

server.py // Pythonprogramm das wie ein Server funktioniert

```
import socket

SERVER_IP = "0.0.0.0"
SERVER_PORT = 5005

sock = socket.socket(socket.AF_INET,
socket.SOCK_DGRAM)
sock.bind((SERVER_IP, SERVER_PORT))

print("UDP Server läuft...")

while True:
    data, addr = sock.recvfrom(1024)
    print(f"Empfangen von {addr}:
{data.decode()}")

    if data.decode() == "ping":
        sock.sendto("pong".encode(), addr)
        print("→ pong gesendet")
```

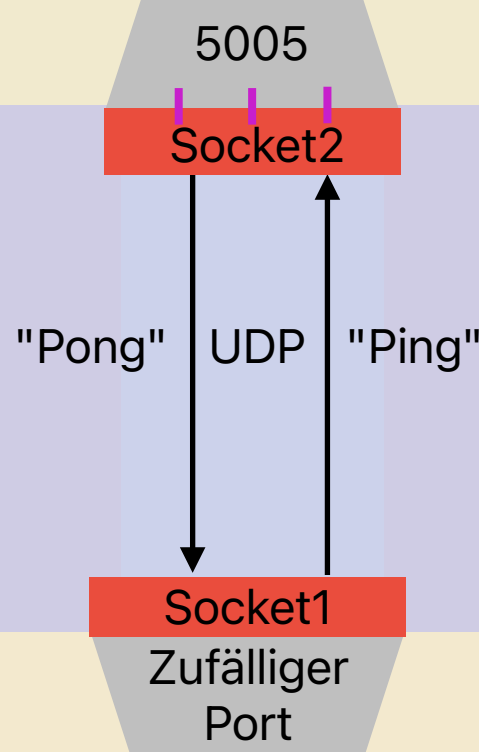
// Funktionalität die Python braucht um über Netzwerk zu sprechen

// Anfrage kann von überall kommen
// Port auf dem der Server lauscht

// ein neues Socket wird erstellt
// Bindet das Socket an angegebene IP und Port
(Nötig damit client weiss, wohin Packet schicken)

// Wartet auf eine Nachricht vom Netzwerk

// Wenn eingehende Nachricht "ping" dann wird "pong" gesendet



```
import socket

SERVER_IP = "127.0.0.1"
SERVER_PORT = 5005

sock = socket.socket(socket.AF_INET,
socket.SOCK_DGRAM)

sock.sendto("ping".encode(), (SERVER_IP,
SERVER_PORT))
print("→ ping gesendet")

data, addr = sock.recvfrom(1024)
print("Antwort:", data.decode())
```

// Funktionalität die Python braucht um über Netzwerk zu sprechen

// Definiert Server IP (in diesem Fall unsere eigene (localhost))
// Definiert Port

// ein neues Socket wird erstellt

// Schickt "ping" an definierte IP und Port

// Wartet auf eine Nachricht vom Netzwerk
// Wandelt die empfangenen Bytes zurück in Text
(decode()) und zeigt die Antwort an

client.py // Pythonprogramm das wie ein Client funktioniert

>_Terminal1

```
python3 server.py
```

>_Terminal2

```
python3 client.py
```