Mc Graw Hill Education

Schildt 的经典编程著作——针对C# 3.0进行了全面修订和更新

# C# 3.0
## The Complete Reference

# C# 3.0
# 完全参考手册

“要找到在现实世界中使用 C# 的答案，只需购买这本不可或缺的参考手册即可！”
—— *Michael Howard, Microsoft*

(美) Herbert Schildt 著
赵利通 译

免费源代码下载

- 全面透彻地介绍 C# 语言
- 涵盖 C# 3.0 的新功能，包括 LINQ、拉姆达表达式和匿名类型
- 包括数百个浅显明了的示例

Herbert Schildt 是最优秀的编程畅销书作者，其创作的编程书籍在全球的销量已逾 350 万册。

清华大学出版社

# C# 3.0 完全参考手册

(美) Herbert Schildt 著

赵利通 译

清华大学出版社

北 京

Herbert Schildt
C# 3.0: The Complete Reference
EISBN: 978-0-07-158841-6


北京市版权局著作权合同登记号： 01-2009-2130



**图书在版编目(CIP)数据**

C# 3.0 完全参考手册/(美) 希尔特(Schildt, H.) 著；赵利通 译. —北京：清华大学出版社，2010.3
书名原文：C# 3.0：The Complete Reference
ISBN 978-7-302-22072-5
Ⅰ. ①C… Ⅱ. ①希… ②赵… Ⅲ. ①C 语言－程序设计－技术手册 Ⅳ. ①TP312-62
中国版本图书馆 CIP 数据核字(2010)第 026257 号

**责任编辑**：王 军 李 阳
**装帧设计**：孔祥丰
**责任校对**：成凤进
**责任印制**：何 芊
**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 刷 者**：清华大学印刷厂
**装 订 者**：三河市新茂装订有限公司
**经 销**：全国新华书店
**开 本**：185×260 **印 张**：54.75 **字 数**：1435 千字
**版 次**：2010 年 3 月第 1 版 **印 次**：2010 年 3 月第1 次印刷
**印 数**：1～3000
**定 价**：128.00 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：031786-01

# 前　言

编程人员往往喜欢精益求精，他们总是不断地想方设法提高程序的性能、效率和可移植性。因此，他们对所用的工具同样也要求甚多，特别是对编程语言的选择。编程语言的种类很多，但只有一小部分可称之为“伟大”。伟大的编程语言往往功能强大而又灵活，语法简洁且清晰，能够为创建正确的代码提供极大的方便，而不是设置障碍；能够不断地支持最新的功能，而不是逐渐被淘汰出局。最重要的是，伟大的编程语言注定会有这样一种无形的品质：让程序员在使用时感觉良好。C#就是这样一种编程语言。

C#是微软为支持.NET Framework 而创建的语言，它建立在丰富的程序设计资源之上，其首席设计师是长期以来公认的软件大师 Anders Hejlsberg。C#继承了至今世界上最成功的两种计算机语言：C 和 C++，它继承了 C 语言的语法、部分关键字和运算符，并以 C++定义的对象模型为基础加以改进。C#还和另一种非常成功的语言 Java 有紧密关系。

C#和 Java 有共同的起源，像双胞胎一样，但是在许多重要方面两者也有所不同，所以 C#和 Java 更像是堂兄弟。例如，两者都支持分布式程序设计，并且都使用中间代码来获得安全性和可移植性，但是两者的实现细节是不同的。它们都提供了大量运行时错误检查、安全性和托管执行，但是同样在细节方面有所区别。然而，与 Java 不同的是，C#提供了对指针的访问——指针是 C++支持的一种功能。因此，C#将 C++的原始功能与 Java 的类型安全性组合在一起。而且，它在安全性和功能之间达到了最佳平衡，且实现了透明化。

在计算技术变革的历史中，为了适应计算环境的变化、计算机语言的发展，以及人们在思维方式和程序设计方式上的改变，程序设计语言得到了不断的发展，C#也不例外。在不断的提炼、适应和创新过程中，C#已经展示了它能不断满足现代编程人员需求的能力。随着 2000 年发布最初的 1.0 版本以来不断向 C#添加许多新的功能就是最好的证明。

C# 2.0 是 C#语言的第一个重大修订版本，该版本通过增加一系列的新功能来使编程人员更容易地编写更有弹性、更可靠和更高效的代码。毫无疑问，C# 2.0 中增加的最重要的功能是泛型。通过使用泛型，在 C#中创建类型安全、可重用的代码成为可能。因此，泛型的添加从根本上扩展了 C#语言的功能和作用范围。

C#语言的第二个重大修订版本就是 C# 3.0，这是 C#的最新版本，也是本书将要介绍的版

本。毫不夸张地说，C# 3.0 添加的功能已经重新定义了 C#的核心部分，同时在此过程中提高了计算机语言开发的门槛。在 C# 3.0 的许多创新功能中，最具代表性的两个功能是 LINQ 和拉姆达表达式。LINQ 代表语言集成查询，它允许通过使用 C#语言的元素创建数据库样式的查询。拉姆达表达式则实现功能样式的语法，它使用=>拉姆达运算符，并且频繁用于 LINQ 表达式中。

在学习本书的过程中可以了解到，LINQ 和拉姆达表达式的组合代表 C#中极端强大的功能子集。此外，它们都是创新的功能，重新定义了为许多不同类型的编程任务(而不仅是数据库查询)设计解决方案的方式。从本质上来说，这些功能允许按照新方式处理老问题，使用它们不仅可以简化解决方案，而且可以帮助从不同的角度形成关于问题的概念。这些功能改变了人们从事编程工作的思维方式。

由于 C#能够快速适应编程领域中不断变化的需求，因此它始终是充满活力的、不断创新的语言。总之，C#已成为现代计算中最为强大的、功能丰富的一种语言，任何程序员都不能忽略它的存在。编写本书的目的就是帮助读者掌握 C#语言。

**内容简介**

本书介绍了 C# 3.0，全书分为两个部分。第 I 部分全面讨论了 C#语言，包括 C# 3.0 版本中增加的新特性。这是全书内容比重最大的一部分，它描述了关键字、语法和一些定义 C#语言的功能，并且介绍了 I/O 操作、文件处理、反射和预处理程序。

第 II 部分研究了 C#类库，同时也是.NET Framework 类库。这是一个巨大的类库，由于篇幅有限，本书不可能深入探讨整个.NET Framework 类库，而只能集中地讨论 System 名称空间中包含的核心类库。另外，本部分还包括对集合、多线程处理和网络连接和 Windows Forms 的介绍。这些是类库中几乎每个 C#编程人员都会用到的部分。

**读者对象**

本书的读者无需具有任何编程经验。如果您已掌握 C++或者 Java 语言，那么阅读本书时将毫不费力，因为 C#和这两种语言有很多共同之处。如果之前没有任何编程经验，您也能够通过对本书的学习逐渐掌握 C#语言，但需要仔细研究每章中的示例。

**编程环境**

必须使用 Visual Studio 2008 或更高版本来编译和运行 C# 3.0 程序。

**在线源代码下载**

本书中所有程序的源代码都可以从 www.mhprofessional.com 和 www.tupwk.com.cn/downpage 网站上免费获得。

**反馈信箱**

本书是 Herbert Schildt 系列编程书籍之一，Herbert Schildt 网站(www.HerbSchildt.com)上展示了他所编著的更多书籍，并提供了他的联系方式。最后，请将您的反馈意见发送至：wkservice@vip.163.com，我们将不胜感激。

# 目　录

## 第Ⅰ部分　C# 语 言

## 第 II 部分 C#类库研究

PART I

# C# 语 言

第 I 部分主要讨论 C#语言的元素，包括：关键字、语法和运算符。另外还描述了一些和 C#语言紧密相关的 C#基本技术，比如：I/O 操作和反射。

# 1 CHAPTER

# C#的起源

C#是微软.NET 开发人员的首选语言，它具有的新功能经受住了时间的考验，且始终位于科技前沿，为现代企业计算环境提供了一种可用性强的高效编程方法。无论从哪个角度看，C#都是21 世纪最重要的编程语言之一。

本章的目的是回顾 C#发展的历程，包括：C#创建的原动力、设计理念以及它是如何受其他计算机语言影响的。本章也解释了 C#和.NET Framework 的相关性。正如您将看到的，C#语言和.NET Framework 协同工作，一起构建了一个高度优雅的编程环境。

## 1.1 C#的族谱

计算机语言并不是凭空存在的，相反，它们彼此相关，新语言都或多或少地受到它之前的语言的影响。类似于异花授粉的过程，一种语言的功能会被另一种语言沿用，但新的创新内容会被集成到现有的环境中，而陈旧的构造则会被消除。就这样，编程语言不断地进化，编程艺术也不断地完善。C#也不例外。

C#继承了多种程序设计语言的精髓，它直接继承了当今最成功的两种计算机语言——C 和 C++语言的功能，并且与 Java 有紧密联系。理解它们之间的关系对于理解 C#语言是很重要的，因此，我们将分析这三种语言的发展环境，以此来开始 C#语言的研究。

### 1.1.1 C 语言：现代程序设计的开端

C 语言的创建标志着现代程序设计时代的开始，它是 Dennis Ritchie 于 20 世纪 70 年代在一台使用 UNIX 操作系统的 DEC PDP-11 机上创建的。尽管一些早期的语言，最著名的如 Pascal 语言，已经取得了相当的成功，然而是 C 语言首先建立了面向过程编程的规范，至今仍适用。

C 语言成长于 20 世纪 60 年代的“结构化程序设计”变革。在结构化程序设计兴起之前，大型程序很难编写，因为程序逻辑容易混乱会呈现所谓的“无头绪的代码”，比如掺杂大量纷乱而难以跟踪的跳转、调用和返回。而结构化语言通过增加定义明确的控制语句、带有局部变量的子程序以及其他方法改进了这个局面。通过使用结构化的技术，程序变得更加组织良好和可靠，并且