04/07/2022

CONCEPTION D’UNE CALCULATRICE SIMPLE REPARTIE

MASTER 1 BDGL

Etudiants : SACKO ALLOU-BADRA ET KAMATE ISSIAKA

Prof : dr ASSOHOUN

1. **Approche définitionnelle RPC**

L’appel de procédure à distance (RPC, Remote Procedure Call) est un protocole qu’un programme peut utiliser pour solliciter un service auprès d’un programme situé sur un autre ordinateur d’un réseau dont il n’a pas besoin de connaître les détails. On l’appelle parfois appel de fonction ou de sous-routine.

Le RPC s’appuie sur le modèle client/serveur. Le programme demandeur est le client et le programme assurant le service, le serveur. Comme un appel de procédure locale ou habituelle, un remote procedure call est une opération synchrone : le programme demandeur attend la fin du traitement de la procédure distante pour reprendre. Toutefois, le recours à des threads ou processus légers à espace d’adressage commun permet l’exécution simultanée de plusieurs appels de procédure à distance.

* **Procédure de message des RPC**

Quand des instructions de programme qui utilisent un Framework RPC sont compilées en un programme exécutable, un fichier stub est inclus dans le code compilé pour représenter le code de la procédure distante. Quand le programme s’exécute et que la procédure est appelée, le fichier stub reçoit la demande et la fait suivre à un programme d’exécution client sur l’ordinateur local

Le programme d’exécution client sait comment s’adresser à l’application serveur de l’ordinateur distant et envoie par le réseau un message demandant la procédure à distance. De même, le serveur comprend un fichier stub et un programme d’exécution qui communique avec la procédure distante proprement dite. Les protocoles de type demande-réponse suivent la même méthode.

1. **Communication transitoire orientée messages avec les sockets**

* De nombreux systèmes et applications distribués sont construits directement sur le modèle simple orienté message proposé par la couche de transport.
* Pour mieux comprendre et apprécier les systèmes orientés message dans les solutions middleware, nous abordons les communications via des sockets de niveau transport
* Une attention particulière a été accordée à la normalisation de l'interface de la couche de transport afin de permettre aux programmeurs d'utiliser toute sa suite de protocoles par le biais d'un simple ensemble d'opérations.
* De plus, les interfaces standard facilitent le portage d'une application sur une autre machine.
* A titre d’exemple, l'interface socket introduite dans les années1970 dans Unix de Berkeley et qui a été adoptée comme standard POSIX (avec très peu d'adaptations).
* Conceptuellement, un socket est un point de communication sur lequel une application peut écrire des données à envoyer sur le réseau sous-jacent et à partir desquelles des données entrantes peuvent être lues.
* Un socket forme une abstraction sur le port réel utilisé par le système d’exploitation local pour un transport spécifique.

1. **Mode applicatif sur une calculatrice simple.**

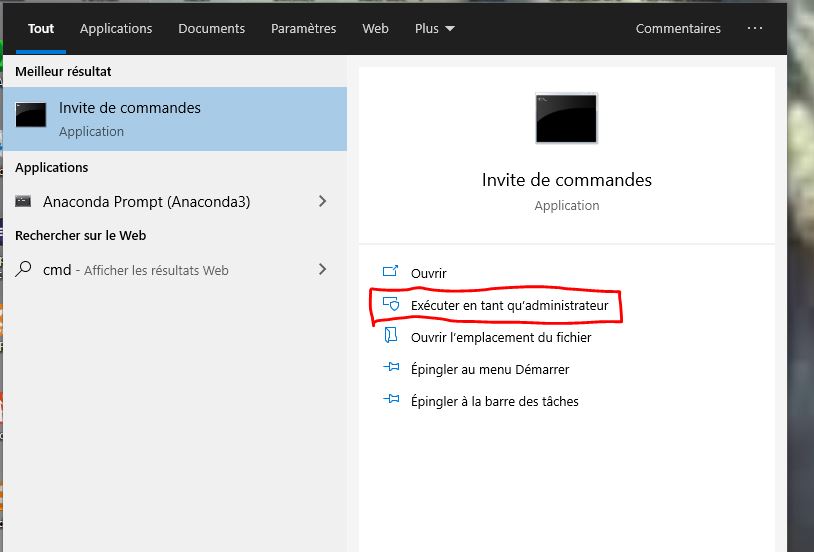
La calculatrice est conçue de façon repartie. C’est-à-dire une partie sur le serveur et une partie chez l’utilisateur (avec l’interface).

Nous avons utilisé l’éditeur VS Code pour la conception de notre code source et son implémentation.

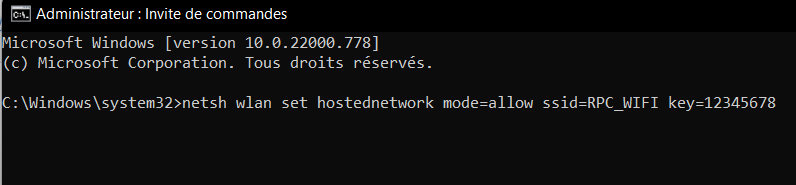
L’implémentation se fait sur deux machines distinctes, l’une servira de serveur et l’autre le client utilisateur. Pour se faire, nous avons certaines conditions à respecter pour la bonne implémentation et une utilisation parfaite par l’utilisateur.

* 1. **Création d’une connexion entre le serveur et les autres machines clientes**

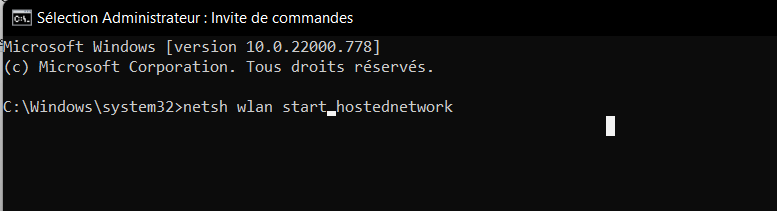
-La machine qui va servir de Serveur va créer une connexion WIFI à l'aide de l'invite de commande exécuté en mode administrateur :



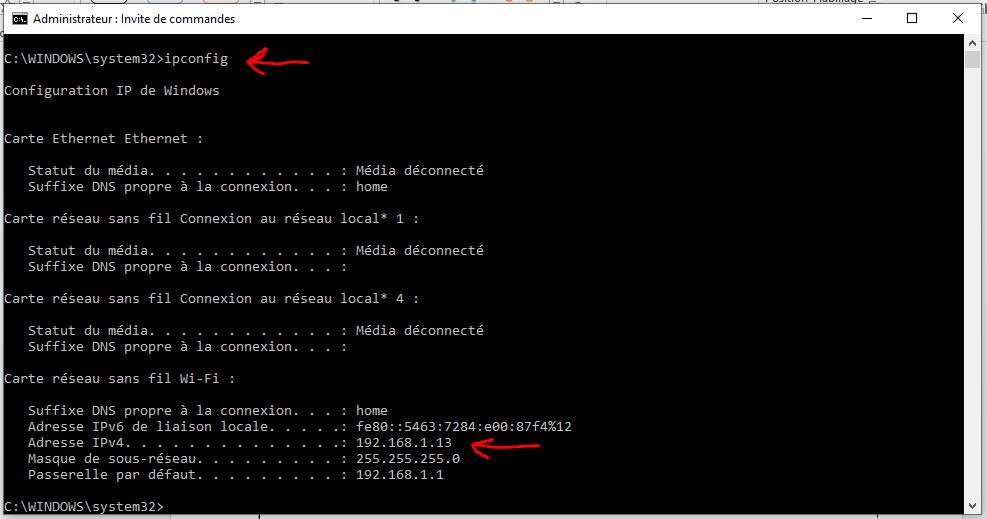
-sur l'invite de commande saisir cette commande (netsh wlan set hostednetwork mode=allow ssid=RPC\_WIFI key=12345678) pour créer le WIFI.



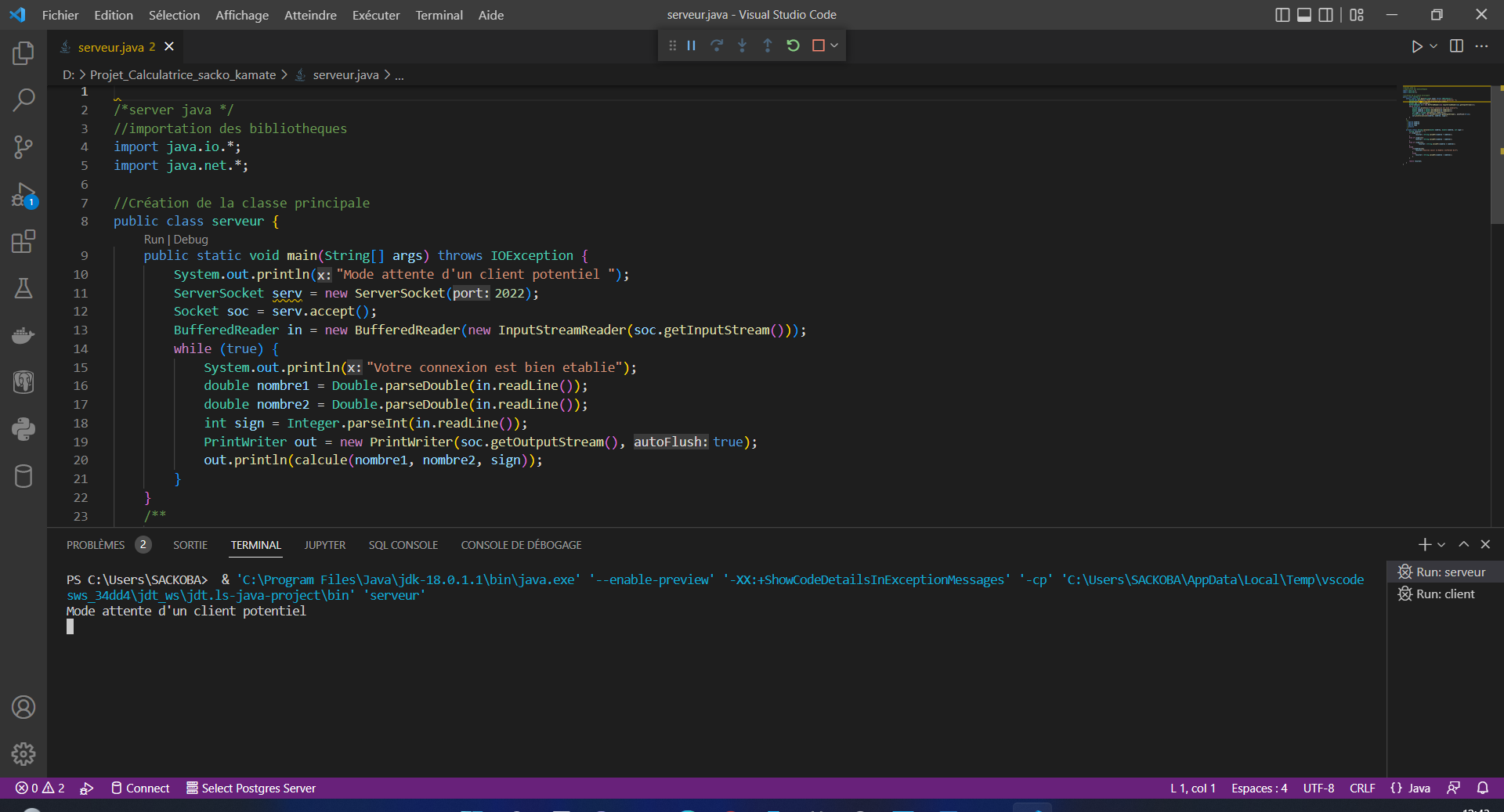
-puis saisir cette commande (netsh wlan start hostednetwork) pour démarrer le WIFI.



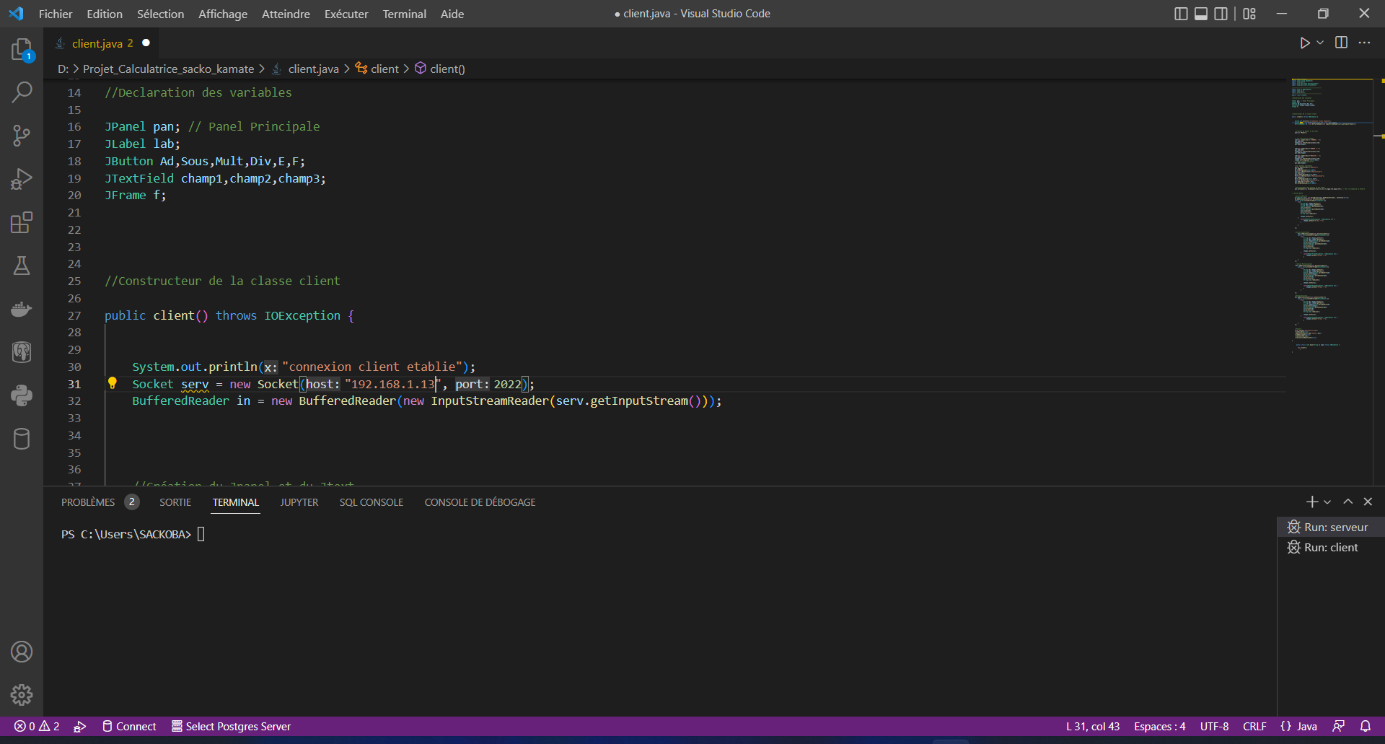
-enfin saisir la commande ipconfig pour obtenir l’adresse IP de la machine ‘’Serveur’’ qui sera utilisé par les autres machines ‘’Client’’ pour se connecter à la machine ‘’Serveur’’



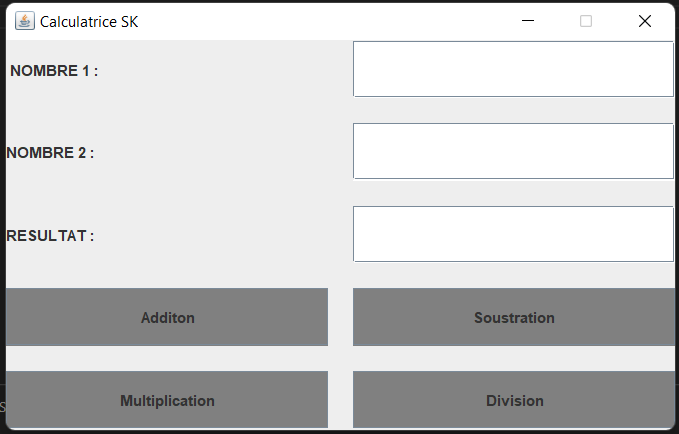
* 1. **Utilisation proprement dit du logiciel.**
* On exécute le code serveur.java sur la machine qui va servir de serveur.



* On exécute ensuite le code client.java sur la machine qui sera le client avec dans le code source nom\_serveur et port qui seront l’adresse IP (192.168.1.13) et le port d’écoute du serveur (2022).

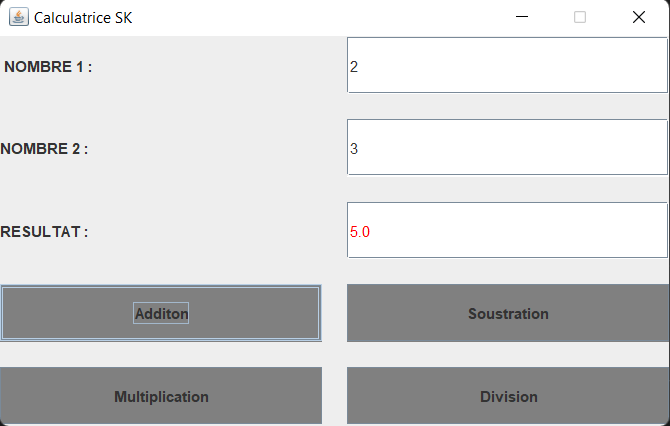


* Après l’exécution, on obtient cette interface utilisateur client

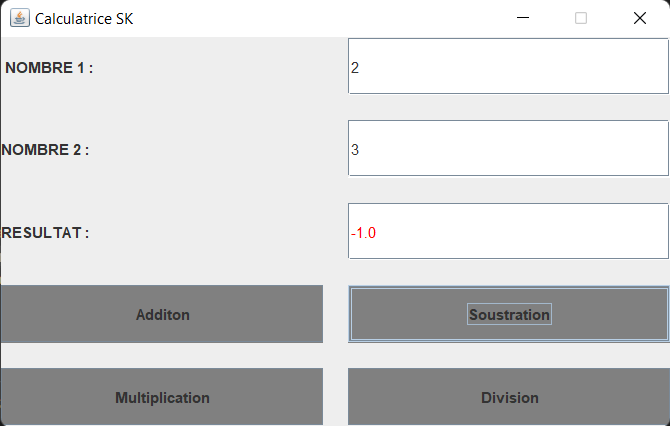


* On saisit la valeur du 1er nombre dans le champ juste devant le Nombre 1
* On saisit également le second nombre dans le champ devant le Nombre 2
* Et on choisit ensuite l’opération que nous voulons en appuyant sur le bouton Addition pour faire une addition, sur Soustraction pour faire une soustraction, sur Multiplication pour faire une multiplication ou sur Division pour faire une division.
* Ensuite l’opération est exécutée aussitôt sur le serveur et le résultat est ramené sur le champ devant Resultat.
* Et même le cas d’une division par zéro est pris en compte dans le programme.

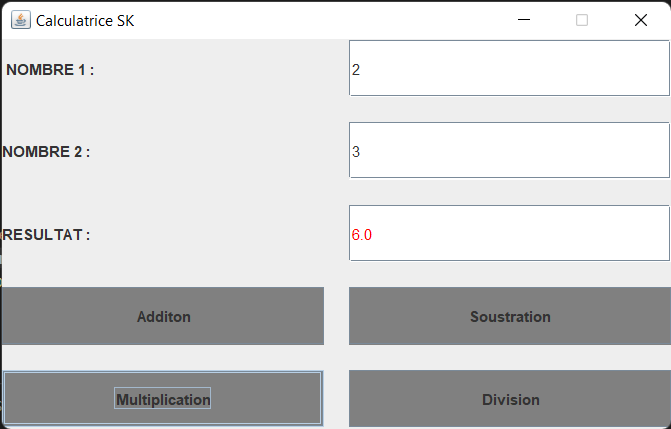
**Cas d’une addition :**



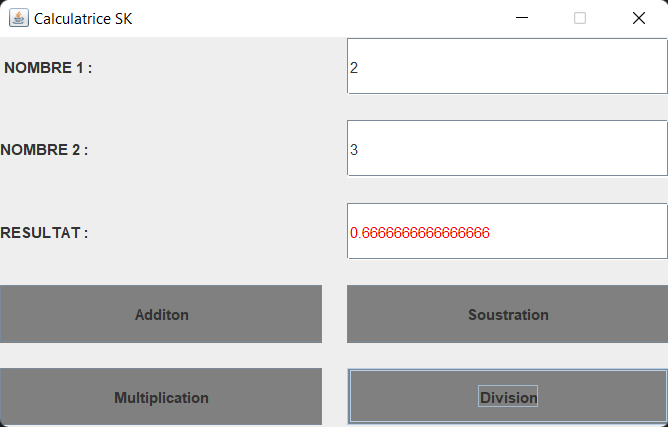
**Cas d’une soustraction :**



**Cas d’une multiplication :**



**Cas d’une division :**



**Cas d’une division par zéro :**

