

## <<C#程序设计语言>>

### 图书基本信息

书名：<<C#程序设计语言>>

13位ISBN编号：9787111347781

10位ISBN编号：7111347781

出版时间：2011-8-1

出版时间：机械工业出版社

作者：（美）Anders Hejlsberg,（美）Mads Torgersen,（美）Scott Wiltamuth,（美）Peter Golde

页数：559

译者：陈宝国,黄俊莲,马燕新

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>

## <<C#程序设计语言>>

### 内容概要

C#语言结合了快速应用开发语言的高效和C/C++语言的强大。

本书全部内容更新到C#

4.0版，提供了C#

4.0语言的完整规范、参考资料、范例代码和来自12位卓越的C#大师的详细注解。

这些注解所达到的深度和广度在其他书中难得一见。

C#

4.0新添加动态绑定、命名和可选参数以及协变和逆变泛型等特性，恰到好处的注解解释了这些特性的重要性、使用方法、与其他语言的关系等。

对任何希望深入理解C#的程序员来说，本书都是不容错过的经典书籍。

## <<C#程序设计语言>>

### 作者简介

Anders Hejlsberg  
是编程界的传奇人物。  
他是C#语言的架构师，同时也是微软技术专家。

Mads Torgersen  
是微软C#的项目经理，负责日常语言设计工作以及维护C#语言的规范。

Scott Wiltamuth 是Visual Studio  
Professional团队的合作项目主管。  
他在微软参与过OLE Automation、Visual  
Basic、VBScript、JScript、Visual J++和Visual C#等开发工具的开发工作。

## <<C#程序设计语言>>

### 书籍目录

- 译者序
- 序言
- 前言
- 作者简介
- 第1章 介绍
  - 1.1 hello , world
  - 1.2 程序结构
  - 1.3 类型和变量
  - 1.4 表达式
  - 1.5 语句
  - 1.6 类和对象
  - 1.7 结构
  - 1.8 数组
  - 1.9 接口
  - 1.10 枚举
  - 1.11 委托
  - 1.12 特性
- 第2章 词法结构
  - 2.1 程序
  - 2.2 文法
  - 2.3 词法分析
  - 2.4 标记
  - 2.5 预处理指令
- 第3章 基本概念
  - 3.1 应用程序起始
  - 3.2 应用程序终止
  - 3.3 声明
  - 3.4 成员
  - 3.5 成员访问
  - 3.6 签名和重载
  - 3.7 作用域
  - 3.8 命名空间和类型名
  - 3.9 自动化内存管理
  - 3.10 执行顺序
- 第4章 类型
  - 4.1 值类型
  - 4.2 引用类型
  - 4.3 装箱和拆箱
  - 4.4 构造类型
  - 4.5 类型参数
  - 4.6 表达式树类型
  - 4.7 dynamic类型
- 第5章 变量
  - 5.1 变量类别
  - 5.2 默认值

## <<C#程序设计语言>>

- 5.3 明确赋值
- 5.4 变量引用
- 5.5 变量引用的原子性
- 第6章 转换
- 第7章 表达式
- 第8章 语句
- 第9章 命名空间
- 第10章 类
- 第11章 结构
- 第12章 数组
- 第13章 接口
- 第14章 枚举
- 第15章 委托
- 第16章 异常
- 第17章 特性
- 第18章 不安全的代码
- 附录a 文档注释
- 附录b 文法
- 附录c 参考

## 章节摘录

版权页：插图：4.4.2开放式和封闭式类型所有类型都可以归结为开放式类型（open types）或封闭式类型（closed types）。

开放式类型指的是带有类型形参的类型。

更具体地说：· 类型形参定义了开放式类型。

· 有且只有在数组的元素类型为开放式类型的时候，数组类型才是开放式类型。

· 有且只有在构造类型的一个或多个类型实参为开放式类型的时候，构造类型才是开放式类型。

有且只有在一个或多个构造嵌套类型的类型参数或者包含其类型的类型实参为开放式类型的时候，构造嵌套类型才是开放式类型。

不是开放式类型的类型自然就是封闭式类型了。

在运行时，泛型类型声明里所有的代码都在由应用了类型实参的泛型声明所创建的封闭式构造类型的上下文里执行。

泛型类型里的每个类型参数都会绑定到一个特殊的运行时类型上。

运行时所有的语句和表达式都一定是处理封闭式类型的，开放式类型只有在编译期处理的时候才会出现。

每一个封闭式构造类型都拥有自己的静态变量集合，它们不会与任何其他封闭式构造类型共享。

因为开放式类型不存在于运行时，所以不会有静态变量和开放式类型关联。

如果两个构造类型都是从同一个未绑定泛型类型构造而来，并且它们相应的类型实参都是相同的类型，那么这两个封闭式类型就是相同的类型。

## <<C#程序设计语言>>

### 媒体关注与评论

据我的经验，我敢说每个程序员在读本书的时候都至少会有一次“啊，原来如此”的感叹，它能让你的编程水平更上一层楼。

——选自Don Box的序言本书为.NET专业开发人员提供了很好的参考。

它涵盖了所有基础知识，从专家观点到参考，再到用法。

本书是想要扩充知识和专业技能的开发人员的重要参考读物。

——John Montgomery，微软开发部首席团队经理

<<C#程序设计语言>>

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>