

# Explorateur de fichiers

## Contexte

Un client souhaite obtenir une version basique d’exploration de fichiers similaire à ce qui pourrait être réalisé à l’aide du protocole FTP, mais à l’aide de sockets Java.

## Le programme

Le programme demandé devra être constitué d’un serveur acceptant de multiples connexions de clients. Ce serveur devra être capable, pour chaque client, de parcourir une arborescence de dossiers et de fichiers à partir d’un dossier de départ fixe (initialisé par le serveur). Chaque client pourra agir en simultané sur le serveur. Deux types d’instructions sont à gérer :

- **ls** : Liste les dossiers et fichiers contenus dans un dossier. Cette instruction prend éventuellement en paramètre le nom d’un dossier ;
- **cd** : Change le dossier courant (pour l’utilisateur en question). Cette instruction prend en paramètre le nom d’un dossier ou `..` (pour remonter au dossier parent).

On pourra supposer que tout client se connectant au serveur est placé par défaut dans un répertoire prédéterminé, le même pour chaque client.

## Fichier fourni

Vous trouverez, dans le répertoire `/share/m1miage/sem1-CSR/TP/TP3/src` le fichier `FileExplorerProtocol.java`. Il contient les méthodes nécessaires à l’implémentation des deux commandes décrites ci-dessus, de sorte que vous puissiez vous concentrer sur le développement de la partie communication client/serveur du programme. Le constructeur prend en paramètre le chemin d’un dossier qui sera le dossier initial par défaut décrit plus haut.

## Squelette d’implémentation

Votre programme sera constitué de 4 classes, incluant la classe protocole qui vous est fournie. Les trois fichiers à écrire sont :

- **FTPClient** : Cette classe gère le côté client des communications. Elle crée les *streams* nécessaires à la communication avec le serveur et créer un canal de communication avec le serveur. Ensuite, en boucle, elle accepte une ligne en entrée du client sur `System.in` et l’envoie au serveur, puis reçoit la réponse du serveur.
- **FTPServeur** : Cette classe gère l’ouverture d’un socket serveur sur un port déterminé sur localhost, puis crée une instance de `FTPServeurThread` à chaque fois qu’un client se connecte, et lance cette instance ;
- **FTPServeurThread** : Cette classe étend `Thread` et doit donc implémenter les méthodes nécessaires. Quand le thread est lancé, les *streams* nécessaires à la communication avec le client sont créés, et une instance de `FileExplorerProtocol` est initialisée avec un chemin par

défaut. Finalement, une boucle d'écoute du client est mise en place. Quand un message du client est reçu, le serveur émet pour réponse la chaîne de caractères produite par la méthode `processInput(inputString)` de `FileExplorerProtocol` qui se charge de l'analyse de la commande ;