

TP2

Rappels : Socket TCP et UDP

Tous les programmes seront écrits en Java.

1. On veut créer une application client/serveur. Cette application utilise des sockets TCP.
 - Le serveur attend une requête du client sur le port 1027
 - Le client lit une chaîne au clavier et l'envoie au serveur. Puis il attend de recevoir des chaînes du serveur.
 - A la lecture d'une chaîne le serveur l'envoie à tous les clients connectés.
 - (a) Comment se synchronisent les threads du serveur ?
 - (b) Donner le code serveur et client.
 - (c) Comment gérer le départ d'un client ?

Cet exercice sera à rendre (cf TP à rendre)
2. On veut créer une application client/serveur. Cette application utilise des sockets UCP.
 - Le serveur attend une requête du client sur le port 9876
 - Le client lit une chaîne au clavier et l'envoie au serveur.
 - A la lecture d'une chaîne le serveur l'affiche.Donner le code serveur et client.

```
import java.io.*;
import java.net.*;
class Client
{
    public static void main(String args[]) throws Exception {
        BufferedReader inFromUser = new BufferedReader(
            new InputStreamReader(System.in));
        DatagramSocket clientSocket = new DatagramSocket();
        InetAddress IPAddress = InetAddress.getByName("localhost");
        byte[] sendData = new byte[1024];
        String sentence = inFromUser.readLine();
        sendData = sentence.getBytes();
        DatagramPacket sendPacket = new DatagramPacket(sendData,
            sendData.length, IPAddress, 9876);
        clientSocket.send(sendPacket);
        clientSocket.close(); }
}
```

```

    }
}
-----
import java.io.*;
import java.net.*;
public class Serveur {
    public static void main(String args[]) throws Exception{
        DatagramSocket serverSocket = new DatagramSocket(9876);
        byte[] receiveData = new byte[1024];
        byte[] sendData = new byte[1024];
        while(true) {
            DatagramPacket receivePacket = new DatagramPacket(receiveData, receiveData.l
            serverSocket.receive(receivePacket);
            String sentence = new String(receivePacket.getData());
            System.out.println(sentence);
        }
    }
}

```