



# Network

## Internet

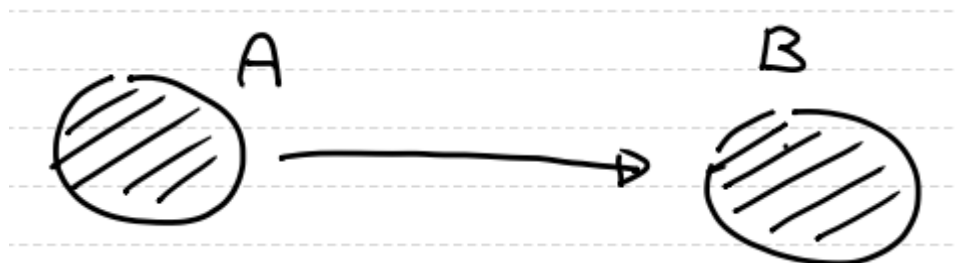
- LAN (Local Area Network)
- WAN (Wide Area Network)
- MAN (Metropolitan Area Network)



아주 쉽게 정의하면 소규모 Network를 전세계적인 Network로 다시 묶은것이라고 할 수있다.

**Network of Network**

## IP Address



A 에서 B 로 데이터 통신을 하려면 **IP Address** 를 알아야 한다.

**IP Address**는 컴퓨터가 가지고 있는 논리적인 주소이다.

그렇다면, IP Address만 알면 컴퓨터끼리 통신이 가능할까??

논리적인 주소는 언제든지 값이 바뀔수 있다.

- **IPv4**
  - 4바이트(=32비트) 주소 체계
- **IPv6**
  - 6바이트(=48비트) 주소 체계

- 4바이트 주소 체계가 조금 부족하다고 해서 만든 것인데, 아직까진 IPv4 를 거의 다 사용하고 있다.

## MAC Address

따라서, 컴퓨터끼리 통신을 할려면 물리적인 주소인 **MAC Address**가 필요하다.

MAC Address는 컴퓨터마다 전부 다른값을 가지고 있는 **Unique**한 값이다.

따라서 MAC Address를 알고있다면, 전세계에서 유일한 그 컴퓨터만 가지고 있는 주소를 아는것이다.

## ARP Protocol (Address Resolution Protocol)

IP Address → MAC Address로 바뀌서 **NIC(Network Interface Card)**로 전달한다. 즉, IP주소만 알아도 **ARP Protocol**을 이용해서 MAC Address를 알아낼 수 있는것이다.

## Protocol



데이터 통신을 위해 지켜야 할 규칙

- Protocol 종류

- HTTP

- HTML과 같은 **하이퍼미디어 문서를 전송**하기위한 애플리케이션 레이어 프로토콜이다.
    - 웹 브라우저와 웹 서버간의 커뮤니케이션을위해 디자인되었다.

- FTP

- 컴퓨터끼리 **File**단위로 **데이터 전송**할때 필요한 규칙

- SMTP

- TELNET

## Port Number



같은 디바이스에 있는 여러 프로세스들을 구분하기 위해 필요한 번호이다.

ex) 내 컴퓨터에서 실행되고있는 Excel, chrome, LOL 등을 구분하기 위한 번호이다.

- 0 ~ 65535 범위의 숫자를 사용한다.
- 0~1023까지는 예약되어있는 PortNumber기 때문에 사용하는것을 지양한다.

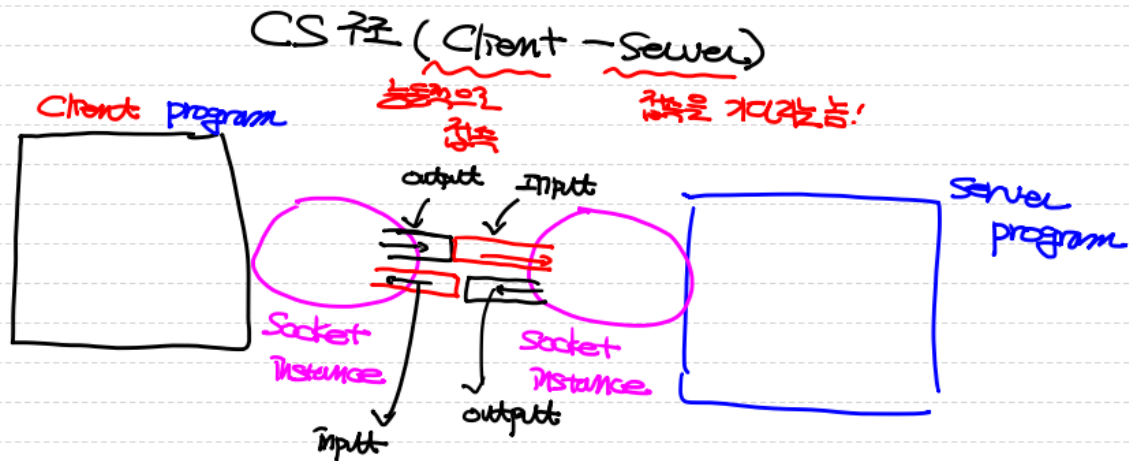


여기까지가 인터넷상에서 데이터통신을 할때 필요한 요소들이다.

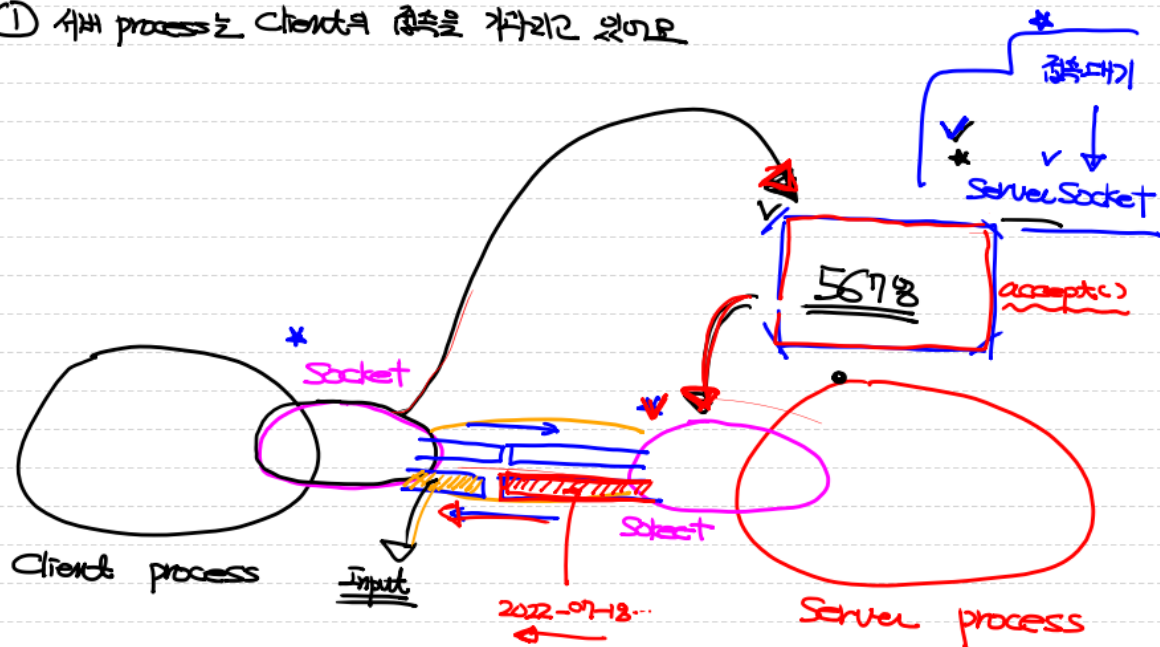
IP Address → ARP를 통해 → MAC Address로 바꾸고 통신을 위한 규칙 Protocol을 따라서 상대방 컴퓨터안에 있는 특정 프로세스의 PortNumber를 통해 접속해서 데이터를 주고 받는것이다.

## Socket

- Network 하단의 복잡한 Protocol에 상관없이 쉽게 프로그램을 작성
- Java에서는 **Socket** class가 있다.
  - Application간의 Data통신을 쉽게 할 수있다.
- CS구조를 가진다 (Client - Server)
  - Client = 능동적으로 접속하는 자식
  - Server = 접속을 기다리는 자식



① 서버 process는 Client의 접속을 기다리고 있음



## 연습문제 1

접속한 client에게 서버는 현재 날짜를 전송!

## 연습문제 2

간단한 Echo program작성

클라이언트가 "/exit" 를 입력하면 종료

## 연습문제 3

방 1개짜리 chatting program 구현

