# TP – Réseau

# License DANT – PITTON Olivier

## Exercice 1 – Echo Server

En utilisant Java IO (Socket / ServerSocket), vous créerez un serveur et un client.

Le serveur accepte une seule connection, lis chaque ligne envoyée par le client, l’affiche et la renvoie directement au client.

Le client lit le flux d’entrée standard (System.in). A chaque ligne écrite dans la console, celle-ci est envoyée au serveur. Le client lit la réponse du serveur, l’affiche et attend la prochaine écriture sur le flux d’entrée.

Conseil : Utiliser les classe PrintWriter / BufferedReader pour écrire / lire une ligne entière

## Exercice 2 – Selector

Reprenez le code situé dans cours.nio contenant les SocketChannel.

Ce code étant bloquant, vous devrez le rendre non bloquant en … recherchant sur internet, comme dans la vraie vie 😊

## Exercice 3 – Echo Server asynchrone bloquant

Reprenez l’exercice 1, mais cette fois en utilisant les AsynchronousSocketChannel. Vous utiliserez les Future de manière à ce que chaque appel soit bloquant, à l’image de l’exercice 1, mais asynchrone.

## Exercice 4 – Echo Server asynchrone non bloquant

Reprenez l’exercice 3, mais cette fois utilisez uniquement les CompletionHandler pour CHAQUE appel.