**How to: Create a Socket**

**.NET Framework 4**

Before you can use a socket to communicate with remote devices, the socket must be initialized with protocol and network address information. The constructor for the [Socket](http://msdn.microsoft.com/en-us/library/system.net.sockets.socket.aspx) class has parameters that specify the address family, socket type, and protocol type that the socket uses to make connections.

[**Example**](javascript:void(0))

The following example creates a Socket that can be used to communicate on a TCP/IP-based network, such as the Internet.

**C#**

Socket s = new Socket(AddressFamily.InterNetwork,

SocketType.Stream, ProtocolType.Tcp);

To use UDP instead of TCP, change the protocol type, as in the following example:

**C#**

Socket s = new Socket(AddressFamily.InterNetwork,

SocketType.Dgram, ProtocolType.Udp);

The [AddressFamily](http://msdn.microsoft.com/en-us/library/system.net.sockets.addressfamily.aspx) enumeration specifies the standard address families used by the **Socket** class to resolve network addresses (for example, the **AddressFamily.InterNetwork** member specifies the IP version 4 address family).

The [SocketType](http://msdn.microsoft.com/en-us/library/system.net.sockets.sockettype.aspx) enumeration specifies the type of socket (for example, the **SocketType.Stream** member indicates a standard socket for sending and receiving data with flow control).

The [ProtocolType](http://msdn.microsoft.com/en-us/library/system.net.sockets.protocoltype.aspx) enumeration specifies the network protocol to use when communicating on the **Socket** (for example, **ProtocolType.Tcp** indicates that the socket uses TCP; **ProtocolType.Udp** indicates that the socket uses UDP).

After a **Socket** is created, it can either initiate a connection to a remote endpoint or receive connections from remote devices.