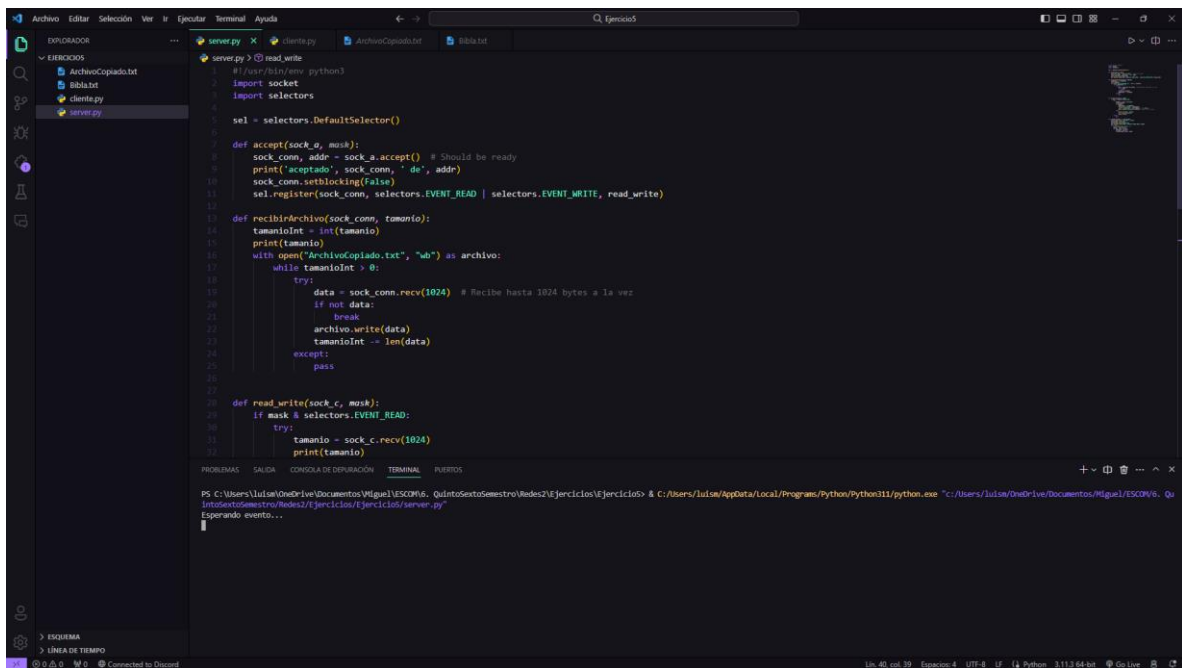
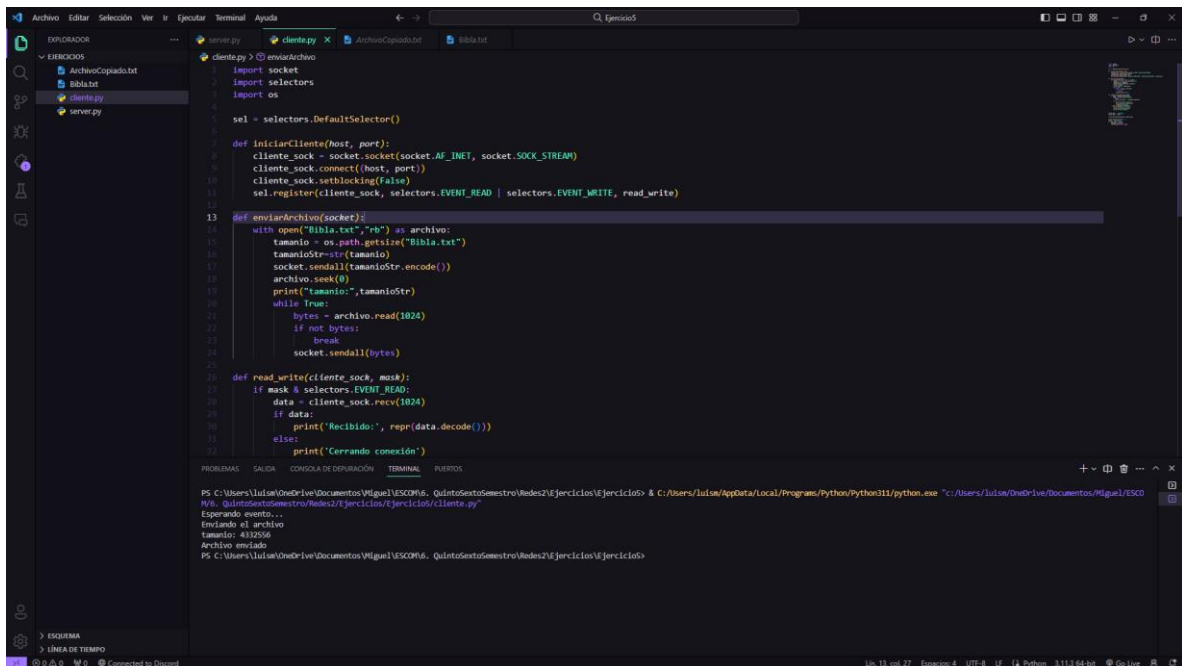


Actividad - Transferencia de archivo con sockets no bloqueantes

Capturas de pantalla.



```
server.py > read_write
1 #!/usr/bin/env python
2 import socket
3 import selectors
4
5 sel = selectors.DefaultSelector()
6
7 def accept(sock, mask):
8     sock_conn, addr = sock.accept() # Should be ready
9     print('aceptado', sock_conn, ' de', addr)
10    sock_conn.setblocking(False)
11    sel.register(sock_conn, selectors.EVENT_READ | selectors.EVENT_WRITE, read_write)
12
13 def recibirArchivo(sock_conn, tamaño):
14     tamañoInt = int(tamaño)
15     print(tamaño)
16     with open('ArchivoCopiado.txt', "wb") as archivo:
17         while tamañoInt > 0:
18             try:
19                 data = sock_conn.recv(1024) # Recibe hasta 1024 bytes a la vez
20                 if not data:
21                     break
22                 archivo.write(data)
23                 tamañoInt -= len(data)
24             except:
25                 pass
26
27 def read_write(sock_c, mask):
28     if mask & selectors.EVENT_READ:
29         try:
30             tamaño = sock_c.recv(1024)
31             print(tamaño)
32
33 PS C:\Users\luis\OneDrive\Documentos\WUguel\ESCOM\6. QuintoSemestre\Redes2\Ejercicios\Ejercicio6> & C:\Users\luis\AppData\Local\Program\Python\Python311\python.exe "C:\Users\luis\OneDrive\Documentos\WUguel\ESCOM\6. QuintoSemestre\Redes2\Ejercicios\Ejercicio6\server.py"
Esperando evento...
```



```
cliente.py > enviarArchivo
1 import socket
2 import selectors
3 import os
4
5 sel = selectors.DefaultSelector()
6
7 def iniciarCliente(host, port):
8     cliente_sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
9     cliente_sock.connect((host, port))
10    cliente_sock.setblocking(False)
11    sel.register(cliente_sock, selectors.EVENT_READ | selectors.EVENT_WRITE, read_write)
12
13 def enviarArchivo(sock):
14     with open('Bibla.txt', "rb") as archivo:
15         tamaño = os.path.getsize("Bibla.txt")
16         tamañoStr = str(tamaño)
17         socket.sendall(tamañoStr.encode())
18         archivo.seek(0)
19         print("tamaño:", tamañoStr)
20         while True:
21             bytes = archivo.read(1024)
22             if not bytes:
23                 break
24             socket.sendall(bytes)
25
26 def read_write(cliente_sock, mask):
27     if mask & selectors.EVENT_READ:
28         data = cliente_sock.recv(1024)
29         if data:
30             print("Recibido:", repr(data.decode()))
31         else:
32             print('Cerrando conexión')
33
34 PS C:\Users\luis\OneDrive\Documentos\WUguel\ESCOM\6. QuintoSemestre\Redes2\Ejercicios\Ejercicio6> & C:\Users\luis\AppData\Local\Program\Python\Python311\python.exe "C:\Users\luis\OneDrive\Documentos\WUguel\ESCOM\6. QuintoSemestre\Redes2\Ejercicios\Ejercicio6\cliente.py"
Esperando evento...
Enviando el archivo
tamaño: 413256
Archivo enviado
PS C:\Users\luis\OneDrive\Documentos\WUguel\ESCOM\6. QuintoSemestre\Redes2\Ejercicios\Ejercicio6>
```

