

## ***java.io.InetAddress* class methods**

*boolean equals(Object obj)*

compares two IP addresses, returns true if there is a match.

*byte[ ] getAddress( )*

returns the IP address in byte format.

*static InetAddress [ ] getAllByName (String hostname) throws java.net.UnknownHostException, java.lang.SecurityException*

returns an array of InetAddress instances representing the hostname.

*static InetAddress getByName (String hostname) throws java.net.UnknownHostException, java.lang.SecurityException*

returns an InetAddress instance representing the hostname.

*String getHostAddress( )*

returns the IP address in dotted decimal format.

*static InetAddress getLocalHost ( ) throws java.net.UnknownHostException, java.lang.SecurityException*

returns the IP address of the localhost machine.

*String getHostName( ) throws java.lang.SecurityManager*

returns the hostname of the InetAddress.

*boolean isMulticastAddress( )*

returns true if the InetAddress is a multicast address (class D address).

### **EXAMPLE 1:**

```
import java.net.*;
public class SampleInetAddress {
    public static void main(String [] args){
        try {    InetAddress local = InetAddress.getLocalHost( );
                System.out.println("Name: " + local.getHostName( ) +
                    "\t IP address: " + local.getHostAddress( ) );
            }
        catch ( UnknownHostException e) {
            System.out.println("Error: " + e);
        }
    } //end of main
} //end of class definition
```

## EXAMPLE 2:

```
import java.net.*;
public class DNSLookup {

    public static void main(String [] args){
        if (args.length != 1){
            System.out.println("Usage: DNSLookup |<name> |<address>");
            System.exit(1);
        }

        String findAddress = args[0];
        boolean isReverseLookup =
            Character.isDigit(findAddress.charAt(0));
        try {
            InetAddress [] allAddress =
                InetAddress.getAllByName(findAddress);

            for (int k=0; k<allAddress.length; k++){

                InetAddress address = allAddress[k];
                if (isReverseLookup)
                    System.out.println("Address: " +
                        address.getHostAddress( ) +
                        "\t resolves to: " +
                        address.getHostName( ) );
                else
                    System.out.println("Hostname: " +
                        address.getHostName( ) +
                        "\t resolves to: " +
                        address.getHostAddress( ) );
            } //end of for
        }
        catch ( UnknownHostException e) {

            System.out.println("Error: " + e);
            System.exit(1);
        }
        catch (java.lang.SecurityException se){

            System.out.println("Socket permission:" + se);
            System.exit(1);
        }
    } //end of main
} //end of class definition
```