

Tarea Individual 22 - Transferencia de objetos en SOCKETS UDP y TCP

Objetivo

El objetivo de esta actividad es desarrollar un programa que permita la transferencia de objetos entre un cliente y un servidor utilizando los protocolos TCP y UDP. Los alumnos deberán reutilizar y adaptar las soluciones proporcionadas como base para crear un programa funcional con las especificaciones requeridas.

Descripción de la Tarea

Con base en las soluciones de ejemplo, se solicita a los alumnos realizar las siguientes modificaciones y desarrollos:

- **Definición de la clase de datos:**
 - Utilizar la clase `Persona` proporcionada, que ya implementa la interfaz `Serializable`.
 - Asegurarse de que los atributos `nombre` y `edad` sean utilizados correctamente.
- **Protocolo TCP:**
 - Modificar las clases `TCPCliente` y `TCPServidor` proporcionadas para realizar lo siguiente:
 - El servidor debe enviar inicialmente un objeto `Persona` con valores predefinidos al cliente.
 - El cliente debe recibir este objeto, modificar sus valores (nombre y edad), y enviarlo de vuelta al servidor.
 - El servidor debe recibir el objeto modificado y mostrar los nuevos valores en la consola.
- **Protocolo UDP:**
 - Adaptar las clases `UDPCliente` y `UDPServidor` proporcionadas para que:
 - El cliente envíe un objeto `Persona` al servidor utilizando un datagrama.
 - El servidor reciba este objeto, lo deserialice y muestre sus valores en la consola.
- **Requisitos adicionales:**
 - Utilizar estructuras `try-with-resources` para gestionar automáticamente los recursos (sockets, streams).
 - Manejar las posibles excepciones (como `IOException`, `ClassNotFoundException`) e incluir mensajes de error claros en la consola.
 - Comentar el código de manera clara para explicar las funcionalidades principales.

La clase
Persona:

```
package data;

import java.io.Serializable;

public class Persona implements Serializable{

    private static final long serialVersionUID=1L;
    private String nombre;
    private int edad;

    public Persona() {
        super();
    }
    public Persona(String nombre,int edad){
        super();
        this.nombre=nombre;
        this.edad=edad;
    }
    public String getNombre() {
        return nombre;
    }
    public void setNombre(String nombre) {
        this.nombre = nombre;
    }
    public int getEdad() {
        return edad;
    }
    public void setEdad(int edad) {
        this.edad = edad;
    }

    @Override
    public String toString() {
        return "Persona{" + "nombre=" + nombre + ", edad=" + edad + '}';
    }
}
```

Intercambio de objetos mediante TCP:

```
public class TCPCliente {

    public static void main(String[] args) {
        String host = "localhost";
        int puerto = 12345;

        try {
            Socket socket = new Socket(host, puerto);
            ObjectOutputStream salida = new ObjectOutputStream(socket.getOutputStream());
            ObjectInputStream entrada = new ObjectInputStream(socket.getInputStream());
        } {
            // Recibir objeto inicial del servidor
            Persona personaRecibida = (Persona) entrada.readObject();
            System.out.println("Persona recibida del servidor: " + personaRecibida);

            // Modificar el objeto
            personaRecibida.setNombre("Maria");
            personaRecibida.setEdad(30);

            // Enviar objeto modificado
            System.out.println("Enviando persona modificada: " + personaRecibida);
            salida.writeObject(personaRecibida);
        } catch (IOException | ClassNotFoundException e) {
            System.err.println("Error en la comunicaci3n: " + e.getMessage());
        }
    }
}
```

Capturas de la ejecución del programa:

- Envío y recepción de objetos mediante UDP.

```

public class UDPServidor {

    public static void main(String[] args) {
        int puerto = 12345;
        byte[] buffer = new byte[1024];

        try (DatagramSocket socket = new DatagramSocket(puerto)) {
            System.out.println("Servidor UDP iniciado en puerto " + puerto);

            while (true) {
                try {
                    // Preparar el paquete para recibir datos
                    DatagramPacket paquete = new DatagramPacket(buffer, buffer.length);
                    socket.receive(paquete);

                    // Deserializar el objeto
                    try (ByteArrayInputStream bais = new ByteArrayInputStream(paquete.getData());
                        ObjectInputStream ois = new ObjectInputStream(bais)) {

                        Persona personaRecibida = (Persona) ois.readObject();
                        System.out.println("Persona recibida: " + personaRecibida);
                    }
                } catch (IOException | ClassNotFoundException e) {
                    System.err.println("Error al procesar el paquete: " + e.getMessage());
                }
            }
        } catch (IOException e) {
            System.err.println("Error al iniciar el servidor: " + e.getMessage());
        }
    }
}

```

```

public class UDPCliente {

    public static void main(String[] args) {
        String host = "localhost";
        int puerto = 12345;

        try (DatagramSocket socket = new DatagramSocket()) {
            InetAddress direccion = InetAddress.getByName(host);

            // Crear y serializar el objeto Persona
            Persona persona = new Persona("Carlos", 35);

            try (ByteArrayOutputStream baos = new ByteArrayOutputStream();
                ObjectOutputStream oos = new ObjectOutputStream(baos)) {

                oos.writeObject(persona);
                byte[] datos = baos.toByteArray();

                // Enviar el objeto serializado
                DatagramPacket paquete = new DatagramPacket(
                    datos, datos.length, direccion, puerto);
                socket.send(paquete);

                System.out.println("Persona enviada: " + persona);
            }
        } catch (IOException e) {
            System.err.println("Error en la comunicación: " + e.getMessage());
        }
    }
}

```

Capturas de la ejecución del programa:

```
PS C:\Users\heily\OneDrive\Escritorio\github\DamPrivado\java\Tarea22\src> java -cp . data.UDPServidor
Servidor UDP iniciado en puerto 12345
Persona recibida: Persona{nombre=Carlos, edad=35}
|
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWin

PS C:\Users\heily> cd C:\Users\heily\OneDrive\Escritorio\github\DamPrivado\java\Tarea22\src
PS C:\Users\heily\OneDrive\Escritorio\github\DamPrivado\java\Tarea22\src> java -cp . data.TCPCliente
Persona recibida del servidor: Persona{nombre=Juan, edad=25}
Enviando persona modificada: Persona{nombre=María, edad=30}
PS C:\Users\heily\OneDrive\Escritorio\github\DamPrivado\java\Tarea22\src> java -cp . data.UDPCliente
Persona enviada: Persona{nombre=Carlos, edad=35}
PS C:\Users\heily\OneDrive\Escritorio\github\DamPrivado\java\Tarea22\src> |
```