**Networking in Java**

Java is a premier language for network programming. **java.net** package encapsulate large number of classes and interface that provides an easy-to use means to access network resources. Here are some important classes and interfaces of java.net package.

**Some Important Classes**

|  |  |
| --- | --- |
| **CLASSES** | |
| CacheRequest | CookieHandler |
| CookieManager | Datagrampacket |
| Inet Address | ServerSocket |
| Socket | DatagramSocket |
| Proxy | URL |
| URLConnection |  |

**Some Important Interfaces**

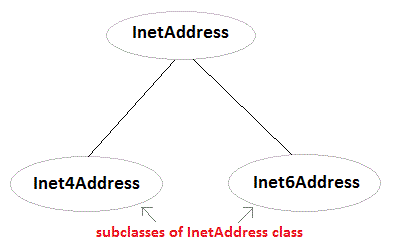
|  |  |
| --- | --- |
| **INTERFACES** | |
| CookiePolicy | CookieStore |
| FileNameMap | SocketOption |
| InetAddress | ServerSocket |
| SocketImplFactory | ProtocolFamily |

**InetAddress**

Inet Address encapsulates both numerical IP address and the domain name for that address. Inet address can handle both IPv4 and Ipv6 addresses. Inet Address class has no visible constructor. To create an inet Address object, you have to use **Factory methods**.

Three commonly used Inet Address factory methods are.

1. static *InetAddress* **getLocalHost()** throws **UnknownHostException**
2. static *InetAddress* **getByName** (*String hostname*) throws **UnknownHostException**
3. static *InetAddress[ ]* **getAllByName** (*String hostname*) throws **UnknownHostException**



**Example using InetAddress class**

import java.net.\*;

class Test

{

public static void main(String[] args)

{

**InetAddress address = InetAddress.getLocalHost()**;

System.out.println(address);

address = **InetAddress.getByName**("www.netparam.com");

System.out.println(address);

InetAddress sw[] = **InetAddress.getAllByName**("www.netparam.com");

for(int i=0; i< sw.length; i++)

{

System.out.println(sw[i]);

}

}

}

**Output:**

Welcome-PC/59.161.87.227

www.netparam.com/74.125.236.115

www.netparam.com/74.125.236.116

www.netparam.com/74.125.236.112

www.netparam.com/74.125.236.113

www.netparam.com/74.125.236.114

www.netparam.com/2404:6800:4009:802:0:0:0:1014