
TD1 RMI

1. Donner la procédure exacte pour lancer le serveur ci-dessous

On commence par lancer le registre
`rmiregistry &`
Ensuite on lance le serveur
`java Server.java`
Puis on lance le client en précisant le port
`java Client.java 1099`

2. Que se passe t-il si on remplace la ligne 15 par la ligne 14 de Server.java ?

Avec la nouvelle ligne, plus besoin de créer de registre avant de lancer le serveur. Cela nous permet également d'avoir le registre sur la même JVM que le serveur. L'inconvénient est que si le serveur tombe, le registre tombe également. Si on a besoin d'actualiser notre registre il va falloir arrêter le serveur.

3. Donnez les affichages chez le client et chez le serveur

— Serveur :
- "Server ready"
- "The server prints : Hello World!"

— Client :
- "response : Hello World!"

4. Dans quelle JVM seront créés les objets de type HelloImpl ?

Dans la JVM contenant le serveur

5. A quoi sert l'interface Hello.java ?

A spécifier les méthodes accessibles à distance pour les classes implémentant cette interface

6. Donnez des exemples d'exceptions pouvant être attrapés à la ligne 22 de Server.java

AlreadyBoundException, NullPointerException

7. Quelle est la différence entre Naming.bind et Naming.rebind

Bind va lier le nom au registre et rebind va remplacer l'ancien alors que si on utilise deux fois bind on aura une exception

8. Que se passe-t-il si on ne passe pas d'argument en ligne de commande quand on lance le client ?

Le registre sera récupéré en utilisant le port par défaut (1099), si notre registre est lancé sur ce dernier alors ça ne posera pas de problème

9. Faites un diagramme d'instances pour le côté serveur et un pour le côté client illustrant les objets présents au démarrage du serveur et du client. Faites un autre diagramme d'objets illustrant le côté client après la ligne 15 du client.

