Note:- All the Mentoned Program Need to Write in the A4 Paper.

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

```java
// Program to Manipulate the IP Address of a System

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

```java
// Program to Manipulate the IP Address of a System
import java.net.*;

public class ManipulateIPAddress {
    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);
        }
    }
}
```

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

```java
1  // Program to Obtain the Information about the (a)Host (b)Port (c) Protocol
2
3  import java.net.*;
4
5  public class GetHostAndPort {
6      public static void main(String args[]){
7          try{
8              URL url = new URL("https://bhandari-santosh.com.np:80");
9              System.out.println("Host Name is = "+url.getHost());
10             System.out.println("Port Number is = "+url.getPort());
11             System.out.println("Protocal Used is = "+url.getProtocol());
12         }catch(Exception ex){
13             System.out.println(ex);
14         }
15     }
16 }
17
```

3. Program to access daytime service from server using socket.

```java
1  // 3. Program to access daytime service from server using socket.
2  import java.net.*;
3  import java.io.*;
4
5  public class AccessDateTime {
6      public static void main(String[] args) {
7          try {
8              Socket s = new Socket("time.nist.gov" , 13);
9              BufferedReader br = new BufferedReader(new InputStreamReader(s.getInputStream()));
10             int line;
11             while ((line = br.read()) != -1) {
12                 System.out.print((char) line);
13             }
14         }catch (IOException e) {
15             System.out.println(e);
16         }
17     }
18 }
```

4. Program to get remote and local socket address.

```java
// Write a Program to Get Remote and Local Socket Address

import java.net.*;
public 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);
        }
    }
}
```

5. Program to find port No. running on server.

```java
// Program to find port No. running on server.

import java.net.*;
public class PortRunningInServer{
    public static void main(String[] args) {
        System.out.println("Start 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.");
    }
}
```

6. Program to read the source Code of the web page.

```java
// 6. Program to read the source Code of the web  page.

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

7. Program to create Socket for sending and receiving data.

```java
1  // 7. Program to create Socket for sending and receiving data.
2  import java.net.*;
3  import java.util.Scanner;
4  import java.io.*;
5
6  public class SocketServer{
7      public static void main(String[] args) {
8          try {
9              ServerSocket ss = new ServerSocket(6666);
10             System.out.println("Waiting for Client Connection....");
11             Socket s = ss.accept();
12             System.out.println("Client Connected.");
13             Scanner sc = new Scanner(System.in);
14             DataInputStream dis = new DataInputStream(s.getInputStream());
15             DataOutputStream dos = new DataOutputStream(s.getOutputStream());
16             while (true) {
17                 System.out.println("Waiting For Client Message...");
18                 String client = (String)dis.readUTF();
19                 System.out.println("Client : "+client);
20                 System.out.print("Enter a Message(e for Exit): ");
21                 String server=sc.nextLine();
22                 if (client.equalsIgnoreCase("e") || server.equalsIgnoreCase("e")){
23                     break;
24                 }
25                 System.out.println("Server : "+server);
26                 dos.writeUTF(server);
27             }
28             s.close();
29         } catch (Exception ex) {
30             System.out.println(ex);
31         }
32     }
33 }
34
```

```java
// 7. Program to create Socket for sending and receiving data.
import java.net.*;
import java.util.Scanner;
import java.io.*;

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

8. RMI Client & Server Side

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

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

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

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