Class java.net.Socket

```
java.lang.Object
   |
   +----java.net.Socket
```

---

public final class **Socket**
extends Object

The client Socket class. It uses a SocketImpl to implement the actual socket operations. It is done this way so that you are able to change socket implementations depending on the kind of firewall that is used. You can change socket implementations by setting the SocketImplFactory.

Version:
1.13, 10/24/95
Author:
Jonathan Payne, Arthur van Hoff

Constructor Index

o **Socket**(String, int)
Creates a stream socket and connects it to the specified port on the specified host.
o **Socket**(String, int, boolean)
Creates a socket and connects it to the specified port on the specified host.
o **Socket**(InetAddress, int)
Creates a stream socket and connects it to the specified address on the specified port.
o **Socket**(InetAddress, int, boolean)
Creates a socket and connects it to the specified address on the specified port.

Method Index

o **close**()
Closes the socket.
o **getInetAddress**()
Gets the address to which the socket is connected.
o **getInputStream**()
Gets an InputStream for this socket.

- **getLocalPort**()
  Gets the local port to which the socket is connected.
- **getOutputStream**()
  Gets an OutputStream for this socket.
- **getPort**()
  Gets the remote port to which the socket is connected.
- **setSocketImplFactory**(SocketImplFactory)
  Sets the system's client SocketImplFactory.
- **toString**()
  Converts the Socket to a String.

Constructors

- **Socket**

```
public Socket(String host,
              int port) throws UnknownHostException, IOException
```

  Creates a stream socket and connects it to the specified port on the specified host.
  Parameters:
  host – the host
  port – the port

- **Socket**

```
public Socket(String host,
              int port,
              boolean stream) throws IOException
```

  Creates a socket and connects it to the specified port on the specified host. The last argument lets you specify whether you want a stream or datagram socket.
  Parameters:
  host – the specified host
  port – the specified port
  stream – a boolean indicating whether this is a stream or datagram socket

- **Socket**

```
public Socket(InetAddress address,
              int port) throws IOException
```

  Creates a stream socket and connects it to the specified address on the specified port.
  Parameters:
  address – the specified address
  port – the specified port

- **Socket**

```
public Socket(InetAddress address,
```

```
int port,
boolean stream) throws IOException
```

Creates a socket and connects it to the specified address on the specified port. The last argument lets you specify whether you want a stream or datagram socket.

Parameters:

address – the specified address
port – the specified port
stream – a boolean indicating whether this is a stream or datagram socket

Methods

o **getInetAddress**

```
public InetAddress getInetAddress()
```

Gets the address to which the socket is connected.

o **getPort**

```
public int getPort()
```

Gets the remote port to which the socket is connected.

o **getLocalPort**

```
public int getLocalPort()
```

Gets the local port to which the socket is connected.

o **getInputStream**

```
public InputStream getInputStream() throws IOException
```

Gets an InputStream for this socket.

o **getOutputStream**

```
public OutputStream getOutputStream() throws IOException
```

Gets an OutputStream for this socket.

o **close**

```
public synchronized void close() throws IOException
```

Closes the socket.

o **toString**

```
public String toString()
```

Converts the Socket to a String.
Overrides:
toString in class Object

o **setSocketImplFactory**

```
public static synchronized void setSocketImplFactory(SocketImplFactory fac) throws IOException
```

Sets the system's client SocketImplFactory. The factory can be specified only once.
Parameters:
fac – the desired factory
Throws: SocketException
If the factory is already defined.