

## L'application "serveur"

```
from socket import socket

# serveur=socket.socket()
serveur=socket()

PORT=3333
HOTE='' # le serveur écoutera toutes les interfaces réseau

try:
    PE=(HOTE,PORT)
    serveur.bind(PE)
    serveur.listen(2)
    while True:
        print ("En cours d'écoute...", serveur.getsockname())
        sc,adr=serveur.accept() # sc est le socket de communication
        print ('client connecté:', adr)
        data=sc.recv(1024) # data est de type byte[]
        chaine = data.decode() # 33 22
        s1, s2= chaine.split() # découpe une chaîne en utilisateur l'espace co
# comme séparateur et
# retourne les valeurs sous forme d'un tableau

        # si le client envoie "33;22" s1,s2= chaine.split(';')
        i1=int(s1) # Conversion string vers int
        i2=int(s2)
        s= i1+ i2
        # conversion int --> string --> byte []
        # str() convertit un nombre en une chaîne
        #encode convertit une chaîne en un tableau d'octet
        data= str(s).encode()
        sc.sendall(data)
        sc.close()
finally:
    serveur.close()
```

## Application "Client"

```
from socket import socket
client=socket()
PORT=3333
HOTE='127.0.0.1'
```

```
MAX=1024
try:
    client.connect((HOTE,PORT))
    print ('connecté ....', client.getpeername())
    v1= input("V1=")
    v2=input("v2=")
    s= v1 + " " + v2
    data=s.encode()
    client.sendall(data)

    # Réception
    data=client.recv(MAX)
    print(v1, "+",v2,"=",data.decode())

finally:
    client.close()

#ex1: ex1 le client peut réaliser des calculs , tant que l'utilisateur n'a pas
saisi la
        # q
#ex2: écrire un serveur et un client qui supporte plusieurs opérations
#      + / - *
```