# Introduction to Socket Programming
# and
# Client-Server Communication in Python

1. **Socket**

A **socket** is an endpoint for communication between two machines. It provides an interface for sending and receiving data across a network using protocols like TCP or UDP. One socket(node) listens on a particular port at an IP, while the other socket reaches out to the other to form a connection. The server forms the listener socket while the client reaches out to the server. Socket programming in Python involves using sockets to establish communication between a server and clients over a network.

2. **Client-Server Architecture**

A **client-server model** is a communication framework where:

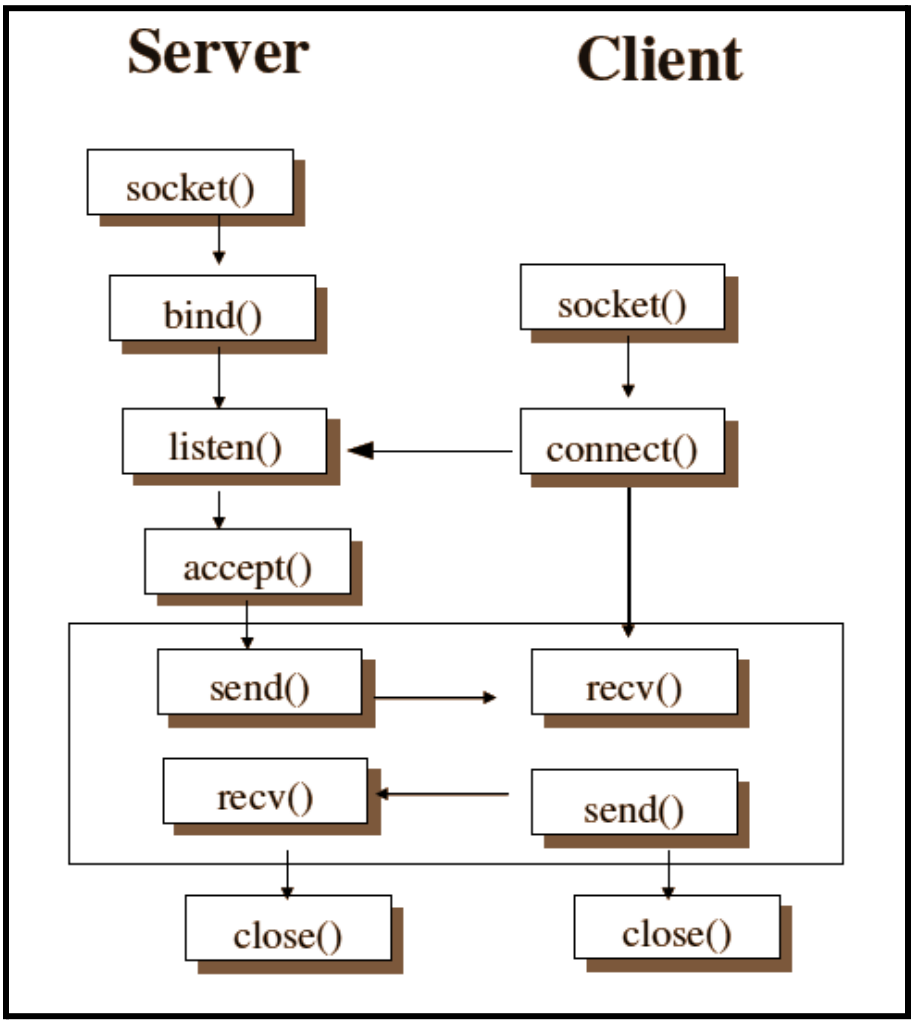
- **Client**: Initiates the request and connects to the server.
- **Server**: Listens for incoming client requests and responds accordingly.

3. **Few functions**

| Function | Description |
|---|---|
| socket() | Create a new socket |
| bind() | Attach socket to IP and port |
| listen() | Prepare server for connections |
| accept() | Accept an incoming client |
| connect() | Connect to a server |
| send()/sendto() | Send data |
| recv()/recvfrom() | Receive data |
| close() | Close the socket |

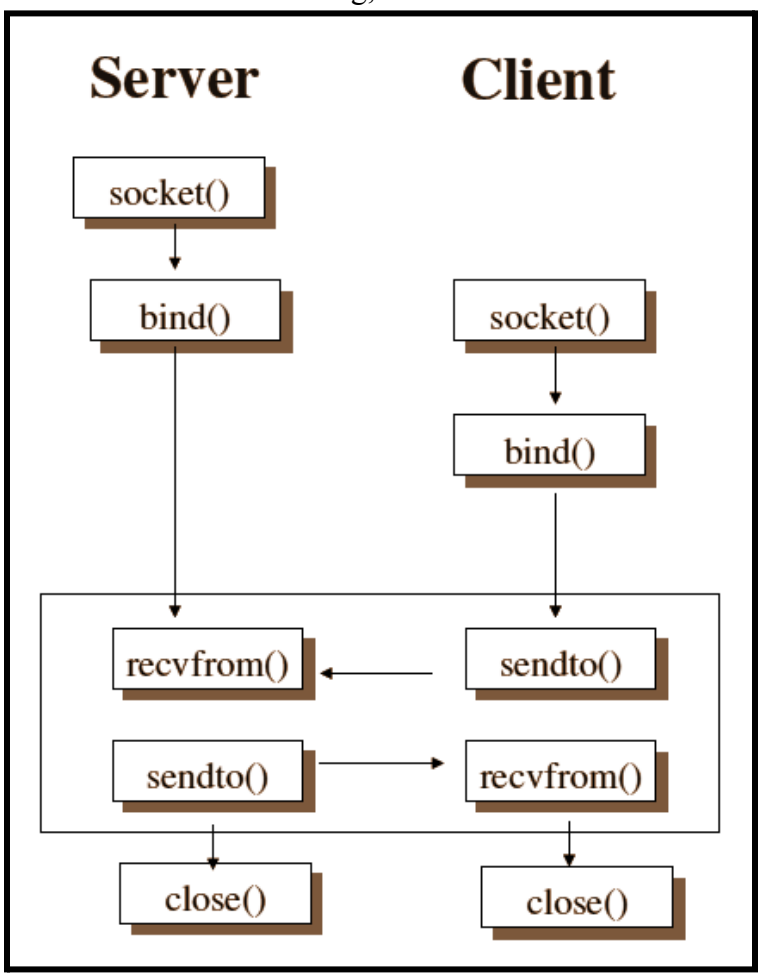
## 4. TCP (Transmission Control Protocol)

- **Connection-oriented** protocol.
- Ensures reliable delivery of data in the correct order.
- Example usage: Web browsing (HTTP), email.



5. **UDP (User Datagram Protocol)**

- **Connectionless** protocol.
- Faster but does **not guarantee** delivery or order.
- Example usage: Live video/audio streaming, DNS.




6. **IP Address**

An **IP address** is a 32-bit (IPv4) or 128-bit (IPv6) unique identifier for a machine on a network. Written in **dot-decimal** form, e.g., 192.168.11.90.

7. **Port Number**

A **port number** identifies a specific application/service on a machine.

- Range: 0 to 65535
- Well-known ports: 80 (HTTP), 443 (HTTPS), 21 (FTP), 53 (DNS)

8. **Hostname**

A **hostname** is a human-readable name for a device (e.g., google.com). It maps to an IP address using DNS**(Domain Name System)**. DNS is a system that translates domain names (like example.com) into IP addresses. Python can use socket.gethostbyname() for this.

9. **Address Family**

Specifies the protocol/address family used by the socket:

- AF_INET: IPv4 (used in this lab)
- AF_INET6: IPv6
- AF_UNIX: Local (same-machine) sockets

10. **Socket Type**

Specifies the communication style of the socket:

- SOCK_STREAM: Reliable byte stream (TCP)
- SOCK_DGRAM: Unreliable datagram service (UDP)

11. **Blocking vs Non-blocking Sockets**

- **Blocking socket**: Waits (blocks) for operations like accept() or recv() to complete.
- **Non-blocking socket**: Returns immediately if no data is available or no client connects.

12. **Loopback Address (localhost)**

IP address 127.0.0.1 used to refer to the local machine. Useful for testing server-client programs on the same computer.