

Mechi Multiple Campus

(Tribhuvan University)

Bhadrapur, Jhapa



Lab Report of Network Programming (CACS-355)

Faculty of Humanities & Social Sciences

Tribhuvan University

Kritipur, Nepal

Submitted By

Name: Santosh Bhandari

Roll No: 58

Submitted To

Mechi Multiple Campus

Department of Bachelor in Computer

Bhadrapur, Jhapa, Nepal

Table of Contents

S.N	Title	Page No.
1.	Program to manipulate the IP address of a system	1
2.	Program to obtain the information about the host, port, protocol	2
3.	Program to access daytime Service from Server using Socket	2-3
4.	Program to get remote and local Socket address	3
5.	Program to find port no running on Server	4
6.	Program to read the Source Code of Web page	5
7.	Program to Create Socket for Sending and receiving data	5-7
8.	RNT Client and Server Side	7-9

1. Program to manipulate the IP address of a System.

```
import java.net.*;
class LocalIPAddres {
    public static void main(String args[]) {
        try {
            InetAddress inet = InetAddress.getLocalHost();
            System.out.println("IP of the System : " + inet.getHostAddress());
        } catch (Exception ex) {
            System.out.println(ex);
        }
    }
}
```

Output

IP of the System : 127.0.0.1

```
import java.net.*;
class InternetAddressOfHost {
    public static void main(String args[]) {
        try {
            InetAddress inet = InetAddress.getByName("www.bhandari-sanshu.com.np");
            System.out.println("IP : " + inet.getHostAddress());
        } catch (Exception ex) {
            System.out.println(ex);
        }
    }
}
```

Output

IP : 13.251.96.10

2. Program to obtain the information about the (a) Host (b) Port (c) Protocol

```

import java.net.*;
class GetHostAndPort{
    public static void main(String args[]){
        try {
            URL url = new URL("https://bhandar-santosh.com.ng:80");
            System.out.println("Host Name : " + url.getHost());
            System.out.println("Port Number : " + url.getPort());
            System.out.println("Protocol : " + url.getProtocol());
        } catch (Exception ex) {
            System.out.println(ex);
        }
    }
}

```

3

3

3

Output

Host Name : bhandar-santosh.com.ng
 Port Number : 80
 Protocol : https

3. Program to access daytime Service from Server using Sockets.

```

import java.net.*;
import java.io.*;
public class AccessDateTime {
    public static void main(String[] args) {
        try {
            Socket s = new Socket("time.nist.gov", 13);
            BufferedReader br = new BufferedReader(new InputStreamReader(
                s.getInputStream()));
            int line,

```

```
while((line=b.read())!= -1) {
    System.out.print((char)line);
}
```

```
3 catch(Exception ex) {
    System.out.println(ex);
}
```

3

3

Output

60503 24-07-12 14:23:54 SD 0 0 784.5 UTC(NTST)*

4. Program to get remote and local Socket Address.

```
import java.net.*;
```

```
class RemoteAndLocalSocketAddress {
```

```
public static void main(String args[]) {
    try {
```

```
        Socket s = new Socket("localhost", 6666);
```

```
        System.out.println("Local Socket Address : " + s.getLocalSocketAddress());
```

```
        System.out.println("Remote Socket Address : " + s.getRemoteSocketAddress());
```

```
    } catch (Exception ex) {
```

```
        System.out.println(ex);
```

3

3

Output

Local Socket Address : 127.0.0.1:57336

Remote Socket Address : localhost/127.0.0.1:6666

5. Program to find Port No. running on Server.

```
import java.net.*;  
class PortRunningInServer {  
    public static void main(String[] args) {  
        System.out.println("Starting Scanning Server....\nOpen Port :");  
        for (int port = 0; port <= 65535; port++) {  
            try {  
                Socket s = new Socket("localhost", port);  
                System.out.println(port);  
            } catch (Exception ex) {  
                //System.out.println(ex);  
            }  
        }  
        System.out.println("Port Scanning Completed.");  
    }  
}
```

Output

Starting Scanning Server

Open Port :

631

49328

60228

Port Scanning Completed.

6. Program to read the Source Code of the Web Page.

```

import java.io.InputStream;
import java.net.*;

public class WebpageSourceCode {
    public static void main(String args[]) {
        try {
            URL url = new URL("https://bhandari-santosh.com.ng");
            URLConnection con = url.openConnection();
            InputStream sourcecode = con.getInputStream();
            int code;
            while ((code = sourcecode.read()) != -1) {
                System.out.println((char) code);
            }
        } catch (Exception ex) {
            System.out.println(ex);
        }
    }
}

```

Output

Print Website Source Code

7. Program to Create Socket for Sending and receiving data.

ServerSocket

```

import java.net.*;
import java.util.Scanner;
import java.io.*;

public class SocketServer {
    public static void main(String[] args) {
        try {
            ServerSocket ss = new ServerSocket(6666);
        }
    }
}

```

```

System.out.println("Waiting for client connection---");
Socket s = ss.accept();
System.out.println("Client connected.");
Scanner sc = new Scanner(System.in);
DataInputStream dos = new DataInputStream(s.getInputStream());
DataOutputStream dos = new DataOutputStream(s.getOutputStream());
while(true){
    System.out.println("Waiting for Client Message ---");
    String client = (String) dos.readUTF();
    System.out.println("Client : "+client);
    System.out.println("Enter a message (e for exit): ");
    String server = sc.nextLine();
    if(client.equalsIgnoreCase("e") || server.equalsIgnoreCase("e")){
        break;
    }
    System.out.println("Server : "+server);
    dos.writeUTF(server);
}
s.close();
}
catch(Exception ex){
    System.out.println(ex);
}
}

```

Client Socket

```

import java.net.*;
import java.io.*;
import java.util.Scanner;

public class SocketClient {
    public static void main(String[] args) {
        try {
            Socket s = new Socket("localhost", 6666);
            System.out.println("Connected To Server.");
            Scanner sc = new Scanner(System.in);

```

```

DataInputStream des = new DataInputStream(s.getInputStream());
DataOutputStream dos = new DataOutputStream(s.getOutputStream());
while (true) {
    System.out.println("Enter a Message (e for Exit): ");
    String Client = sc.nextLine();
    if (Client.equals("Ignore (e)")) {
        break;
    }
    dos.writeUTF(Client);
    System.out.println("Waiting Server Message... ");
    String server = (String) des.readUTF();
    System.out.println("Server :" + server);
}
s.close();
} catch (Exception ex) {
    System.out.println(ex);
}
}

```

8. RMI Client and Server Side

RMIInterface.java

```

import java.rmi.*;
public interface RMIInterface extends Remote {
    public String HelloWorld() throws RemoteException,
}

```

3

RMIInterfaceImpl.java

```

import java.rmi.*;
import java.rmi.server.*;
public class RMIInterfaceImpl extends UnicastRemoteObject implements RMIInterface {
    RMIInterfaceImpl() throws RemoteException {
        super();
    }
    public String HelloWorld() {
        return "Hello World";
    }
}

```

RMIServer.java

```

import java.rmi.registry.*;
public class RMIServer {
    public static void main(String args[]) {
        try {
            RMIInterfaceImpl obj = new RMIInterfaceImpl();
            Registry reg = LocateRegistry.createRegistry(1099);
            reg.rebind("localhost", obj);
        } catch (Exception ex) {
            System.out.println(ex);
        }
    }
}

```

RMIclient.java

```
import java.rmi.registry.*;
public class RMIclient {
    public static void main(String args[]) {
        try {
            RMIClientInterfaceImp obj = new RMIClientInterfaceImp();
            Registry reg = LocateRegistry.getRegistry(1888);
            reg.rebind("localhost", obj);
            System.out.println(obj.HelloWorld());
        } catch (Exception ex) {
            System.out.println(ex);
        }
    }
}
```

3

3