| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/java/net/SocketAddress.html) | **Use** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV   NEXT | [**FRAMES**](http://docs.google.com/index.html?java/net//class-useSocketAddress.html)    [**NO FRAMES**](http://docs.google.com/SocketAddress.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

**Uses of Class**

**java.net.SocketAddress**

| Packages that use [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | |
| --- | --- |
| [**java.net**](#3znysh7) | Provides the classes for implementing networking applications. |
| [**java.nio.channels**](#2et92p0) | Defines channels, which represent connections to entities that are capable of performing I/O operations, such as files and sockets; defines selectors, for multiplexed, non-blocking I/O operations. |

| Uses of [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) in [java.net](http://docs.google.com/java/net/package-summary.html) | |
| --- | --- |

| Subclasses of [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) in [java.net](http://docs.google.com/java/net/package-summary.html) | |
| --- | --- |
| class | [**InetSocketAddress**](http://docs.google.com/java/net/InetSocketAddress.html)            This class implements an IP Socket Address (IP address + port number) It can also be a pair (hostname + port number), in which case an attempt will be made to resolve the hostname. |

| Methods in [java.net](http://docs.google.com/java/net/package-summary.html) that return [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | |
| --- | --- |
| [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | **Proxy.**[**address**](http://docs.google.com/java/net/Proxy.html#address())()            Returns the socket address of the proxy, or null if its a direct connection. |
| [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | **Socket.**[**getLocalSocketAddress**](http://docs.google.com/java/net/Socket.html#getLocalSocketAddress())()            Returns the address of the endpoint this socket is bound to, or null if it is not bound yet. |
| [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | **ServerSocket.**[**getLocalSocketAddress**](http://docs.google.com/java/net/ServerSocket.html#getLocalSocketAddress())()            Returns the address of the endpoint this socket is bound to, or null if it is not bound yet. |
| [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | **DatagramSocket.**[**getLocalSocketAddress**](http://docs.google.com/java/net/DatagramSocket.html#getLocalSocketAddress())()            Returns the address of the endpoint this socket is bound to, or null if it is not bound yet. |
| [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | **Socket.**[**getRemoteSocketAddress**](http://docs.google.com/java/net/Socket.html#getRemoteSocketAddress())()            Returns the address of the endpoint this socket is connected to, or null if it is unconnected. |
| [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | **DatagramSocket.**[**getRemoteSocketAddress**](http://docs.google.com/java/net/DatagramSocket.html#getRemoteSocketAddress())()            Returns the address of the endpoint this socket is connected to, or null if it is unconnected. |
| [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | **DatagramPacket.**[**getSocketAddress**](http://docs.google.com/java/net/DatagramPacket.html#getSocketAddress())()            Gets the SocketAddress (usually IP address + port number) of the remote host that this packet is being sent to or is coming from. |

| Methods in [java.net](http://docs.google.com/java/net/package-summary.html) with parameters of type [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | |
| --- | --- |
| void | **Socket.**[**bind**](http://docs.google.com/java/net/Socket.html#bind(java.net.SocketAddress))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) bindpoint)            Binds the socket to a local address. |
| void | **ServerSocket.**[**bind**](http://docs.google.com/java/net/ServerSocket.html#bind(java.net.SocketAddress))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) endpoint)            Binds the ServerSocket to a specific address (IP address and port number). |
| void | **DatagramSocket.**[**bind**](http://docs.google.com/java/net/DatagramSocket.html#bind(java.net.SocketAddress))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) addr)            Binds this DatagramSocket to a specific address & port. |
| void | **ServerSocket.**[**bind**](http://docs.google.com/java/net/ServerSocket.html#bind(java.net.SocketAddress,%20int))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) endpoint, int backlog)            Binds the ServerSocket to a specific address (IP address and port number). |
| void | **Socket.**[**connect**](http://docs.google.com/java/net/Socket.html#connect(java.net.SocketAddress))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) endpoint)            Connects this socket to the server. |
| void | **DatagramSocket.**[**connect**](http://docs.google.com/java/net/DatagramSocket.html#connect(java.net.SocketAddress))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) addr)            Connects this socket to a remote socket address (IP address + port number). |
| protected abstract  void | **SocketImpl.**[**connect**](http://docs.google.com/java/net/SocketImpl.html#connect(java.net.SocketAddress,%20int))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) address, int timeout)            Connects this socket to the specified port number on the specified host. |
| void | **Socket.**[**connect**](http://docs.google.com/java/net/Socket.html#connect(java.net.SocketAddress,%20int))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) endpoint, int timeout)            Connects this socket to the server with a specified timeout value. |
| abstract  void | **ProxySelector.**[**connectFailed**](http://docs.google.com/java/net/ProxySelector.html#connectFailed(java.net.URI,%20java.net.SocketAddress,%20java.io.IOException))([URI](http://docs.google.com/java/net/URI.html) uri, [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) sa, [IOException](http://docs.google.com/java/io/IOException.html) ioe)            Called to indicate that a connection could not be established to a proxy/socks server. |
| void | **MulticastSocket.**[**joinGroup**](http://docs.google.com/java/net/MulticastSocket.html#joinGroup(java.net.SocketAddress,%20java.net.NetworkInterface))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) mcastaddr, [NetworkInterface](http://docs.google.com/java/net/NetworkInterface.html) netIf)            Joins the specified multicast group at the specified interface. |
| protected abstract  void | **DatagramSocketImpl.**[**joinGroup**](http://docs.google.com/java/net/DatagramSocketImpl.html#joinGroup(java.net.SocketAddress,%20java.net.NetworkInterface))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) mcastaddr, [NetworkInterface](http://docs.google.com/java/net/NetworkInterface.html) netIf)            Join the multicast group. |
| void | **MulticastSocket.**[**leaveGroup**](http://docs.google.com/java/net/MulticastSocket.html#leaveGroup(java.net.SocketAddress,%20java.net.NetworkInterface))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) mcastaddr, [NetworkInterface](http://docs.google.com/java/net/NetworkInterface.html) netIf)            Leave a multicast group on a specified local interface. |
| protected abstract  void | **DatagramSocketImpl.**[**leaveGroup**](http://docs.google.com/java/net/DatagramSocketImpl.html#leaveGroup(java.net.SocketAddress,%20java.net.NetworkInterface))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) mcastaddr, [NetworkInterface](http://docs.google.com/java/net/NetworkInterface.html) netIf)            Leave the multicast group. |
| void | **DatagramPacket.**[**setSocketAddress**](http://docs.google.com/java/net/DatagramPacket.html#setSocketAddress(java.net.SocketAddress))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) address)            Sets the SocketAddress (usually IP address + port number) of the remote host to which this datagram is being sent. |

| Constructors in [java.net](http://docs.google.com/java/net/package-summary.html) with parameters of type [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | |
| --- | --- |
| [**DatagramPacket**](http://docs.google.com/java/net/DatagramPacket.html#DatagramPacket(byte%5B%5D,%20int,%20int,%20java.net.SocketAddress))(byte[] buf, int offset, int length, [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) address)            Constructs a datagram packet for sending packets of length length with offset ioffsetto the specified port number on the specified host. |
| [**DatagramPacket**](http://docs.google.com/java/net/DatagramPacket.html#DatagramPacket(byte%5B%5D,%20int,%20java.net.SocketAddress))(byte[] buf, int length, [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) address)            Constructs a datagram packet for sending packets of length length to the specified port number on the specified host. |
| [**DatagramSocket**](http://docs.google.com/java/net/DatagramSocket.html#DatagramSocket(java.net.SocketAddress))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) bindaddr)            Creates a datagram socket, bound to the specified local socket address. |
| [**MulticastSocket**](http://docs.google.com/java/net/MulticastSocket.html#MulticastSocket(java.net.SocketAddress))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) bindaddr)            Create a MulticastSocket bound to the specified socket address. |
| [**Proxy**](http://docs.google.com/java/net/Proxy.html#Proxy(java.net.Proxy.Type,%20java.net.SocketAddress))([Proxy.Type](http://docs.google.com/java/net/Proxy.Type.html) type, [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) sa)            Creates an entry representing a PROXY connection. |

| Uses of [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) in [java.nio.channels](http://docs.google.com/java/nio/channels/package-summary.html) | |
| --- | --- |

| Methods in [java.nio.channels](http://docs.google.com/java/nio/channels/package-summary.html) that return [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | |
| --- | --- |
| abstract  [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | **DatagramChannel.**[**receive**](http://docs.google.com/java/nio/channels/DatagramChannel.html#receive(java.nio.ByteBuffer))([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html) dst)            Receives a datagram via this channel. |

| Methods in [java.nio.channels](http://docs.google.com/java/nio/channels/package-summary.html) with parameters of type [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | |
| --- | --- |
| abstract  boolean | **SocketChannel.**[**connect**](http://docs.google.com/java/nio/channels/SocketChannel.html#connect(java.net.SocketAddress))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) remote)            Connects this channel's socket. |
| abstract  [DatagramChannel](http://docs.google.com/java/nio/channels/DatagramChannel.html) | **DatagramChannel.**[**connect**](http://docs.google.com/java/nio/channels/DatagramChannel.html#connect(java.net.SocketAddress))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) remote)            Connects this channel's socket. |
| static [SocketChannel](http://docs.google.com/java/nio/channels/SocketChannel.html) | **SocketChannel.**[**open**](http://docs.google.com/java/nio/channels/SocketChannel.html#open(java.net.SocketAddress))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) remote)            Opens a socket channel and connects it to a remote address. |
| abstract  int | **DatagramChannel.**[**send**](http://docs.google.com/java/nio/channels/DatagramChannel.html#send(java.nio.ByteBuffer,%20java.net.SocketAddress))([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html) src, [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) target)            Sends a datagram via this channel. |

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/java/net/SocketAddress.html) | **Use** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV   NEXT | [**FRAMES**](http://docs.google.com/index.html?java/net//class-useSocketAddress.html)    [**NO FRAMES**](http://docs.google.com/SocketAddress.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).