

응용 계층

1. 응용 계층의 역할

: 어플리케이션은 서비스를 요청하는 측(사용자 측)에서 사용하는 어플리케이션과 서비스를 제공하는 측의 어플리케이션으로 분류된다.

일반적으로 서비스를 요청하는 측을

클라이언트, 서비스를 제공하는 측을 **서버**라고 한다.

웹브라우저나 메일 프로그램은 사용자 측에서 사용하는 어플리케이션이니 클라이언트에 속한다. 반면, 서비스를 제공하는 측인 서버에는 웹 서버 프로그램과 메일 서버 프로그램 등이 있다. 이런 어플리케이션은 **응용 계층**에서 동작한다.

응용 계층은 클라이언트의 요청을 전달하기 위해 통신 대상(서버)이 이해할 수 있는 메시지로 변환하고 전송 계층으로 전달하는 역할을 한다.



클라이언트 측 어플리케이션(웹 브라우저, 메일 프로그램)이 서버 측 어플리케이션(웹 서버 프로그램, 메일 프로그램)과 통신하려면 **응용 계층의 프로토콜**을 이용해야 한다.

웹 사이트를 볼 때는 **HTTP**, 파일을 전송할 때는 **FTP**, 메일을 보낼 때는 **SMTP**, 메일을 받을 때는 **POP3**라는 프로토콜을 사용한다.

2. Web

WWW(World Wide Web)란?

거대한 통신망인 인터넷은 수많은 사이트, 데이터, 정보를 갖고 있으며, 통신 회선이 거미줄처럼 서로 연결되어 있어서 언제 어디서든 필요한 곳에 접근하거나 정보를 공유하고 주고 받을 수 있는 멀티미디어 인터넷 서버다.

줄여서

W3이나 **웹(Web)**이라고도 부른다.

- WWW에는 HTML, URL, HTTP라는 세가지 기술이 사용된다.

1. **HTML (Hyper Text Markup Language)** : 인터넷 서비스의 하나인 WWW를 통해 볼 수 있는 문서를 만들 때 사용하는 프로그래밍 언어다. 하이퍼텍스트를 작성하기 위해 개발되었다.

웹 페이지에서 문장 구조나 문자를 꾸미는 태그를 사용하여 작성하는 마크업 언어다. 제목이나 목록과 같은 문장 구조를 지정하거나 이미지 파일을 보여줄 때도 태그를 사용한다. HTML로 작성하면 웹 브라우저로 볼 수 있다.

2. **URL (Uniform Resource Locator)** : 인터넷에서 파일 위치를 지정하기 위해 기술된 주소다. 웹 사이트 주소를 지정하기 위해 사용한다.

3. **HTTP**

3. HTTP

클라이언트(웹 브라우저)는 웹 사이트를 보기 위해 서버(웹 서버 프로그램)의 **80번 포트**를 사용하여 HTTP 통신을 한다.

클라이언트에서 **HTTP 요청(request)**을 보내고, 서버에서 **HTTP 응답(response)**을 반환한다.

- 요청 예시 : GET/index.html HTTP/1.1
- 응답 예시 : HTTP/1.1 200 OK

클라이언트가 데이터를 요청할 때는 "GET"이라 하는 요청 정보, 파일 이름, 버전 등을 서버에 전송한다. 그러면 서버는 응답으로 요청을 정상적으로 처리했다는 "OK"라는 정보를 반환하고 index.html을 클라이언트에게 보낸다.



지금은 HTTP/2나 HTTP/1.1 버전을 사용하지만 예전의 HTTP/1.0 버전에서는 요청을 보낼 때마다 연결했다가 끊는 작업을 반복해서 웹 페이지가 느리게 보였다.

- HTTP/1.0 예시 : 연결 수립 → 요청 → 응답 → 연결 끊기 → 연결 수립 → 요청 → 응답 → 연결 끊기

그러나 HTTP/1.1 버전에서는 **keepalive** 라는 기능이 추가되어, 연결을 한 번 수립하면 데이터 교환을 마칠 때까지 유지하고, 데이터 교환을 모두 끝내면 연결을 끊는 구조다. keepalive는 요청도 순서대로 처리하는 특징이 있다. 1요청에 대한 1응답 반환 후, 2요청과 2응답을 반환한다.

- HTTP/1.1 예시 : 연결 수립 → 1요청 → 1응답 → 2요청 → 2응답 → 연결 끊기

최근에는 HTTP/1.1 버전의 단점을 보완해서 빠르게 콘텐츠를 표시할 수 있는 프로토콜인 HTTP/2 버전도 사용할 수 있게 되었다. HTTP/1.1 버전은 요청을 보낸 순서대로 응답을 반환하기 때문에 콘텐츠 표시가 늦어지는 문제가 있었는데, **HTTP/2 버전은 보낸 순서대로 응답을 반환하지 않아도 되기 때문에**, 콘텐츠를 빠르게 표시할 수 있다.

- HTTP/2 예시 : 연결 수립 → 1요청 → 2요청 → 3요청 → 2응답 → 3응답 → 1응답 - 연결 끊기

4. DNS 서버

- DNS 서버란? 서버에 접속하려면 IP 주소를 입력해야 하는데, 우리는 보통 URL을 입력한다. 이는 DNS가 **URL을 IP 주소로 변환해주는 서비스**를 제공하기 때문이고, 이를 DNS의 이름해석이라고도 한다.

- **컴퓨터와 DNS 서버간에 일어나는 IP 주소 교환**

1. 컴퓨터에서 “이 도메인 이름의 IP 주소는 무엇인가요?”라고 DNS 서버에 질문한다.
2. DNS 서버는 해당 요청에 해당하는 도메인 이름의 IP 주소를 알려준다.
3. 컴퓨터는 해당 IP 주소로 웹 서버에 접속한다.

5. 메일 서버 (SMTP, POP2)

: 메일을 송수신하려면 클라이언트 측의 메일 프로그램과 서버 측의 메일 서버 프로그램 간에 통신을 해야 한다. 이때 사용되는 프로토콜에는 두가지 종류가 있다.

- **메일 송신(SMTP) : 25번 포트**

- **메일 수신(POP3) : 110번 포트**

- **과정**

1. SMTP를 사용하여 컴퓨터1에서 메일 서버1로 메일을 보낸다.
2. SMTP를 사용하여 메일 서버1에서 메일 서버2로 메일을 보낸다.
3. POP3를 사용하여 메일 서버2에서 컴퓨터 2로 메일 데이터를 보낸다.
 - 컴퓨터1 → 메일 서버1 → 메일 서버2 → 컴퓨터2

컴퓨터1의 사용자는 메일 서버1을 사용하는데 메일 서비스를 제공하는 포털 사이트나 회사의 메일 서버를 사용하는 경우가 많다. 포털 사이트에 가입하거나 회사에서 메일 주소를 받으면 메일 서버를 사용할 수 있게 된다.

1. 메일을 보낼때는 메일 서비스를 제공하는 포털 사이트나 회사의 메일 서버에 메일을 보내는 것이다.
2. 메일 서버1은 SMTP를 사용하여 메일 서버2로 메일을 전송한다.
3. 메일 서버2는 POP3를 사용하여 메일 서버2의 박스에서 메일을 가져와서 컴퓨터2로 전송한다.