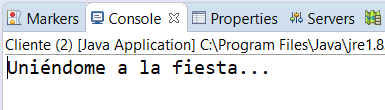
**Ejemplos ejercicios:**

**Ejercicio 2:** *(2,5 puntos)*. MulticastSocket. Para este ejercicio crearemos **únicamente una clase**. Esa clase lo primero que deberá hacer al iniciarse es enviar un mensaje multicast y posteriormente esperar un mensaje del mismo grupo multicast. De esta forma, cada vez que se lance una instancia del programa, todos los demás iniciados con anterioridad recibirán el mismo mensaje.

* Enviar mensaje (1 punto)
* Recibir mensaje (1 punto)
* Cerrar todos los programas (0,5 puntos)

Lo que debe mostrar:

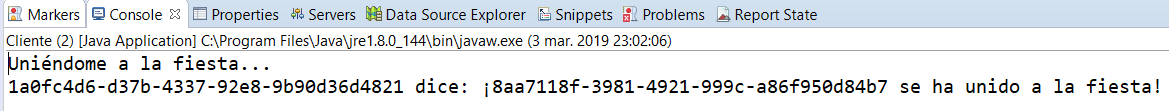
1. Mensaje al lanzar la primera instancia:



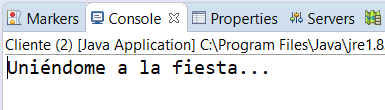
Y se queda a la espera de recibir mensaje…

1. Al lanzar dos instancias:

*Programa 1*:

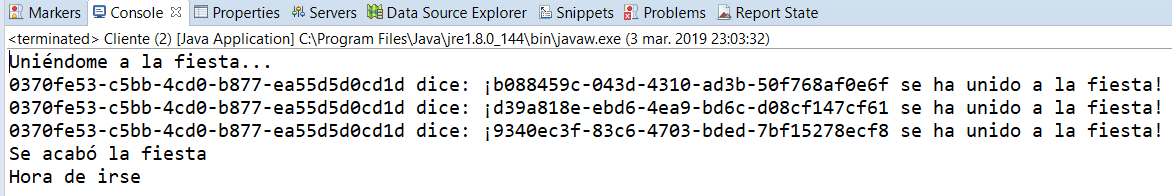


*Programa 2:*



Y así sucesivamente.

Cuando se lancen 3 programas, el primer programa lanzará un mensaje finalizando los demás:



**NOTA**: para identificar cada ejecución puedes utilizar:



**Ejercicio 3:** *(5 puntos)*. Servidor de correo electrónico. Para este ejercicio se requiere programar un cliente y servicio de correo electrónico.

En la parte del **servidor**:

* (0,5 punto)Esperará **conexiones entrantes** de clientes de correo. Deberá permitir muchas conexiones simultáneas.
* (1 punto) Deberá **almacenar** dos tipos de datos: credenciales (usuarios y contraseñas) y los emails correspondientes a cada usuario.
  + Credenciales: Un correo como usuario y una contraseña. Al lanzar el servidor se crearán algunos de ejemplo que usarán los clientes.
  + Emails: Serán enviados por los usuarios que se conecten y los almacenará el servidor. Guardaremos dirección de origen y de destino, asunto y cuerpo del mensaje.
* (0,5 puntos) Cuando un **cliente se conecte,** el servidor esperará recibir un usuario y contraseña. Si son correctos devolverá un mensaje afirmativo al cliente, en caso contrario uno de error de autenticación y cerrará el socket.
* Si la autenticación ha sido válida podrá recibir algunas instrucciones del cliente.

En la parte del **cliente**:

* (0,5 puntos) El cliente deberá de **conectarse al servidor**. Cuando se conecte deberá de enviar un correo y contraseña propios.
* Si la conexión devuelve un **error** acaba la ejecución. En caso contrario, el cliente tendrá un menú con tres opciones:
  + *(1 punto)* Enviar correo: El cliente escribirá en la consola e-mail de destino, cuerpo y mensaje y lo enviará al servidor. El servidor lo almacenará.
  + *(1 punto)* Recibir correos: El cliente recibe todos correos que le correspondan.
  + *(0,5 puntos)* Salir: Se cierra el cliente.

**Ejercicio 1:** *(3 puntos)*. Conexión a servidor ftp abierto. El programa se conectará a un servidor ftp abierto definido por el usuario a través de la consola (por ejemplo, *ftp.rediris.es*). Una vez conectado, el usuario podrá introducir los siguientes comandos:

* **cd directorio** : Cambiamos el directorio actual por directorio (1 puntos).
  + Por ejemplo: cd sites cambiaría el directorio a la carpeta sites.
* **cd ..** : Nos movemos a la carpeta superior (1 puntos).
* **ls** : Listamos el contenido del directorio actual (0,5 puntos).
* **exit** : Nos desconectamos de servidor (0,5 puntos).

**Ejercicio 2:** *(3,5 puntos)*. Firma y conexión a Filezilla. En este ejercicio el usuario deberá indicar por consola la ruta de un fichero. A continuación, se deberán implementar las siguientes opciones:

* **Opción 1:** Subir tanto el fichero original como el fichero con la firma al servidor localhost de Filezilla Server configurado los datos de conexión en el código (1,5 puntos).
* **Opción 2:** Descargar tanto el fichero original como el fichero firma. Se deberá comprobar que la firma es correcta indicándolo por la consola (1,5 puntos).
* **Opción 3:** Subir el fichero con la clave pública utilizada para la firma (0,5 puntos).

**Ejercicio 3:** *(3,5 puntos)*. SMTP, Gmail y cifrado de ficheros. Primeramente, el usuario deberá de especificar los siguientes datos a través de la consola:

1. E-mail de destino.
2. Ruta de un fichero cualquiera del equipo (crear un .txt con texto).

Utilizando el protocolo SMTP y los servidores de Google, queremos enviar un e-mail con el siguiente contenido:

* **Línea 1 del e-mail:** Hola, el código hash del texto del fichero requerido es: [hash MD5 del fichero dado la ruta] (1 punto).
* **Línea 2 del e-mail:** y contenía el siguiente mensaje el cual cifro: [mensaje cifrado AES] (1 punto).
* A continuación, se enviará el email (1,5 puntos).

**Nota:** Los datos de conexión se podrán dejar en blanco si no se desea entregar la cuenta de Gmail. Hay que indicar claramente en el código donde introducirlos.