***Exercice1***

**Objectif de l’exercice :** développer un serveur TCP/IP qui écoute sur le port 10000 et qui accepte des connexions clientes. Le client se connecte au serveur, puis lui envoie un message. Le message est analysé par le serveur, qui lui répond.

**Côté Serveur :**

**1.** Demande à l'utilisateur de saisir le port sur lequel le serveur écoutera.S'il saisit un non-entier, une exception est attrapée et un message d'erreur est affiché, expliquant l'utilisation correcte du programme.

**2. Création du ServerSocket :** Crée un objet ServerSocket lié au port spécifié par l'utilisateur.

**3. Attente et acceptation de la connexion :** Le serveur Attend qu'une connexion soit établie depuis un client en utilisant la méthode accept() du ServerSocket.. Lorsqu'une connexion est acceptée, un nouveau socket est crée dédié à la communication avec le client.

**4. Création des flux de communication :** Crée des flux d'entrée et de sortie pour envoyer et recevoir des objets sur le socket.

**5. Lecture :** Lit une chaîne envoyée par le client à travers le socket, puis affiche cette chaîne sur la console.

**6. Affichage :** Affiche l'adresse IP et le port du client à partir duquel la connexion a été reçue

**7. Confirmation :** Envoie une confirmation au client que le message a été bien reçu.

**8. Gestion des erreurs :** Attrape toute exception pouvant survenir pendant l'exécution du code et affiche un message d'erreur correspondant.

**Côté client**

**1.** **Saisie du nom du serveur et du port :** demandant à l'utilisateur de saisir le nom et le port du serveur.

**2. Obtention de l'adresse IP du serveur :** À partir du nom du serveur, l'adresse IP du serveur est obtenue à l'aide de InetAddress.

**3. Création du Socket client et connexion au serveur :** Un objet Socket est créé, se connectant au serveur en utilisant l'adresse IP et le port fournis.

**4. Création des flux de communication :** Des flux de sortie (ObjectOutputStream) et d'entrée (ObjectInputStream) sont instanciés pour permettre la transmission d'objets entre le client et le serveur.

**5. Envoi d'une chaîne au serveur :** Le client envoie une chaîne au serveur via le flux de sortie.

**6. Lecture d'une chaîne du serveur :** Le client lit une chaîne envoyée par le serveur via le flux d'entrée.

**7. Affichage de la chaîne reçue du serveur :** La chaîne reçue du serveur est affichée côté client.

**8.** **Gestion des erreurs :** capture toutes les exceptions qui pourraient se produire lors de l'exécution du code dans le bloc try et les affiche dans la console.