

# SISTEMAS DISTRIBUIDOS

## Practica 1

Introducción a la tecnología de Sockets Java

Grado en ingeniería informática

Francisco Joaquín Murcia Gómez 48734281H

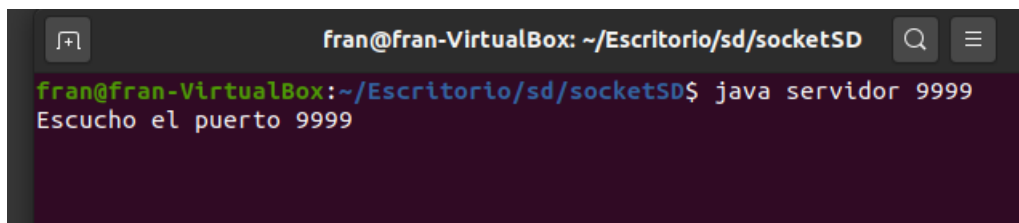
Grupo 1

## Introducción

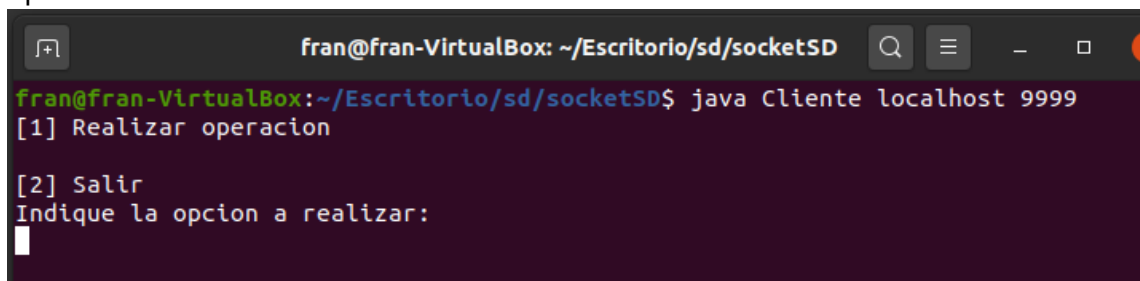
El Programa en cuestión es una estructura de cliente servidor, el cliente manda una petición de conexión y al conectarse el servidor le “envía” el programa que es una calculadora que o bien te suma o bien te multiplica, en mi caso he añadido una función que te hace el cuadrado de un número.

## Funcionamiento del programa

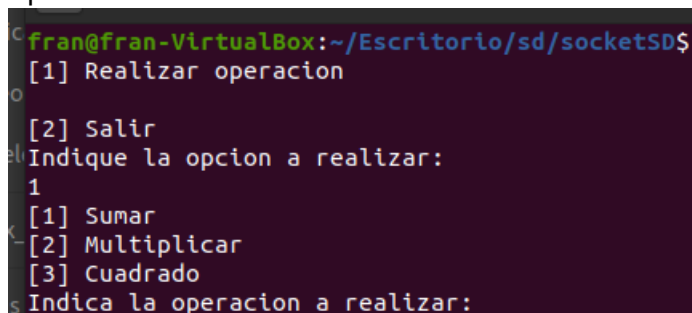
Primero tenemos que ejecutar el servidor y le pasamos el puerto para abrirlo, por argumento, al instante el servidor estará en modo escucha.

A terminal window titled 'fran@fran-VirtualBox: ~/Escritorio/sd/socketSD'. The prompt is 'fran@fran-VirtualBox:~/Escritorio/sd/socketSD\$'. The command entered is 'java servidor 9999'. The output is 'Escucho el puerto 9999'.

A continuación, ejecutamos el servidor y le decimos por argumentos que nos queremos conectar por el puerto local y colocamos el mismo puerto que el servidor, el cliente envía una petición y el servidor envía los datos al cliente y aparecerá el menú

A terminal window titled 'fran@fran-VirtualBox: ~/Escritorio/sd/socketSD'. The prompt is 'fran@fran-VirtualBox:~/Escritorio/sd/socketSD\$'. The command entered is 'java Cliente localhost 9999'. The output is '[1] Realizar operacion', '[2] Salir', and 'Indique la opcion a realizar:' followed by a cursor.

Cuando interactúas con el programa al darle a la opción 1 sale el menú de operaciones:

A terminal window titled 'fran@fran-VirtualBox: ~/Escritorio/sd/socketSD'. The prompt is 'fran@fran-VirtualBox:~/Escritorio/sd/socketSD\$'. The command entered is 'java Cliente localhost 9999'. The output is '[1] Realizar operacion', '[2] Salir', and 'Indique la opcion a realizar:'. The user has entered '1'. The output is '[1] Sumar', '[2] Multiplicar', '[3] Cuadrado', and 'Indica la operacion a realizar:'.

en la terminal del servidor indica que esta sirviendo al cliente.

A continuación, se puede ver el funcionamiento del programa

```
fran@fran-VirtualBox: ~/Escritorio/sd/socketSD$ java C
[1] Realizar operacion
[2] Salir
Indique la opcion a realizar:
1
[1] Sumar
[2] Multiplicar
[3] Cuadrado
Indica la operacion a realizar:
3
Introduzca el primer operando [0-9]:
5
El resultado es: 25
Desea realizar otra operacion? [s,n]:
s
[1] Sumar
[2] Multiplicar
[3] Cuadrado
Indica la operacion a realizar:
1
Introduzca el primer operando [0-9]:
8
Introduzca el segundo operando [0-9]:
4
El resultado es: 12
Desea realizar otra operacion? [s,n]:
n
Conexion cerrada.
fran@fran-VirtualBox:~/Escritorio/sd/socketSD$
```

En el servidor aparece el registro de las operaciones:

```
fran@fran-VirtualBox:~/Escritorio/sd/socketSD$ j
Escucho el puerto 5000
Sirviendo cliente...
SRV: La operacion es: cuadr
SRV: El operando 1 es 5 y el operando 2 es 0
SRV: El resultado es: 25
SRV: La operacion es: suma
SRV: El operando 1 es 8 y el operando 2 es 4
SRV: El resultado es: 12
SRV: La operacion es: fin
fran@fran-VirtualBox:~/Escritorio/sd/socketSD$
```

## Mi modificación

Como ya he comentado en la introducción he añadido una opción mas al menú, esta te hace el cuadrado del numero que introduzcas y a diferencia de las otras opciones solo te pide un solo operador.

En el servidor he añadido la función siguiente:

```
public int cuadrado(int p_a)//realiza el cuadrado
{
    return p_a*p_a;
}
```

He modificado la función realizarOperacion de la siguiente manera:

```
System.out.println("SRV: La operacion es: " + operacion[0]);
if(operacion.length != 1)
{
    System.out.println("SRV: El operando 1 es " + operacion[1] + " y el operando 2 es " + operacion[2]);
    if(operacion[0].compareTo("suma")==0)
    {
        res = sumar(Integer.parseInt(operacion[1]),Integer.parseInt(operacion[2]));
    }
    else
    {
        if(operacion[0].compareTo("mult")==0)
        {
            res = multiplicar(Integer.parseInt(operacion[1]),Integer.parseInt(operacion[2]));
        }
        else if(operacion[0].compareTo("cuadr")==0)//realizamos el cuadrado del numero
        {
            res = cuadrado(Integer.parseInt(operacion[1]));
        }
        else{
            res = -1;
        }
    }
}
System.out.println("SRV: El resultado es: " + res);
}else
{
    res = -1;
}
return (res);
```

He añadido un if else controlando la opción del menú 3, esta llama a la función cuadrado.

En el cliente he añadido la función siguiente para solo solicitar un operador al usuario:

```
public int pedirUnNumero(String p_operacion, int p_resultado, String p_Cadena, Socket p_Socket_Con_Servidor)
{
    int op1 = 10;
    int op2 = 10;
    InputStreamReader isr = new InputStreamReader(System.in);
    BufferedReader br = new BufferedReader (isr);
    try
    {
        while (op1 < 0 || op1 > 9)
        {
            System.out.println("Introduzca el primer operando [0-9]:");
            op1 = Integer.parseInt(br.readLine());
        }
        p_Cadena = p_operacion + "," + op1 + "," + 0;
        escribeSocket (p_Socket_Con_Servidor, p_Cadena);
        p_Cadena = "";
        p_Cadena = leeSocket (p_Socket_Con_Servidor, p_Cadena);
        p_resultado = Integer.parseInt(p_Cadena);
    }
    catch(Exception e)
    {
        System.out.println("Error: " + e.toString());
    }
    return p_resultado;
}
```

A parte, he modificado la función pedir operación para añadir la opción 3 al menú:

```
operacion = 0;
while (operacion !=1 && operacion !=2 && operacion !=3)
{
    System.out.println("[1] Sumar");
    System.out.println("[2] Multiplicar");
    System.out.println("[3] Cuadrado");//para realizar el cuadrado de un numero
    System.out.println("Indica la operacion a realizar: ");
    operacion = Integer.parseInt(br.readLine());
}
if (operacion == 1){
    op = "suma";
    resultado = pedirNumeros(op, resultado, Cadena, skCliente);
}
else if(operacion == 2){
    op = "mult";
    resultado = pedirNumeros(op, resultado, Cadena, skCliente);
}
else{
    op ="cuadr";
    resultado = pedirUnNumero(op, resultado, Cadena, skCliente);//pido solo un numero
}
resp='x';
```

## Extras

He probado el programa con un servidor concurrente, el funcionamiento es el mismo, pero al cerrar el cliente el servidor finaliza la conexión, pero no termina la ejecución:

```
SRV: La operacion es: fin
fran@fran-VirtualBox:~/Escritorio/sd/socketSD$ java ServidorConcurrente 9999
Escucho el puerto 9999
Sirviendo cliente...
SRV: La operacion es: mult
SRV: El operando 1 es 3 y el operando 2 es 3
SRV: El resultado es: 9
SRV: La operacion es: fin
█
```