Práctica 2: RMI

1. **Remote Method Invocation**

   Siga el siguiente tutorial:

   https://docs.oracle.com/javase/7/docs/technotes/guides/rmi/hello/hello-world.html#create

   Para el archivo hello.tar, seguir la información que está en las diapositiva con el siguiente link:
   http://www.sc.ehu.es/acwlaalm/sdi.html

   # Parte I: Fundamentos

   ## Tema 1. Introducción a los sistemas distribuidos

   Apuntes

   Transparencias

   Laboratorio de programación en red **(Ficheros: udp_ipcast.zip)**

   Laboratorio de programación distribuida **(Ficheros: hello.tar)**

   ## Tema 2. Tiempo, causalidad y estado global

   Actividad: Desarrolle el ejercicio 1

2. **Chat**

Se anexa un programa basado en "Chat" escrito en Java RMI. Ejecute la siguiente instrucción para compilar el programa:

./run n A B C

Donde "n" es el número del terminal (1-3) en el que se debe ejecutar el programa. Cuando todos los programas hayan comenzado, debe presionar entre en cada uno de los terminales. Entonces, deberían estar listos para enviar mensajes entre los programas. Cada mensaje que escriba debe ir precedido por el número del terminal en el que se desea que aparezca el mensaje. Por ejemplo:

2 Este es el terminal para el terminal dos

Puede observar que en este programa no se hace distinción entre clientes y servidores: todos los programas son idénticos y cumplen el rol de cliente y servidores.

Actividad: **Comente cada línea de código.**

Información Adicional

Ejemplos de GitHub con chat, juegos, e-commerce

- Game of the rope
  https://github.com/respinha/game-of-the-rope
- Java RMI – Server Client, Peer to Peer
  https://github.com/rrqq/java-rmi-chat-p2p
- Room –based chat Server and Client
  https://github.com/ajmorton/distributed-chat-system
- Message Broadcaster Demo
  https://github.com/xunyunliu/MessageBroadcasterDemo
-