Network Programming

IP Address in java

IP Address

- An **IP(Internet Protocol) address** is an identifier assigned to each computer and another device (e.g., router, mobile, etc) connected to a TCP/IP network that is used to locate and identify the node in communication with other nodes on the network.

IP Address

- An IP address serves two principal functions: host or network interface identification and local addressing.
- Its role has been characterized as follows: “A **name** indicates **what we seek**. An **address** indicates **where it is**. A **route** indicates **how to get there**.”

IP Address

- An IP address is an address having information about how to reach a specific host which is a 32-bit unique address number having an address space of 2^32.

IP Address

- Every device has an IP address with two pieces:
  1. The client or host address
  2. And the server or network address.

IP Address

- The subnet mask splits the IP address into the host and network addresses, thereby defining which part of the IP address belongs to the device and which part belongs to the network.

IPv4 Addressing

- **IP addresses**
  - Identify each individual machine on the internet
  - 32 bits in length.
  - Four 8-bit decimal values between 0 and 255 separated by periods (octets).
    - octet set.
  - network part/prefix (high order bits)
  - host part (low order bits)

- **Subnet Mask**
  - 32-bit value of 0’s and 1’s
  - 1’s designate network bits, 0’s are host bits

IPv4 Addressing

Network Host

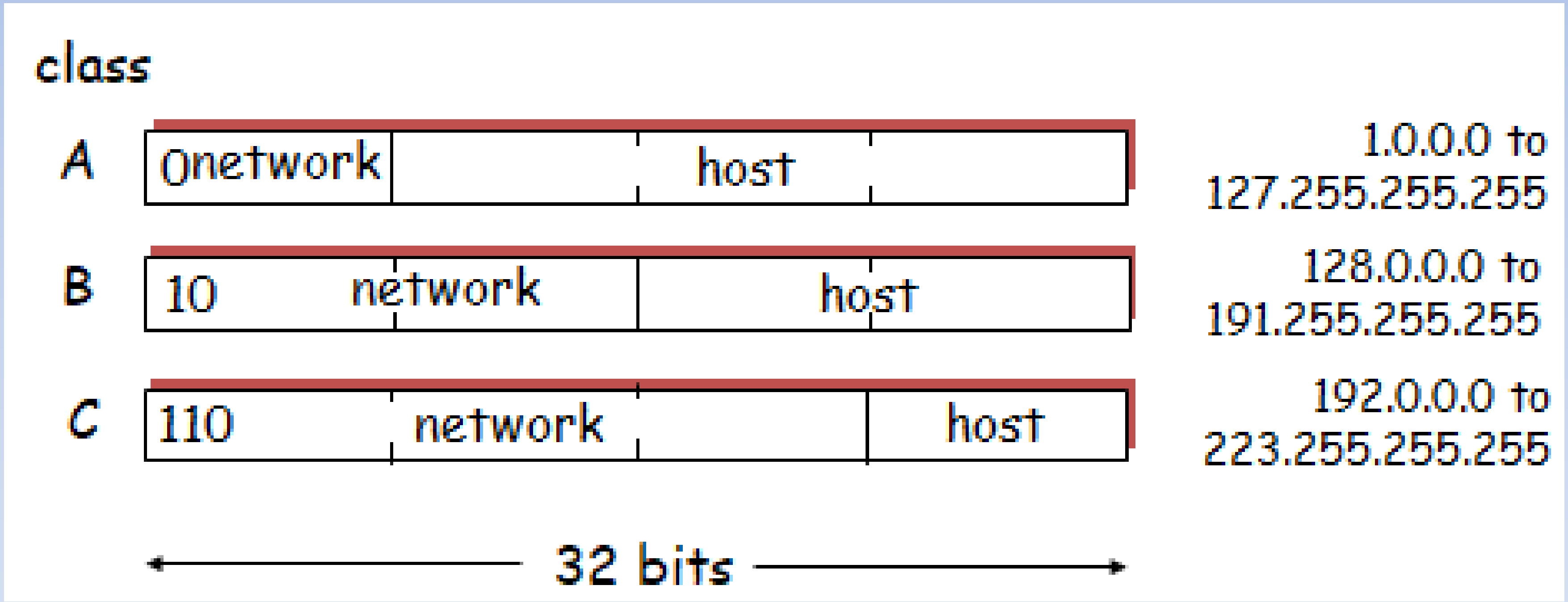
Examples: IP Address 192.168.43.100

Subnet Mask 255.255.255.0

IPv4 Addressing

- IP addresses have 4 billion combinations
  - Not really
  - Allocated in groups called address blocks
  - 3 sizes, based on the class of the address
  - Class A, Class B, and Class C

IPv4 Class-full Addressing



class

A Onetwork host 1.0.0.0 to 127.255.255.255

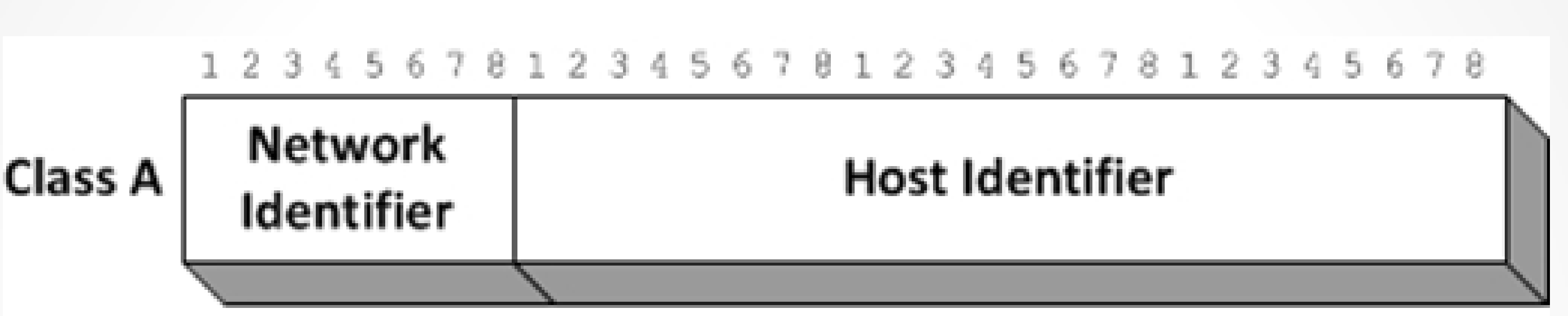
B 10 network host 128.0.0.0 to 191.255.255.255

C 110 network host 192.0.0.0 to 223.255.255.255

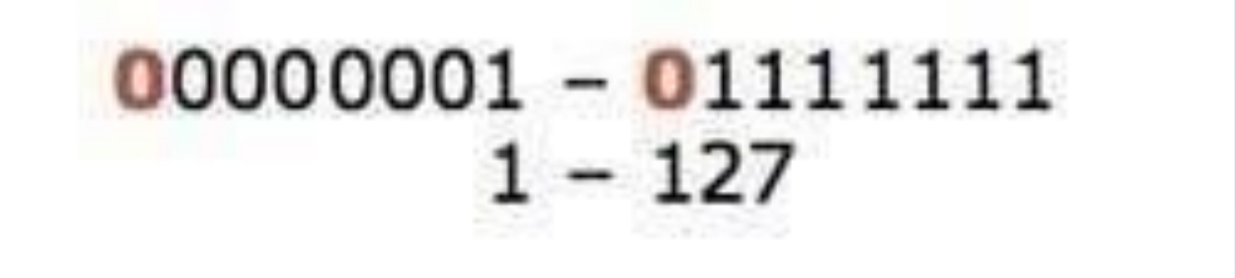
32 bits

Class A

- The first octet set represents the network and the rest octet sets represents the host.



- the first octet ranges from 1 – 127



00000001 – 01111111

1 – 127

Class A

- Number of network identifier bits : 8 (first octet set)
- Number of host identifier bits: 24 (rest octet sets)
- Number of possible networks: 126 (2^7-2)
  - 127.x.x.x is reserved for loopback IP addresses.
- Number of possible host 2^{24}-2
  - x.0.0.0 (network address).
  - x.255.255.255 (Broadcast address).
- Subnet Mask **255**.0.0.0

Class A example

- **IP address : 90.36.5.11**

prefix   Network Host

90.36.5.11/8 (class A) → 90.36.5.11

Network address: 90.0.0.0

Broadcast address: 90.255.255.255

Number of possible host 2^{24}-2

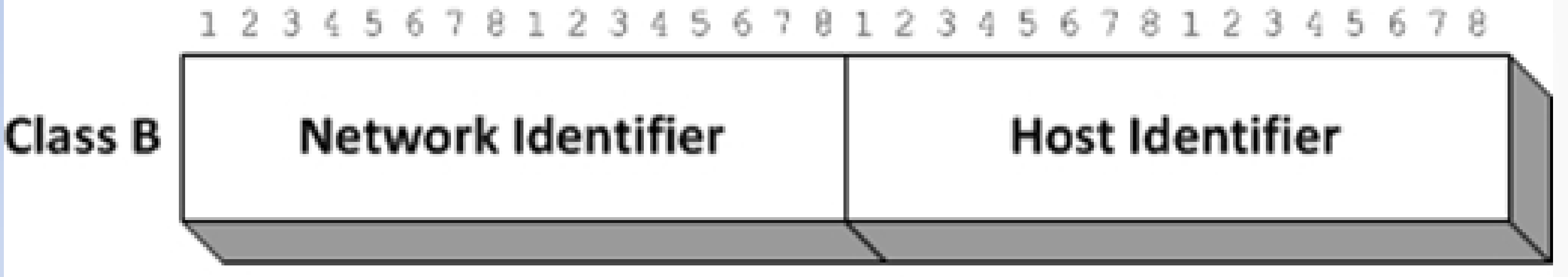
Number of possible networks: 126 (2^{7}-2)

Subnet Mask: **255.0.0.0**

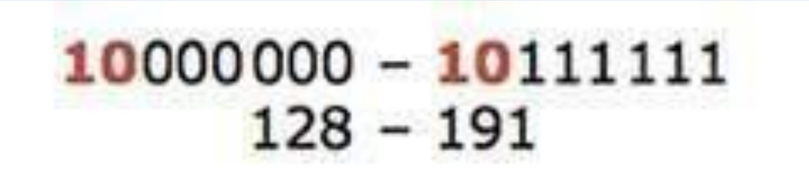
```
          90.36.5.11
And
          255.0.0.0
_______________________
           90.0.0.0
```

Class B

- The first two octet sets represents the network and the rest octet sets represents the host.



- The first octet ranges from 128 – 191



10000000 – 10111111
128 – 191

Class B

- Number of network identifier bits : 16 (first two octet set)
- Number of host identifier bits: 16 (rest two octet sets)

$63 \approx 2^6$

- Number of possible networks: 2^{14} ((191-128)*2^8)
- Number of possible host $2^{16}-2$
  - x.x.0.0 (network address).
  - x.x.255.255 (Broadcast address).
- Subnet Mask 255.255.0.0

Class B example

- **IP address : 150.36.5.11**

prefix

150.36.5.11/16 (class B) → 150.36.5.11

Network Host

Network address: 150.36.0.0

Broadcast address: 150.36.255.255

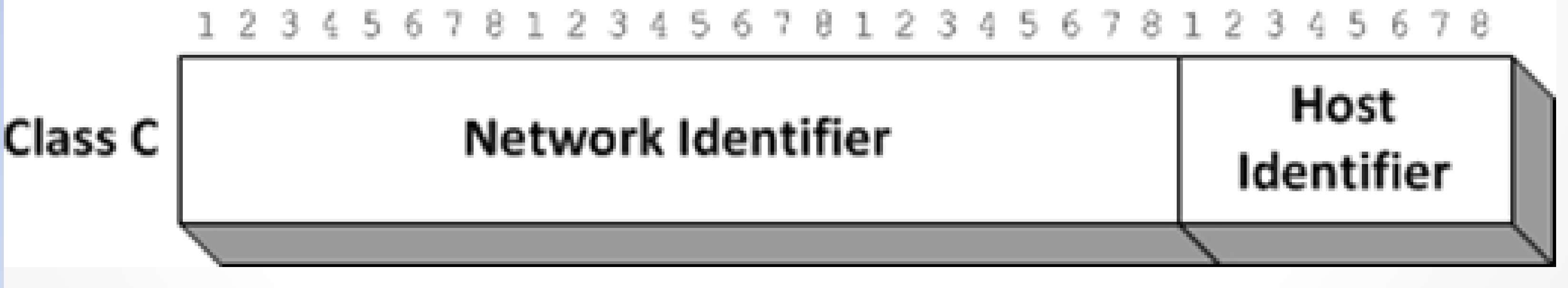
Number of possible host 2^{16}-2

Number of possible networks: 2^{14}

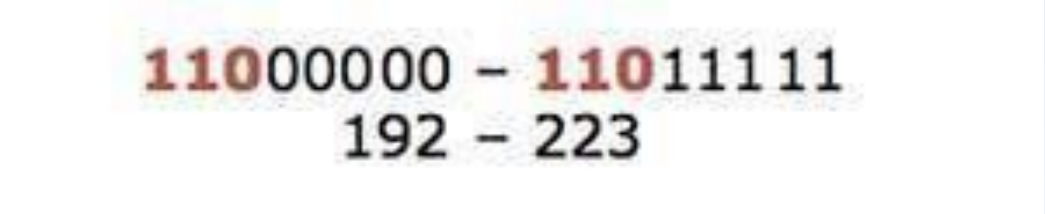
Subnet Mask: **255.255.0.0**

Class C

- The first three octet sets represents the network and the last octet sets represents the host.



- The first octet ranges from 192 – 223



11000000 – 11011111
192 – 223

Class C

- Number of network identifier bits : 24(first 3 octet sets)
- Number of host identifier bits: 8 (last octet sets)
- Number of possible networks: 2^{21} ((223-192)*2^{16})

  31 ≈ 2^5
- Number of possible host 2^8-2 (254)
  - x.x.x.0 (network address).
  - x.x.x.255 (Broadcast address).
- Subnet Mask 255.255.255.0

May Zakarneh

Class C example

- **IP address : 192.36.5.11**

prefix Network Host

192.36.5.11/24 (class C) → 192.36.5.11

Network address: 192.36.5.0

Broadcast address: 192.36.5.255

Number of possible host 254 (2^8-2)

Number of possible networks: 2^{21}

Subnet Mask: 255.255.255.0

Reserved Addresses

- Private Networks (no public connections)
  - 10.x.x.x
  - 172.16.x.x
  - 192.168.x.x
- 127.x.x.x – local network (loopback)
- 255.255.255.255 – broadcast – sends to everyone on the network

Domain Name System (DNS)

- The Domain Name System (DNS) turns domain names into IP addresses, which browsers use to load internet pages.
- Every device connected to the internet has its own IP address, which is used by other devices to locate the device.

Domain Name System (DNS)

- DNS servers make it possible for people to input normal words into their browsers, such as Fortinet.com, without having to keep track of the IP address for every website.

Domain Name System (DNS)

- A DNS server is a computer with a database containing the public IP addresses associated with the names of the websites an IP address brings a user to.

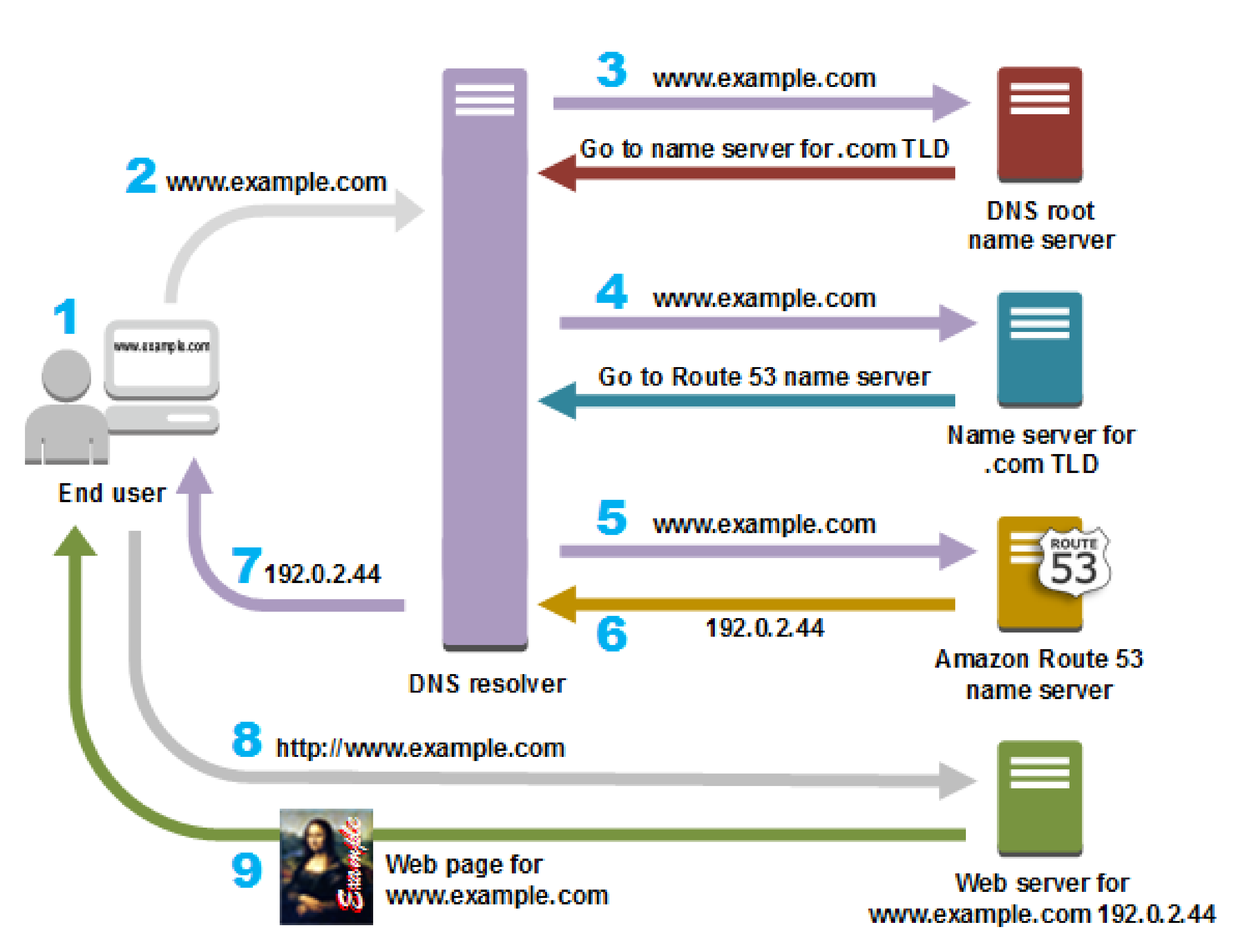
Domain Name System (DNS)

- DNS acts like a phonebook for the internet.
  - Whenever people type domain names, like Fortinet.com or Yahoo.com, into the address bar of web browsers, the DNS finds the right IP address.
  - The site's IP address is what directs the device to go to the correct place to access the site's data.

Domain Name System (DNS)

- Once the DNS server finds the correct IP address, browsers take the address and use it to send data to content delivery network (CDN) edge servers or origin servers.
- Once this is done, the information on the website can be accessed by the user.
- The DNS server starts the process by finding the corresponding IP address for a website's uniform resource locator (URL).

How Does DNS Work?



InetAddress class

- InetAddress class is a representation of an IP address.
- It represents both the 32-bit IPv4 address and the 128-bit IPv6 address.
- It is the superclass of Inet6Address and Inet4Address classes.

InetAddress class

- An instance of this class consists of an IP address and usually a hostname depending on whether hostname resolution was performed during the creation.

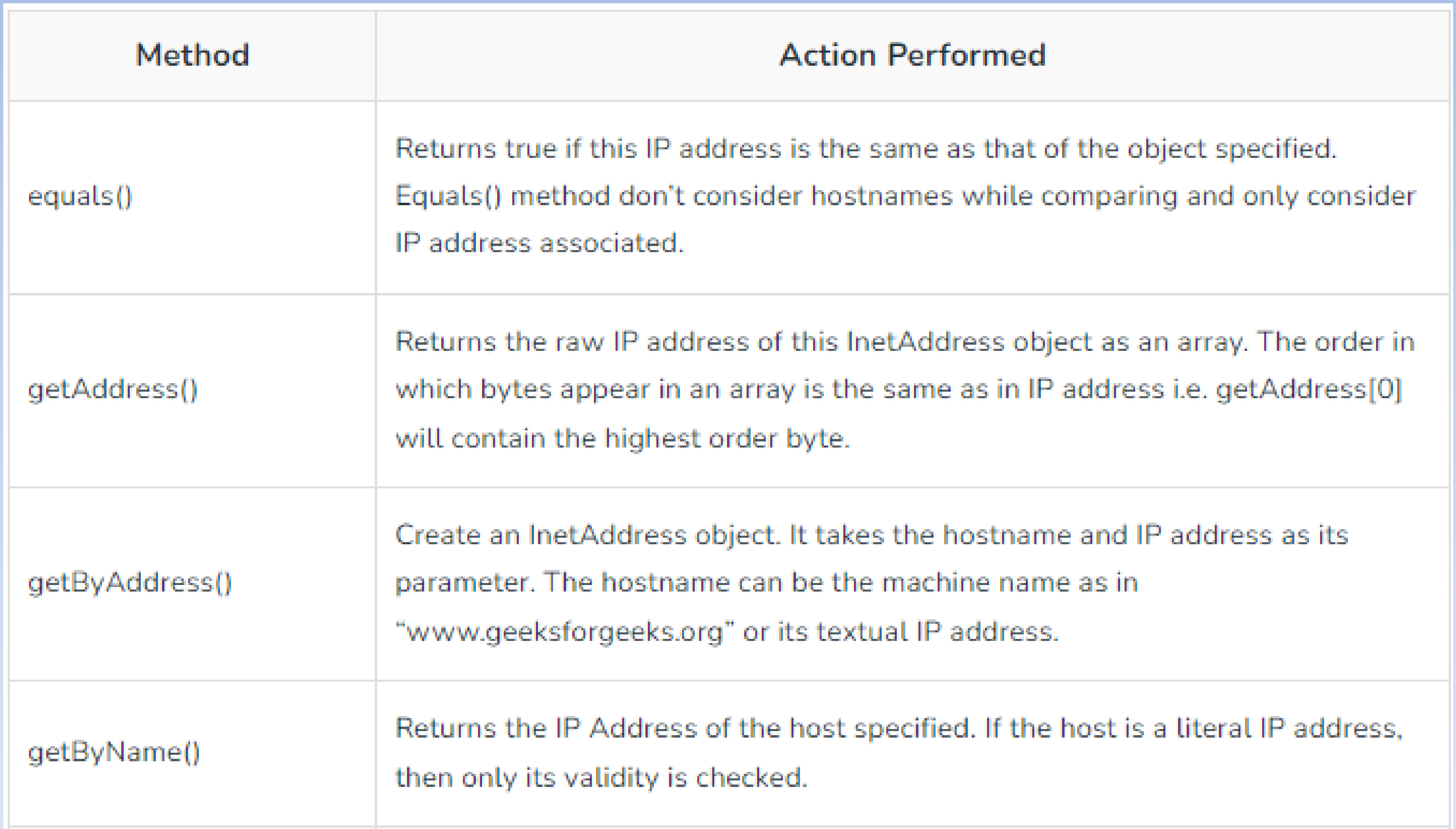
InetAddress class

- ***There are no constructors for this class but static methods which return instances of InetAddress class for general use.***

InetAddress class

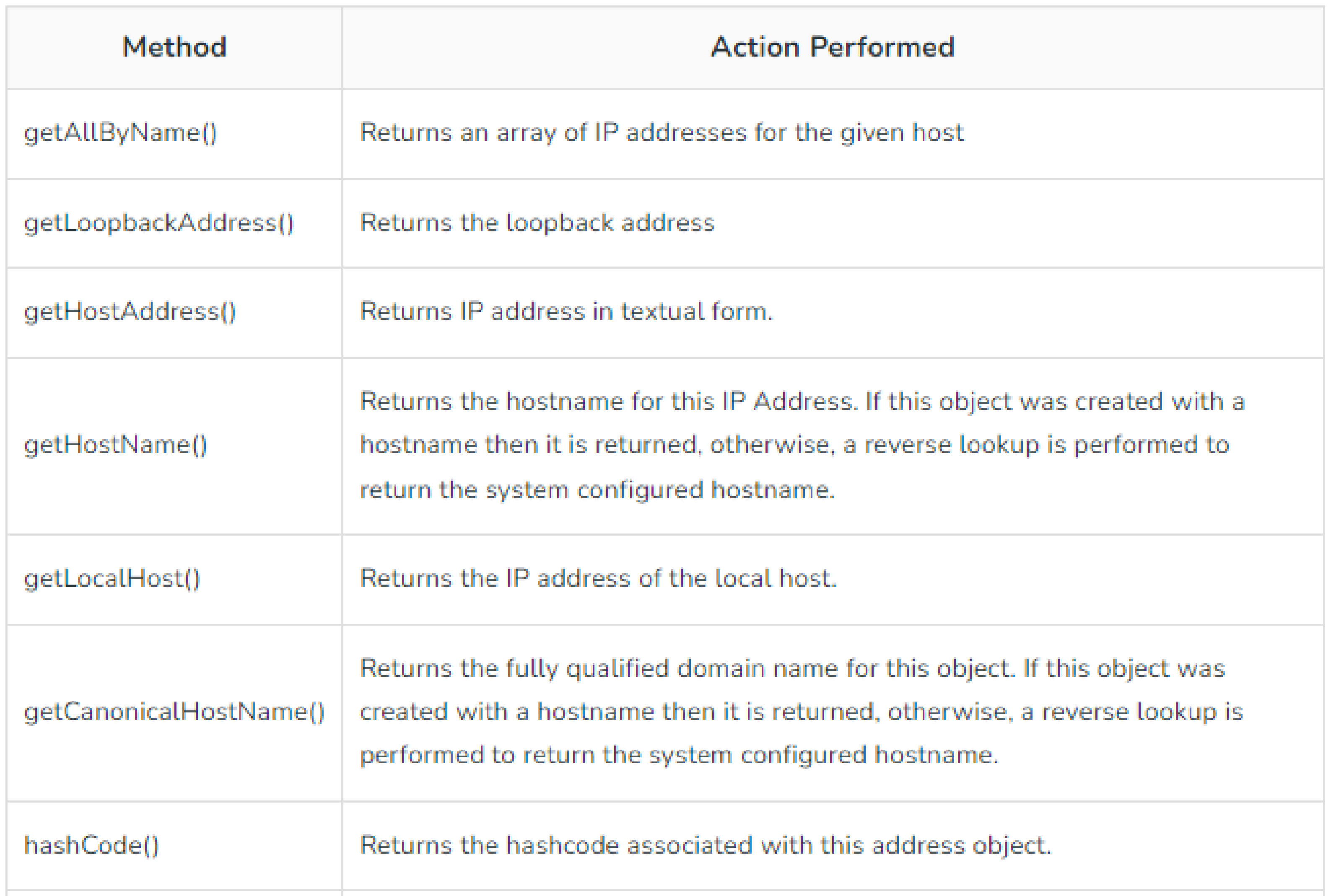
- **The are no constructors but “class factory”**
  - InetAddress n = InetAddress.getLocalHost()
  - InetAddress n = InetAddress.getByName(nombre)
- **The most important methods:**
  - String nombre = n.getHostName()
  - String direccion = n.getHostAddress()

Methods of InetAddress Class



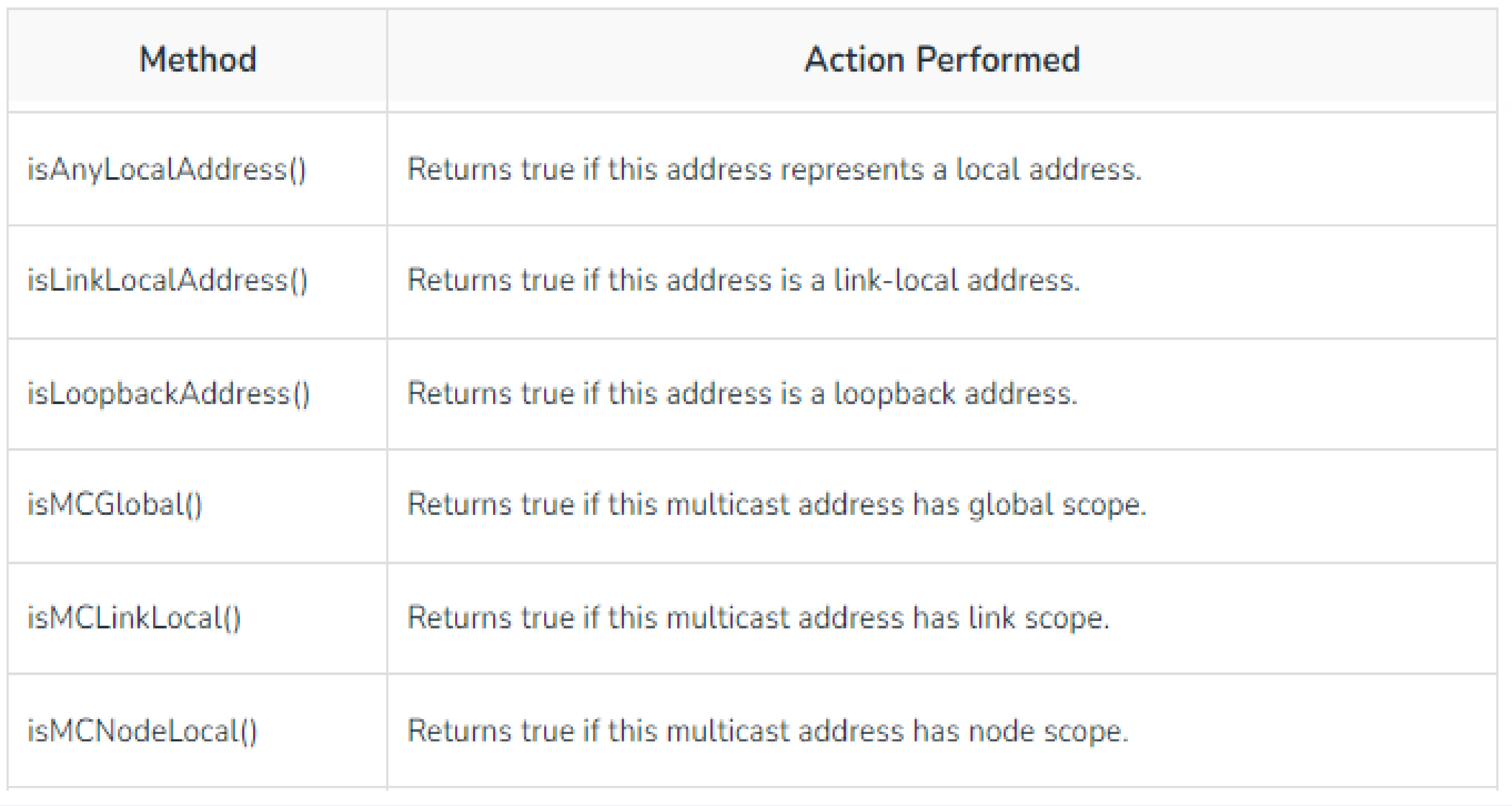
Method	Action Performed
equals()	Returns true if this IP address is the same as that of the object specified. Equals() method don't consider hostnames while comparing and only consider IP address associated.
getAddress()	Returns the raw IP address of this InetAddress object as an array. The order in which bytes appear in an array is the same as in IP address i.e. getAddress[0] will contain the highest order byte.
getByAddress()	Create an InetAddress object. It takes the hostname and IP address as its parameter. The hostname can be the machine name as in "www.geeksforgeeks.org" or its textual IP address.
getByName()	Returns the IP Address of the host specified. If the host is a literal IP address, then only its validity is checked.

Methods of InetAddress Class



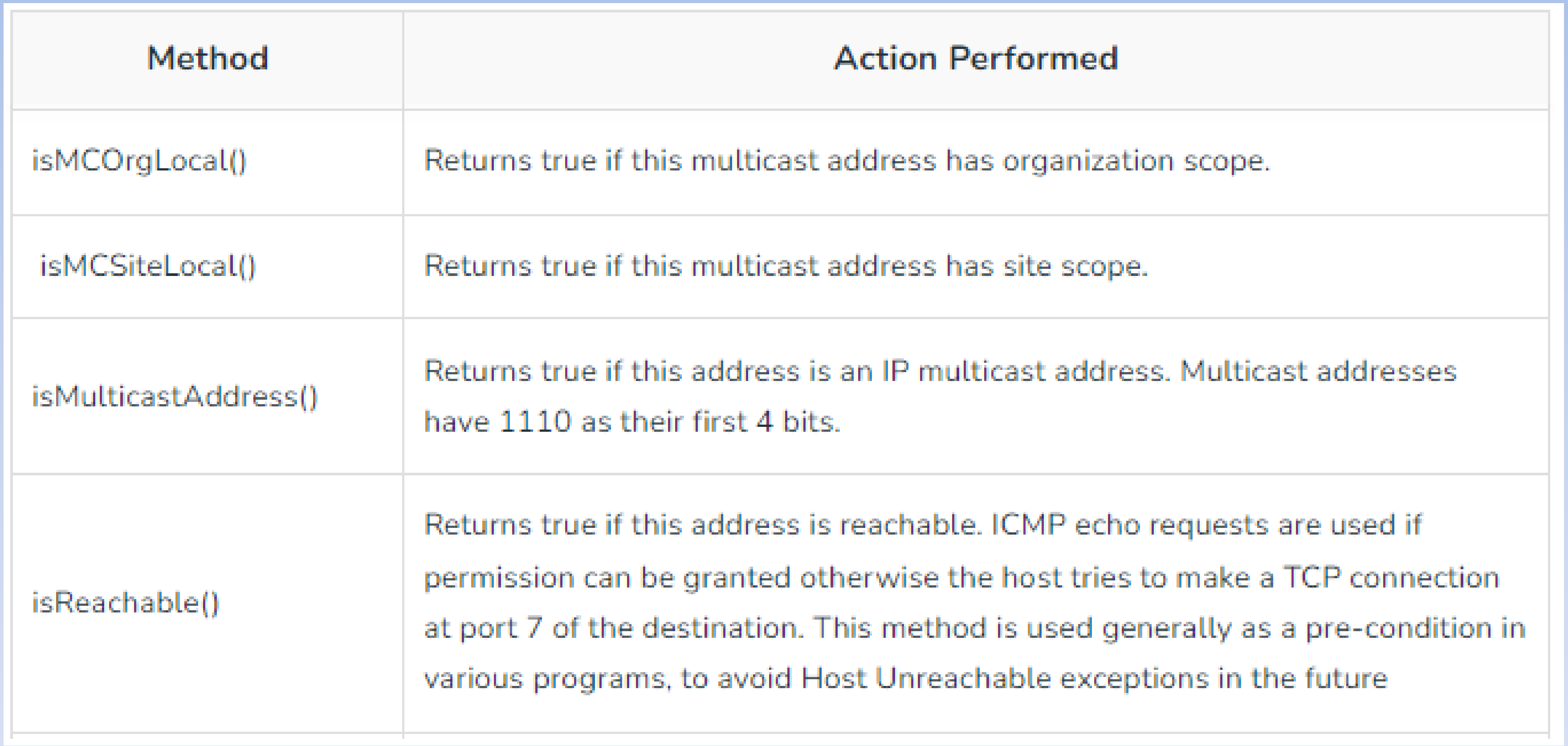
Method	Action Performed
getAllByName()	Returns an array of IP addresses for the given host
getLoopbackAddress()	Returns the loopback address
getHostAddress()	Returns IP address in textual form.
getHostName()	Returns the hostname for this IP Address. If this object was created with a hostname then it is returned, otherwise, a reverse lookup is performed to return the system configured hostname.
getLocalHost()	Returns the IP address of the local host.
getCanonicalHostName()	Returns the fully qualified domain name for this object. If this object was created with a hostname then it is returned, otherwise, a reverse lookup is performed to return the system configured hostname.
hashCode()	Returns the hashcode associated with this address object.

Methods of InetAddress Class



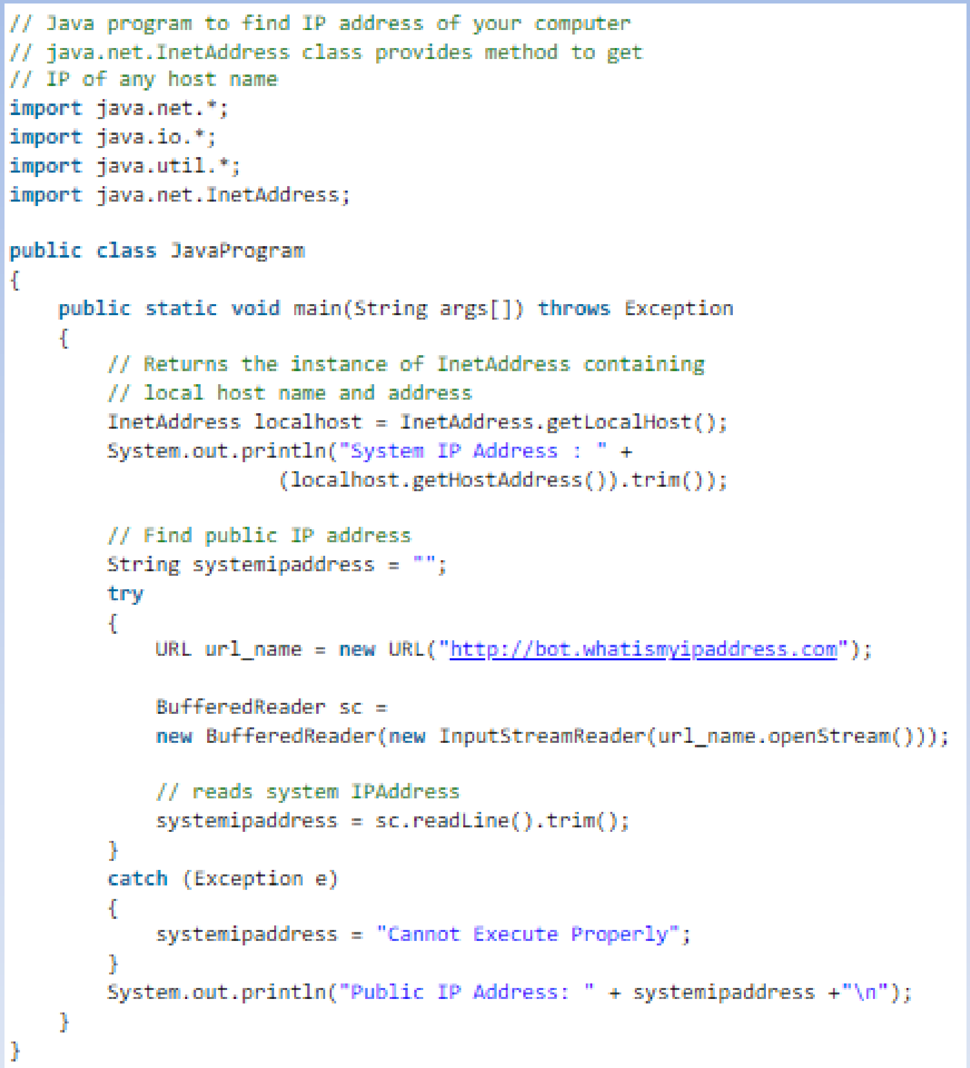
Method	Action Performed
isAnyLocalAddress()	Returns true if this address represents a local address.
isLinkLocalAddress()	Returns true if this address is a link-local address.
isLoopbackAddress()	Returns true if this address is a loopback address.
isMCGlobal()	Returns true if this multicast address has global scope.
isMCLinkLocal()	Returns true if this multicast address has link scope.
isMCNodeLocal()	Returns true if this multicast address has node scope.

Methods of InetAddress Class



Method	Action Performed
isMCOrgLocal()	Returns true if this multicast address has organization scope.
isMCSiteLocal()	Returns true if this multicast address has site scope.
isMulticastAddress()	Returns true if this address is an IP multicast address. Multicast addresses have 1110 as their first 4 bits.
isReachable()	Returns true if this address is reachable. ICMP echo requests are used if permission can be granted otherwise the host tries to make a TCP connection at port 7 of the destination. This method is used generally as a pre-condition in various programs, to avoid Host Unreachable exceptions in the future

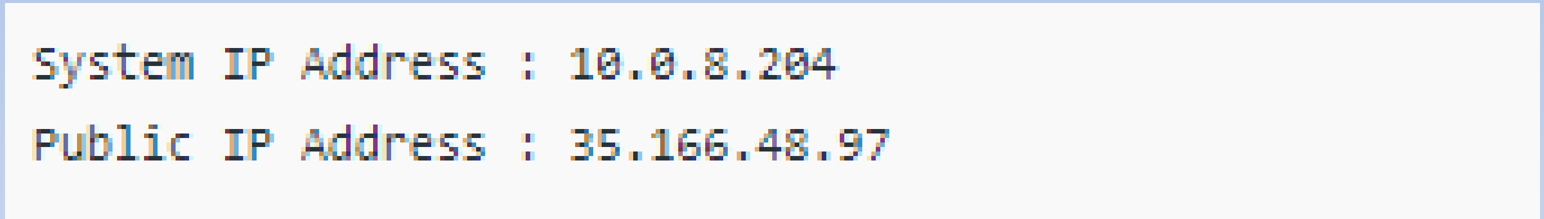
Methods of InetAddress Class



```
// Java program to find IP address of your computer
// java.net.InetAddress class provides method to get
// IP of any host name
import java.net.*;
import java.io.*;
import java.util.*;
import java.net.InetAddress;

public class JavaProgram
{
    public static void main(String args[]) throws Exception
    {
        // Returns the instance of InetAddress containing
        // local host name and address
        InetAddress localhost = InetAddress.getLocalHost();
        System.out.println("System IP Address : " +
                      (localhost.getHostAddress()).trim());

        // Find public IP address
        String systemipaddress = "";
        try
        {
            URL url_name = new URL("http://bot.whatismyipaddress.com");

            BufferedReader sc =
            new BufferedReader(new InputStreamReader(url_name.openStream()));

            // reads system IPAddress
            systemipaddress = sc.readLine().trim();
        }
        catch (Exception e)
        {
            systemipaddress = "Cannot Execute Properly";
        }
        System.out.println("Public IP Address: " + systemipaddress +"\n");
    }
}
```

Methods of InetAddress Class

- Output:



```
System IP Address : 10.0.8.204
Public IP Address : 35.166.48.97
```