



“十二五”职业教育国家规划教材（经全国职业教育教材审定委员会审定）
高等职业教育精品示范教材 **信息安全系列**

Web开发与安全防范

主 编 武春岭
副主编 胡 凯 熊 伟 陈杏环

本书特色：

- 以就业为导向，以能力为本位
- 项目案例引导，任务需求驱动
- 生活实例链接知识点，案例增加趣味性
- 通用教学内容与特殊教学内容协调配置



中国水利水电出版社
www.waterpub.com.cn

“十二五”职业教育国家规划教材（经全国职业教育教材审定委员会审定）

高等职业教育精品示范教材（信息安全系列）

Web 开发与安全防范

主 编 武春岭

副主编 胡 凯 熊 伟 陈杏环



中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书使用 C#和 SQL Server 介绍 ASP.NET 应用程序开发技术,以 Visual Studio 2005 为开发环境,通过制作一个完整的个人网站,将大量开发实例及安全实例融入到开发过程中,在实际操作中介绍 ASP.NET 应用程序安全开发技巧,通过编码介绍网站安全防护,在国内教材中具有开创性,有利于培养安全编码技术人才。

本书采用完整项目贯穿全书的方式,体现行动导向、任务驱动,主要介绍 ASP.NET 开发环境部署、ASP.NET 应用程序开发基础、ASP.NET 标准控件母版页与页面导航、ADO.NET 数据访问技术、数据控件、主题与皮肤设置、网站安全开发流程等。

本书可作为高职高专院校动态网站开发课程的教材,也可供广大网站设计爱好者学习参考。

图书在版编目(CIP)数据

Web开发与安全防范 / 武春岭主编. -- 北京: 中国水利水电出版社, 2014. 11

“十二五”职业教育国家规划教材 高等职业教育精品示范教材. 信息安全系列

ISBN 978-7-5170-2624-2

I. ①W… II. ①武… III. ①网页制作工具—高等教育—教材②互联网络—安全技术—高等教育—教材 IV. ①TP393.092②TP393.408

中国版本图书馆CIP数据核字(2014)第240330号

策划编辑: 祝智敏

责任编辑: 张玉玲

封面设计: 李 佳

书 名	“十二五”职业教育国家规划教材(经全国职业教育教材审定委员会审定) 高等职业教育精品示范教材(信息安全系列) Web 开发与安全防范
作 者	主 编 武春岭 副主编 胡 凯 熊 伟 陈杏环
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路1号D座 100038) 网址: www.waterpub.com.cn E-mail: mchannel@263.net(万水) sales@waterpub.com.cn
经 售	电话: (010) 68367658(发行部)、82562819(万水) 北京科水图书销售中心(零售) 电话: (010) 88383994、63202643、68545874 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	三河市鑫金马印装有限公司
规 格	184mm×240mm 16开本 17.25印张 385千字
版 次	2015年2月第1版 2015年2月第1次印刷
印 数	0001—4000册
定 价	36.00元

凡购买我社图书,如有缺页、倒页、脱页的,本社发行部负责调换

版权所有·侵权必究

高等职业教育精品示范教材（信息安全系列）

丛书编委会

主任 武春岭

副主任 雷顺加 唐中剑 史宝会 张平安 胡国胜

委员

李进涛 李延超 王大川 李宝林 杨 辰

鲁先志 张 湛 路 亚 甘 辰 徐雪鹏

唐继勇 梁雪梅 李贺华 何 欢 张选波

杨智勇 乐明于 赵 怡 胡光永 李峻屹

周璐璐 胡 凯 王世刚 匡芳君 郭兴社

何 倩 李剑勇 陈 剑 刘 涛 杨 飞

冯德万 江果颖 熊 伟 徐钢涛 徐 红

冯前进 胡海波 李莉华 王 磊 陈顺立

武 非 王全喜 王永乐 迟恩宇 胡方霞

王 超 王 刚 陈云志 高灵霞 王文莉

秘书 祝智敏

序 言

随着信息技术和社会经济的快速发展，信息和信息系统成为现代社会极为重要的基础性资源。信息技术给人们的生产、生活带来巨大便利的同时，计算机病毒、黑客攻击等信息安全事故层出不穷，社会对于高素质技能型计算机网络技术和信息安全人才的需求日益旺盛。党的十八大明确指出“高度关注海洋、太空、网络空间安全”，信息安全被提到前所未有的高度。加快建设国家信息安全保障体系，确保我国的信息安全，已经上升为我国的国家战略。

发展我国信息安全技术与产业，对确保我国信息安全有着极为重要的意义。信息安全领域的快速发展，亟需大量的高素质人才。但与之不相匹配的是，在高等职业教育层次信息安全技术专业的教学中，还更多地存在着沿用本科专业教学模式和教材的现象，对于学生的职业能力和职业素养缺乏有针对性的培养。因此，在现代职业教育体系的建立过程中，培养大量的技术技能型信息安全专业人才成为我国高等职业教育领域的重要任务。

信息安全是计算机、通信、数学、物理、法律、管理等学科的交叉学科，涉及计算机、通信、网络安全、电子商务、电子政务、金融等众多领域的知识和技能。因此，探索信息安全专业的培养模式、课程设置和教学内容就成为信息安全人才培养的首要问题。高等职业教育信息安全与管理专业丛书编委会的众多专家、一线教师和企业技术人员，依据最新的专业教学目录和教学标准、结合就业实际需求，组织了以就业为导向的高等职业教育精品示范教材（信息安全系列）的编写工作。该系列教材由《网络安全产品调试与部署》、《网络安全系统集成》、《Web 开发与安全防范》、《数字身份认证技术》、《计算机取证与司法鉴定》、《操作系统安全（Linux）》、《网络安全攻防技术实训》、《大型数据库应用与安全》、《信息安全工程与管理》、《信息安全法规与标准》、《信息安全风险评估》等组成，在紧跟当代信息安全研究发展的同时，全面、系统、科学地培养信息安全类技术技能型人才。

本系列教材在组织规划的过程中，遵循以下几个基本原则：

（1）体现就业为导向、产学研结合的发展道路。学科和专业同步加强，按企业需要、按岗位要求来对接培养内容。既能反映信息安全学科的发展趋势，又能结合信息安全专业教育的改革，且及时反映教学内容和教学体系的调整更新。

（2）采用项目驱动、案例引导的编写模式。打破传统的以学科体系设置课程体系、以知识点为核心的框架，更多地考虑学生所学知识与行业需求及相关岗位、岗位群的需求相一致，坚持“工作流程化”、“任务驱动式”，突出“走向职业化”的特点，努力培养学生的职业素养、职业能力，实现教学内容与实际工作的高仿真对接，真正以培养技术技能型人才为核心。

（3）专家和教师共建团队，优化编写队伍。由来自信息安全领域的行业专家、院校教师、企业技术人员组成编写队伍，跨区域、跨学校进行交叉研究、协调推进，把握行业发展和创新

教材发展方向，融入信息安全专业的课程设置与教材内容。

(4) 开发课程教学资源，推进专业信息化建设。从充分关注人才培养目标、专业结构布局等入手，开发补充性、更新性和延伸性教辅资料，开发网络课程、虚拟仿真实训平台、工作过程模拟软件、通用主题素材库以及名师讲义等多种形式的数字化教学资源，建立动态、共享的课程教材信息化资源库，服务于系统培养技术技能人才。

信息安全类教材建设是提高信息安全专业技术技能型人才培养质量的关键环节，是深化职业教育教学改革的有效途径。为了促进现代职业教育体系的建设，使教材建设全面对接教学改革、行业需求，更好地服务区域经济和社会发展，我们殷切希望各位职教专家和老师提出建议，并加入到我们的编写队伍中来，共同打造信息安全领域的系列精品教材！

丛书编委会

2014年6月

前 言

ASP.NET 是 Microsoft 公司推出的建立动态 Web 应用程序的开发平台,它为开发人员提供了完整的可视化开发环境,具有使用方便、灵活、性能好、安全性高、完整性强、面向对象等特性,是目前主流的网络编程工具之一。

本书以 C#为编程工具,以 SQL Server 2005 为数据平台,将一个经典案例——个人网站的开发作为贯穿项目,实现一个小型动态网站项目开发的全过程,在开发的过程中,兼顾 Web 安全开发技术,就常见的服务器端应用安全问题进行了阐述。

书中将实现一个网站功能所需要的知识分散到各个章节,让读者通过分析项目结构及功能进行具体的页面实施,让读者在“做中学,学中做”,从而能够逐步实现一个既完整又注重安全性的个人网站。

本书阅读指南

本书分为 9 章,由浅入深,每一章完成个人网站开发过程中相对独立的模块。其中前 8 章添加了“技能基础”C#基础模块,是为没有基础的读者准备的,教学时可以先讲解第 1 章到第 8 章的“技能基础”模块部分。

第 1 章首先介绍如何部署开发环境、安装和配置 IIS,然后介绍 Visual Studio 2005 的语言开发环境;分析本项目网站的总体结构,说明各个页面的功能,以便读者对本项目的功能有一个系统了解。

第 2 章介绍如何根据项目网站的需求分析设计相关数据库,以便存储网站项目中的相关相册资料;介绍基于图片的存放目录结构,书写自定义 HTTP 处理程序,实现图片的显示功能,并介绍如何进行反射性 XSS 防御。

第 3 章介绍如何使用母版统一及简化页面制作,并设计页面导航功能,具体包括站点地图的创建、Tree 控件及 SiteMapPath 控件等的使用。

第 4 章是本书的重点,实现个人网站中的重要功能,即相册及照片的显示。通过实现这一功能,介绍了 ADO.NET 数据操作技术、DataList 控件及 FormView 控件的基本使用,并专门就目前流行的数据库注入式攻击进行分析和举例。

第 5 章是本书的关键内容,实现了相册管理的基本功能,对如何编辑相册及照片,实现对相册和照片的显示、增加、修改及删除进行了详细说明,在具体实现的过程中使用了 DataList 控件、FormView 控件、GridView 控件,最后针对前一章所提及的注入式攻击提出防御方案。

第 6 章介绍本项目网站的主题设置,包括主题文件夹、主题文件的创建,以及如何使用主题;对相关主题下的皮肤创建进行说明,包括新建、设置及使用皮肤。

目 录

前言

第1章 配置 ASP.NET 网页运行和开发环境..... 1	2.2 常量与变量..... 35
任务目标..... 1	任务实施..... 36
技能目标..... 1	2.3 任务一：建立数据库..... 36
任务导航..... 1	2.3.1 新建数据库..... 36
技能基础..... 2	2.3.2 分析数据库..... 39
1.1 Visual C#简介..... 2	2.4 任务二：建立 Web 页面查询照片名..... 41
1.2 Visual C#数据类型..... 3	2.4.1 新建 Web 页面..... 41
1.3 数据类型转换..... 5	2.4.2 编写 Web 页面的代码..... 44
1.4 C#的字符集和词汇集..... 6	2.5 任务三：建立 Http 处理程序..... 46
任务实施..... 7	2.5.1 认识 Http 处理程序..... 46
1.5 任务一：认识 ASP.NET..... 7	2.5.2 建立 Http 处理程序..... 47
1.5.1 ASP.NET 开发环境搭建..... 7	2.5.3 运行 Http 处理程序..... 50
1.5.2 IIS 安装及安全配置..... 12	2.6 任务四：页面间传值的安全防范..... 52
1.5.3 Visual Studio.NET 开发环境介绍..... 15	2.6.1 页面间传值的安全问题..... 52
1.6 任务二：数据库安装与设计..... 19	2.6.2 本项目页面间传值的安全解决方法..... 54
1.6.1 SQL Server Management Studio Express 的安装..... 19	综合练习..... 56
1.6.2 启动 SQL Server Management Studio Express..... 22	第3章 创建母版页及页面导航..... 57
1.7 任务三：网站项目规划设计..... 23	任务目标..... 57
1.7.1 网站结构分析..... 23	技能目标..... 57
1.7.2 网站功能分析..... 24	任务导航..... 57
综合练习..... 32	技能基础..... 58
第2章 建立 Web 页面及 Http 处理程序..... 33	3.1 条件语句..... 58
任务目标..... 33	3.2 循环语句..... 61
技能目标..... 33	3.3 跳转结构..... 63
任务导航..... 33	任务实施..... 64
技能基础..... 34	3.4 任务一：实现母版页..... 64
2.1 C#程序代码的基本书写规则..... 34	3.4.1 母版页的优点..... 64
	3.4.2 设计母版页..... 66
	3.4.3 在项目中使用母版页..... 69

3.5 任务二：实现页面导航	81	任务导航	139
3.5.1 创建站点地图文件	82	技能基础	140
3.5.2 使用 TreeView 控件实现导航	84	5.1 定义类	140
3.5.3 使用 SiteMapPath 控件显示导航路径	85	5.2 定义和使用字段	141
3.5.4 使用 Menu 控件实现导航菜单	86	任务实施	142
3.5.5 在母版页中实现站点导航	88	5.3 任务一：管理相册	142
3.5.6 在项目中实现页面导航	91	5.3.1 使用 SqlDataSource 连接相册数据库	143
3.6 任务三：Web.config 的安全防御	94	5.3.2 使用 FormView 新建相册	148
综合练习	98	5.3.3 使用 GridView 显示并编辑相册目录	150
第 4 章 显示相册	99	5.4 任务二：管理照片	155
任务目标	99	5.4.1 使用 FormView 新建相片	156
技能目标	99	5.4.2 使用 DataList 批量上传照片	159
任务导航	99	5.4.3 使用 GridView 实现照片的显示、更改和删除	163
技能基础	100	5.5 任务三：数据库攻击技巧及防御方法	166
4.1 一维数组	100	5.5.1 数据库攻击技巧	166
4.2 多维数组	102	5.5.2 正确地防御 SQL 注入	167
任务实施	102	综合练习	168
4.3 任务一：使用 ADO.NET 操作数据库	102	第 6 章 设置主题和皮肤	169
4.3.1 ADO.NET 及命名空间	102	任务目标	169
4.3.2 使用 Connection 对象连接数据库	104	技能目标	169
4.3.3 使用 Command 对象操作数据库	105	任务导航	169
4.4 任务二：数据控件的使用	106	技能基础	170
4.4.1 使用 SqlDataSource 连接相册数据库	106	6.1 属性的定义和访问	170
4.4.2 使用 DataList 显示相册目录	115	6.2 方法的定义和调用	171
4.4.3 使用 DataList 显示所有照片	117	任务实施	174
4.4.4 使用 FormView 显示某张照片	123	6.3 任务一：新建主题和皮肤	174
4.5 任务三：对数据库进行 SQL 注入攻击	125	6.3.1 新建主题	174
4.5.1 SQL 注入的含义	125	6.3.2 创建主题文件	175
4.5.2 何谓“盲注”	126	6.3.3 应用主题	176
4.5.3 实施 SQL 注入攻击	126	6.4 任务二：在项目中使用时主题	178
综合练习	138	6.4.1 创建主题	178
第 5 章 管理相册	139	6.4.2 使用主题	179
任务目标	139		
技能目标	139		

6.5 任务三：在项目中实现皮肤设置	180	任务目标	232
6.5.1 创建皮肤	180	技能目标	232
6.5.2 使用皮肤	182	任务导航	232
6.6 任务四：HTML 跨站脚本安全分析	188	技能基础	232
6.6.1 跨站脚本介绍	188	8.1 文件的读写	232
6.6.2 造成跨站攻击的流行因素分析	188	任务实施	236
6.6.3 用户安全输入	189	8.2 任务一：网站发布	236
6.6.4 跨站攻击的危害分析	190	8.2.1 注册用户	236
综合练习	194	8.2.2 创建网站	238
第 7 章 成员与角色管理	195	8.2.3 上传网站文件	239
任务目标	195	8.2.4 附加数据库	245
技能目标	195	8.2.5 在互联网上运行网站	248
任务导航	195	8.3 任务二：XSS 攻击与防御	249
技能基础	196	8.3.1 XSS 攻击	250
7.1 文件管理	196	8.3.2 XSS 防御	253
7.2 目录和路径管理	198	综合练习	255
任务实施	202	第 9 章 安全开发流程 (SDL)	256
7.3 任务一：成员与角色管理	202	任务目标	256
7.3.1 新建成员管理页面	202	技能目标	256
7.3.2 配置成员与角色管理	203	任务导航	256
7.3.3 实现用户登录	207	任务实施	256
7.3.4 注册新用户	211	9.1 SDL 简介	256
7.3.5 在项目中实现成员管理	215	9.1.1 微软 SDL 过程阶段	257
7.3.6 在项目中实现角色管理	224	9.1.2 敏捷 SDL	260
7.4 任务二：成员与角色的安全防御	228	9.2 SDL 实战经验	260
7.4.1 成员与角色的安全概念	228	9.2.1 需求分析与设计阶段	262
7.4.2 密码安全防御	229	9.2.2 开发阶段	263
综合练习	231	9.2.3 测试阶段	264
第 8 章 网站发布	232	综合练习	265

1

配置 ASP.NET 网页运行和 开发环境

任务目标

- 安装及配置 Visual Studio 2005 开发环境。
- 安装及配置 IIS。
- 安装 SQL Server Management Studio Express。
- 规划及设计项目网站功能。

技能目标

- 学会安装 Visual Studio 2005，了解安装 Visual Studio 2005 所必需的系统配置，熟悉 Visual Studio 2005 开发环境，了解网站的开发过程。
- 为不同的操作系统选择对应的 IIS 并安装，掌握 IIS 安全配置方法。
- 获取并安装可视化的 SQL Server 数据管理工具，了解其基本使用。
- 规划并设计本项目网站，对网站的总体结构和页面功能进行分析。

任务导航

本书从这里开始 ASP.NET 技术的学习之旅。

ASP.NET 技术是 Microsoft Web 开发史上的一个重要的里程碑，使用 ASP.NET 开发 Web 应用程序并维持其运行比以前更加简单。通过本章的学习，读者会对 ASP.NET 有进一步的认识，安装、搭建和熟悉 ASP.NET 环境；配置 IIS；安装在网页开发中必然会用到的数据库系统；了解一些网页相关的基本知识，了解并设计本书中项目网站的功能，当然也可以利用 ASP.NET 帮助系统更加深入地学习 ASP.NET。

技能基础

1.1 Visual C#简介

C#是 Microsoft 在 Visual Studio .NET 中推出的一种新型程序设计语言，具有面向组件、功能强大和灵活等特点。C#语言与 C++和 Java 非常类似，样式清晰，可读性很强，易于掌握。

C#是由 C/C++语言发展而来的，C/C++编程语言功能强大，但与诸如 Delphi、Visual Basic 等语言相比，C/C++语言较为复杂，需要花费更多的时间、精力。而诸如 Visual Basic 之类的语言虽然编程效率较高，但底层开发功能较差。相比之下，C#能较好地平衡功能与效率之间的关系：一方面，C#与 C/C++具有继承关系，保留和扩展了 C/C++的功能，C++开发人员易于熟悉掌握；另一方面，使用 C#可以快速地编写各种基于 Microsoft.NET 平台的应用程序，由于 Microsoft.NET 提供了一系列工具和服务，因此 C#程序开发具有更高的效率。

Visual C#组织结构的关键概念是程序、命名空间、类型 (Type)、成员 (Member) 和程序集 (Assembly)。Visual C#程序由一个或多个源文件组成。程序中声明类型，类型包含成员，按照命名空间进行组织。类型包括类和接口，成员包括字段、方法、属性和事件。程序编译后生成程序集，文件扩展名通常为 .exe 或 .dll。

程序集是一个自描述的功能单元，既包含代码又包含元数据，使用时不需要 #include 指令和头文件。程序集包含中间语言 (Intermediate Language, IL) 指令形式的可执行代码和元数据 (Metadata)。执行时，CLR 的实时编译器 (JIT) 将程序集中的 IL 代码转换为特定于处理器的代码。

Visual C#程序是用命名空间组织起来的，命名空间类似于文件夹，一个命名空间可以包含类型声明和嵌套的命名空间声明。类型声明用于定义类、结构、接口、枚举和委托。在一个类型声明中可以使用哪些类型作为其成员，取决于该类型声明的形式。

C#能够开发控制台应用程序 (Console)、Windows 窗体应用程序、Web 应用程序和 Web 服务等。

在控制台应用程序中，人机交互操作主要是通过输入输出语句进行的。System.Console 类的 Read() 和 ReadLine() 方法用来实现控制台输入，Write() 和 WriteLine() 方法用来实现控制台输出，例如代码 1-1 所示。

代码 1-1 第一个 C#控制台程序

```
using System;
using System.Collections.Generic;
using System.Text;
namespace ex1_1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("请输入您的姓名: ");
            string name=Console.ReadLine(); //输入姓名字符串赋值给 name 变量
            Console.WriteLine("Hello," + name+" WriteLine");
            Console.WriteLine("Hello," + name+" Write");
            Console.WriteLine("第一个 C#控制台程序");
            Console.ReadLine();
        }
    }
}
```

1.2 Visual C#数据类型

C#是一种完全面向对象的语言，其所有内容都必须放置在类中，C#的数据类型也不例外，每一种类型均是一个类，都具有相应的属性和方法，这也是它与其他语言不同的地方。实现数据类型的类都处于 System 命名空间中，也称为预定义结构类型。

C#的数据类型分为值类型、引用类型和指针类型三大类。值类型直接存储它的数据内容，包括简单数据类型（如 char、int 和 float）、结构（struct types）和枚举（enum）三种；引用类型不存储实际数据内容，而是存储对实际数据的引用即地址，包括类（class）、接口（interface）、委托（delegate，也称代表）和数组（array）；指针类型一般使用得较少，并且只能用于不安全模式。

1. 值类型

所有的值类型都隐式派生自 Object 类，因此每种值类型都与 .NET 框架类库（FCL）中的类型有直接的对应关系，比如整型 int 对应为 System.Int32 类。

在 C#中，整数类型、浮点类型、decimal 类型、布尔类型和字符类型统称为简单数据类型。各简单类型的大小和取值范围如表 1-1 所示。

表 1-1 C#的简单数据类型

简单数据类型	表示数据	字节长度	取值范围	默认值	后缀
bool	布尔型	2	True 或 False	False	
sbyte	字节型	1	0~127	0	

续表

简单数据类型	表示数据	字节长度	取值范围	默认值	后缀
byte	无符号字节型	1	0~255	0	
short	短整型	2	-32768~+32767	0	
ushort	无符号短整型	2	0~65535	0	
int	整型	4	$-2^{31} \sim 2^{31}-1$	0	
uint	无符号整型	4	$0 \sim 2^{32}-1$	0	u
long	长整型	8	$-2^{63} \sim 2^{63}-1$	0	l
ulong	无符号长整型	8	$0 \sim 2^{64}-1$	0	ul
char	字符型	2	0~65535	null	
float	单精度浮点数	4	1.40E-45~3.40E+38	0	f
double	双精度浮点数	8	4.940E-324~1.798E+308	0.0	d
decimal	十进制数类型	16	$1.0 \times 10^{-28} \sim 7.9 \times 10^{28}$	0.0	m

注意：在 C# 中使用布尔型数据时应该特别注意，布尔型变量与其他类型变量之间不能互相赋值。在 C# 中，布尔逻辑量只有 True 和 False 两个值，在 C# 中不存在“非零值等效于 True”和“零值等效于 False”的用法。

decimal 类型是适合财务和货币计算的 128 位数据类型。同浮点类型相比，decimal 类型具有更高的精度和更小的范围，其精度为 28 个或 29 个有效数字，预定义结构类型为 System.decimal。当给 decimal 变量赋值时，使用 m 后缀来表明它是一个 decimal 类型，例如：

```
decimal monthSalary=4500.5m;
```

在 C# 程序中，如果书写的十进制的数值常数不带有小数，就默认该常数的类型是整型。

在 C# 中，对字符型变量使用整数进行赋值和运算是不允许的，但是字符型和整型之间可以进行显式转换，例如：

```
char c=(char)8; int x=(int)'a';
```

2. 引用类型

引用类型存储对它们的数据的引用，称为对象，值类型的赋值是重新创建一个副本，而引用类型的赋值是共享同一块内存（副本），是指向同一块内存。

C# 的引用类型包括：数组、类、接口、委托、对象、字符串。

数组将在后面详细介绍，本节介绍字符串和日期类。

(1) 字符串类型。

字符串是使用 string 关键字声明的、由一个或多个字符构成的一组字符。关键字 string 实际上就是 System.String 类的别名。C# 中的字符串有两种表示方式：用双引号括起来；用 @ 引起来，它可以把字符串中的特殊字符的特殊性去掉，字符串中的所有字符均被认为是普通字符。例如：

```
string s="Hello";  
string dirname=@"c:\Documents";
```

字符串 `dirname` 中的“\D”不是转义字符。

`System.String` 类实例的唯一属性是 `Length`，语法格式如下：

```
int x=s.Length; //s 是字符串类型变量
```

注意：C#中的字符串常量是建立在 Unicode 字符集基础之上的，它所包含的任何单个字符都由 2 个字节来表示。计算字符串长度时，无论中文字符、英文大小写字母、数字、标点符号还是其他特殊符号，都按一个字符计算。

此外，`string` 拥有大量处理字符串的方法，可以完成诸如查找字符、取子串等任务。

(2) DateTime 类。

在 C#中，日期和时间信息主要是由 `System.DateTime` 类来表达和处理的，`System.DateTime` 类提供了常用日期与时间属性，其中 `Now`、`Today`、`UtcNow` 为 `DateTime` 类的静态属性，其余为实例属性。这些属性大部分为数值类型，作为字符串输出时需要用 `ToString()` 方法进行类型转换。

使用 `DateTime` 结构表示日期和时间。可以使用 `DateTime` 结构来创建日期，也可以使用 `Now` 属性获取系统日期。例如：

```
DateTime Birthday=new DateTime(1980,3,14);  
DateTime today=DateTime.Now;
```

有了日期以后就可以使用 `Year`、`Month`、`Day`、`DayOfWeek`、`Hour`、`Minute`、`Second` 属性访问日期中的年份、月份、日、星期、小时、分钟和秒。例如：

```
int month= Birthday.month;  
int hour=today.hour;
```

1.3 数据类型转换

在编写 C#程序过程中，经常会碰到类型转换问题。例如，将整数类型数据和浮点类型数据相加，C#会进行隐式转换。

在简单类型中，除了字符类型和布尔类型以外，总是存在从低精度数值类型到高精度数值类型的自动转换，这称为隐式转换；否则，必须使用显式转换，也称为强制类型转换。例如：

```
double pi=3.14;  
int x=5;  
pi=x; //隐式转换  
x=(int)pi; //显式转换
```

每种类型均可以转换为字符串，因为每种类型都有 `ToString` 方法。例如：

```
int num=125;  
string text=num.ToString();
```

字符串向其他类型转换可以使用 `Parse` 方法。例如：

```
string txt="178.5";  
double shengao=Double.Parse(txt);
```


还有一种数据类型转换方法，就是使用 `Convert` 类，它可以在所有数据类型间转换。例如：

```
string txt="175.8";  
double shengao=Convert.ToDouble(txt);
```

1.4 C#的字符集和词汇集

1. 字符集

为了实现 Microsoft 全球通用的战略目标，C#中所有字符都是使用 Unicode 编码表示的。在 Microsoft 推出的 Windows 2000 以上版本的操作系统中，所有的核心函数也都要求使用 Unicode 编码。

按照 Unicode 的编码规定，每个字符都由两个字节（16 位二进制数）来表示，编码范围为 0~65535，所以 Unicode 字符集最多可以表示 65536 个字符。因此在 C#中，字符型（`char`）数据占用两个字节的内存，可以用来存储 Unicode 字符集其中的一个字符。用一对单引号括起来的单个字符，如 'A'、'h'、'学'、'校'等，称为字符常量，可以用来向字符型变量赋值。但是，反斜杠（\）、双引号或单引号特殊字符等需要用转义字符来表示。

2. 词汇集

C#的词汇集主要包括关键字、标识符和文字常量等。C#的关键字共有 77 个，在 C#程序中，C#的标识符的命名必须遵循如下规则：

- 第一个字符必须是英文字母或下划线（事实上也可以是汉字、希腊字母、俄文字母等其他 Unicode 字符，但不推荐，一般不要这样用）。
- 从第二个字符开始，可以使用英文字母、数字和下划线，但不能包含空格、标点符号、运算符等字符，不能与关键字重名，如果在关键字前面加上 @ 前缀，也可以成为合法标识符（不推荐，一般不要这样用）。
- 长度不能超过 255 个字符。

在实际应用中，为了改善程序的可读性，标识符最好使用具有实际意义的英文单词或其缩写，做到见名知义。

目前软件开发中使用较多的标识符命名样式主要有以下 3 种：

- Pascal 样式。在 Pascal 命名样式中，直接组合用于命名的英语单词或单词缩写形式，每个单词的首字母大写，其余字母小写。例如 `TextBox`、`FileOpen` 等。
- Camel 样式。除了第一个单词小写外，其余单词的首字母均采用大写形式。例如 `myName`、`myAddress` 等。
- Upper 样式。每个字母均采用大写形式，此种形式一般用于标识具有固定意义的缩写形式。例如 `XML`、`GUI` 等。