

Bloque 5. Paradigmas de la Programación

Tarea 3: Primos con sockets TCP

Enunciado:

3.- Crear una aplicación distribuida con sockets TCP, a la que se le envíe un número y nos devuelva como resultado si el número es primo o no.

A tener en cuenta:

- Tendremos un Cliente y un Servidor, ambos clase Main.
- Se implementará con Socket Stream.
- Pasos:
 1. Creamos el ServerSocket (en servidor).
 2. Abrimos el socket.
 3. Creamos los streams de entrada y salida DataInputStream y DataOutputStream
 4. Cerramos streams y sockets.

Código Servidor:

El servidor primero lee y luego escribe.

```
import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.IOException;
import java.net.ServerSocket;
import java.net.Socket;
import java.time.LocalDate;
import java.time.Period;

public class Servidor {

    public static void main(String[] args) {

        ServerSocket servidor;
        Socket conexion;
        DataOutputStream salida;
        DataInputStream entrada;

        int n, esPrimo = 0;

        try {
            servidor = new ServerSocket(5000); //Creamos un ServerSocket en el Puerto 5000, aquí es donde
            escucharemos al cliente
            System.out.println("Servidor Arrancado....");

            conexion = servidor.accept(); //Esperamos una conexión
            System.out.println("Conexión establecida desde: " + conexion.getInetAddress().getHostName());
            entrada = new DataInputStream(conexion.getInputStream()); //Abrimos los canales de E/S
            salida = new DataOutputStream(conexion.getOutputStream());

            //Leemos los mensajes del
            n = entrada.readInt();
            System.out.println("Contenido del mensaje recibido . . . Número a evaluar: "+ n );
```

Bloque 5. Paradigmas de la Programación

Tarea 3: Primos con sockets TCP

```
if (n <= 1) { // Si es 1 o menor, directamente no será primo
    esPrimo = 0;
} else { // Si es mayor, realizo la comprobación
    for (int i = 2; i <= n / 2; i++) {
        if ((n % i) == 0) {
            esPrimo = 0;
        } else {
            esPrimo = 1;
        }
    }
}

System.out.println("Contenido del mensaje a enviar: " + esPrimo);

salida.writeInt(esPrimo); //Le respondemos
entrada.close(); //Cerramos los flujos de entrada y salida
salida.close();
conexion.close(); //Y cerramos la conexión

} catch (IOException e) {
}
}
```

Código Cliente:

El cliente primero escribe y luego lee.

```
import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.IOException;
import java.net.InetAddress;
import java.net.Socket;
import java.util.Scanner;

public class Cliente {

    public static void main(String[] args) {
        Socket cliente;
        DataInputStream entrada;
        DataOutputStream salida;

        int n;
        int esPrimo; //1 es primo, 0 no es primo
        Scanner sc = new Scanner(System.in);

        try {
            cliente = new Socket(InetAddress.getLocalHost(), 5000); //Creamos el socket para conectarnos al
            puerto 5000 del servidor
            entrada = new DataInputStream(cliente.getInputStream()); //Creamos los canales de E/S
            salida = new DataOutputStream(cliente.getOutputStream());
```

Bloque 5. Paradigmas de la Programación

Tarea 3: Primos con sockets TCP

```
System.out.println("Vamos a enviar un mensaje . . .");

System.out.println("**Introduce un número entero: **");
n = sc.nextInt();
System.out.println("Vamos a comprobar si " + n + " es primo");

//Enviamos el numero al servidor
salida.writeInt(n);

esPrimo = entrada.readInt(); //Leemos la respuesta
String resp;

if (esPrimo == 1){
    resp = " Es PRIMO";
}else{
    resp = " NO es PRIMO";
}
System.out.println("El servidor nos devuelve si es primo . . . El número " + n + resp);
entrada.close(); //Cerramos los flujos de entrada y salida
salida.close();
cliente.close(); //Cerramos la conexión
} catch (IOException e) {
    System.out.println("Error: " + e.getMessage());
}
}
```

Resultado:

Desde Cliente...	Desde Servidor...
run: Vamos a enviar un mensaje . . . **Introduce un número entero: ** 79 Vamos a comprobar si 79 es primo El servidor nos devuelve si es primo . . . El número 79 Es PRIMO BUILD SUCCESSFUL (total time: 11 seconds)	run: Servidor Arrancado.... Conexión establecida desde: PC-TOSH Contenido del mensaje recibido . . . Número a evaluar: 79 Contenido del mensaje a enviar: 1 BUILD SUCCESSFUL (total time: 21 seconds)