

### Calculadora cliente-servidor:

Para la implementación de una calculadora usando un paradigma de programación cliente-servidor, hemos hecho dos ficheros (el cliente y el servidor):

1. **Cliente.java:** En este archivo, el usuario escribe una operación (por ejemplo: 3+4, o cualquier otra) que se le envía al servidor. Éste, después de realizar dicha operación, devuelve el resultado y el usuario puede ver dicho resultado en el cliente.
2. **Servidor.java:** En este archivo se recibe el mensaje de los clientes (ya que acepta varios clientes) debido a que utiliza un hilo por cada cliente nuevo que se conecta al servidor. Dicho hilo, será seleccionado de un pool de threads (del tamaño de los cores que haya disponible en el sistema) que hay en el servidor. Para la selección de la operación, hacemos uso del método `split("operador")` de `String` para dividir el mensaje del usuario en "operando 1", "operador" y "operando 2". A continuación, se realiza la operación seleccionada por el usuario en un `switch` y se devuelve el resultado. Si el primer número es negativo, tenemos una bandera llamada "primeroNegativo" que se establecerá a "true" en ese caso y volverá a reiterar el `split("operador")` para seleccionar de nuevo los operandos de la operación. En caso de dividir por cero, el servidor devolverá "Infinito" y en caso de la división 0/0, el servidor devolverá "Indeterminación".

Aplicaciones ▾ Lugares ▾ Terminal ▾

jue, 5 de abr, 12:37

root@Kali: ~/Universidad/PPD/Prácticas/Práctica 4/PPD4\_1

Archivo Editar Ver Buscar Terminal Ayuda

root@Kali:~/Universidad/PPD/Prácticas/Práctica 4/PPD4\_1# java servidor.Servidor  
El servidor de la calculadora esta listo.  
Espera una conexion.  
Un cliente ha hecho una conexion.  
Espera una conexion. decrementAndget();  
Un cliente ha hecho una conexion.  
Espera una conexion.  
Un cliente ha hecho una conexion. ring[] args) {  
Espera una conexion. // El servidor de la calculadora esta lista  
Un cliente ha hecho una conexion. // socket stream para aceptar las conexiones  
Espera una conexion. // ServerSocket para aceptar las conexiones  
Un cliente ha hecho una conexion. // El servidor de la calculadora esta lista  
Espera una conexion. // socket stream para aceptar las conexiones  
Un cliente ha hecho una conexion. // El servidor de la calculadora esta lista  
Espera una conexion. // El servidor de la calculadora esta lista  
// ahora: no hay necesidad de leer una peticion o la pet  
System.out.println("Un cliente ha hecho una conexion.");  
MiSocketStream miSocketDatos = new MiSocketStream(miSo  
System.out.println("Espera una conexion");  
// ahora manda la respuesta al solicitante  
miSocketDatos.close();  
} // fin de while  
} // fin de try  
catch (Exception ex) {  
ex.printStackTrace();  
}  
} // fin de main  
} // fin de clase

root@Kali:~/Universidad/PPD/Prácticas/Práctica 4/PPD4\_1

Archivo Editar Ver Buscar Terminal Ayuda

root@Kali:~/Universidad/PPD/Prácticas/Práctica 4/PPD4\_1# java cliente.Cliente  
Bienvenido al cliente de la calculadora.  
  
Bienvenido. Eres el cliente número 1  
Introduce la operacion a realizar: 3.5+6.2  
El resultado es 9.7  
root@Kali:~/Universidad/PPD/Prácticas/Práctica 4/PPD4\_1# java cliente.Cliente  
Bienvenido al cliente de la calculadora.  
  
Bienvenido. Eres el cliente número 1  
Introduce la operacion a realizar: 3.1-6  
El resultado es -2.9  
root@Kali:~/Universidad/PPD/Prácticas/Práctica 4/PPD4\_1# java cliente.Cliente  
Bienvenido al cliente de la calculadora.  
  
Bienvenido. Eres el cliente número 1  
Introduce la operacion a realizar: -2.6+6.1  
El resultado es 3.4999999999999996  
root@Kali:~/Universidad/PPD/Prácticas/Práctica 4/PPD4\_1# java cliente.Cliente  
Bienvenido al cliente de la calculadora.  
  
Bienvenido. Eres el cliente número 1  
Introduce la operacion a realizar: 1.9+3.5  
El resultado es -1.6  
root@Kali:~/Universidad/PPD/Prácticas/Práctica 4/PPD4\_1# java cliente.Cliente  
Bienvenido al cliente de la calculadora.  
  
Bienvenido. Eres el cliente número 1  
Introduce la operacion a realizar: 3.3\*6.2  
El resultado es 32.86  
root@Kali:~/Universidad/PPD/Prácticas/Práctica 4/PPD4\_1#

