

Exercices d'informatique - Corrigé de la série n° 3

Cours 4INOC01

1. Les *sockets*.

Les objets *sockets* permettent la communication entre différentes machines. Les méthodes utilisées sont décrites dans le livre de G. Swinnen aux pages 278 à 281.

Pour pouvoir communiquer avec un camarade, il suffit d'inscrire dans les deux scripts l'adresse IP du poste du camarade.

2. Conversation.

Voici un exemple de solution pour le serveur

```
1 import threading
2 import socket , sys
3 host='127.0.0.1'
4 port=5000
5 c=socket.socket(socket.AF_INET, socket.SOCK_STREAM)
6 try:
7     c.bind((host , port))
8 except:
9     print "Connexion impossible"
10    sys.exit ()
11 c.listen(10)
12 connexion , adresse=c.accept ()
13 class reception(threading.Thread):
14     def run(self):
15         while True:
16             m=connexion.recv(100)
17             if 'stop' in m:
18                 self.ext._Thread__stop ()
19                 break
20             print m
21 class envoi(threading.Thread):
22     def run(self):
23         while True:
24             t=raw_input ()
25             connexion.send(t)
26 j1=reception ()
27 j2=envoi ()
28 j1.ext=j2
29 j1.start ()
30 j2.start ()
```

et pour le client

```
1 import threading
2 import socket , sys
3 host='127.0.0.1'
4 port=5000
5 c=socket.socket(socket.AF_INET, socket.SOCK_STREAM)
6 try:
7     c.connect((host , port))
8 except:
9     print "Connexion au serveur impossible"
10    sys.exit ()
11 class reception(threading.Thread):
12     def run(self):
13         while True:
14             m=c.recv(100)
15             print m
```

```

16 class envoi(threading.Thread):
17     def run(self):
18         while True:
19             t=raw_input()
20             c.send(t)
21             if 'stop' in t:
22                 self.ext._Thread__stop()
23                 c.close()
24                 break
25 j1=reception()
26 j2=envoi()
27 j2.ext=j1
28 j1.start()
29 j2.start()

```

3. Conversations.

On peut, par exemple, utiliser le script

```

1 import threading
2 import socket , sys
3 host='127.0.0.1'
4 port=5000
5 class client(threading.Thread):
6     def run(self):
7         while True:
8             m=self.connexion.recv(100)
9             if 'stop' in m:
10                break
11             message='('+self.getName()+') '+m
12             print message
13             for x in connexions:
14                 connexions[x].send(message)
15             self.connexion.close()
16             del connexions[self.getName()]
17             message=self.getName()+ ' est sorti !'
18             for x in connexions:
19                 connexions[x].send(message)
20             print message
21 c=socket.socket(socket.AF_INET, socket.SOCK_STREAM)
22 try:
23     c.bind((host , port))
24 except:
25     print "Connexion impossible"
26     sys.exit()
27 c.listen(10)
28 connexions={}
29 while True:
30     connexion , adresse=c.accept()
31     job=client()
32     job.connexion=connexion
33     connexions[job.getName()]=connexion
34     print 'Nouveau client : ',job.getName() , adresse[0] , adresse[1]
35     job.start()
36     connexion.send('Envoyez vos messages !')

```

pour le serveur et le script

```

1 import threading
2 import socket , sys
3 host='127.0.0.1'
4 port=5000
5 c=socket.socket(socket.AF_INET, socket.SOCK_STREAM)
6 try:
7     c.connect((host , port))

```

```
8 except:
9     print "Connexion au serveur impossible"
10    sys.exit()
11 class reception(threading.Thread):
12     def run(self):
13         while True:
14             m=c.recv(100)
15             print m
16 class envoi(threading.Thread):
17     def run(self):
18         while True:
19             t=raw_input()
20             c.send(t)
21             if 'stop' in t:
22                 self.ext._Thread__stop()
23                 c.close()
24                 break
25 j1=reception()
26 j2=envoi()
27 j2.ext=j1
28 j1.start()
29 j2.start()
```

pour les utilisateurs.