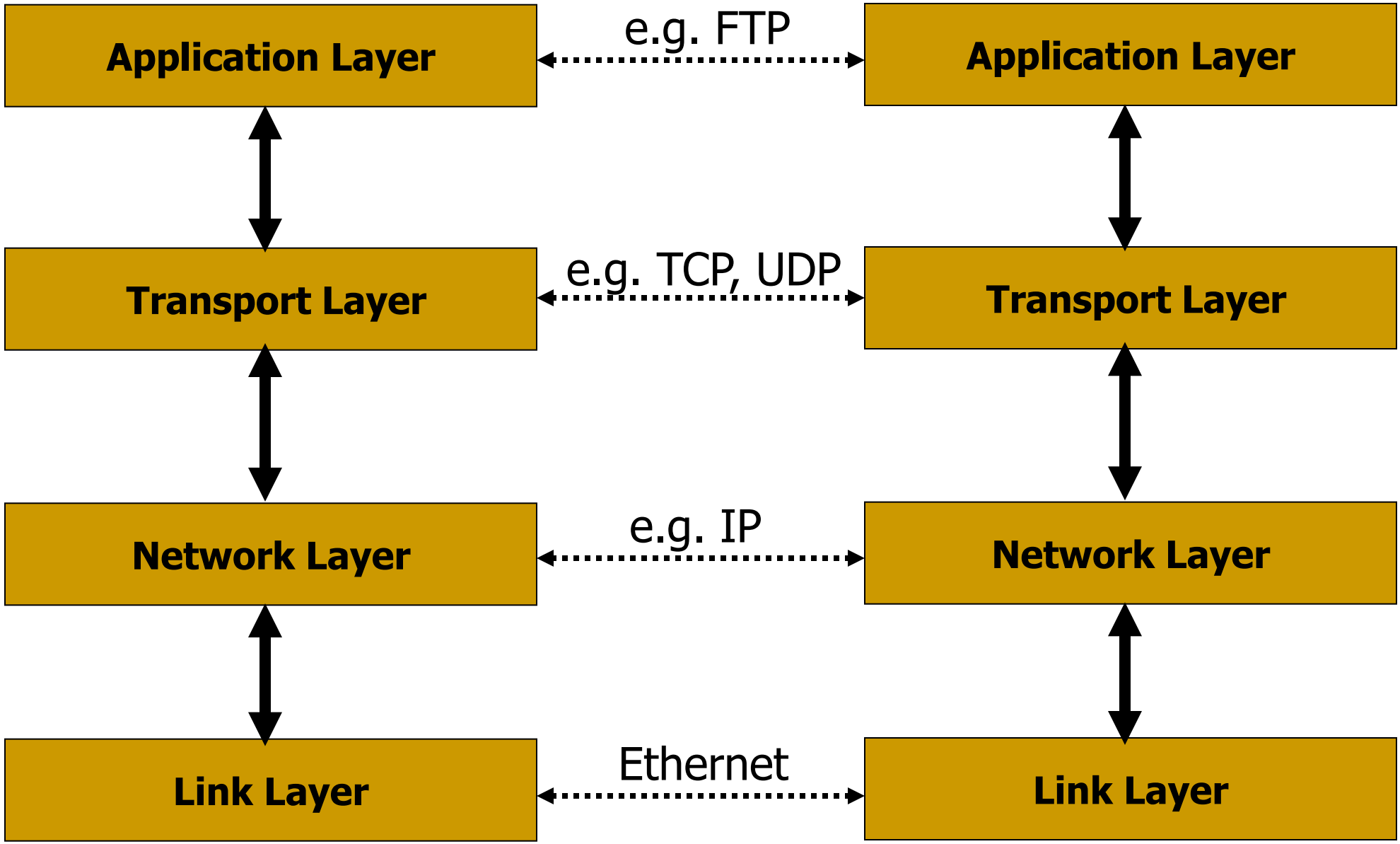
NETWORK PROGRAMMING IN JAVA

4th Aug 2020

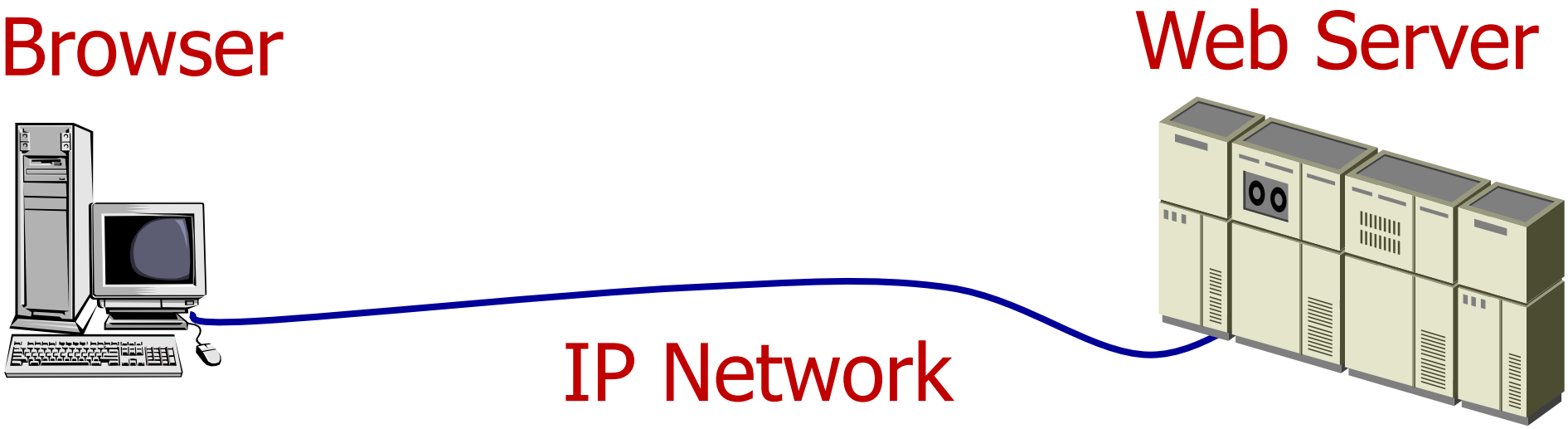
Centre for Development of Advanced Computing
(C-DAC), Hyderabad

Layers of the TCP/IP Protocol Suite



Network Applications

- Accessing Web (http, https)



Similarly FTP, Telnet SMTP etc

Types of Internet Sockets

- Different types of sockets implement different communication types (stream vs. datagram)
- Type of socket: stream socket
  - connection-oriented
  - two way communication
  - reliable (error free), in order delivery
  - can use the Transmission Control Protocol (TCP)
  - e.g. telnet, ssh, http, ftp etc
- Type of socket: datagram socket
  - connectionless, does not maintain an open connection, each packet is independent
  - can use the User Datagram Protocol (UDP)
  - e.g. IP telephony, RTP, NTP etc
- Other types exist: similar to the one above

Naming and Addressing

- Host name
  - identifies a single host
  - variable length string (e.g. www.berkeley.edu)
  - is mapped to one or more IP addresses
- IP Address
  - written as dotted octets (e.g. 10.0.0.1)
  - 32 bits. Not a number! But often needs to be converted to a 32-bit to use.
- Port number
  - identifies a process on a host
  - 16 bit number

About TCP and UDP Ports

- TCP and UDP Ports are 16-bit numbers.

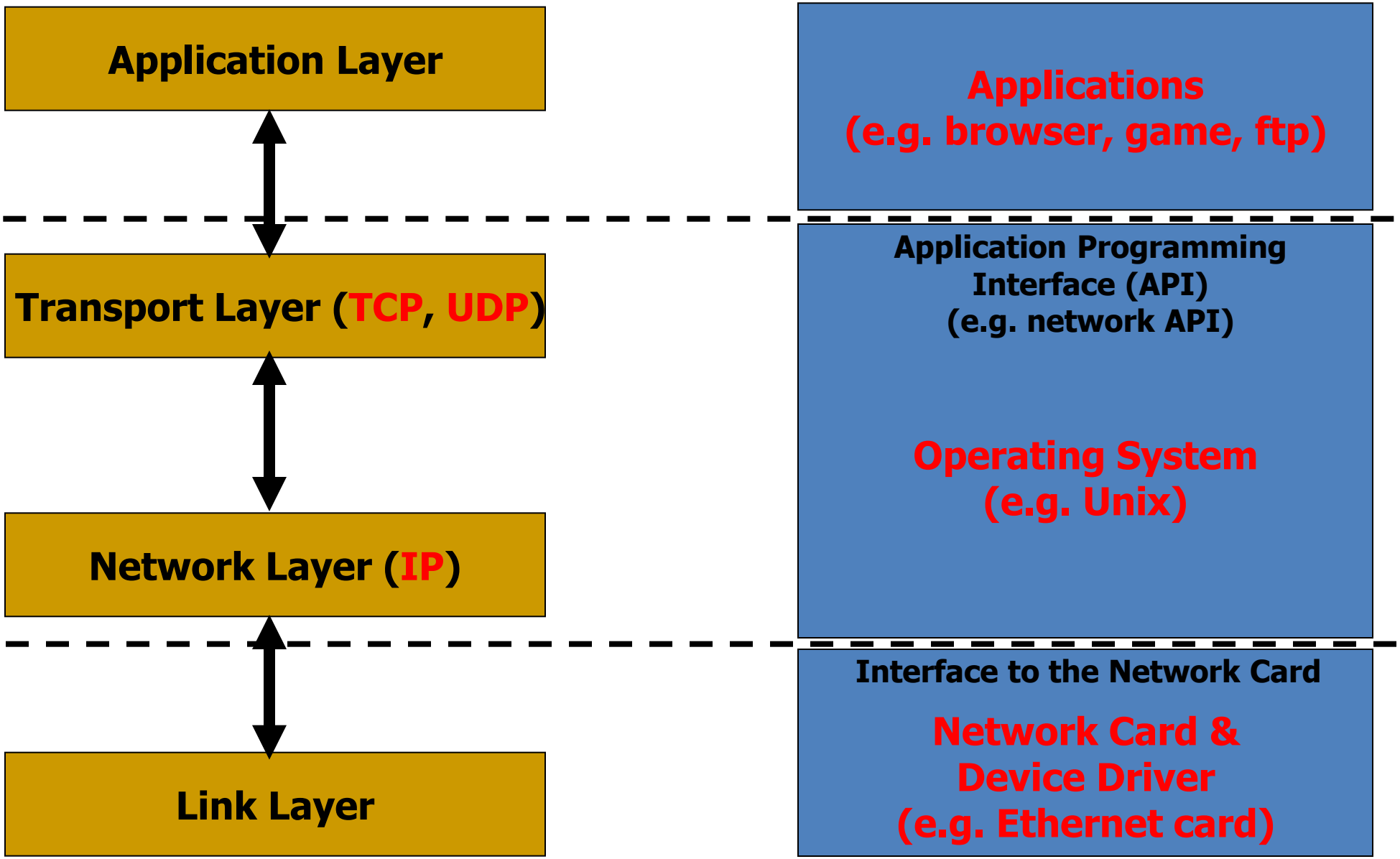
- They are of three types:
  - Well-known Ports (0-1023: Controlled by the IANA)
  - Registered Ports (1024-49159) and
  - Ephemeral / Dynamic Ports (49152-65535). (RFC 1700 shows a list suggested initially)

- FTP over TCP uses 21 whereas TFTP over UDP uses 69 for instance.

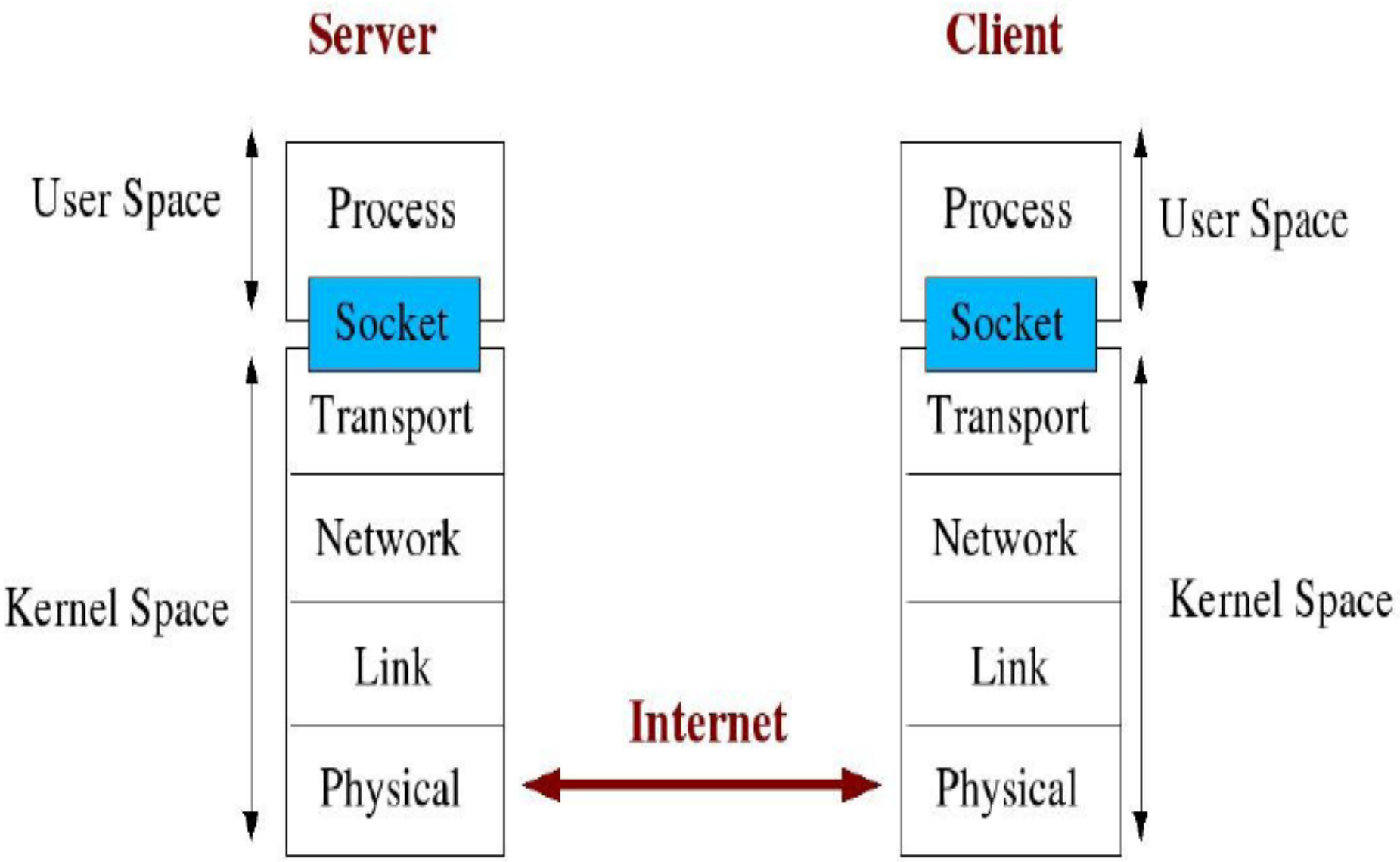
PROTOCOL SUITE LOCATION

- Internet Protocol Layer

Location



Socket Description

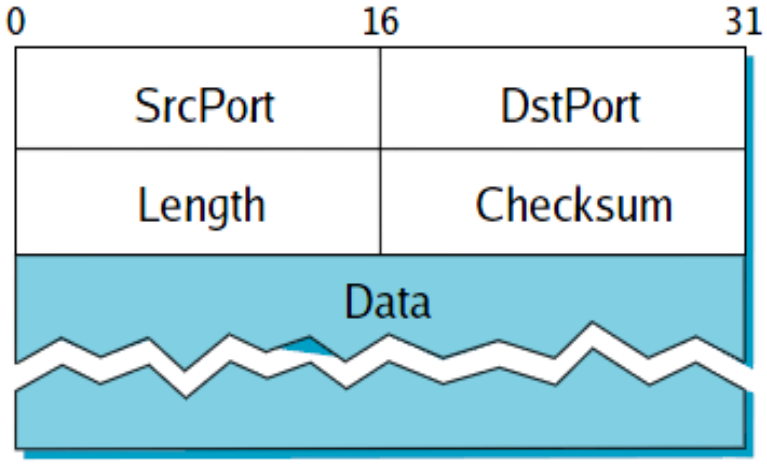


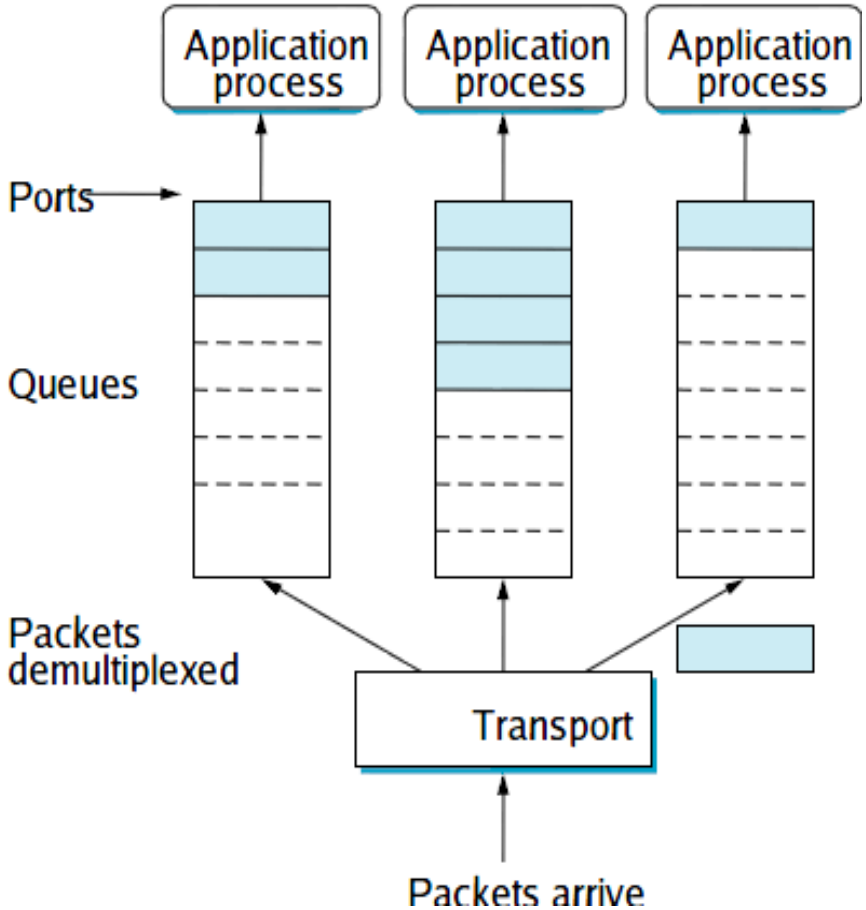
What is a Socket?

- Socket is a resource to access network services
- Socket APIs are used by Server and Client applications to communicate over the network
- Internet sockets characterized by IP Address (4 bytes) & port number (2 bytes)

Demultiplexing

- Convert host-to-host packet delivery service into a process-to-process communication channel





Client - Server Paradigm

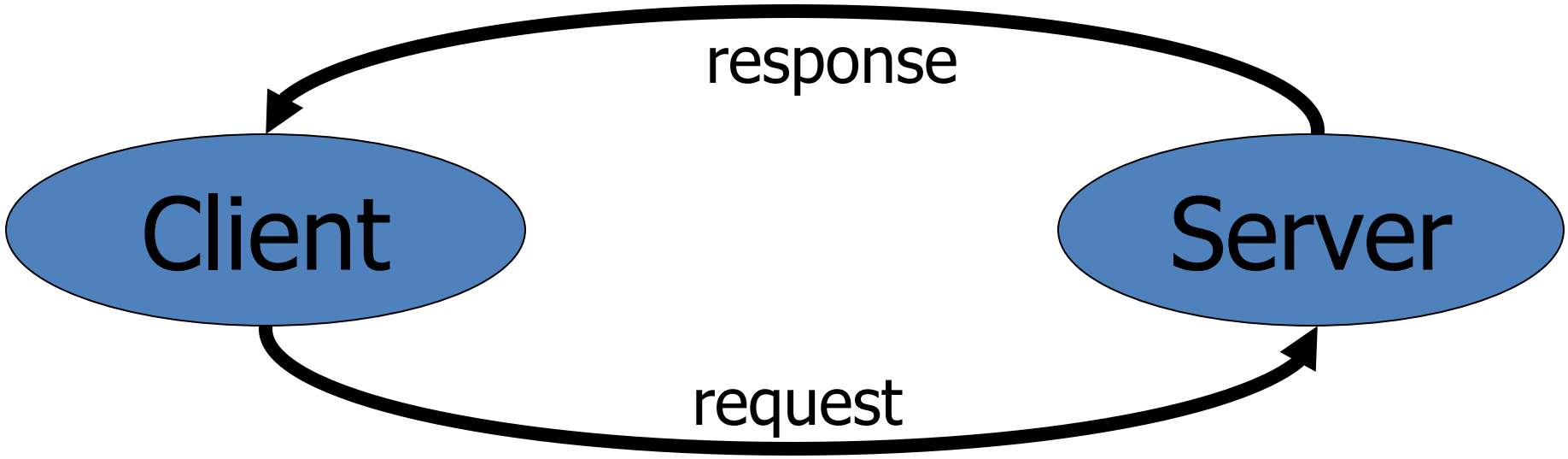
Application Server Component:

- Provides any set of predefined services to one or more requesting clients

Application Client Component:

- Requests for any specific service from any designated server

Client-Server Architecture



- Client requests service to server
- Server responds with sending service or error message to client

SOCKET PROGRAMMING WITH TCP

Client must contact server

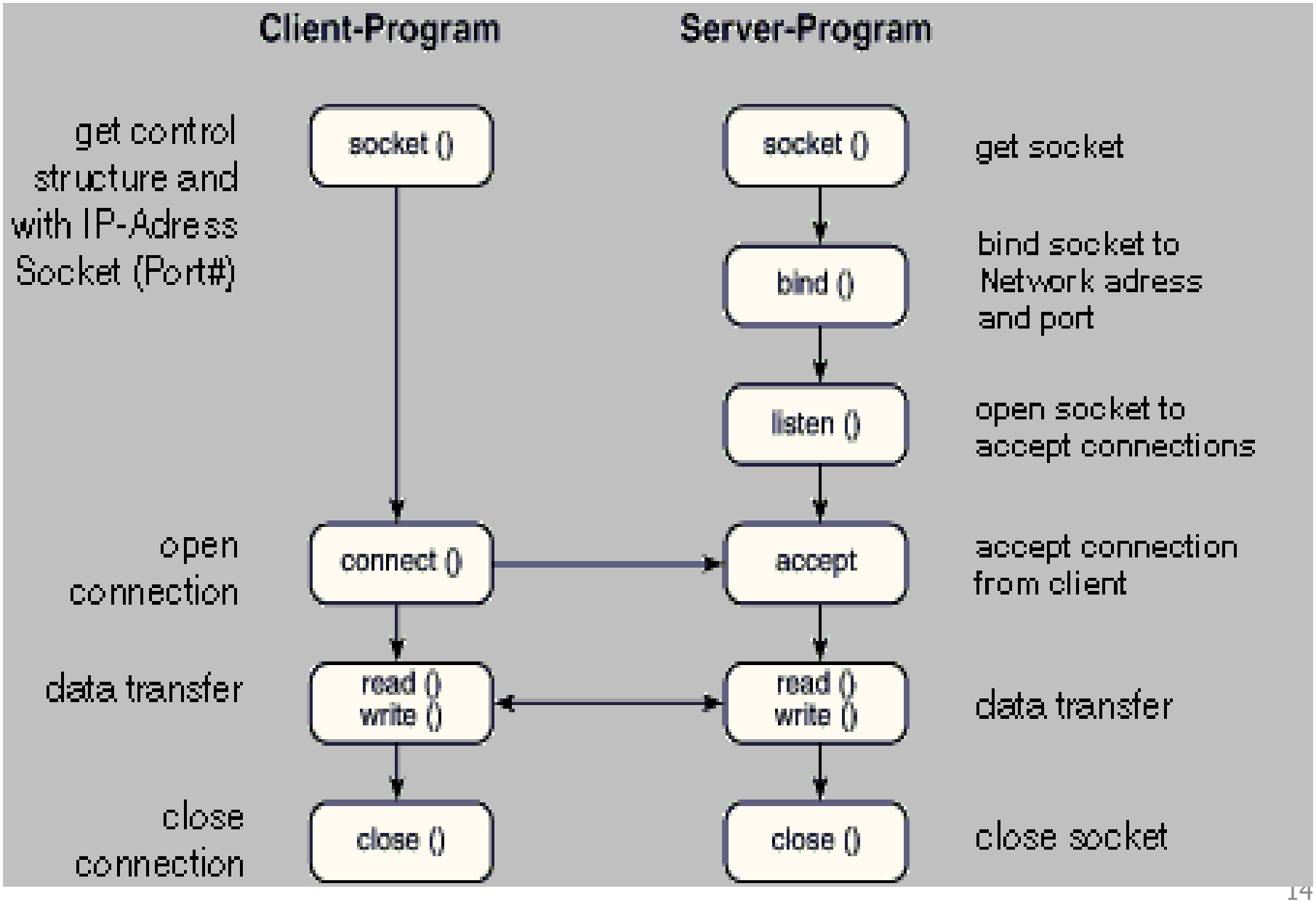
- server process must first be running
- server must have created socket (door) that welcomes client's contact

Client contacts server by:

- creating client-local TCP socket
- specifying IP address, port number of server process
- When client creates socket: client TCP establishes connection to server TCP

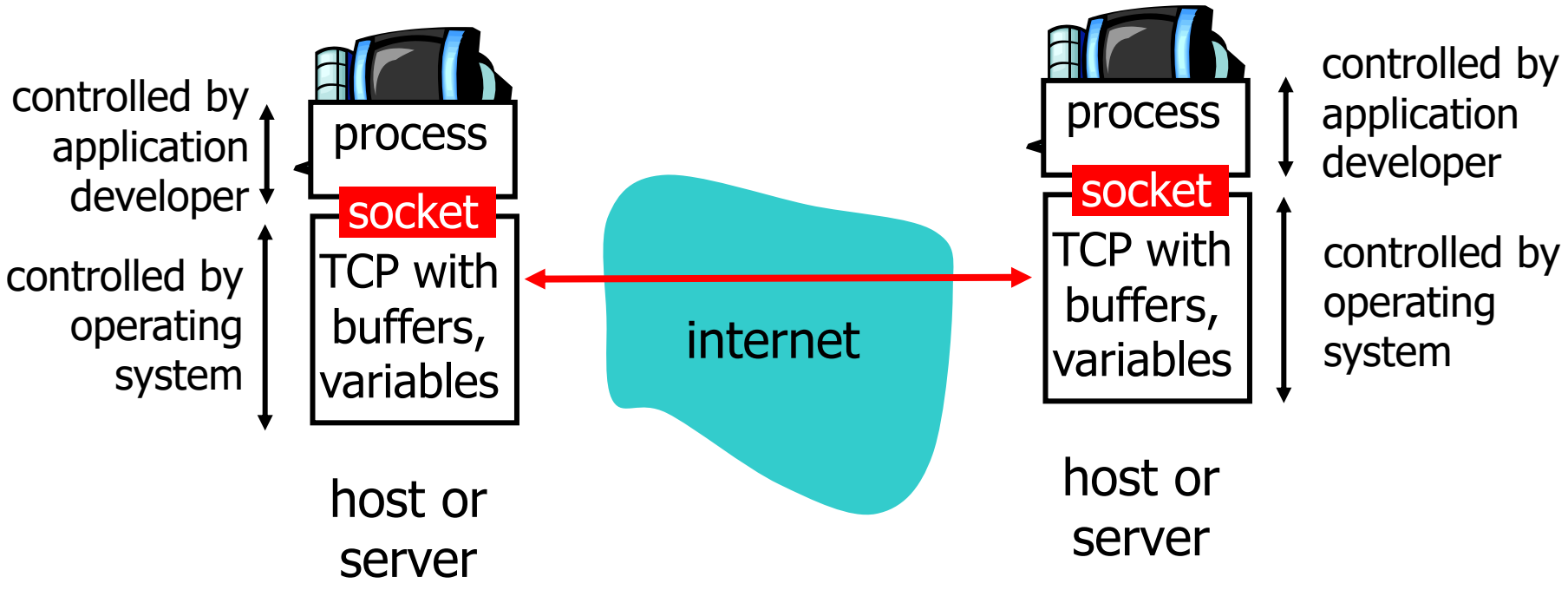
- When contacted by client, server TCP creates new socket for server process to communicate with client
  - allows server to talk with multiple clients
  - source port numbers used to distinguish clients

TCP based Application



SOCKET-PROGRAMMING USING TCP

TCP service: reliable transfer of **bytes** from one process to another



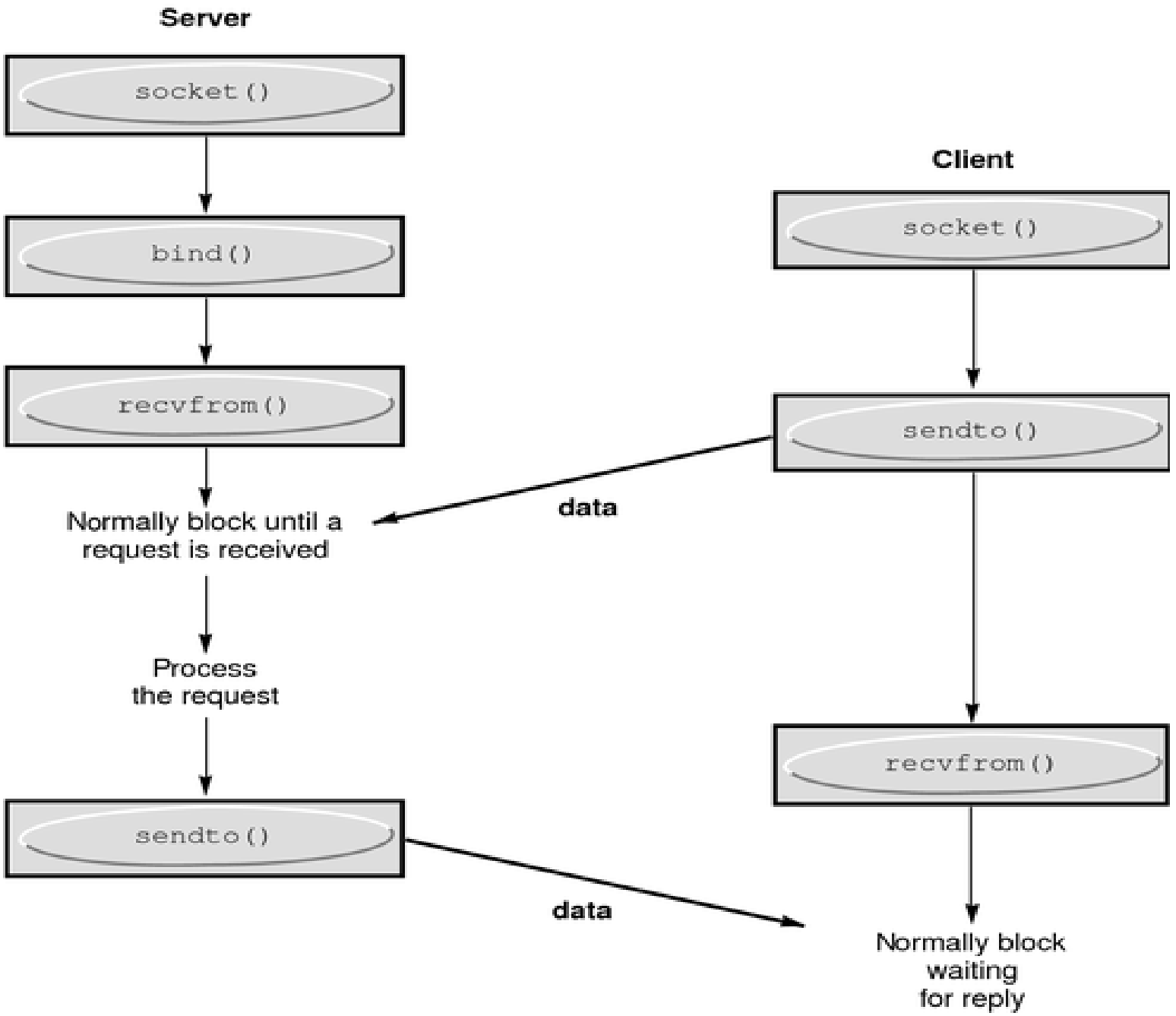
SOCKET PROGRAMMING WITH UDP

UDP: no “connection” between client and server

- no handshaking
- sender explicitly attaches IP address and port of destination to each packet

UDP: transmitted data may be received out of order, or lost

UDP based Application



Types of Application Servers

Concurrent Server

» It simultaneously provides any set of predefined services to one or more requesting clients is called a Concurrent Server Process.

Iterative Server

» It provides any set of predefined services to only one requesting client at any point of time is called an Iterative Server Process.

Network API

- Operating system provides Application Programming Interface (API) for network application
- API is defined by a set of function types, data structures, and constants
- Desirable characteristics of the network interface
  - Simple to use
  - Flexible
    - independent from any application
    - allows program to use all functionality of the network
  - Standardized
    - allows programmer to learn once, write anywhere
- Application Programming Interface for networks is called socket

Class ServerSocket

ServerSocket

- The ServerSocket class can be used to create a server socket. This object is used to establish communication with the clients.

Methods

- public Socket accept() - returns the socket and establish a connection between server and client.

- public synchronized void close() - closes the server socket.

Class Socket

Socket(String host, int port) or (InetAddress addr, int port)

- A socket is simply an endpoint for communications between the machines. The Socket class can be used to create a socket.

Methods

- InputStream getInputStream() – returns the InputStream attached with this socket.
- OutputStream getOutputStream() – returns the OutputStream attached with this socket.
- InetAddress getInetAddress()
- InetAddress getLocalAddress()
- int getPort()
- int getLocalPort()
- void close() – closes this socket

TCP Socket - Server Side

- Create server socket (*ServerSocket()*)
  - Bind port to socket and make it a listen for connections
- While (Condition)
  - accept user connection request which returns new socket which is used to have session with the client (*accept()*)
  - Get Input and Output streams for the session socket
  - Communicate with the Client
- Close TCP connection and Socket (close())

TCP Socket - Client Side

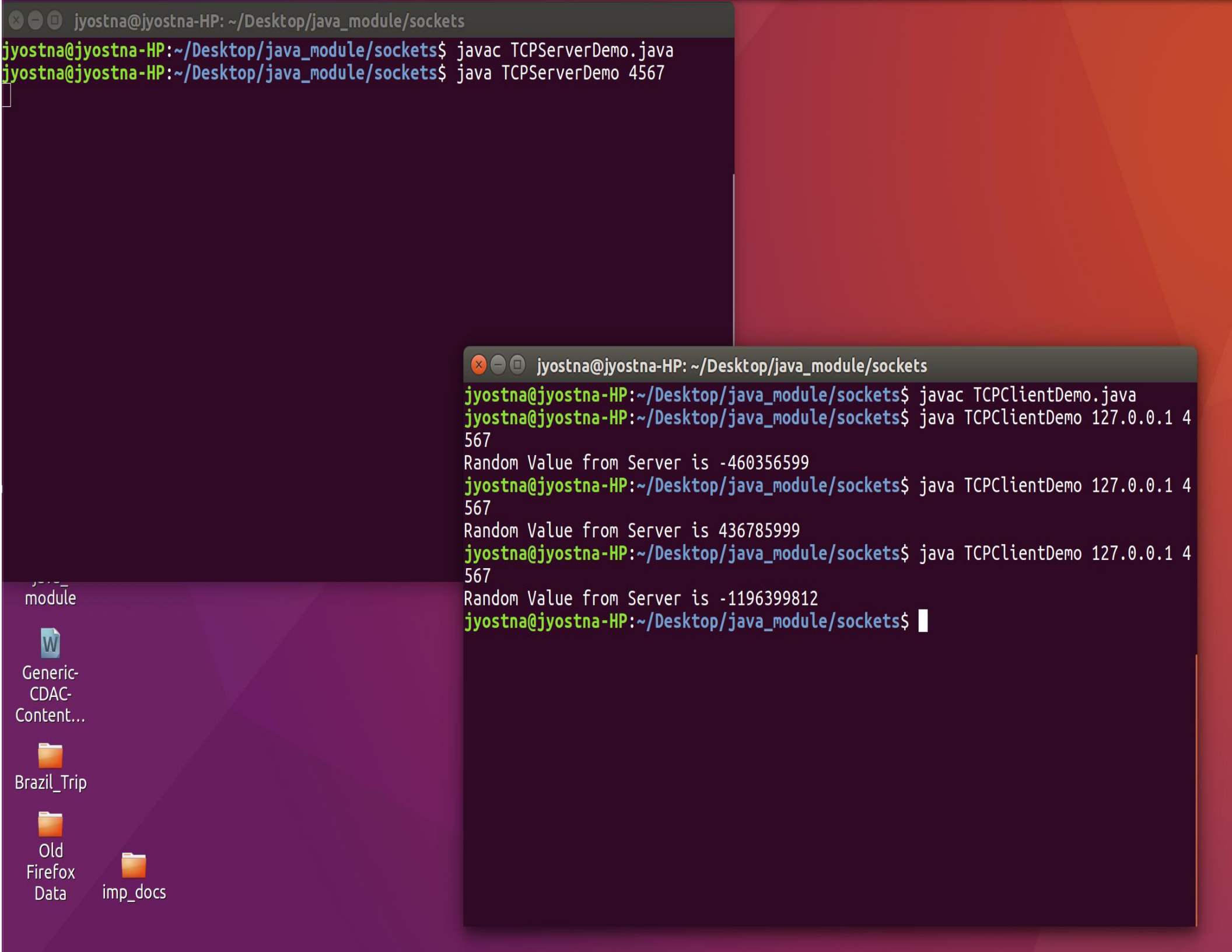
- Create stream socket (*socket()*) and connect to relevant server – IP & Port No
- Get Input and Output streams for communicating over socket
- Send and Receive data from the server
- Close TCP connection and Socket (close())

Example - TCPClientDemo

```
import java.io.*;
import java.util.*;import
java.net.*;
class TCPClientDemo {
public static void main(String
args[])
{try
{
String server=args[0];
int port=Integer.parseInt(args[1]);
Socket s=new Socket(server,port);
InputStream
is=s.getInputStream();
DataInputStream dis=new
DataInputStream(is);
int i=dis.readInt();
System.out.println("Random
Value from Server is "+i);
s.close();
}
catch(Exception e) {
System.out.println(e);
}
}
}
```

Example - TCPServerDemo

```
import java.io.*;
import java.net.*;
import java.util.*;
class TCPServerDemo
{
public static void main(String
args[])
{
Try {
int port
=Integer.parseInt(args[0]);
Random random=new
Random();
ServerSocket ss=new
ServerSocket(port);
while(true)
{
Socket s=ss.accept();
OutputStream
os=s.getOutputStream();
DataOutputStream dos=new
DataOutputStream(os);
dos.writeInt(random.nextInt());
s.close();
}}
catch(Exception e)
{
System.out.println("Exception: "+e);
}}}
```


jyostna@jyostna-HP: ~/Desktop/java_module/sockets
jyostna@jyostna-HP:~/Desktop/java_module/sockets$ javac TCPServerDemo.java
jyostna@jyostna-HP:~/Desktop/java_module/sockets$ java TCPServerDemo 4567
jyostna@jyostna-HP: ~/Desktop/java_module/sockets
jyostna@jyostna-HP:~/Desktop/java_module/sockets$ javac TCPClientDemo.java
jyostna@jyostna-HP:~/Desktop/java_module/sockets$ java TCPClientDemo 127.0.0.1 4
567
Random Value from Server is -460356599
jyostna@jyostna-HP:~/Desktop/java_module/sockets$ java TCPClientDemo 127.0.0.1 4
567
Random Value from Server is 436785999
jyostna@jyostna-HP:~/Desktop/java_module/sockets$ java TCPClientDemo 127.0.0.1 4
567
Random Value from Server is -1196399812
jyostna@jyostna-HP:~/Desktop/java_module/sockets$
module
Generic-CDAC-Content...
Brazil_Trip
Old Firefox Data
imp_docs

Class DatagramSocket

DatagramSocket(int port)

- Create DatagramSocket object with port number

Methods

- void receive(DatagramPacket packet) –
- send(DatagramPacket packet) –
- void close() –

Class DatagramPacket

DatagramPacket

- A class that represents a datagram packet containing packet data, packet length, internet addresses and port

Methods

- **DatagramPacket**(byte[], int) - This constructor is used to create a DatagramPacket object used for receiving datagrams

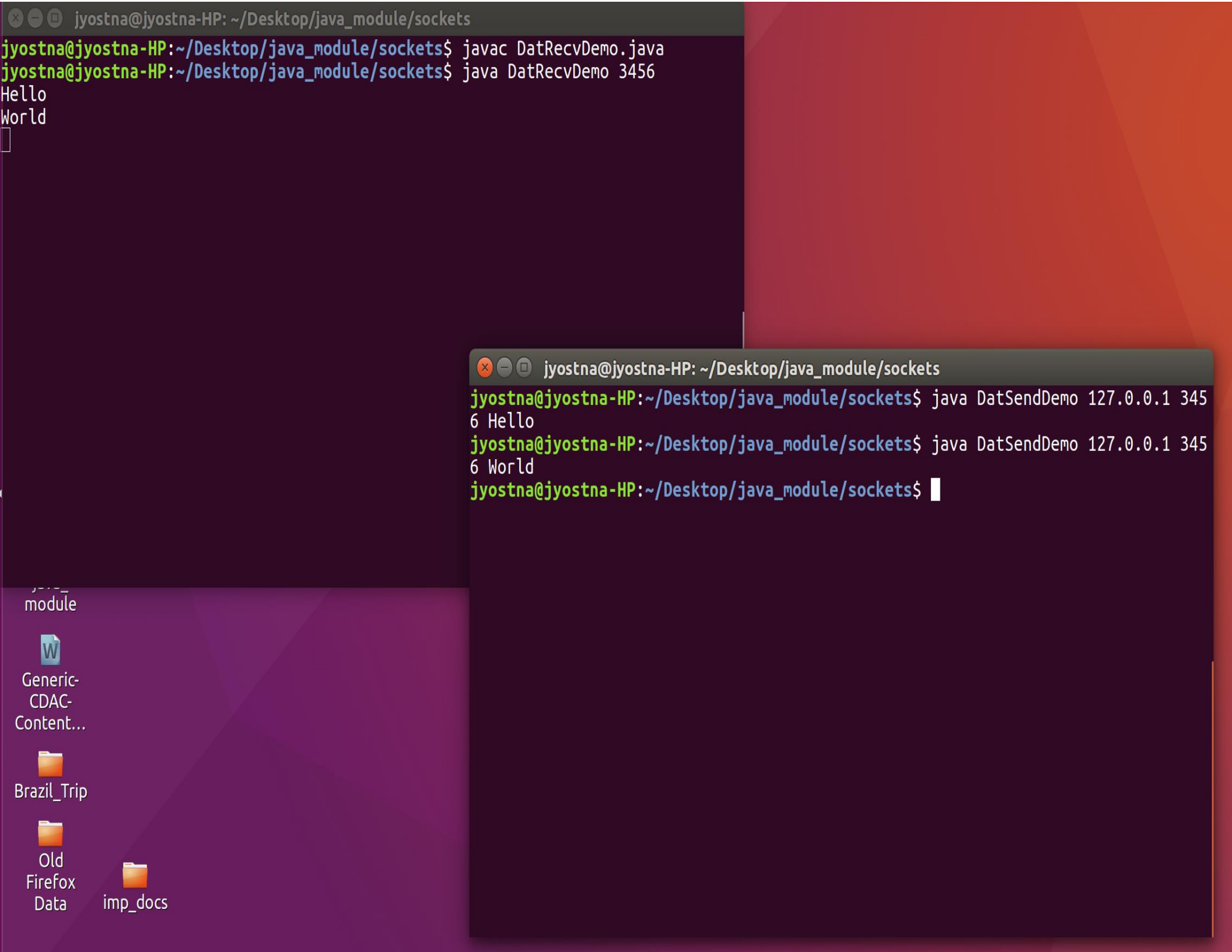
- **DatagramPacket**(byte[], int, InetAddress, int) - This constructor is used construct the DatagramPacket to be sent

UDP Socket - DatSendDemo

```
import java.io.*;
import java.util.*;
import java.net.*;
class DatSendDemo {
public static void main(String
args[]){
try{
InetAddress
ia=InetAddress.getByName(ar
gs[0]);
int
port=Integer.parseInt(args[1]);
DatagramSocket ds=new
DatagramSocket();
byte
buffer[]=args[2].getBytes();
DatagramPacket dp=new
DatagramPacket(buffer,buffer.
length,ia,port);
ds.send(dp);
}
catch(Exception e)
{
e.printStackTrace();
}
}
}
```

UDP Socket - DatRecvDemo

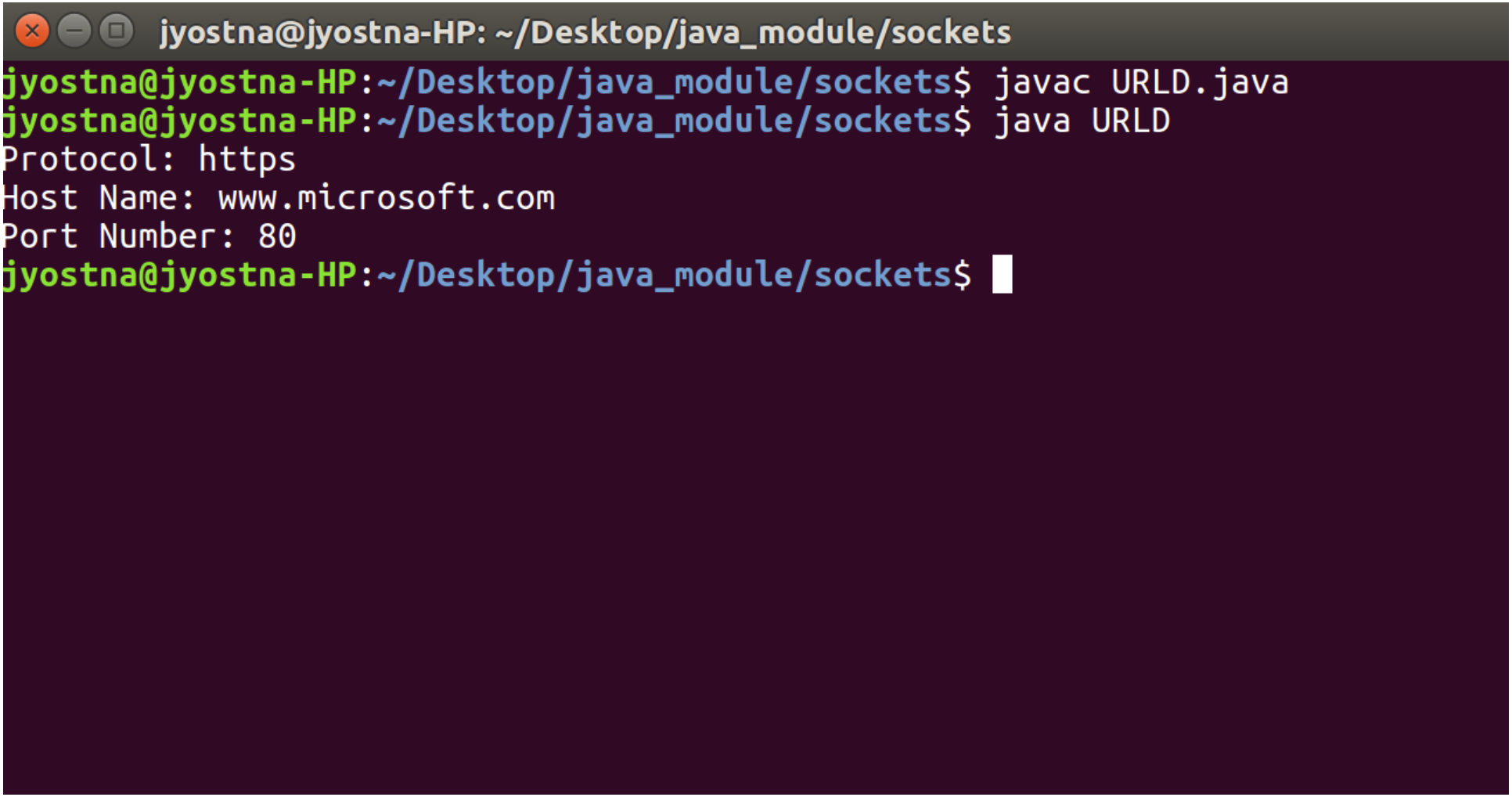
```
import java.io.*;
import java.util.*;
import java.net.*;
class DatRecvDemo {
private final static int BUFSIZE=20;
public static void main(String args[])
{
Try {
int port =Integer.parseInt(args[0]);
DatagramSocket ds=new DatagramSocket(port);
byte buffer[]=new byte[BUFSIZE];
while(true)
{
DatagramPacket dp=new DatagramPacket(buffer,buffer.length);
ds.receive(dp);
String str=new String(dp.getData());
System.out.println(str);
} }
catch(Exception e)
{ System.out.println(e);
}}}
```


jyostna@jyostna-HP: ~/Desktop/java_module/sockets
jyostna@jyostna-HP:~/Desktop/java_module/sockets$ javac DatRecvDemo.java
jyostna@jyostna-HP:~/Desktop/java_module/sockets$ java DatRecvDemo 3456
Hello
World
jyostna@jyostna-HP: ~/Desktop/java_module/sockets
jyostna@jyostna-HP:~/Desktop/java_module/sockets$ java DatSendDemo 127.0.0.1 345
6 Hello
jyostna@jyostna-HP:~/Desktop/java_module/sockets$ java DatSendDemo 127.0.0.1 345
6 World
jyostna@jyostna-HP:~/Desktop/java_module/sockets$
module
Generic-CDAC-Content...
Brazil_Trip
Old Firefox Data
imp_docs

URL Class - Demo 1

```
import java.io.*;
import java.net.*;
public class URLD
{
public static void main(String[] args)
{
Try
{
URL url=new URL("https://www.microsoft.com:80/index.html");
System.out.println("Protocol: "+ url.getProtocol());
System.out.println("Host Name: "+ url.getHost());
System.out.println("Port Number: "+ url.getPort());
}
catch(Exception e)
{
System.out.println(e);
} } }
```

URL Class – Demo 1 Output



URL Class - Demo 2

```
import java.io.*;
import java.net.*;
class URLDemo
{
public static void main(String args[])   {
Try
{
URL url=new URL(args[0]);
InputStream is=url.openStream();
byte buffer[]=new byte[1024];
int i;
while((i=is.read(buffer)) != -1) { System.out.write(buffer,0,i);}
}
catch(Exception e){System.out.println(e);
}}}
```

```
jyostna@jyostna-HP:~/Desktop/java_module/sockets$ javac URLDemo.java
jyostna@jyostna-HP:~/Desktop/java_module/sockets$ java URLDemo https://www.google.com
```


```
<!doctype html><html itemscope="" itemtype="http://schema.org/WebPage" lang="en-IN"><head><meta content="text/html; charset=UTF-8" http-equiv="Content-Type"><meta content="/images/branding/googleg/1x/googleg_standard_color_128dp.png" itemprop="image"><title>Google</title><script nonce="2uxBn0779H4/nIGLBX6epg==">(function(){window.google={kEI:'kykoX4C0AcKamgen8JGQDQ',kEXPI:'0,202123,3,4,32,1151585,5662,730,224,5105,206,3204,10,1226,364,1499,612,205,383,246,5,1015,339,648,995,51,1935,471,314,3,370,680,90,193,865,116,104,342,7,1120582,1197787,410,329020,13727,4855,32692,15247,867,28684,9188,8384,4859,1361,9290,3022,3896,850,11033,1808,4020,978,7931,5297,2054,920,873,1217,5667,3739,1140,6291,1221,5875,4518,1395,1381,919,2277,8,2796,887,707,1278,2212,530,149,1103,840,517,1522,4258,109,203,1132,4,3,2063,606,2025,1775,143,377,1947,2209,113,328,1284,16,2927,2246,1813,1787,3227,1990,855,7,5599,469,6286,4454,642,2450,3684,1742,4929,108,3409,906,2,940,2615,2399,9645,1098,3,576,970,865,7,3911,707,148,189,3313,2488,2252,3942,47,58,1035,651,4,1528,2304,1228,10,269,874,405,42,1818,445,1650,298,73,1718,52,214,761,346,95,961,463,460,120,755,468,212,4067,293,1972,86,3,1493,1787,1417,9,69,305,2310,1514,1297,940,8,1,1,803,690,1968,245,300,1940,949,621,106,81,321,198,25,151,736,277,217,70,689,118,313,30,463,6,29,410,395,759,6,153,122,3,53,1370,222,1,152,504,205,2,215,214,23,46,8,170,145,758,96,127,430,280,63,529,28,142,484,51,92,1148,50,2669,997,8,680,169,206,5783632,3375,8798540,549,333,444,1,2,80,1,900,896,1,9,2,2551,1,748,141,59,736,563,1,4265,1,1,2,1017,9,305,3299,248,68,23,37,36,67,4,4,4,4,4,4,4,3501938,20458659,2704831',kBL:'THJ6'};google.sn='webhp';google.kHL='en-IN';})();(function(){google.lc=[];google.li=0;google.getEI=function(a){for(var c;a&&(!a.getAttribute||!(c=a.getAttribute("eid")));)a=a.parentNode;return c||google.kEI};google.getLEI=function(a){for(var c=null;a&&(!a.getAttribute||!(c=a.getAttribute("leid")));)a=a.parentNode;return c};google.ml=function(){return null};google.time=function(){return Date.now()};google.log=function(a,c,b,d,g){if(b=google.logUrl(a,c,b,d,g)){a=new Image;var e=google.lc,f=google.li;e[f]=a;a.onerror=a.onload=a.onabort=function(){delete e[f]};google.vel&&google.vel.lu&&google.vel.lu(b);a.src=b;google.li=f+1}};google.logUrl=function(a,c,b,d,g){var e="",f=google.ls||"";b||-1!=c.search("&ei=")||(e="&ei="+google.getEI(d),-1==c.search("&lei=")&&(d=google.getLEI(d))&&(e+="&lei="+d));d="";!b&&google.cshid&&-1==c.search("&cshid=")&&"slh"!=a&&(d="&cshid="+google.cshid);b=b||"/"+(g||"gen_204")+"?atyp=i&ct="+a+"&cad="+c+e+f+"&zx="+google.time()+d;/^http:/i.test(b)&&"https:"==window.location.protocol&&(google.ml(Error("a"),!1,{src:b,glmm:1}),b="");return b};}).call(this);(function(){google.y={};google.x=function(a,b){if(a)var c=a.id;else{do c=Math.random();while(google.y[c])}google.y[c]=[a,b];return!1};google.lm=[];google.plm=function(a){google.lm.push.apply(google.lm,a)};google.lq=[];google.load=function(a,b,c){google.lq.push([[a],b,c])};google.loadAll=function(a,b){google.lq.push([a,b])};}).call(this);google.f={};(function(){
document.documentElement.addEventListener("submit",function(b){var a;if(a=b.target){var c=a.getAttribute("data-submitfalse");a="1"==c||"q"==c&&!a.elements.q.value?!0:!1}else a=!1;a&&(b.preventDefault(),b.stopPropagation())},!0);document.documentElement.addEventListener("click",function(b){var a;a:{for(a=b.target;a&&a!=document.documentElement;a=a.parentElement)if("A"==a.tagName){a="1"==a.getAttribute("data-nohref");break a}a=!1}a&&b.preventDefault()},!0);}).call(this);
var a=window.location,b=a.href.indexOf("#");if(0<=b){var c=a.href.substring(b+1);/(^|&)q=/.test(c)&&-1==c.indexOf("#")&&a.replace("/search?"+c.replace(/(^|&)fp=[^&]*/g,"")+"&cad=h")};</script><style>#gbar,#guser{font-size:13px;padding-top:1px !important;}#gbar{height:22px}#guser{padding-bottom:7px !important;text-align:right}.gbh,.gbd{border-top:1px solid #c9d7f1;font-size:1px}.gbh{height:0;position:absolute;top:24px;width:100%}@media all{.gb1{height:22px;margin-right:.5em;vertical-align:top}#gbar{float:left}}a.gb1,a.gb4{text-decoration:underline !important}a.gb1,a.gb4{color:#00c !important}.gbi .gb4{color:#dd8e27 !important}.gbf .gb4{color:#900 !important}
</style><style>body,td,a,p,.h{font-family:arial,sans-serif}body{margin:0;overflow-y:scroll}#gog{padding:3px 8px 0}td{line-height:.8em}.gac_m td{line-height:17px}form{margin-bottom:20px}.h{color:#36c}.q{color:#00c}em{font-weight:bold;font-style:normal}.lst{height:25px;width:496px}.gsfi,.lst{font:18px arial,sans-serif}.gsfs{font:17px arial,sans-serif}.ds{display:inline-box;display:inline-block;margin:3px 0 4px;margin-left:4px}input{font-family:inherit}body{background:#fff;color:#000}a{color:#11c;text-decoration:none}a:hover,a:active{text-decoration:underline}.fl a{color:#36c}a:visited{color:#551a8b}.sblc{padding-top:5px}.sblc a{display:block;margin:2px 0;margin-left:13px;font-size:11px}.lsbb{background:#eee;border:solid 1px;border-color:#ccc #999 #999 #ccc;height:30px}.lsbb{display:block}#fll a{display:inline-block;margin:0 12px}.lsb{background:url(/images/nav_logo229.png) 0 -261px repeat-x;border:none;color:#000;cursor:pointer;height:30px;margin:0;outline:0;font:15px arial,sans-serif;vertical-align:top}.lsb:active{background:#ccc}.lst:focus{outline:none}</style><script nonce="2uxBn0779H4/nIGLBX6epg==">
```


URL Class – Demo 2 Output

Thank You