

TP 4 : mise en place de résolution DNS, PING, Socket et Thread

Documentation

1. <https://openclassrooms.com/courses/java-et-la-programmation-reseau>
2. Fichier du TP : IP-M1-20132014.pdf
3. <http://www.computing.dcu.ie/~humphrys/Notes/Networks/java.html>
4. <http://cs.lmu.edu/~ray/notes/javanetexamples/>

A. IP Address

1. Exercice : Lecture

- Commencer par lire IP-M1-20132014.pdf

2. Exercice : Get IP address of hostname

- Commencer par éditer et commenter le fichier ip.java
- Compilation et exécution sous linux
 - Pour compiler : javac ip.java
 - Pour exécuter : java ip www.computing.dcu.ie
- Compiler et exécuter la classe. Faire plusieurs tests sur des sites différents

3. Exercice : Get the hostname of an IP address

- Utilisation de l'objet `InetAddress` pour récupérer, en fonction de ce que vous allez saisir, soit l'adresse IP d'une adresse internet, soit le nom de domaine d'une adresse IPv4.
Write program to find hostname given IP. Note `getByName()` is flexible in its input.

4. Exercice : Ping

- Faire un sort de ping avec la methode `isReachable()` : tente un envoi de ECHO REQUEST en ICMP (si privilège ok), ou alors tente d'établir une connexion TCP avec le port 7 (Echo).

B. Socket

1. Exercice : Date Server

- Commencer par éditer et commenter les lignes de la classe `DateServer.java`.
- Compiler et Exécuter cette classe. Commenter le résultat.
- Editer et commenter les lignes de la classe `DateClient.java`
- Compiler et Exécuter cette classe. Commenter le résultat.
- Est-ce que le serveur peut traiter plusieurs clients simultanément ?

2. Exercice : Port Scanner

- En lisant la classe `ports.java`, proposer un programme qui permet de lister l'ensemble des ports ouverts sur une machine passée en argument. Vous pouvez faire une petite console pour saisir le hostname cible

C. Thread

1. Exercice : PrintThread1

- Commencer par éditer et commenter les lignes de la classe PrintThread1.java.
- Compiler et Exécuter cette classe. Commenter le résultat.

2. Exercice : PrintThread2

- Commencer par éditer et commenter les lignes de la classe PrintThread2.java.
- Compiler et Exécuter cette classe. Commenter le résultat. Expliquer la différence entre les deux classes.

3. Exercice : Ping-Pong

- Ecrire une classe Ping qui réalise un Thread qui affiche « Ping » à intervalle irrégulier
- Ecrire une classe Pong qui réalise un Thread qui affiche « Pong » à intervalle irrégulier
- Ecrire une classe Go qui lance les Threads Ping et Pong

D. Threaded Server

1. Exercice : Date Server

- Proposer une version parallèle du serveur de temps précédent. Procéder aux tests.

2. Exercice : Hello Server

- Editer, commenter et exécuter le serveur-client Hello