# 6 Python Sockets Example

Jacques Mock Schindler

03.09.2025

In this section, we will explore how to create a simple client-server application using Python sockets. This example will demonstrate the basic concepts of socket programming, including how to establish a connection, and send and receive data.

> **Network Socket**
>
> A network socket is a software structure within a network node of a computer network that serves as an endpoint for sending and receiving data across the network.

## Communication Structure

The example will consist of a server that listens for incoming connections and a client that connects to the server. The communication will be modelled as a kind of chat service.

## Server Code

```
# socket_server.py

import socket

def server_program():
    # define host name and port
    host = 'localhost'    # for coummunication on the local machine
    port = 5000           # initiate port no above 1024

    # create socket
    server_socket = socket.socket()

    # bind the socket to the host and port
```

```
    server_socket.bind((host, port))

    # set the server to listen for connections
    server_socket.listen(2) # max 2 clients can connect

    # accept new connection from client
    conn, address = server_socket.accept()
    print("Connection from: " + str(address))

    while True:
        # receive data stream. it won't accept data packet greater than 1024 bytes
        data = conn.recv(1024).decode()
        if not data:
            # if data is not received break
            break
        print("Recived from connected client: " + str(data))
        # prompt the user to enter a message
        data = input(' -> ')
        # send data to the client as bytes
        conn.send(data.encode())

    # close the connection
    conn.close()

if __name__ == '__main__':
    server_program()
```