

UNIVERSIDAD MAYOR DE SAN ANDRES
FACUTLAD DE CIENCIAS PURAS Y NATURALES
CARRERA: INFOMATICA

MATERIA: LAB-273 LABORATORIO DE TELEMATICA

UNIVERSITARIO: Meneces Zelaya David Josue.

CI: 13119000 LP.

DOCENTE: Vargas Blaccut Roberto

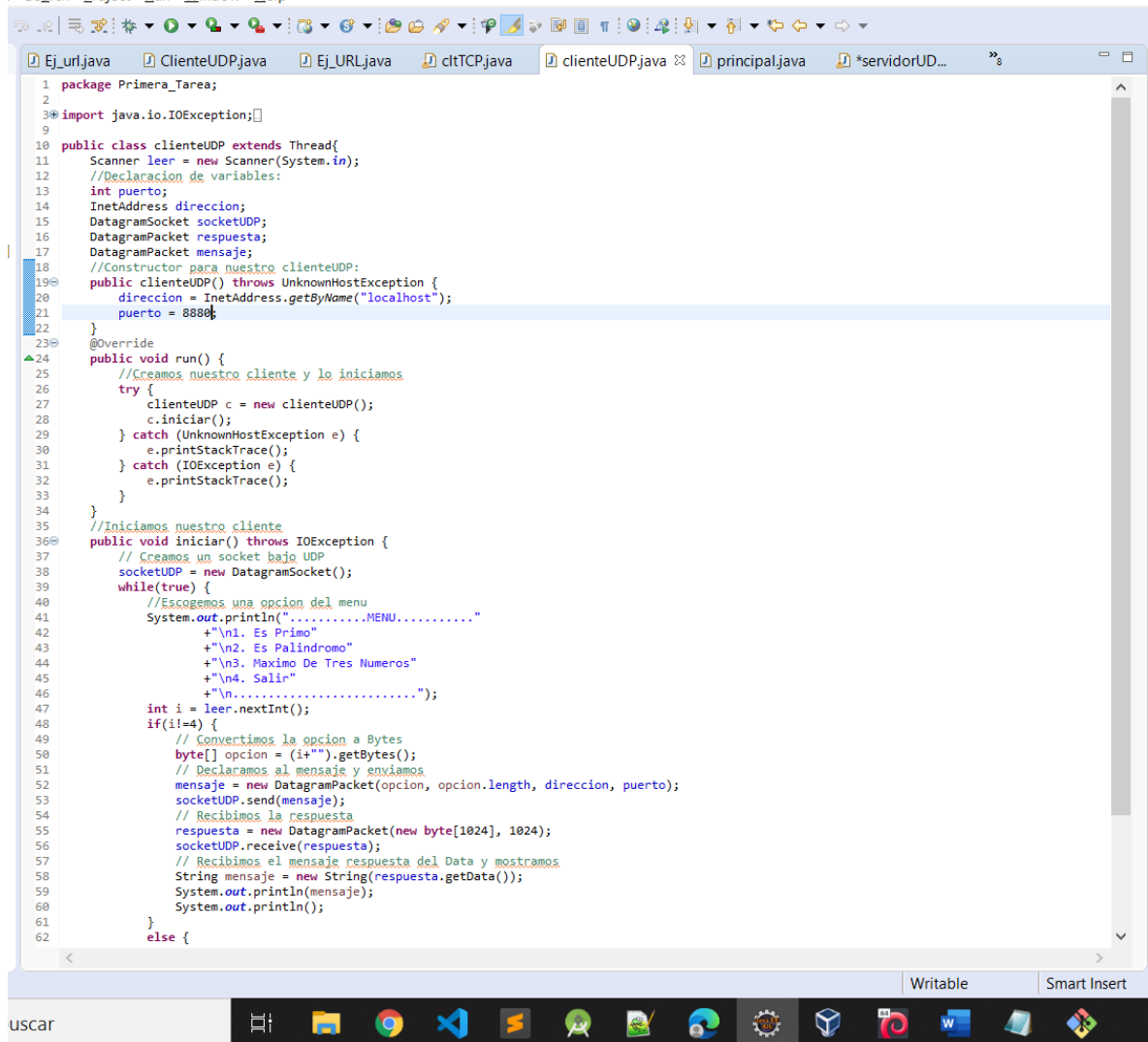
PARALELO: "C"

PRACTICA AUXILIATURA LAB-273
SERVIDOR UDP – TAREA

Cliente UDP:

nera_Tarea/clienteUDP.java - Eclipse IDE

: Search Project Run Window Help



```
1 package Primera_Tarea;
2
3 import java.io.IOException;
4
5
6
7
8
9
10 public class clienteUDP extends Thread{
11     Scanner leer = new Scanner(System.in);
12     //Declaracion de variables:
13     int puerto;
14     InetAddress direccion;
15     DatagramSocket socketUDP;
16     DatagramPacket respuesta;
17     DatagramPacket mensaje;
18     //Constructor para nuestro clienteUDP:
19     public clienteUDP() throws UnknownHostException {
20         direccion = InetAddress.getByName("localhost");
21         puerto = 8880;
22     }
23
24     @Override
25     public void run() {
26         //Creamos nuestro cliente y lo iniciamos
27         try {
28             clienteUDP c = new clienteUDP();
29             c.iniciar();
30         } catch (UnknownHostException e) {
31             e.printStackTrace();
32         } catch (IOException e) {
33             e.printStackTrace();
34         }
35     }
36     //Iniciamos nuestro cliente
37     public void iniciar() throws IOException {
38         // Creamos un socket bajo UDP
39         socketUDP = new DatagramSocket();
40         while(true) {
41             //Escogemos una opcion del menu
42             System.out.println(".....MENU.....");
43             +"\n1. Es Primo"
44             +"\n2. Es Palindromo"
45             +"\n3. Maximo De Tres Numeros"
46             +"\n4. Salir"
47             +"\n.....";
48             int i = leer.nextInt();
49             if(i!=4) {
50                 // Convertimos la opcion a Bytes
51                 byte[] opcion = (i+"").getBytes();
52                 // Declaramos al mensaje y enviamos
53                 mensaje = new DatagramPacket(opcion, opcion.length, direccion, puerto);
54                 socketUDP.send(mensaje);
55                 // Recibimos la respuesta
56                 respuesta = new DatagramPacket(new byte[1024], 1024);
57                 socketUDP.receive(respuesta);
58                 // Recibimos el mensaje respuesta del Data y mostramos
59                 String mensaje = new String(respuesta.getData());
60                 System.out.println(mensaje);
61                 System.out.println();
62             }
63         }
64     }
65 }
```

Servidor UDP:

```

1 package Primera_Tarea;
2
3 import java.io.IOException;
4
5
6
7
8
9 public class servidorUDP extends Thread {
10     Scanner leer = new Scanner(System.in);
11     // Declaracion de variables:
12     int puerto;
13     DatagramSocket socketUDP;
14     DatagramPacket mensaje;
15     DatagramPacket respuesta;
16
17     // Constructor para nuestro server
18     public servidorUDP() {
19         puerto = 8880;
20     }
21
22     @Override
23     public void run() {
24         try {
25             // Creamos el socket del server
26             socketUDP = new DatagramSocket(puerto);
27             while (true) {
28                 // Declaramos mensaje peticion y lo recibimos
29                 mensaje = new DatagramPacket(new byte[1024], 1024);
30                 socketUDP.receive(mensaje);
31                 String cadenasrespuesta = "";
32                 // Convertimos la opcion de peticion enviada a un entero
33                 String opcion = new String(mensaje.getData());
34                 int i = Integer.parseInt(opcion.substring(0,1));
35                 switch (i) {
36                     case 1:
37                         System.out.println("Ingrese numero:");
38                         int num = leer.nextInt();
39                         int cont = 0;
40                         if (num > 1) {
41                             for (int j = 2; j <= num-1; j++)
42                                 if (num % j == 0)
43                                     cont++;
44                             if (cont == 0)
45                                 cadenasrespuesta = "es primo";
46                             else
47                                 cadenasrespuesta = "no es primo";
48                         }
49                         else
50                             cadenasrespuesta = "no es primo";
51                         break;
52                     case 2:
53                         System.out.println("Ingrese la cadena");
54                         String cadena = leer.next();
55                         String cadaux = "";
56                         for (int j = cadena.length()-1; j >= 0; j--)
57                             cadaux = cadaux + cadena.charAt(j);
58                         if (cadena.compareToIgnoreCase(cadaux) == 0)
59                             cadenasrespuesta = "Es palindromo";
60                         else
61                             cadenasrespuesta = "No es palindromo";
62

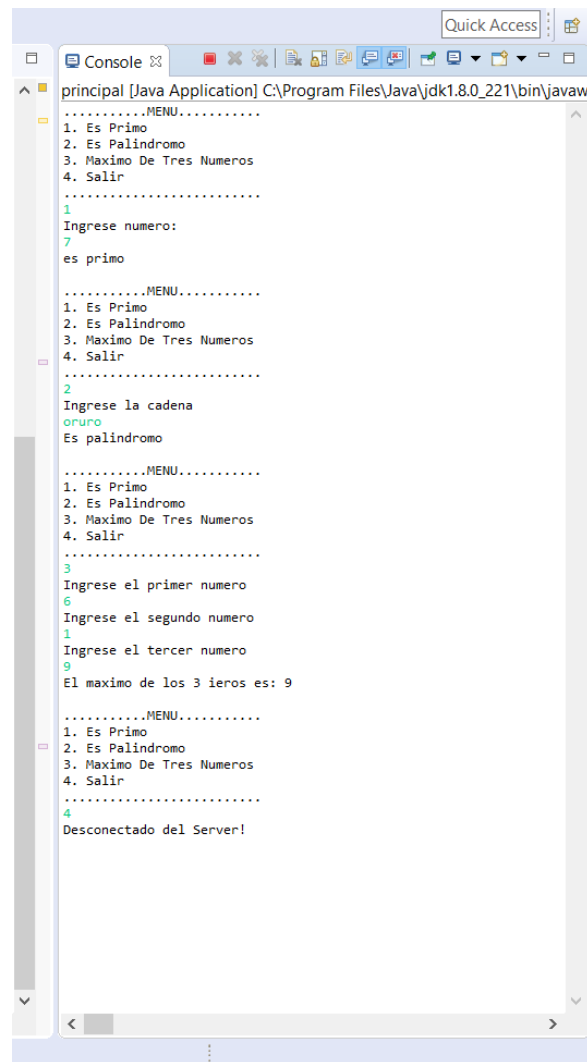
```

```

58         if (cadena.compareToIgnoreCase(cadaux) == 0)
59             cadenasrespuesta = "Es palindromo";
60         else
61             cadenasrespuesta = "No es palindromo";
62         break;
63     case 3:
64         System.out.println("Ingrese el primer numero");
65         int a = leer.nextInt();
66         System.out.println("Ingrese el segundo numero");
67         int b = leer.nextInt();
68         System.out.println("Ingrese el tercer numero");
69         int c = leer.nextInt();
70         int resp = Math.max(a, Math.max(b, c));
71         cadenasrespuesta = "El mayor de los 3 numeros es: "+resp;
72         break;
73     default:
74         break;
75     }
76     byte mensajeEnviar[] = new byte[1024];
77     // Convertimos cadenasrespuesta a bytes y lo ponemos en el datagrama respuesta
78     mensajeEnviar = cadenasrespuesta.getBytes();
79     respuesta = new DatagramPacket(mensajeEnviar, mensajeEnviar.length, mensaje.getAddress(), mensaje.getPort());
80     // Enviamos la respuesta a la peticion del cliente
81     socketUDP.send(respuesta);
82 }
83 } catch (Exception e) {
84     e.printStackTrace();
85 }
86 }
87 // Funcion para finalizar la conexion
88 public void end() {
89     try {
90         socketUDP.close();
91         System.out.println("Conexion finalizada");
92     } catch (Exception e) {
93         e.printStackTrace();
94     }
95 }
96 }

```

Screenshots de consola del problema:



```
principal [Java Application] C:\Program Files\Java\jdk1.8.0_221\bin\javaw
.....MENU.....
1. Es Primo
2. Es Palindromo
3. Maximo De Tres Numeros
4. Salir
.....
1
Ingrese numero:
7
es primo

.....MENU.....
1. Es Primo
2. Es Palindromo
3. Maximo De Tres Numeros
4. Salir
.....
2
Ingrese la cadena
oruro
Es palindromo

.....MENU.....
1. Es Primo
2. Es Palindromo
3. Maximo De Tres Numeros
4. Salir
.....
3
Ingrese el primer numero
6
Ingrese el segundo numero
1
Ingrese el tercer numero
9
El maximo de los 3 ieros es: 9

.....MENU.....
1. Es Primo
2. Es Palindromo
3. Maximo De Tres Numeros
4. Salir
.....
4
Desconectado del Server!
```