

Krzysztof Pancierz
Programowanie współbieżne i rozproszone
Programowanie z wykorzystaniem gniazd

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

public class Serwer
{
    public Serwer()
    {
    }

    public static void main(String[] args)
    {
        ServerSocket gniazdoSerwera=null;
        Socket gniazdo=null;
        BufferedReader odczyt=null;
        DataOutputStream strwy=null;
        String linia;

        try
        {
            gniazdoSerwera=new ServerSocket(6000);
        }
        catch(Exception e)
        {
            System.out.println("Błąd - gniazdo serwera");
            System.exit(-1);
        }

        System.out.println("Serwer uruchomiony");

        try
        {
            gniazdo=gniazdoSerwera.accept();
        }
        catch(Exception e)
        {
            System.out.println("Błąd - połączenie");
            System.exit(-1);
        }

        try
        {
            InputStream strweGniazda=gniazdo.getInputStream();
            odczyt=new BufferedReader(new InputStreamReader(strweGniazda));

            OutputStream strwyGniazda=gniazdo.getOutputStream();
            strwy=new DataOutputStream(strwyGniazda);
        }
        catch(IOException e)
        {
            System.out.println("Błąd - strumienie");
            System.exit(-1);
        }

        while(true)
        {
            try
            {
                linia=odczyt.readLine();

                if(linia.equals("quit"))
                {
                    System.out.println("koniec pracy serwera");
                    gniazdoSerwera.close();
                    System.exit(0);
                }

                System.out.println("Otrzymane: "+linia);

                linia=linia.toUpperCase();

                strwy.writeBytes(linia+"\n");

                //opróżnienie strumienia wyjściowego:
                strwy.flush();
            }
            catch(IOException e)
            {
                System.out.println("Błąd - operacje we/wy");
                System.exit(-1);
            }
        }
    }
}
```

Krzysztof Pancerz
Programowanie współbieżne i rozproszone
Programowanie z wykorzystaniem gniazd

```
    }  
}  
  
*****  
  
import java.net.*;  
import java.io.*;  
  
public class Klient  
{  
    public Klient()  
    {  
    }  
  
    public static void main(String[] args)  
    {  
        Socket gniazdo=null;  
        String linia;  
        BufferedReader gniazdoOdczyt=null;  
        BufferedReader klawiaturaOdczyt=null;  
        DataOutputStream strwy=null;  
  
        try  
        {  
            gniazdo=new Socket("localhost", 6000);  
        }  
        catch(Exception e)  
        {  
            System.out.println("Błąd - gniazdo klienta");  
            System.exit(-1);  
        }  
  
        System.out.println("Połączenie z "+gniazdo);  
  
        try  
        {  
            OutputStream strwyGniazda=gniazdo.getOutputStream();  
            strwy=new DataOutputStream(strwyGniazda);  
  
            InputStream strweGniazda=gniazdo.getInputStream();  
            gniazdoOdczyt=new BufferedReader(new InputStreamReader(strweGniazda));  
  
            InputStream strweKlaw=System.in;  
            klawiaturaOdczyt=new BufferedReader(new InputStreamReader(strweKlaw));  
        }  
        catch(Exception e)  
        {  
            System.out.println("Błąd - strumienie");  
            System.exit(-1);  
        }  
  
        while(true)  
        {  
            try  
            {  
                linia=klawiaturaOdczyt.readLine();  
                strwy.writeBytes(linia+"\n");  
                strwy.flush();  
  
                if(linia.equals("quit"))  
                {  
                    System.out.println("koniec pracy klienta");  
                    gniazdo.close();  
                    System.exit(0);  
                }  
  
                linia=gniazdoOdczyt.readLine();  
  
                System.out.println("Otrzymane: "+linia);  
            }  
            catch(Exception e)  
            {  
                System.out.println("Błąd - operacje we/wy");  
                System.exit(-1);  
            }  
        }  
    }  
}
```