

1 System.Net.Dns Class

2
3

```
4 [ILASM]  
5 .class public sealed Dns extends System.Object  
6 [C#]  
7 public sealed class Dns
```

8 Assembly Info:

- 9
- Name: System
 - 10 • Public Key: [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00]
 - 11 • Version: 1.0.x.x
 - 12 • Attributes:
 - 13 ○ CLSCompliantAttribute(true)

14 Summary

15

16 Obtains domain information from the Domain Name System as defined
17 by IETF RFC 1035 and RFC 1036.

18 Inherits From: System.Object

19

20 **Library:** Networking

21

22 **Thread Safety:** All public static members of this type are safe for multithreaded
23 operations. No instance members are guaranteed to be thread safe.

24

25 Description

26 The **System.Net.Dns** class creates and sends queries to obtain
27 information about a host server from the Internet Domain Name
28 System (DNS). In order to access DNS, the machine executing the
29 query is required to be connected to a network. If the query is
30 executed on a machine that does not have access to a domain name
31 server, a **System.Net.Sockets.SocketException** is thrown.

32

33 Information from the DNS query is returned in an instance of the
34 **System.Net.IPHostEntry** class. If the specified host has more than
35 one entry in the DNS database, the **System.Net.IPHostEntry**
36 instance contains multiple IP addresses and aliases.

37

38 [Note: See the **System.Net.IPHostEntry** class page for an example
39 that uses the **System.Net.Dns** class.]

40

1 Dns.BeginGetHostByName(System.String, 2 System.AsyncCallback, System.Object) 3 Method

```
4 [ILASM]  
5 .method public hidebysig static class System.IAsyncResult  
6 BeginGetHostByName(string hostName, class  
7 System.AsyncCallback requestCallback, object stateObject)  
  
8 [C#]  
9 public static IAsyncResult BeginGetHostByName(string  
10 hostName, AsyncCallback requestCallback, object  
11 stateObject)
```

12 Summary

13 Begins the asynchronous execution of a DNS query to obtain address
14 information about the specified host.

15 Parameters

16
17

Parameter	Description
<i>hostName</i>	A System.String containing the DNS name of the host.
<i>requestCallback</i>	A System.AsyncCallback delegate, or null .
<i>stateObject</i>	An application-defined object, or null .

18
19
20

19 Return Value

21 A **System.IAsyncResult** instance that contains information about the
22 asynchronous operation.

23 Description

24 This method starts an asynchronous request for information about the
25 specified host. To retrieve the results of the query and release
26 resources allocated by this method, call the
27 **System.Net.Dns.EndGetHostByName** method and specify the
28 **System.IAsyncResult** object returned by this method. [Note: The
29 **System.Net.Dns.EndGetHostByName** method should be called
30 exactly once for each call to
31 **System.Net.Dns.BeginGetHostByName**.]
32

33 If the *requestCallback* parameter is not **null**, the method referenced
34 by *requestCallback* is invoked when the asynchronous operation
35 completes. The **System.IAsyncResult** object returned by this method
36 is passed as the argument to the method referenced by

1 *requestCallback*. The method referenced by *requestCallback* can
2 retrieve the results of the query by calling
3 **System.Net.Dns.EndGetHostByName**.

4
5 The *stateObject* parameter can be any object that the caller wishes to
6 have available for the duration of the asynchronous operation. This
7 object is available via the **System.IAsyncResult.AsyncState**
8 property of the object returned by this method.

9 **Exceptions**

10

11

Exception	Condition
System.ArgumentNullException	<i>hostName</i> is null .
System.Net.Sockets.SocketException	An error was encountered executing the DNS query.
System.Security.SecurityException	The caller does not have permission to access DNS information.

12

13

14

15

Permissions

Permission	Description
System.Net.DnsPermission	Requires unrestricted permission for accessing DNS information. See System.Net.DnsPermission and System.Security.Permissions.PermissionState.Unrestricted .

16

17

18

Dns.BeginResolve(System.String, System.AsyncCallback, System.Object) Method

```
[ILASM]  
.method public hidebysig static class System.IAsyncResult  
BeginResolve(string hostName, class System.AsyncCallback  
requestCallback, object stateObject)
```

```
[C#]  
public static IAsyncResult BeginResolve(string hostName,  
AsyncCallback requestCallback, object stateObject)
```

Summary

Begins the asynchronous execution of a DNS query to resolve a host name or IP address.

Parameters

Parameter	Description
<i>hostName</i>	A System.String containing the DNS name or IP address of the host.
<i>requestCallback</i>	A System.AsyncCallback delegate or null .
<i>stateObject</i>	An application-defined object, or null .

Return Value

A **System.IAsyncResult** instance that contains information about the asynchronous operation.

Description

This method starts an asynchronous request for DNS information about the specified host. To retrieve the results of the query and release resources allocated by this method, call the **System.Net.Dns.EndResolve** method, and specify the **System.IAsyncResult** object returned by this method. [Note: The **System.Net.Dns.EndResolve** method should be called exactly once for each call to **System.Net.Dns.BeginResolve**].

If the *requestCallback* parameter is not **null**, the method referenced by *requestCallback* is invoked when the asynchronous operation completes. The **System.IAsyncResult** object returned by this method is passed as the argument to the method referenced by

1 *requestCallback*. The method referenced by *requestCallback* can
2 retrieve the results of the query by calling
3 **System.Net.Dns.EndResolve**.

4
5 The *stateObject* parameter can be any object that the caller wishes to
6 have available for the duration of the asynchronous operation. This
7 object is available via the **System.IAsyncResult.AsyncState**
8 property of the object returned by this method.

9 **Exceptions**

10
11

Exception	Condition
System.ArgumentNullException	<i>hostName</i> is null .
System.Net.Sockets.SocketException	An error was encountered executing the DNS query.
System.Security.SecurityException	The caller does not have permission to access DNS information.

12
13
14
15

Permissions

Permission	Description
System.Net.DnsPermission	Requires unrestricted permission for accessing DNS information. See System.Net.DnsPermission and System.Security.Permissions.PermissionState.Unrestricted .

16
17
18

1 Dns.EndGetHostByName(System.IAsyncResult) 2 result) Method

```
3 [ILASM]  
4 .method public hidebysig static class  
5 System.Net.IPHostEntry EndGetHostByName(class  
6 System.IAsyncResult asyncResult)  
  
7 [C#]  
8 public static IPHostEntry EndGetHostByName(IAsyncResult  
9 asyncResult)
```

10 Summary

11 Ends an asynchronous query to obtain address information about the
12 specified host.

13 Parameters

14
15

Parameter	Description
<i>asyncResult</i>	The System.IAsyncResult object that holds the state information for the asynchronous operation.

16
17
18

Return Value

19 A **System.Net.IPHostEntry** object containing DNS address
20 information about a host.

21 Description

22 This method blocks if the asynchronous operation has not completed.
23
24 The **System.Net.Dns.EndGetHostByName** method completes an
25 asynchronous request for DNS information that was started with a call
26 to **System.Net.Dns.BeginGetHostByName**. The object specified for
27 the *asyncResult* parameter is required to be the same object as was
28 returned by the **System.Net.Dns.BeginGetHostByName** method
29 call that began the request.
30
31 If the **System.Net.Dns.EndGetHostByName** method is invoked via
32 the **System.AsyncCallback** delegate specified to the
33 **System.Net.Dns.BeginGetHostByName** method, the *asyncResult*
34 parameter is the **System.IAsyncResult** argument passed to the
35 delegate's method.

1 **Exceptions**

2

3

Exception	Condition
System.ArgumentNullException	<i>asyncResult</i> is null .

4

5

6

1 Dns.EndResolve(System.IAsyncResult) 2 Method

```
3 [ILASM]  
4 .method public hidebysig static class  
5 System.Net.IPHostEntry EndResolve(class System.IAsyncResult  
6 asyncResult)  
  
7 [C#]  
8 public static IPHostEntry EndResolve(IAsyncResult  
9 asyncResult)
```

10 Summary

11 Ends an asynchronous query to resolve a host name or IP address.

12 Parameters

13
14

Parameter	Description
<i>asyncResult</i>	The System.IAsyncResult object that holds the state information for the asynchronous operation.

15
16
17

16 Return Value

18 A **System.Net.IPHostEntry** object containing address information
19 about a host.

20 Description

21 This method blocks if the asynchronous operation has not completed.

22

23 The **System.Net.Dns.EndResolve** method completes an
24 asynchronous request for DNS information that was started with a call
25 to **System.Net.Dns.BeginResolve**. The object specified for the
26 *asyncResult* parameter is required to be the same object as was
27 returned by the **System.Net.Dns.BeginResolve** method call that
28 began the request.

29

30 If the **System.Net.Dns.EndResolve** method is invoked via the
31 **System.AsyncCallback** delegate specified to the
32 **System.Net.Dns.BeginResolve** method, the *asyncResult* parameter
33 is the **System.IAsyncResult** argument passed to the delegate's
34 method.

1 **Exceptions**

2

3

Exception	Condition
System.ArgumentNullException	<i>asyncResult</i> is null .

4

5

6

1 Dns.GetHostByAddress(System.String)

2 Method

```
3 [ILASM]  
4 .method public hidebysig static class  
5 System.Net.IPHostEntry GetHostByAddress(string address)  
  
6 [C#]  
7 public static IPHostEntry GetHostByAddress(string address)
```

8 Summary

9 Queries DNS for the DNS host name of the specified IP address.

10 Parameters

11
12

Parameter	Description
<i>address</i>	A System.String containing an IP address.

13
14
15

14 Return Value

16 A **System.Net.IPHostEntry** instance containing the host information.

17 Description

18 The IP address specified by the *address* parameter is required to be in
19 dotted-quad notation (for example, "192.168.1.2").

20 Exceptions

21
22

Exception	Condition
System.ArgumentNullException	<i>address</i> is null .
System.FormatException	<i>address</i> is not a valid IP address.
System.Net.Sockets.SocketException	An error was encountered executing the DNS query.
System.Security.SecurityException	The caller does not have permission to access DNS information.

23
24
25
26

24 Permissions

Permission	Description
------------	-------------

1
2
3

System.Net.DnsPermission	Requires unrestricted permission for accessing DNS information. See System.Net.DnsPermission and System.Security.Permissions.PermissionState.Unrestricted .
---------------------------------	---

1 Dns.GetHostByAddress(System.Net.IPAddress) Method

```
3 [ILASM]
4 .method public hidebysig static class
5 System.Net.IPHostEntry GetHostByAddress(class
6 System.Net.IPAddress address)
7
8 [C#]
9 public static IPHostEntry GetHostByAddress(IPAddress
address)
```

10 Summary

11 Queries DNS for the DNS host name of the specified IP address.

12 Parameters

13
14

Parameter	Description
<i>address</i>	A System.Net.IPAddress instance.

15
16
17

Return Value

18 A **System.Net.IPHostEntry** instance containing the host information.

19 Exceptions

20
21

Exception	Condition
System.ArgumentNullException	<i>address</i> is null .
System.Net.Sockets.SocketException	An error was encountered executing the DNS query.
System.Security.SecurityException	The caller does not have permission to access DNS information.

22
23
24
25

Permissions

Permission	Description
System.Net.DnsPermission	Requires unrestricted permission for accessing DNS information. See System.Net.DnsPermission and System.Security.Permissions.PermissionState.Unrestricted .

1
2
3

1 Dns.GetHostByName(System.String)

2 Method

```
3 [ILASM]  
4 .method public hidebysig static class  
5 System.Net.IPHostEntry GetHostByName(string hostName)  
  
6 [C#]  
7 public static IPHostEntry GetHostByName(string hostName)
```

8 Summary

9 Queries DNS for address information about the specified host.

10 Parameters

11
12

Parameter	Description
<i>hostName</i>	A System.String containing the DNS name of the host.

13
14
15

14 Return Value

16 A **System.Net.IPHostEntry** object containing host information for
17 the address specified in *hostName*.

18 Description

19 [Note: To retrieve host information asynchronously, use the
20 **System.Net.Dns.BeginGetHostByName** and
21 **System.Net.Dns.EndGetHostByName** methods.
22

23 See the **System.Net.IPHostEntry** class page for an example that
24 uses the **System.Net.Dns.GetHostByName** method.]

25 Exceptions

26
27

Exception	Condition
System.ArgumentNullException	<i>hostName</i> is null .
System.Net.Sockets.SocketException	An error was encountered executing the DNS query.
System.Security.SecurityException	The caller does not have permission to access DNS information.

28
29

29 Permissions

1
2

Permission	Description
System.Net.DnsPermission	Requires unrestricted permission for accessing DNS information. See System.Net.DnsPermission and System.Security.Permissions.PermissionState.Unrestricted .

3
4
5

1 Dns.GetHostName() Method

```
2 [ILASM]  
3 .method public hidebysig static string GetHostName()  
4 [C#]  
5 public static string GetHostName()
```

6 Summary

7 Gets the DNS host name of the local machine.

8 Return Value

9

10 A **System.String** containing the DNS host name of the local machine.

11 Exceptions

12

13

Exception	Condition
System.Net.Sockets.SocketException	An error was encountered resolving the local host name.
System.Security.SecurityException	The caller does not have permission to access DNS information.

14

15 Permissions

16

17

Permission	Description
System.Net.DnsPermission	Requires unrestricted permission for accessing DNS information. See System.Net.DnsPermission and System.Security.Permissions.PermissionState.Unrestricted .

18

19

20

1 Dns.Resolve(System.String) Method

```
2 [ILASM]  
3 .method public hidebysig static class  
4 System.Net.IPHostEntry Resolve(string hostName)  
  
5 [C#]  
6 public static IPHostEntry Resolve(string hostName)
```

7 Summary

8 Resolves a DNS host name or IP address to a
9 **System.Net.IPHostEntry** instance.

10 Parameters

11
12

Parameter	Description
<i>hostName</i>	A System.String containing a DNS-style host name or IP address.

13
14
15

14 Return Value

16 A **System.Net.IPHostEntry** instance containing address information
17 about the host specified in *hostName*.

18 Description

19 The **System.Net.Dns.Resolve** method queries a DNS server for the
20 IP address associated with a host name or an IP address in dotted-
21 quad notation.

22
23 When *hostName* is a DNS-style host name associated with multiple IP
24 addresses, only the first IP address that resolves to that host name is
25 returned.

26 Exceptions

27
28

Exception	Condition
System.ArgumentNullException	<i>hostName</i> is null .
System.Net.Sockets.SocketException	An error was encountered executing the DNS query.
System.Security.SecurityException	The caller does not have permission to access DNS information.

1
2 **Example**
3

4 The following example demonstrates the use of the
5 **System.Net.Dns.Resolve** method.

```
6 [C#  
7  
8 using System;  
9 using System.Net;  
10  
11 public class DnsTest {  
12     public static void Main() {  
13         IPHostEntry hostInfo1 = Dns.Resolve("www.contoso.com");  
14         DisplayHostInfo(hostInfo1);  
15         Console.WriteLine();  
16     }  
17  
18     public static void DisplayHostInfo(IPHostEntry hostInfo) {  
19         string[] aliases = hostInfo.Aliases;  
20         IPAddress[] addresses = hostInfo.AddressList;  
21         Console.WriteLine("The host name is: {0}",  
22 hostInfo.HostName);  
23  
24         for(int x = 0; x < aliases.Length; x++)  
25             Console.WriteLine("Alias {0} == {1}", aliases[x],  
26 addresses[x]);  
27     }  
28 }
```

29 The output is

30 The host name is: contoso.com

31
32
33
34 Alias www.contoso.com == 207.46.230.186
35

36 **Permissions**
37
38

Permission	Description
System.Net.DnsPermission	Requires unrestricted permission for accessing DNS information. This method also asserts unrestricted DnsPermission . See System.Net.DnsPermission

1
2

	and System.Security.Permissions.PermissionState. Unrestricted.
--	--