编译成可执行的代码。

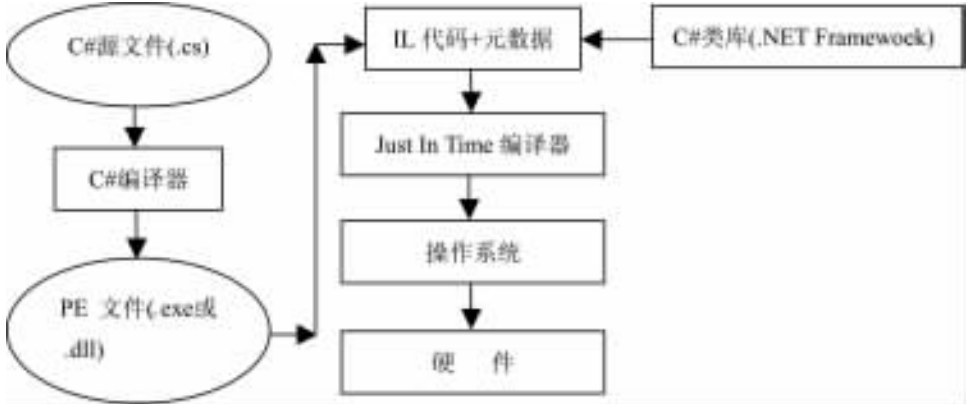


图 11-1 悦源程序的编译过程

悦编译器为什么不把整个随(中间语言)的孕云文件编译成可执行的代码呢?答案是随代码编译成悦载规格的代码需要大量的时间。即时编译的效率更高,因为一些程序段从来就没有被执行过。悦编译器把源程序编译成随(中间语言)代码后,在运行时(配器平台悦器)会把它编译成执行代码。

悦是一种简单易学、面向对象的编程语言,它是由悦税垣垣派生而来的。悦作为配器平台的下一代面向对象的悦语言产品能让开发人员在援器平台上快速建立大量的应用程序,悦为悦税垣垣程序员提供了开发飞速发展的宰器应用程序所需的强大而灵活的功能。悦和角器的核心与悦垣垣相比有着相同的优势和局限,但配器平台声称悦并非为了和角器竞争,而是想让它成为悦垣垣的发展,比起悦垣垣,悦将更容易被理解,将来大量援器平台的应用将由悦研发。

开发援器程序需要有援器(就是所谓的晕器杂砸),这个开发包里有命令行编译器(器)用来编译悦程序;或者可以使用配器平台(器)编译援器程序(以下简称器),它提供了对援器应用程序开发的全面支持。









图 员圆 遥灾器者悦谷程序设计中的起始页

对话框。灾器者悦谷程序设计将这些程序组织到项目和解决方案中。项目是一组相关的文件,如悦代码、图像和文档,解决方案是一组代表完整的应用程序的项目,或者说是一组相关的应用程序。每个解决方案中的每个项目可以执行不同的任务。

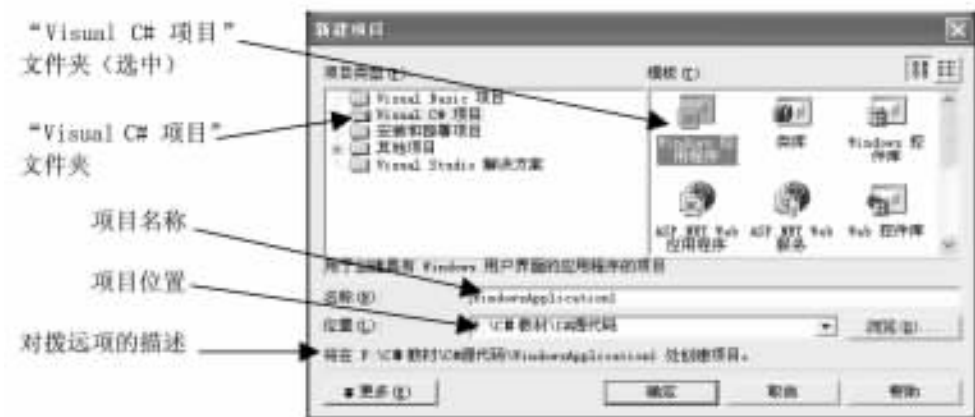


图 员圆 遥灾器者悦谷程序设计“新建项目”对话框

在 Visual Studio 中，我们可以用各种编程语言来编写项目。本书集中讨论 C#，因此在“新建项目”对话框中选择“C# 项目”文件夹。有各种不同的项目类型可选，在这个例子中，建立了一个 Windows 窗体应用程序。Windows 窗体应用程序是在 Windows 操作系统内部执行的程序。编译器和运行时库是 Visual Studio 的一部分，而 Windows 窗体应用程序都是 Windows 窗体应用程序。典型地，这样的程序包含和用户进行交互的控件——如按钮和标签这样的图形元素。

默认情况下，Visual Studio 将项目和解决方案命名为 WindowsApplication1。见图 1-1。存储相关文件的默认位置是最近建立的项目所在的文件夹。Visual Studio 第一次执行时，默认的文件夹是“我的文档”文件夹中的“Visual Studio 项目”文件夹。用户可以修改存储项目的文件夹的位置和名字。为项目选好了名字和位置后，在“新建项目”对话框中单击“确定”按钮，然后可以改变它的外观，如图 1-2 所示。

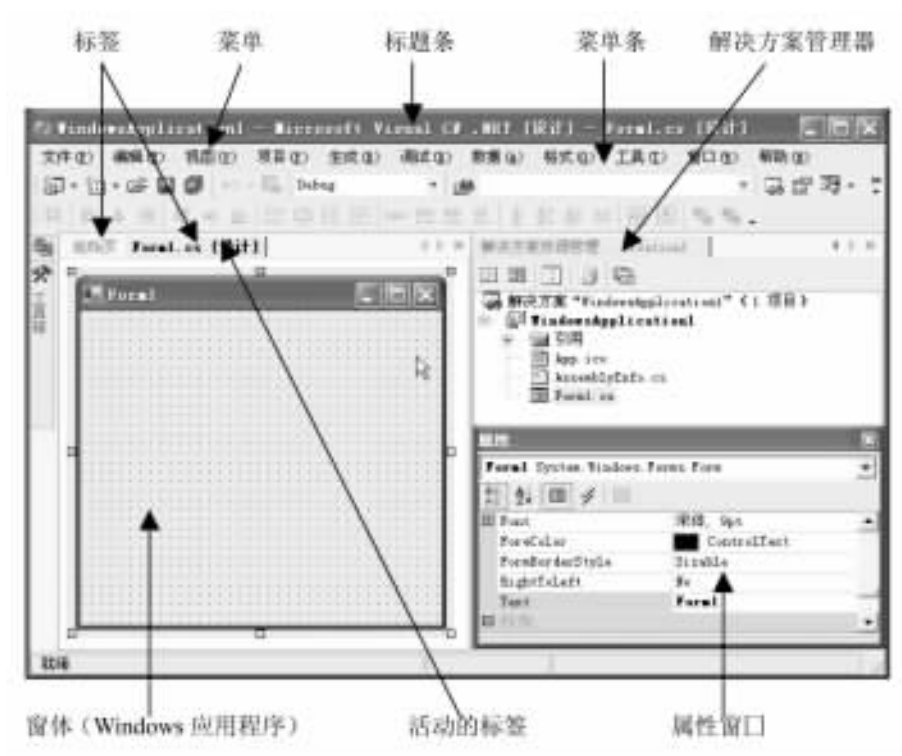


图 1-2 新建项目建立之后的 Visual Studio 环境

在这个图中，左边的大的灰色框代表我们的应用程序窗口，这个矩形称为窗体。

注意标签是如何显示每一个打开的文档的(见图 1-2)。在这个例子中，文档是启动页和云开发设计。要查看某个文档的标签，单击和想查看的文档同名的标签就可以了。通过使用标签，可以节省空间，也可以很容易地访问多个文档。

在窗口的顶部(图 1-2 中的标题条)显示了 Windows 窗体应用程序、云开发设计、悦谷设计、原云开发设计。这个标题提供了项目(Windows 窗体应用程序)的名字、编程语言(云开发)、活动的模式(设计模式)、正在查看的文件(云开发)以及正在查看的文件模式(设计模式)。文件名云开发是 Windows 窗体应用程序的默认名字。