

Sockets TCP en Java

Ejercicios

Ejercicio 1: Servidor de Factoriales

Implementa un sistema cliente-servidor donde el cliente envíe un número entero al servidor, y este le devuelva el factorial del número.

- El **servidor** debe ser una aplicación monohilo que escuche conexiones en un puerto específico. Una vez que un cliente se conecta, el servidor recibe el número, calcula su factorial y envía el resultado de vuelta. Si el número recibido es negativo, el servidor devolverá -1.
- El **cliente** debe solicitar al usuario un número entero mediante la consola. Una vez obtenido el número, se conectará al servidor, enviará el dato y mostrará en la consola el resultado recibido.

$$n! \equiv n(n-1) \cdots 2 \cdot 1$$

Ejercicio 2: Servidor de Fecha y Hora

Implementa un servidor que utilizará el protocolo TCP para enviar la fecha y hora actuales a los clientes que se conecten.

El servidor al recibir una conexión de un cliente, debe obtener la fecha y hora actuales y enviarlas al cliente en formato legible (por ejemplo, `dd/MM/yyyy HH:mm:ss`).

Ejercicio 3: Adivinar el número

Implementa un juego interactivo de adivinanza de números utilizando sockets TCP. El servidor generará un número aleatorio y los clientes se conectarán para intentar adivinarlo. El servidor proporcionará pistas como "mayor" o "menor" hasta que el cliente acierte el número.

Ejercicio 4: Chat Simple

Desarrolla una aplicación de chat sencilla en la que un cliente y un servidor puedan comunicarse enviándose mensajes de texto. La comunicación debe continuar hasta que cualquiera de las dos partes escriba el mensaje `"salir"`, momento en el que la conexión se cerrará.

Ejercicio 5: Operaciones Matemáticas

Implementa una aplicación cliente-servidor utilizando sockets TCP en la que el cliente envía un objeto con los datos necesarios para realizar una operación matemática. El servidor procesa la solicitud, realiza la operación indicada y devuelve un objeto con el resultado y un indicador de si la operación se realizó correctamente.

Cliente:

- Crea un objeto que contenga los siguientes campos:
 - **operacion**: un String que indica la operación a realizar (**"suma"**, **"resta"**, **"multiplicacion"** o **"division"**).
 - **operando1**: un número (entero o decimal).
 - **operando2**: un número (entero o decimal).
- Envía el objeto al servidor a través del socket.
- Recibe un objeto de respuesta que contiene:
 - **exito**: un booleano que indica si la operación se realizó con éxito (**true** o **false**).
 - **resultado**: el resultado de la operación si fue exitosa.
- Muestra en la consola el mensaje recibido del servidor.

Servidor:

- Recibe el objeto enviado por el cliente.
- Realiza la operación especificada en el campo **operacion**:
 - **"suma"**: suma los dos operandos.
 - **"resta"**: resta los dos operandos.
 - **"multiplicacion"**: multiplica los dos operandos.
 - **"division"**: divide los dos operandos, pero debe retornar **exito = false** si el divisor (**operando2**) es 0.
- Envía un objeto de respuesta al cliente con los siguientes campos:
 - **exito**: **true** si la operación fue válida, **false** si hubo algún error (por ejemplo, una división por cero).
 - **resultado**: el valor de la operación si **exito = true**, o **null** en caso contrario.