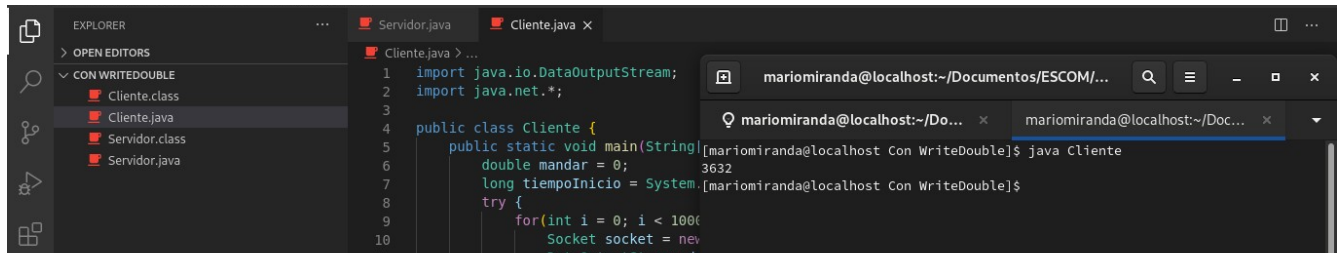


Miranda Sandoval Mario Alberto

4CM5.

Usando WriteDouble.

Se puede observar que con Write Double tarda un total de 3632 mili segundos.

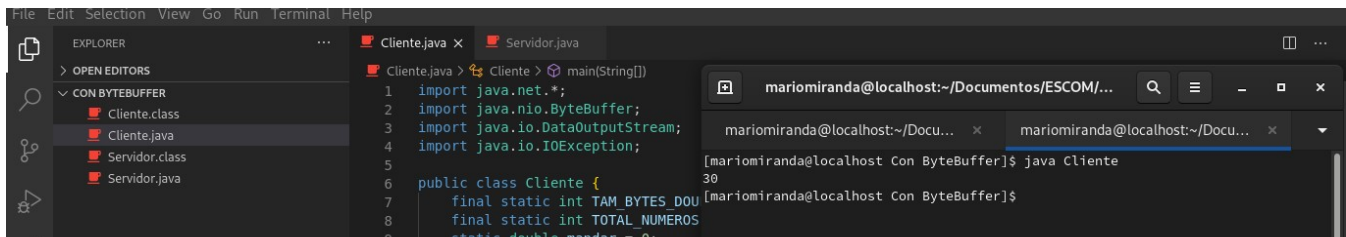


```
1 import java.io.DataOutputStream;
2 import java.net.*;
3
4 public class Cliente {
5     public static void main(String[] args) {
6         double mandar = 0;
7         long tiempoInicio = System.currentTimeMillis();
8         try {
9             for(int i = 0; i < 1000; i++) {
10                 Socket socket = new Socket("localhost", 8080);
11                 DataOutputStream dos = new DataOutputStream(socket.getOutputStream());
12                 dos.writeDouble(mandar);
13                 dos.flush();
14             }
15         } catch (IOException e) {
16             e.printStackTrace();
17         }
18     }
19 }
```

```
[mariomiranda@localhost Con WriteDouble]$ java Cliente
3632
[mariomiranda@localhost Con WriteDouble]$
```

Usando ByteBuffer.

Se puede observar que empaquetando los double en un arreglo de bytes se manda de una manera mucha más rápida, con un increíble tiempo de 30 mili segundos.



```
1 import java.net.*;
2 import java.nio.ByteBuffer;
3 import java.io.DataOutputStream;
4 import java.io.IOException;
5
6 public class Cliente {
7     final static int TAM_BYTES_DOU = 8;
8     final static int TOTAL_NUMEROS = 1000;
9     static double mandar = 0;
10
11     public static void main(String[] args) {
12         Socket socket = new Socket("localhost", 8080);
13         DataOutputStream dos = new DataOutputStream(socket.getOutputStream());
14         ByteBuffer bb = ByteBuffer.allocate(TAM_BYTES_DOU);
15         for(int i = 0; i < TOTAL_NUMEROS; i++) {
16             bb.putDouble(mandar);
17             dos.write(bb.array());
18             dos.flush();
19             mandar++;
20         }
21     }
22 }
```

```
[mariomiranda@localhost Con ByteBuffer]$ java Cliente
30
[mariomiranda@localhost Con ByteBuffer]$
```

Haciendo la comparación se puede concluir que enviar información mediante el uso de bytes es más eficiente en comparación a mandarla como lo es su tipo de dato.