# Socket Programming

Computer network programming involves writing computer programs that enable processes to communicate with each other across a computer network

Socket programming is a way of connecting two nodes on a network to communicate with each other. One socket (node) listens on a particular port at an IP, while other socket reaches out to the other to form a connection. Server forms the listener socket while client reaches out to the server

What is socket and how it works?

The server socket listens for incoming connections. A server creates a socket, binds the socket to an IP address and port number (for TCP and UDP), and then listens for incoming connections. When a client connects to the server, a new socket is created for communication with the client (TCP only).
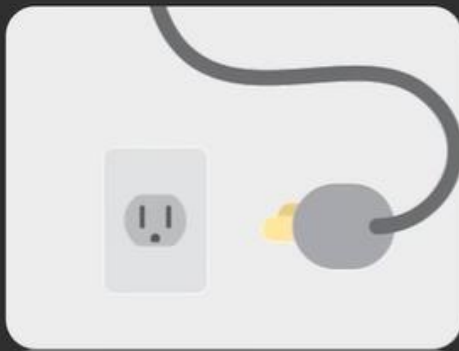
Difference between Socket and Port

Socket and Port are two terms used in computer networks. The difference between socket and port is that the socket is the interface of sending and receiving data on a specific port while the port is a numerical value assigned to a specific process or an application in the device.

Is Socket A hardware or software?

Socket. When a computer program needs to connect to a local or wide area network such as the Internet, it uses a software component called a socket. ... It is important to note that these sockets are software

Why do we need socket programming?

Socket programs are used to communicate between various processes usually running on different systems. It is mostly used to create a client-server environment

- Sockets are interior endpoints built for sending and receiving data
- A single network will have two sockets
- Sockets are a combination of an IP address and a Port

# Common Port Numbers

## Common port numbers and the related protocols

| Protocol | Port Number | Python Library | Function |
|----------|-------------|----------------|----------|
| HTTP | 80 | httplib, urllib,xmlrpclib | Web pages |
| FTP | 20 | ftplib, urllib | File transfers |
| NNTP | 119 | nntplib | Unsent news |
| SMTP | 25 | smtplib | Sending email |
| Telnet | 23 | telnetlib | Command lines |
| POP3 | 110 | poplib | Fetching email |

# Important methods of the **socket** module

## Server Socket Methods:

| Method | Description |
|---|---|
| s.bind() | This method binds address (hostname, port number pair) to socket. |
| s.listen() | This method sets up and start TCP listener. |
| s.accept() | This passively accept TCP client connection, waiting until connection arrives (blocking). |

*Table-1: Server socket methods in python & their description*

## Client Socket Method:

| Method | Description |
|---|---|
| s.connect() | This method actively initiates TCP server connection. |

*Table-2: Client socket method in python & their description*

## General Socket Method:

| Method | Description |
|---|---|
| s.recv() | This method receives TCP message |
| s.send() | This method transmits TCP message |
| s.recvfrom() | This method receives UDP message |
| s.sendto() | This method transmits UDP message |
| s.close() | This method closes socket |
| socket.gethostname() | Returns the hostname. |

*Table-3: General socket methods in python & their description*