

“十五”国家重点电子出版物规划项目·计算机知识普及和软件开发系列
专业程序员编程实作丛书 (10)

C#

Programming Instances

网络编程 实例教程

北京希望电子出版社 总策划
周存杰 编写

本版特点

- * 内容丰富，实用性与指导性强，以近90个实例介绍用C#语言进行网络编程的技巧

适用范围

- * 面向初中级编程人员
- * 高校相关专业师生
- * 社会相关培训班学员

北京希望电子出版社
Beijing Hope Electronic Press
www.bhpe.com.cn



“十五”国家重点电子出版物规划项目·计算机知识普及和软件开发系列
专业程序员编程实作丛书 (10)

896

C#

Programming
Instances

网络编程 实例教程

北京希望电子出版社 总策划
周存杰 编写

本版特点

- * 内容丰富，实用性与指导性强，以近90个实例介绍用C#语言进行网络编程的技巧

适用范围

- * 面向初中级编程人员
- * 高校相关专业师生
- * 社会相关培训班学员

北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

内 容 简 介

本书是一本用 C# 进行网络编程的指导书。作者从基础知识与简单实例入手，到综合实例的开发，从易到难，循序渐进，使读者学习轻松，上手容易。

本书由 12 章构成，主要内容包括 Visual C# DNS 开发；Visual C# 套接字编程；Visual C# TCP 协议编程；Visual C# FTP 编程；Visual C# HTTP 协议编程；UDP 协议与 SMTP 协议编程；ASP.NET 应用程序开发；ASP.NET 服务开发；数据库开发基础；Visual C# 数据库开发实例——图书销售服务系统；套接字网络数据库；ASP.NET 数据库开发——网络学籍管理系统。

本书内容丰富，实用性与指导性强，用近 90 个实例介绍了用 C# 语言进行网络编程的技巧。

本书的读者面向初中级编程人员、高校相关专业师生以及社会相关培训班学员。

说明：需要本书程序的源代码文件 3835.zip 的读者，请从 www.b-xr.com 免费下载，或直接与 zhou868@263.net 联系。

系 列 书 名： “十五”国家重点电子出版物规划项目 计算机知识普及和软件开发系列
专业程序员编程实作丛书（10）

书 名： C#网络编程实例教程

总 策 划： 北京希望电子出版社

文 本 著 作 者： 周存杰 编写

责 任 编 辑： 周 艳

出版、发 行 者： 北京希望电子出版社

地 址： 北京市海淀区知春路 63 号（100080）

网 址： www.bhp.com.cn E-mail: lwm@bhp.com
010-62520290, 62521724, 62528991, 62630301, 62524940, 62521921, 82610344
（发行） 010-62613322-215（门市） 010-82675588-558（编辑部）。

经 销： 各地新华书店、软件连锁店

排 版： 希望图书输出中心 张月岭

文 本 印 刷 者： 北京双青印刷厂

开 本 / 规 格： 787 毫米×1092 毫米 16 开本 28 印张 643.2 千字

版 次 / 印 次： 2002 年 7 月第 1 版 2002 年 7 月第 1 次印刷

本 版 号： ISBN 7-900101-37-3

印 数： 1-4000

定 价： 42.00 元

说明：凡我社产品如有残缺，可执相关凭证与本社调换。

前言

C#推出2年以来,因其简洁的语法和强大的功能,已经为广大程序员所关注,许多商业软件已经开始尝试用C#开发。

因为,C#要比C++方便、高效得多。C#抛弃了许多旧的技术,象ATL, MFC, C运行库,标准模板库(STL)等等类库都被C#抛弃的干干净净。C#不但可以开发传统的Windows程序,而且可以将Windows程序轻易转换成XML Web应用程序或服务,从而实现与Web的紧密结合。

.NET和C#的最进步之处,在于对网络的支持,尤其是对Web的支持。.NET推出的目的之一,就是要将分散的站点变成可以互相交换组件的地方。用C#开发的Web组件是基于XML和SOAP协议的,而XML和SOAP是共有的协议和标准,这样,用C#开发的Web资源就可以为其他支持XML和SOAP应用所调用,这样就可以使各个站点成为可以互相交换组件的地方。这符合现代开发理念。

本书首先介绍了网络基础开发法,包括DNS,套接字,TCP,FTP,HTTP,UDP,SMTP等内容,然后介绍了C#的Web应用,最后又介绍了网络的高级应用——数据库在网络中的应用。本书要求读者已经熟悉掌握C#语法并且对C# Win32开发有一定的了解。

本书的主要程序的源代码,请从www.bhp.com.cn免费下载。如果读者觉得光盘中的代码还不够完善,可来信索取最新版本的源代码。信箱:zhou868@263.net(长期有效)。另外需要注意的是,笔者还可以免费赠送C# Windows高级开发的资料(笔者编写的),初级读者可以向笔者索取这方面的资料。笔者对免费提供的资料享有全部著作权,读者只能用于个人学习目的,不得传播,不得用于商业目的。

本书在编写过程中,得到了天津大学电子信息工程学院的宋健宝先生的大力支持和热情帮助,在此表示感谢。希望出版社的赵文博老师,也给笔者提供了指导和帮助,本书的责任编辑周艳小姐也对本书提出了宝贵的意见,在此一并表示感谢。

由于时间仓促,本书肯定还会有许多错误之处,望读者指正。参与本书编写的还有周丽、王燕、赵跃鸿、李小鹏、宋元时、张明明等人。

作者

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

I

第三章 Visual C# TCP 协议编程	83
3.1 Visual C# TCP 协议编程基础	83
3.1.1 TCP 协议层次结构	83
3.1.2 TCP 协议规范	85
3.2 TCP 协议相关类简介	86
3.2.1 TcpListener 类	86
3.2.2 TcpClient 类	87
3.2.3 NetworkStream 类	89
3.3 HTTP 协议常用方法详解	93
3.3.1 监听	93
3.3.2 请求连接	95
3.3.3 接受连接	96
3.3.4 Socket 类的 Send 方法在 Tcp 编程中的应用	98
3.3.5 NetworkStream 类的 Read 方法程序示例	100
3.3.6 NetworkStream 类的 Write、Flush 方法示例	103
3.3.7 Socket 类的 Receive 方法在 Tcp 编程中的应用	103
3.4 远程控制开发	105
3.4.1 控制端开发	106
3.4.2 服务端开发	114
3.4.3 必要设置	129
3.4.4 演示	131
3.5 本章小结	135
第四章 Visual C# FTP 编程	136
4.1 FTP 协议	136
4.1.1 FTP 简介	136
4.1.2 FTP 命令	137
4.1.3 FTP 响应码	139
4.2 FTP 站点建立	140
4.3 FTP 浏览器开发	145
4.3.1 AxDHTMLEdit 控件简介	145
4.3.2 FTP 浏览器开发	146
4.4 文件传输系统开发	149
4.4.1 文件传输服务器开发	150
4.4.2 文件传输客户端开发	154
4.5 本章小结	160
第五章 Visual C# HTTP 协议编程	161
5.1 HTTP 协议简介	161

5.1.1	HTTP 协议简介.....	161
5.1.2	HTTP 基本语法.....	162
5.2	与 HTTP 相关的类简介.....	165
5.2.1	HttpRequest 类.....	165
5.2.2	HttpResponse 类.....	168
5.2.3	WebRequest 类.....	169
5.2.4	WebResponse 类.....	173
5.2.5	Uri 类.....	174
5.2.6	WebClient 类.....	177
5.3	Visual C# Http 协议编程常用方法详解.....	180
5.3.1	HttpRequest 流操作.....	180
5.3.2	WebRequest 流操作.....	183
5.3.3	上传文件.....	185
5.3.4	下载文件.....	188
5.3.5	认证.....	189
5.3.6	SSL(安全套接字层).....	189
5.3.7	代理.....	189
5.4	Internet 浏览器 SHARP EXPLORER 1.0 开发.....	190
5.4.1	界面设计.....	190
5.4.2	编写代码.....	194
5.4.3	演示.....	206
5.5	本章小结.....	208
第六章	UDP 协议与 SMTP 协议编程.....	209
6.1	UDP 协议编程.....	209
6.1.1	UDP (用户数据报协议) 简介.....	209
6.1.2	.NET 平台与 UDP 相关的类.....	209
6.1.3	UDP 开发实例.....	213
6.2	SMTP 协议编程.....	216
6.2.1	SMTP 介绍.....	216
6.2.2	SmtpMail 类.....	219
6.2.3	MailMessage 类.....	220
6.2.4	MailAttachment 类.....	221
6.2.5	Visual C# SMTP 协议编程实例.....	221
6.3	本章小结.....	228
第七章	ASP.NET 应用程序开发.....	229
7.1	ASP.NET 基础.....	229
7.1.1	Microsoft.NET 革命.....	229

7.1.2	ASP.NET 运行机制	230
7.1.3	ASP.NET 进步	230
7.1.4	在本地主机上建立站点	231
7.1.5	ASP.NET 应用程序	237
7.1.6	Visual C# ASP.NET 开发入门	238
7.1.7	第一个 ASP.NET 应用程序开发	240
7.2	ASP.NET 网页控件	242
7.2.1	TextBox 控件	242
7.2.2	Label 控件	244
7.2.3	Button 控件	245
7.2.4	LinkButton 控件	246
7.2.5	ImageButton 控件和 Image 控件	246
7.2.6	DropDownList 控件	248
7.2.7	ListBox 控件	250
7.2.8	Panel, RadioButton, CheckBox 控件	252
7.2.9	RadioButton List 控件和 CheckBoxList 控件	252
7.2.10	Calendar 控件	253
7.2.11	超链接控件 HyperLink 控件	254
7.2.12	HTML 工具	256
7.3	ASP.NET 开发实例	257
7.3.1	主页设计	257
7.3.2	注册页设计	258
7.3.3	编码	259
7.3.4	演示	261
7.4	本章小结	263
第八章	ASP.NET 服务开发	264
8.1	第一个 ASP.NET 服务项目开发	264
8.1.1	ASP.NET 服务简介	264
8.1.2	第一个 ASP.NET 服务开发	265
8.1.3	在 ASP.NET 应用程序里使用 ASP.NET 服务	267
8.2	ASP.NET 服务开发实例	269
8.2.1	带参数的 ASP.NET 服务开发	269
8.2.2	在 ASP.NET 服务项目中使用 ASP.NET 服务	283
8.2.3	登录服务与注册服务开发	284
8.2.4	超级服务开发	286
8.2.5	一个项目里编写多个方法	287
8.2.6	在 Windows 程序中使用 ASP.NET 服务	288
8.2.7	在 ASP.NET 服务中使用 Windows 组件	289

8.3 重建我的酷网	290
8.3.1 主页设计	290
8.3.2 子网页 WebForm2 设计	292
8.3.3 子网页 WebForm3 设计	292
8.3.4 子网页 WebForm4 设计	293
8.3.5 演示	294
8.4 本章小结	295
第九章 数据库开发基础	296
9.1 数据库建立	296
9.1.1 建立 SQL 数据库	296
9.1.2 建立 OLE 数据库	299
9.2 数据库连接	301
9.2.1 SQL 数据库的连接	302
9.2.2 OLE 数据库的连接	306
9.3 数据浏览与查询	311
9.3.1 DataGrid 控件	311
9.3.2 SQL 数据库数据浏览与查询	312
9.3.3 OLE 数据库数据浏览与查询	315
9.4 数据插入、删除与更新	316
9.4.1 数据插入	316
9.4.2 数据删除	322
9.4.3 数据更新（替换）	328
9.5 数据库的其他常用操作	332
9.5.1 创建数据表和修改数据表	332
9.5.2 使用 SqlDataReader 读取数据	335
9.5.3 多数据库协同操作	337
9.6 本章小结	341
第十章 Visual C#数据库开发实例——图书销售服务系统	342
10.1 界面设计	343
10.1.1 主窗体设计	343
10.1.2 子窗体 1 (Form2) 设计	343
10.1.3 子窗体 2 (Form 3) 设计	346
10.1.4 子窗体 3 (Form4) 设计	346
10.1.5 子窗体 5 (Form5) 设计	347
10.2 编写代码	348
10.2.1 主窗体代码	348
10.2.2 子窗体 Form2 代码	356

10.2.3	子窗体 Form3 代码	358
10.2.4	子窗体 Form4 代码	359
10.2.5	子窗体 Form5 代码	367
10.3	演 示	376
10.3.1	查询功能演示	376
10.3.2	超级用户插入功能演示	376
10.3.3	超级用户删除功能演示	377
10.3.4	超级用户更新功能演示	378
10.4	本章小结	378
第十一章	套接字网络数据库	379
11.1	服务端开发	379
11.2	客户端开发	389
11.3	演示	397
11.4	本章小结	398
第十二章	ASP.NET 数据库开发——网络学籍管理系统	399
12.1	数据库的 ASP.NET 服务开发	400
12.2	ASP.NET 应用程序开发	416
12.2.1	主页开发	416
12.2.2	子网页 WebForm2 开发	417
12.2.3	子网页 WebForm3 开发	418
12.2.4	子网页 WebForm4 开发	422
12.3	演 示	429
12.3.1	演示注册新用户功能	429
12.3.2	演示用户登录功能	430
12.3.3	查询数据	430
12.3.4	演示插入数据	431
12.3.5	演示删除功能	432
12.3.6	演示更新功能	433
12.4	本章小结	434

第一章 Visual C# DNS 开发

所谓 DNS 是 Domain Name System 或者 Domain Name Service 的缩写。也就是将主机域名和 IP 地址之间相互转换的系统或应用程序。这在网络编程中是十分必要的，它可以方便地获取主机的 IP 地址和域名信息，使程序界面更加友好，从而方便人们的应用。

1.1 IP 协议和 DNS 简介

1.1.1 IP 协议简介

1. IP 基本概念

要熟悉 DNS 编程，了解一些 IP 协议知识也是十分必要的。IP 协议 (**Internet Protocol**) 是用于将多个包交换网络连接起来的，将数据报从源地址到目的地址之前传送的基本网络协议。IP 协议位于 ISO 层次结构中的网络层，它实现了 Internet 中自动路由的功能。IP 的任务就是把数据从源传送到目的地。它不负责保证传送可靠性、流控制、包顺序。IP 实现两个基本功能是：寻址和分段，它可以根据数据报报头中的目的地址将数据报传送到目的地址，在此过程中 IP 负责选择传送的道路，这种选择道路称为路由功能。如果有些网络内只能传送小数据报，IP 可以将数据报重新组装并在报头域内注明。IP 是 TCP、UDP 的载体，IP 协议好比是轮船，而 TCP 或 UDP 则是货物，只要告诉船长送达的目的地，具体他怎样去，选择什么路就交给 IP 协议自动处理。IP 协议在网络协议家族的地位如图 1.1 所示。

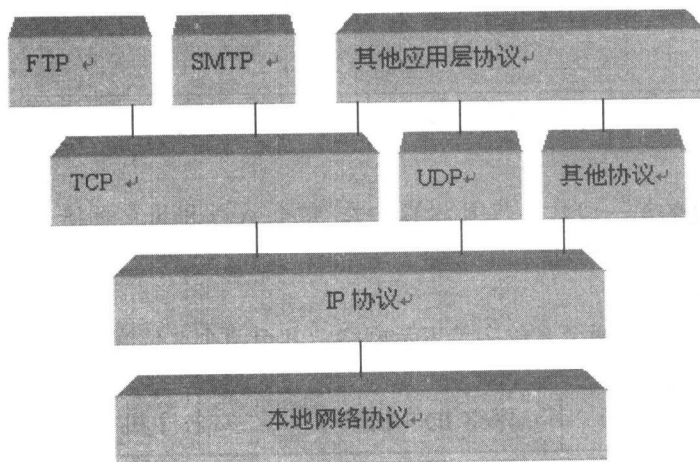


图 1.1

2. IP 基本操作

IP 主要使用四项技术进行工作和提供服务：服务类型，生存时间，选项和报头校验码。

(1) 服务类型是网络服务质量的选择。服务类型是由抽象参数确定的优先级，时延，吞吐量和可靠性的结合物。这些参数和一些实际对应的网络服务对应。这种服务类型由网关使用，用于特定的网络，或是用于下下一个要经过的网络，或是下一个要对这个数据报进行路由的网关上选择实际的传送参数。

(2) 生存时间是数据报可以生存的时间上限，它由发送者设置，是数据报在网络中最长的生存时间，如果超时会抛弃数据报。网络中每个结点都会处理存在于报头中的生存时间，单位是秒，最长的生存时间为 255 秒。

(3) 选项包括时间戳、安全和特殊路由，发送者决定有没有选项。

(4) 报头校验码保证数据的正确传输。如果校验出错，抛弃数据报。

IP 的工作原理是：IP 数据报到达网关后，本地网络接口去掉网络头（如果存在网络头），将结果传送给 IP 模块。由这个 IP 模块决定本地网络地址，然后传送数据报到本地网络接口。该本地网络接口创建一个本地网络头加在数据报上，将数据报传送到目的主机。在目的主机上，其网络接口去掉数据报上的网络头，将结果交给 IP 模块。IP 模块决定把数据报向哪一个应用程序传送，然后系统发出系统调用，IP 模块返回源地址和其他参数。

3. IP 地址

IP 寻址是通过搜寻 IP 地址来完成的。众所周知，在 Internet 上，每一台主机至少要拥有一个 IP 地址。一般来说，一台机器的 IP 地址数和网络接口数是相同的，但有些情况下，一个接口可能会有两个或多个 IP 地址，这些情况是很少的。正因为网上的计算机都拥有 IP 地址，所以 IP 才能通过事先约定的变化规则，找到目标计算机。

IP 地址的主要类型有五种：A，B，C，D，E 类，每类地址都是由 4 个字节组成。

(1) A 类地址

在 A 类地址中，第一个 8 位字节表示网络，其余 3 个 8 位字节用来标识主机，如图所示。

0	网络 ID (7 位)	主机 ID (24 位)
---	-------------	--------------

A 类 IP 地址的第一段数字范围为 1~127，每个 A 类地址可连接 163877064 台主机，Internet 上有 126 个 A 类地址。

(2) B 类地址

在 B 类地址中，前两个 8 位字节表示网络，其余两个 8 位字节表示主机，如图所示，

1	0	网络 ID (14 位)	主机 ID (16 位)
---	---	--------------	--------------

B 类 IP 地址的第一段数字范围为 128~191，每个 B 类地址可连接 64516 台主机，Internet 上有 16256 个 B 类地址。

(3) C 类地址

C 类地址使用前 3 个 8 位字节作为网络部分，只有一个 8 位字节留给主机，如下所示。

1	1	0	网络 ID (21 位)	主机 ID (8 位)
---	---	---	--------------	-------------

C 类 IP 地址的第一段数字范围为 192~223，每个 C 类地址连接 254 台主机，Internet 上有 2054512 个 C 类地址。

(4) D 类 IP 地址的第一段数字范围为 224~239，作为备用。

(5) E 类地址是保留，其第一段数字范围为 240~254。

通过上面的学习，我们发现，从理论上讲 A 类或 B 类主机可多达数千台或上亿台，这在实践中是不切合实际的，因为不可能有任何一个网络的主机数会有这么多。为了解决这个问题，人们使用了子网 (Subnet) 的概念，就是把 A、B 类地址进一步地细化。如下表格就是一个子网化的 B 类地址。

1	0	网络 ID (14 位)	子网 ID (8 位)	主机 ID (8 位)
---	---	--------------	-------------	-------------

为了将子网号和主机号分离开来，我们还需要使用子网掩码。子网掩码是一个 32 位的值，其中网络 ID 和子网 ID 部分全部被置 1，主机的部分被置零。当知道了子网掩码和一个主机的 IP 地址，要想得到网络号和子网号，可以把子网掩码和 IP 地址进行位运算中的 (AND) 运算，这样就去掉了主机号，剩下的网络号和子网号可以通过地址类型进行分离。

例如：有一个 C 类地址为：

192. 9. 200. 2

其缺省的子网掩码为：

255. 255. 255. 0

则它的网络号和主机号可按如下方法得到：

① IP 地址 192. 9. 200. 2 转换为二进制

11000000 00001001 11001000 00000010

② 将子网掩码 255. 255. 255. 0 转换为二进制

11111111 11111111 11111111 00000000

③ 将两个二进制数逻辑与 (AND) 运算后得出的结果即为网络部分

11000000 00001001 11001000 00000010

AND

11111111 11111111 11111111 00000000

11000000 00001001 11001000 00000000

结果为 192. 9. 200. 0，即网络号为 192. 9. 200. 0。

④ 将子网掩码取反再与 IP 地址逻辑与 (AND) 后得到的结果即为主机部分

11000000 00001001 11001000 00000010

AND

00000000 00000000 00000000 11111111

00000000 00000000 00000000 00000010

结果为 0. 0. 0. 2, 即主机号为 2。

需要注意的是, IP 地址除了标识一台主机外, 还有几种具有特殊意义的特殊形式。

- 广播地址

TCP/IP 规定, 主机号全为“1”的网络地址用于广播之用, 叫做广播地址。所谓广播, 指同时向网上所有主机发送报文。

- 有限广播

TCP/IP 规定, 32 字节全为“1”的网间网地址用于本网广播, 该地址叫做有限广播地址 (limitedbroadcast address)。

- “0”地址

TCP/IP 协议规定, 各位全为“0”的网络号被解释成“本”网络。

- 回送地址

A 类网络地址 127 是一个保留地址, 用于网络软件测试以及本地机进程间通信, 叫做回送地址 (loopback address)。无论什么程序, 一旦使用回送地址发送数据, 协议软件立即返回之, 不进行任何网络传输。TCP/IP 协议规定: 含网络号 127 的分组不能出现在任何网络上; 主机和网关不能为该地址广播任何寻径信息。

1.1.2 域名简介

域名是表示一个特定的 IP 地址的唯一标识名, 由于 IP 地址仅仅是一串数值, 所以利用域名系统就使得站点地址更容易识别和记忆。每个域名都由几部分组成, 每部分我们称之为域, 域与域之间也是用点号隔开, 最末的一组叫做根域, 前面的组叫做子域。目前 Internet 中常用的根域含义为:

COM——商业组织	GOV 政府组织
NET——主要网络支持中心	INT 国际组织
FIRM——商业公司	REC 强调消遣和娱乐的实体
ATRS——强调文件和娱乐的实体	EDU 教育机构
MIL——军事部门	INFO 提供信息服务的组织
NOM——用于个人或人体	STORE 从事商业销售的企业
WEB——与 WWW 特别相关的实体	ORG 其他组织

与 IP 地址一样, 一些域名也是有保留的, 下面四组域名已经被保留:

“.test”, “.example”, “.invalid”, “.localhost”

DNS 则用于域名与 IP 地址的解析, 其工作原理如下:

(1) DNS 客户向本地的 DNS 服务器发出个解析请求

(2) 如果该 DNS 本身含有客户需要的数据, 则直接返回给客户, 如果没有, 则服务器与其他 DNS 服务器联系, 从其他 DNS 服务器上获取数据, 然后返回给用户。

(3) 如果查找不到, 返回解析失败的异常信息。

1.2 与 DNS 相关类的简介

在.NET 平台上, Dns 类, IPHostEntry 类, IPAddress 类, DnsPermission 类实现了 DNS 的功能。下面简单介绍一下这几个类。

1.2.1 Dns 类

在.NET 平台上,对 DNS 的支持是靠 DNS 类实现的,其名字空间为 System.Net。DNS 类中常用的方法如下(为做到全书统一,凡本书中需要举例的方法前面一律标有“●”符号,凡本书没有举例的方法前面一律标有“○”符号。):

● GetHostName()方法

方法功能: 该方法用于获取本地主机名称

方法原型: `public static string GetHostName();`

返回值: 字符串,字符串里包含主机名称信息

异常信息:

异常类型	条件
SocketException (套接字异常)	解析本地主机名称时发生错误

● Resolve()方法

方法功能: 该方法用于将域名转化为 IP 地址

方法原型:

```
public static IPHostEntry Resolve(  
string hostName//主机名,比如“www.aaa.bbb.com”或者“123.221.111.100”  
);
```

返回值: IPHostEntry 类型值,该值中包含主机信息

异常信息:

异常类型	条件
ArgumentNullException (参数空异常)	主机名空
SocketException (套接字异常)	解析本地主机名称时发生错误

● BeginResolve()方法

方法功能: 该方法用于异步将域名解析为 IP 地址

```
public static IAsyncResult BeginResolve(  
string hostName, // 主机名 (域名)  
AsyncCallback requestCallback, // 异步回调  
object stateObject 自定义对象  
);
```

返回值: IAsyncResult 类型值

异常信息:

异常类型	条件
SocketException (套接字异常)	解析本地主机名称时发生错误

● EndResolve()

方法功能: 该方法用于结束 BeginResolve()方法

方法原型:

```
public static IPEndPoint EndResolve(
    IAsyncResult asyncResult // BeginResolve()方法返回值
);
```

返回值: 无

异常信息:

异常类型	条件
ArgumentNullException (参数空异常)	AsyncResult 空

● GetHostByName()方法

方法功能: 该方法用于通过主机名获取主机信息

方法原型:

```
public static IPEndPoint GetHostByName(
    string hostName // 主机域名
);
```

返回值: IPEndPoint 该值包含主机信息

异常信息:

异常类型	条件
ArgumentNullException (参数空异常)	hostName 空
SocketException (套接字异常)	解析主机时出错

● BeginGetHostByName()方法

方法功能: 该方法用于通过域名异步获取主机信息

方法原型:

```
public static IAsyncResult BeginGetHostByName(
    string hostName, // 主机名称
    AsyncCallback requestCallback, // 异步回调
    object stateObject // 状态对象
);
```

返回值: IAsyncResult

异常信息:

异常类型	条件
SocketException (套接字异常)	解析主机时出错

● EndGetHostByName()方法

方法功能：该方法用于结束 BeginGetHostByName()方法

```
public static IPEndPoint EndGetHostByName(
    IAsyncResult asyncResult // BeginGetHostByName() 返回值
);
```

返回值：IPHostEntry 类型值

● GetHostByAddress()方法

方法功能：该方法用于通过 IP 地址获取主机信息

方法原型一：

```
public static IPHostEntry GetHostByAddress(
    IPAddress address // 主机 IP 地址
);
```

返回值：IPHostEntry 类型值

异常信息：

异常类型	条件
ArgumentNullException (参数空异常)	address 空
SocketException (套接字异常)	解析主机时出错

方法原型二：

```
public static IPHostEntry GetHostByAddress(
    string address // 字符串地址
);
```

返回值：IPHostEntry 类型值

异常信息：

异常类型	条件
ArgumentNullException (参数空异常)	address 空
SocketException (套接字异常)	解析主机时出错

● IpToString()方法

方法功能：该方法用于将整型数值转化为字符串 IP 地址表现形式

方法原型：

```
public static string IpToString(
    int address // 整型数值, 如 22334
);
```

返回值：字符串