

# Hello Null4U, **Orientation**

부장

김준영

여러분들의 피드백을 적극적으로 환영합니다. (기존 부원 포함)

## 수요조사 링크



<https://forms.gle/LgWDV4HSjWCWDMfPA>

## 디스코드 링크



<https://discord.gg/RSmXz5dUjG>

# Introduction



## Kim Jun Young 김준영

現 세명컴퓨터고등학교 스마트보안솔루션과 3학년 (11기)

現 세명컴퓨터고등학교 스마트보안솔루션과 대표

現 교내 개발/클라우드 컴퓨팅/DevOps 전공동아리 Null4U 부장

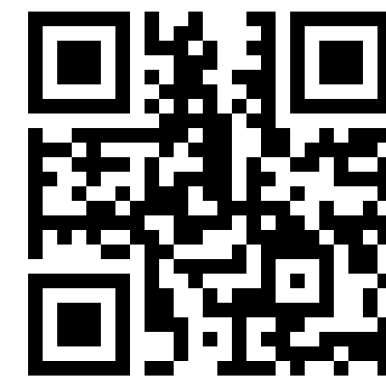
정보처리기능사(프로그래밍기능사), 정보기기운용기능사 및 몇몇 대회 경력 보유

등등 ...

- [github.com/yulmwu](https://github.com/yulmwu)
- Discord @rlawnsdud
- 010-2980-1336
- [me@swua.kr](mailto:me@swua.kr)
- [linkedin.com/in/yulmwu](https://www.linkedin.com/in/yulmwu)

개인 홈페이지

<https://swua.kr>



# Null4U 커리큘럼

| 분야                | 내용  |
|-------------------|---|
| 개발 지식             | 프로그래밍 언어, 백엔드/프론트엔드 웹 개발, 버전 관리(Git) 등            |
| 서버/운영체제 운영        | 리눅스 서버 운영, 네트워크 기초, VM 및 컨테이너 기초 등                |
| 클라우드 컴퓨팅          | AWS, GCP 등의 클라우드 서비스 활용 및 운영, IaC 및 매니지드 서비스 활용 등 |
| 컨테이너 오케스트레이션      | Docker, Kubernetes 등의 도구 활용                       |
| CI/CD             | Jenkins, GitHub Actions 등의 CI/CD 도구 활용            |
| 옵저버빌리티, 모니터링 및 로깅 | Prometheus, Grafana, ELK 스택 등의 도구 활용              |
| 인프라 운영 자동화        | GitOps, Ansible, Terraform 등의 도구 활용               |

# Null4U 커리큘럼

| 분야                | 내용  |
|-------------------|---|
| 개발 지식             | 프로그래밍 언어, 백엔드/프론트엔드 웹 개발, 버전 관리(Git) 등            |
| 서버/운영체제 운영        | 리눅스 서버 운영, 네트워크 기초, VM 및 컨테이너 기초 등                |
| 클라우드 컴퓨팅          | AWS, GCP 등의 클라우드 서비스 활용 및 운영, IaC 및 매니지드 서비스 활용 등 |
| 컨테이너 오케스트레이션      | Docker, Kubernetes 등의 도구 활용                       |
| CI/CD             | Jenkins, GitHub Actions 등의 CI/CD 도구 활용            |
| 옵저버빌리티, 모니터링 및 로깅 | Prometheus, Grafana, ELK 스택 등의 도구 활용              |
| 인프라 운영 자동화        | GitOps, Ansible, Terraform 등의 도구 활용               |

| G          | H                      | I     | J      | K     |
|------------|------------------------|-------|--------|-------|
| 4월 동아리 활동일 |                        |       |        |       |
|            | 월                      | 화     | 수      | 목     |
| 날짜         |                        |       | 1      | 2     |
|            |                        | -     | 넬포유 1회 | -     |
| 날짜         | 6                      | 7     | 8      | 9     |
|            | 스카 1회                  | -     | 닷핵, 기능 | -     |
| 날짜         | 13                     | 14    | 15     | 16    |
|            | 넬포유 1.5 회차<br>스카 1.5회차 | -     | 지필 대비  | 지필 대비 |
| 날짜         | 20                     | 21    | 22     | 23    |
|            | 지필 대비                  | 지필 대비 | 지필평가   | 지필평가  |
| 날짜         | 27                     | 28    | 29     | 30    |
|            | 휴식기                    | 소테교   | 소테교    | 소테교   |

| G                 | H                      | I     | J      | K     |
|-------------------|------------------------|-------|--------|-------|
| <b>4월 동아리 활동일</b> |                        |       |        |       |
|                   | 월                      | 화     | 수      | 목     |
| 날짜                |                        |       | 1      | 2     |
|                   |                        | -     | 넬포유 1회 | -     |
| 날짜                | 6                      | 7     | 8      | 9     |
|                   | 스카 1회                  |       | 다해 기느  |       |
| 날짜                | 13                     |       |        |       |
|                   | 넬포유 1.5 회차<br>스카 1.5회차 | -     | 지필 대비  | 지필 대비 |
| 날짜                | 20                     | 21    | 22     | 23    |
|                   | 지필 대비                  | 지필 대비 | 지필평가   | 지필평가  |
| 날짜                | 27                     | 28    | 29     | 30    |
|                   | 휴식기                    | 소테교   | 소테교    | 소테교   |

4월달 Null4U 강의 가능 시간은 3시간

---

## 1학년 교습활동

- 1,2주 김준영(웹이란? 프론트/백엔드)
- 3,4주(HTML 진재화, CSS 경대영)
- 5,6주(JS 2학년)
- python(김준영, 초반 여건이 될지 안될지? )

---

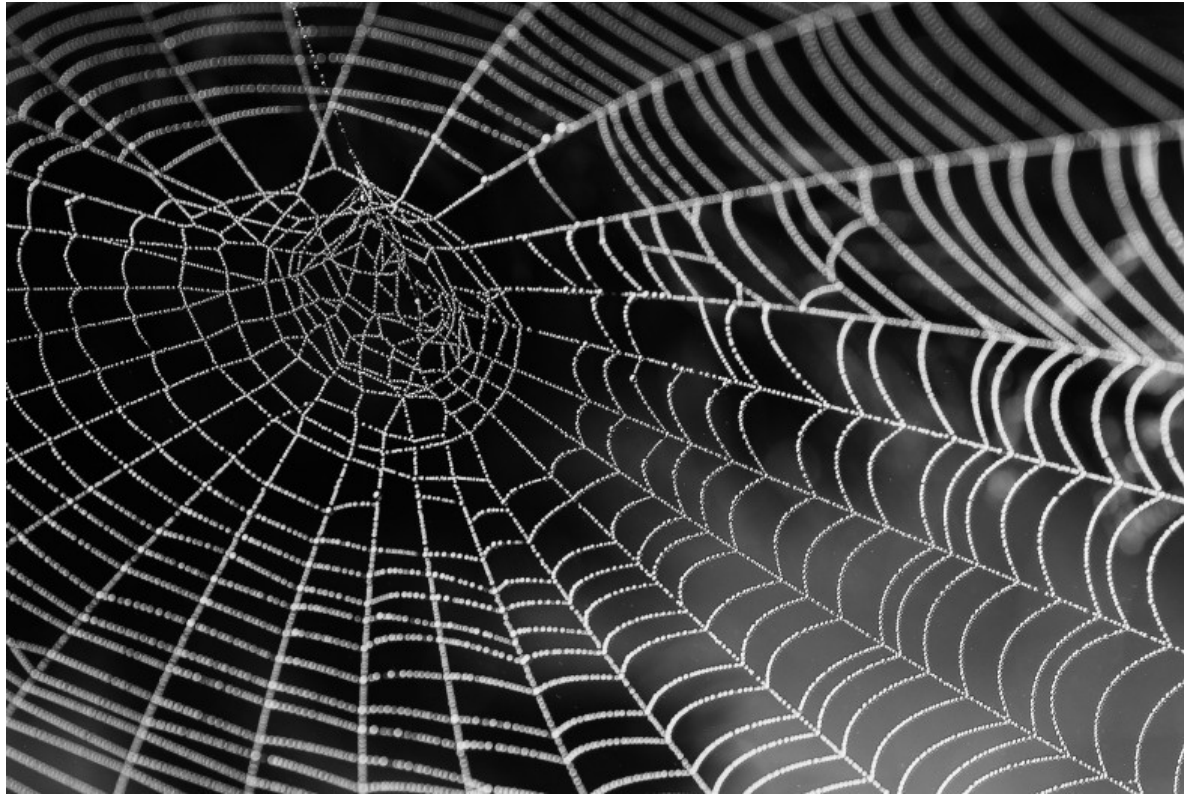
초반 1~2주 (4월 + 5월)엔 웹에 대해서 기초를 다룰 예정

| G                 | H                      | I     | J      | K     |
|-------------------|------------------------|-------|--------|-------|
| <b>4월 동아리 활동일</b> |                        |       |        |       |
|                   | 월                      | 화     | 수      | 목     |
| 날짜                |                        |       | 1      | 2     |
|                   |                        | -     | 넬포유 1회 | -     |
| 날짜                | 6                      | 7     | 8      | 9     |
|                   | 스카 1회                  | -     | 닷핵, 기능 | -     |
| 날짜                | 13                     | 14    | 15     | 16    |
|                   | 넬포유 1.5 회차<br>스카 1.5회차 | -     | 지필 대비  | 지필 대비 |
| 날짜                | 20                     | 21    | 22     | 23    |
|                   | 지필 대비                  | 지필 대비 | 지필평가   | 지필평가  |
| 날짜                | 27                     | 28    | 29     | 30    |
|                   | 휴식기                    | 소테교   | 소테교    | 소테교   |

공백 기간엔 과제 등으로 채울 예정

웹(Web)이란?

## 웹(Web)이란?

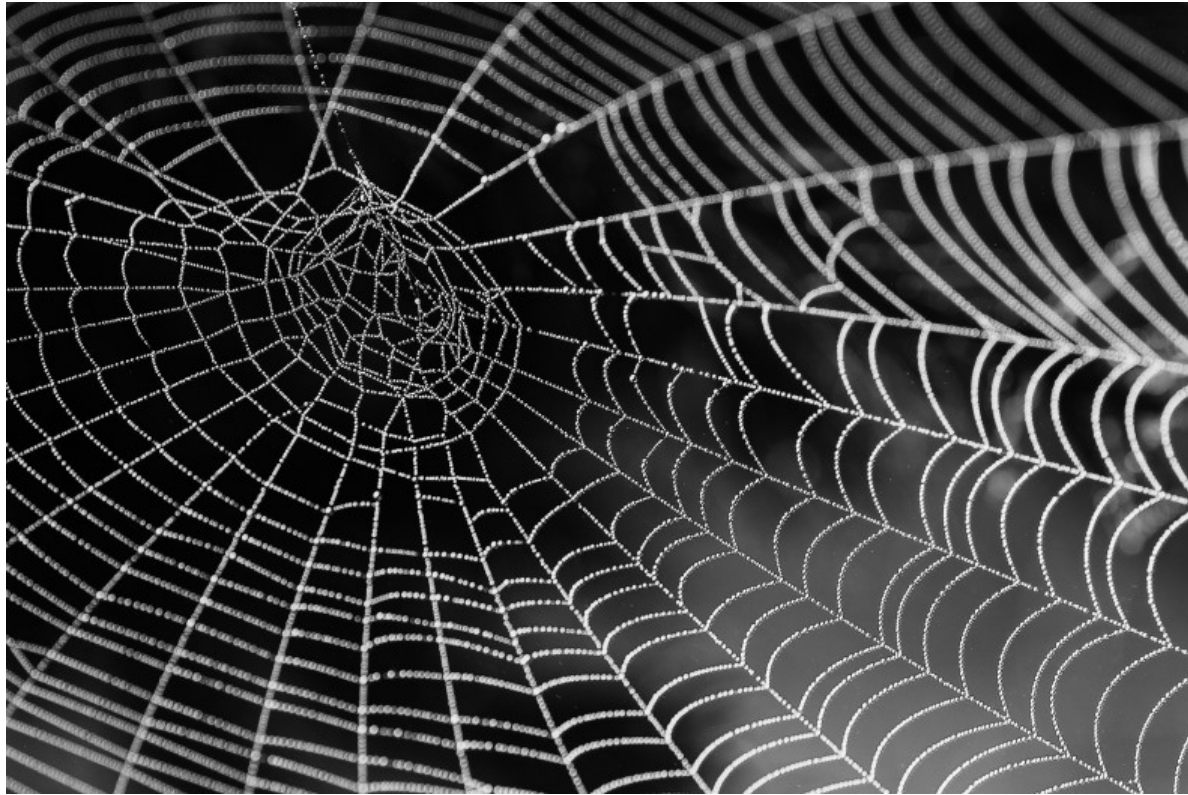


원래 "거미줄"을 의미함

사실 풀네임은 World Wide Web (WWW), 흔히 웹이라 부름

인터넷 등에 연결된 전세계 사용자들이 서로의 정보를 공유할 수 있는 공간  
컴퓨터들이 연결되는 것을 네트워크라고 함. (그물망 처럼 연결됐다고 해서)

# 웹(Web)이란?



원래 "거미줄"을 의미함

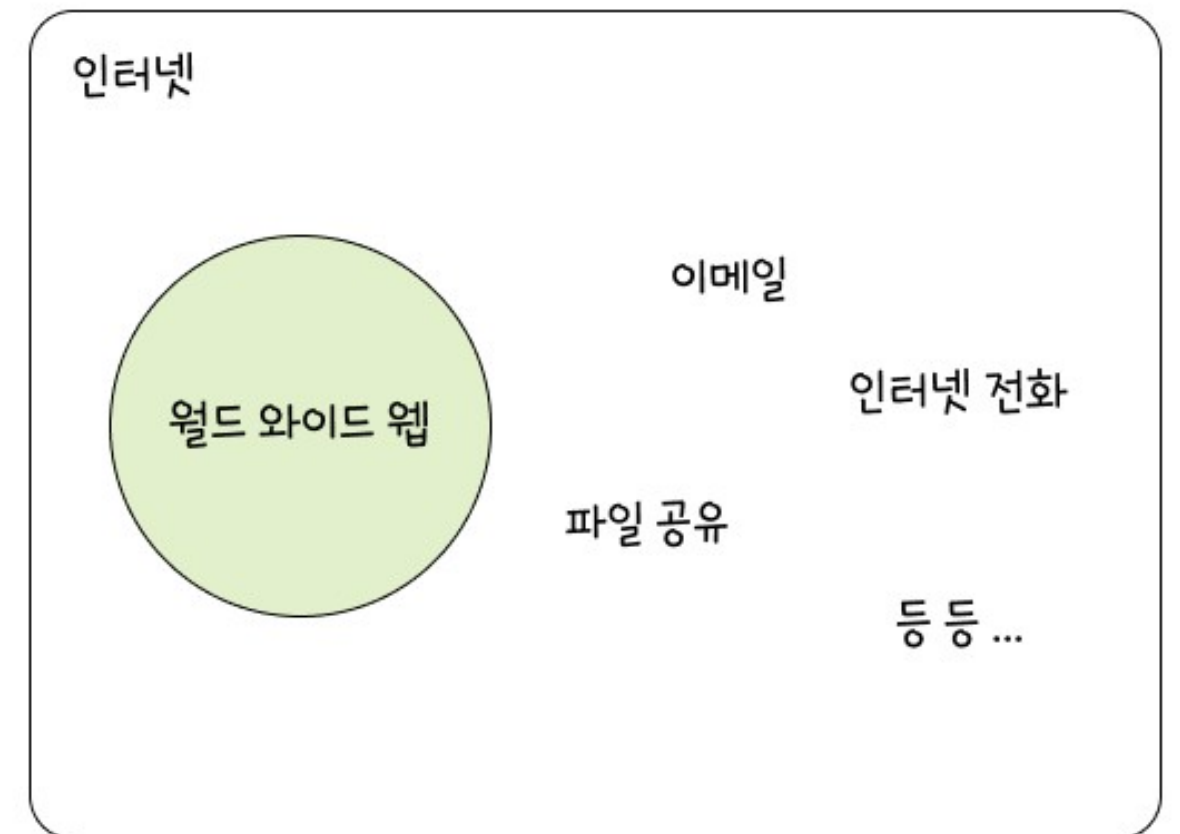
사실 풀네임은 World Wide Web (WWW), 흔히 웹이라 부름

인터넷 등에 연결된 전세계 사용자들이 서로의 정보를 공유할 수 있는 공간  
컴퓨터들이 연결되는 것을 네트워크라고 함. (그물망 처럼 연결됐다고 해서)

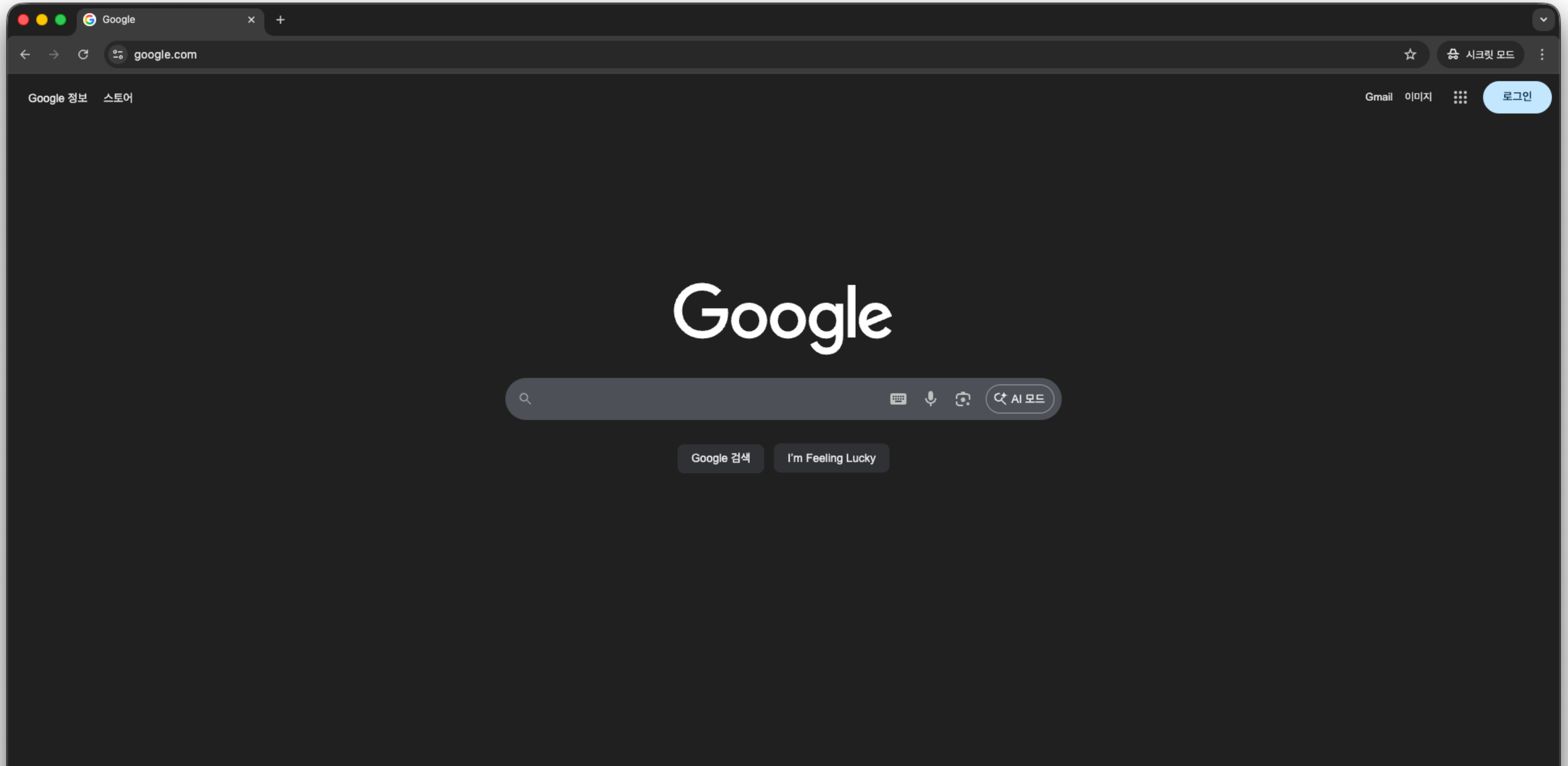
컴퓨터들이 연결되는 것을 네트워크라고 함. (그물망 처럼 연결됐다고 해서)

인터넷 = 전 세계 컴퓨터들을 하나로 연결하는 거대한 컴퓨터 통신망을 의미함

(인터넷 != 네트워크 != 웹)



구글에 접속하면 무슨 일이 벌어질까?



# 구글에 접속하면 무슨 일이 벌어질까?



유저

검색 결과 주세요.



네. "연신내역 근처 맛집"에 대한 검색 결과입니다.



Google

# 구글에 접속하면 무슨 일이 벌어질까?



유저

클라이언트  
(받는 쪽)

검색 결과 주세요.



네. "연신내역 근처 맛집"에 대한 검색 결과입니다.



Google

서버  
(주는 쪽)

너무나도 중요한 개념!!!!



# 퀴즈) 서버가 거절할 수 있을까?



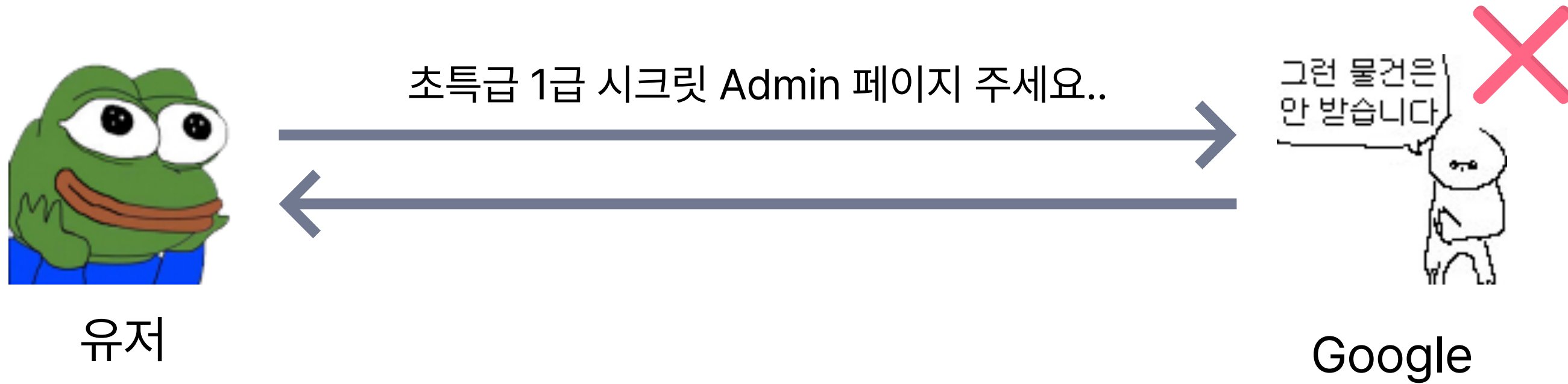
유저

초특급 1급 시크릿 Admin 페이지 주세요..



Google

## 퀴즈) 서버가 거절할 수 있을까?



정답: 가능하다

ex) 잘못된 경로 접근, 잘못된 요청, 로그인 없이 특정 서비스 이용 등등

(단 네트워크 응답 없음/에러와는 별개로 응답 자체는 정상적으로 받음)

부록) 이름만 들어본 URL, 뜯어보자.

검색 결과 주세요.



흔히 사이트 주소라고 부르는게 URL이라고 함.

URL 말고도 URI, URN이 있는데 이걸 추후 시간나면 설명해주겠음. + URL 인코딩

# 부록) 이름만 들어본 URL, 뜯어보자.



부록) 이름만 들어본 URL, 뜯어보자.



프로토콜은 네트워크 설명할때 설명하겠지만, 일단 통신 규약/규격과 같다고 판단하면 됨.

웹은 대부분 HTTP/HTTPS 기반임.

부록) 이름만 들어본 URL, 뜯어보자.



도메인은 google.com, naver.com 같이 IP 주소를 유저가 쉽게 기억할 수 있도록 문자열로 바꾸는 것

이것도 네트워크 설명할때 DNS와 함께 설명함. 네트워크/웹 개발, 운영에서 매우 중요한 개념임.

서브 도메인 개념도 나중에 설명

## 부록) 이름만 들어본 URL, 뜯어보자.



경로(Path)는 파일 디렉토리의 경로와 동일하게 생각하면 됨. 특정 리소스(페이지, 이미지) 등의 위치를 가리키는 주소임.

경우에 따라 진짜 파일 디렉토리를 쓰는 경우도 있지만 직접 쓰는건 보안상 권장되지 않고, 이후에 다룰 API 백엔드 서버는 경로(이 경우엔 엔드포인트)를 개발자가 직접 지정해서 만듦.

## 부록) 이름만 들어본 URL, 뜯어보자.



일반적으로 GET Method에서만 사용되며, 이는 나중에 HTTP 메서드를 다룰때 다시 언급하겠음.

부록) 이름만 들어본 URL, 뜯어보자.



한국어로://은평롯데점.빍스.레스토랑/세트메뉴/파티세트.html&인원=3명#메뉴판-섹션

영어로://백엔드.서버:8080/api/고객조회/김준영/즐거찾기-목록

부록) 이름만 들어본 URL, 뜯어보자.



<https://www.ivips.co.kr/menu?categoryIdx=6>

<https://api.ivips.co.kr/v1/users/3/favorites>

웹은 레스토랑에 비유할 수 있다.

내용: 참깨빵, 순쇠고기, 패티두장, 특별한소스, 양상추, 치즈, 피클, 양파

주문서 = 요청 본문

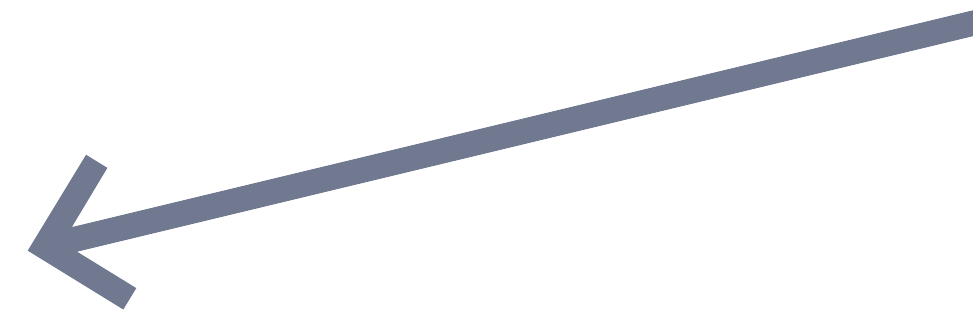
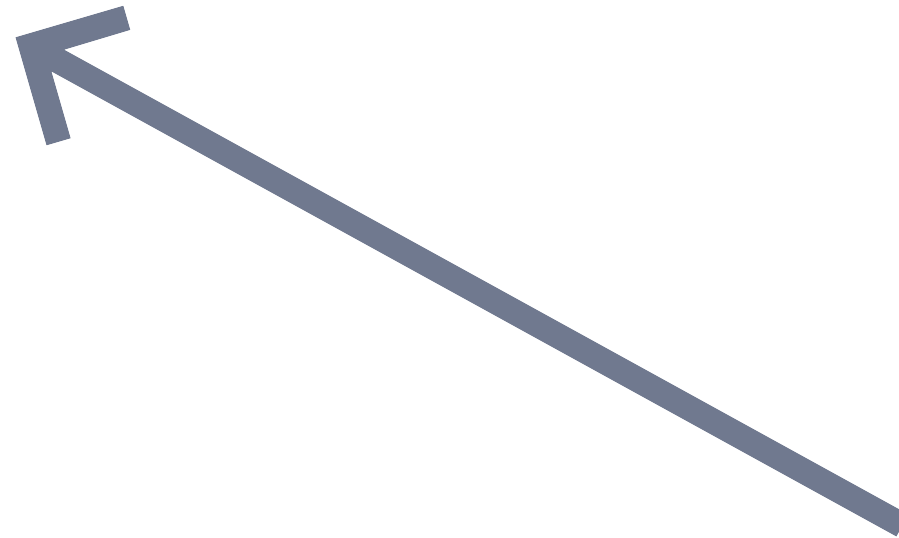


(요청)

햄부기 주세요



네 ^^



주문하신 음식 나왔습니다. 맛있게 드세요 ^^

(응답 본문)

(응답)

퀴즈) 서버가 없는 경우도 있을까?



응답주세요



서버가 없는데요..



응답주세요



퀴즈) 서버가 없는 경우도 있을까?



응답주세요



서버가 없는데요..

응답주세요



정답: 있다

퀴즈) 서버가 없는 경우도 있을까?



응답주세요



응답주세요



정답: 있다

P2P(Peer to Peer) 통신 등등..

(클라이언트가 서버 역할도 수행하는 것. 게임 서버, 파일 전송 등에서 간혹 쓰임)

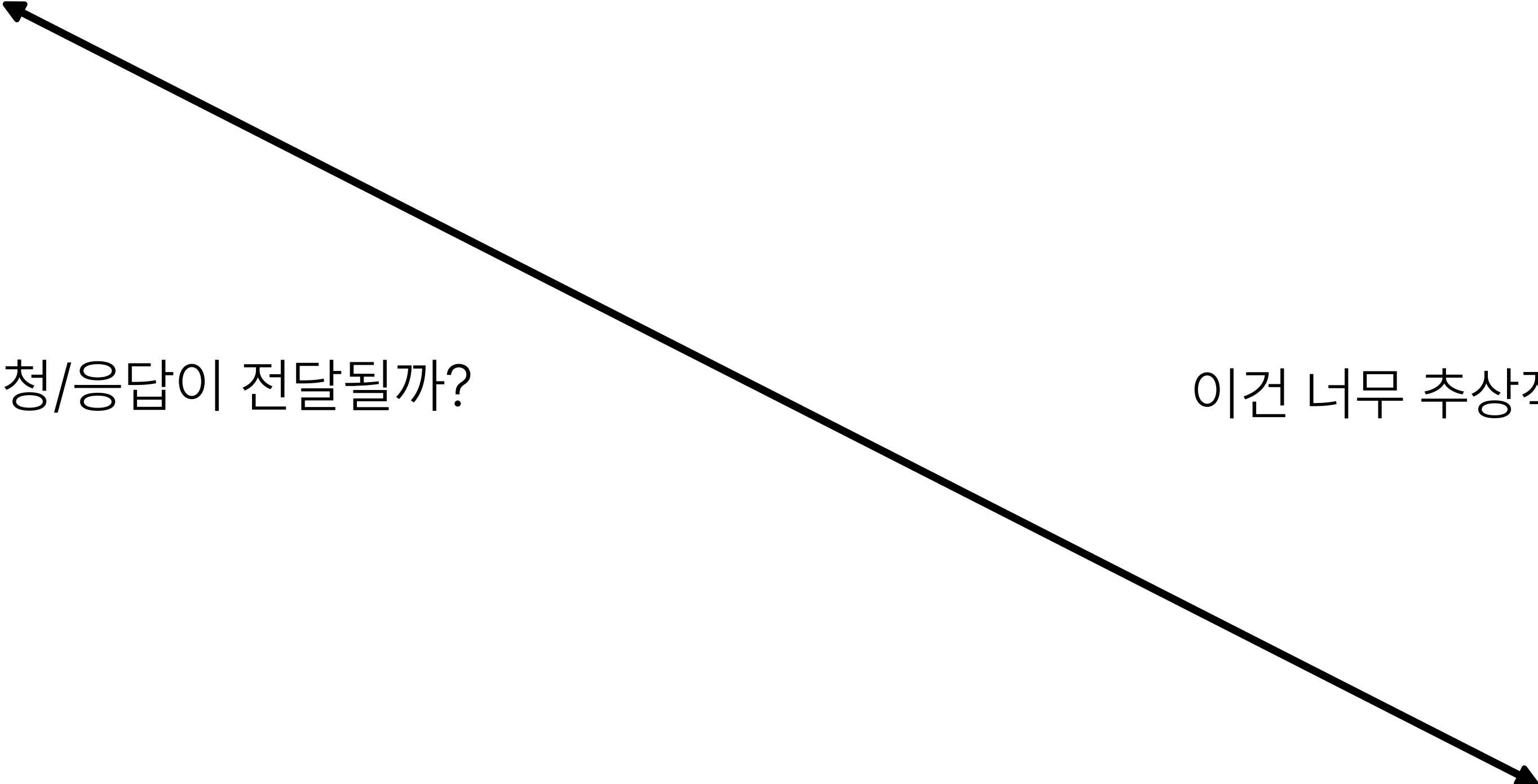
# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초)

**(클라이언트)**

어떻게 요청/응답이 전달될까?

이건 너무 추상적임...

**(서버)**



# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 도메인, DNS

(클라이언트)



# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 도메인, DNS

**(클라이언트)**



<https://google.com>

# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 도메인, DNS

(클라이언트)



https://google.com

→ 172.217.209.101 ...

```
/Users/user [ky0422@MacBookPro] [10:34]
```

```
> nslookup google.com
```

```
Server:          1.0.0.1
```

```
Address:         1.0.0.1#53
```

```
Non-authoritative answer:
```

```
Name:   google.com
```

```
Address: 172.217.209.101
```

```
Name:   google.com
```

```
Address: 172.217.209.102
```

```
Name:   google.com
```

```
Address: 172.217.209.100
```

```
Name:   google.com
```

```
Address: 172.217.209.138
```

```
Name:   google.com
```

```
Address: 172.217.209.139
```

```
Name:   google.com
```

```
Address: 172.217.209.113
```

# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 도메인, DNS

(클라이언트)



"도메인"

https://google.com

→ 172.217.209.101 ...

```
/Users/user [ky0422@MacBookPro] [10:34]
```

```
> nslookup google.com
```

```
Server:          1.0.0.1
```

```
Address:         1.0.0.1#53
```

```
Non-authoritative answer:
```

```
Name:   google.com
```

```
Address: 172.217.209.101
```

```
Name:   google.com
```

```
Address: 172.217.209.102
```

```
Name:   google.com
```

```
Address: 172.217.209.100
```

```
Name:   google.com
```

```
Address: 172.217.209.138
```

```
Name:   google.com
```

```
Address: 172.217.209.139
```

```
Name:   google.com
```

```
Address: 172.217.209.113
```

## "DNS"

사람이 기억하기 쉬운 도메인 주소를 네트워크에서 사용하는 IP 주소로 바꾸는 과정

# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 도메인, DNS

(클라이언트)



" 도메인 "

https://google.com

→ 172.217.209.101 ...

```
/Users/user [ky0422@MacBookPro] [10:34]
```

```
> nslookup google.com
```

```
Server:          1.0.0.1
```

```
Address:         1.0.0.1#53
```

```
Non-authoritative answer:
```

```
Name:   google.com
```

```
Address: 172.217.209.101
```

```
Name:   google.com
```

```
Address: 172.217.209.102
```

```
Name:   google.com
```

```
Address: 172.217.209.100
```

```
Name:   google.com
```

```
Address: 172.217.209.138
```

```
Name:   google.com
```

```
Address: 172.217.209.139
```

```
Name:   google.com
```

```
Address: 172.217.209.113
```

## " DNS "

사람이 기억하기 쉬운 도메인 주소를 네트워크에서 사용하는 IP 주소로 바꾸는 과정

010-2980-1336 (기억하기 어려움)

(연락처 등록) " 김준영 " : 010-2980-1336 (기억하기 쉬움)

# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 도메인, DNS

(클라이언트)



https://google.com

"도메인"

→ 172.217.209.101 ...

```
/Users/user [ky0422@MacBookPro] [10:34]
```

```
> nslookup google.com
```

```
Server:          1.0.0.1
```

```
Address:         1.0.0.1#53
```

```
Non-authoritative answer:
```

```
Name:   google.com
```

```
Address: 172.217.209.101
```

```
Name:   google.com
```

```
Address: 172.217.209.102
```

```
Name:   google.com
```

```
Address: 172.217.209.100
```

```
Name:   google.com
```

```
Address: 172.217.209.138
```

```
Name:   google.com
```

```
Address: 172.217.209.139
```

```
Name:   google.com
```

```
Address: 172.217.209.113
```

그런데 옆 처럼 DNS 결과(IP)가 여러개인 경우는 뭘까?

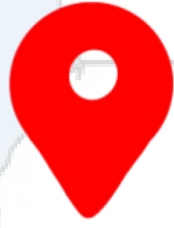
(스스로 조사해보고 원하면 발표)

# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 라우팅, 라우터



# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 라우팅, 라우터

(서버)



(클라이언트)



멀리 있는 서버에 어떻게 가지?



# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 라우팅, 라우터

(서버)

(클라이언트)



라우터 = 둘 이상의 서로 다른 네트워크를 연결하고, 목적지까지의 최적의 경로로 전송해주는 네트워크 장비

# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 라우팅, 라우터

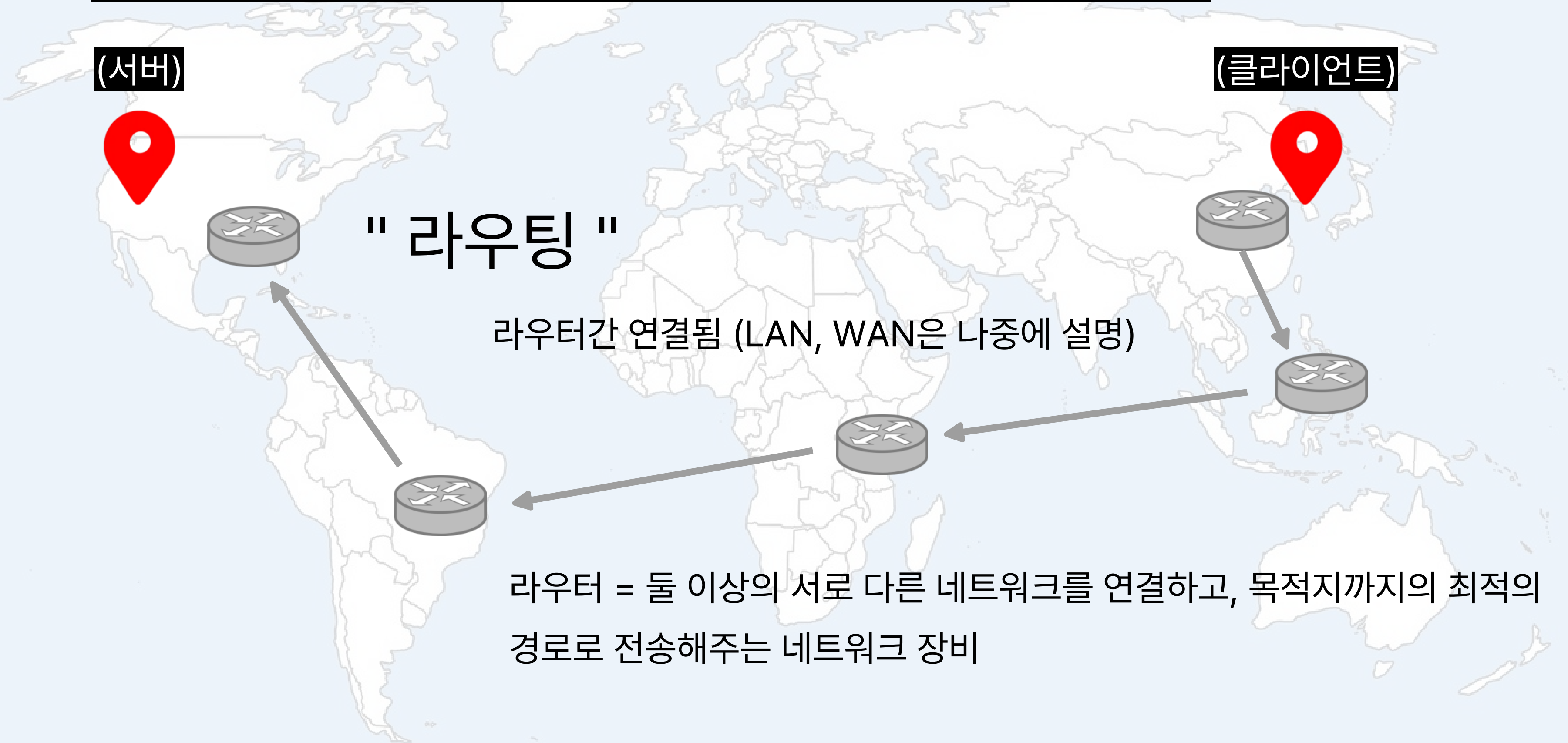
(서버)

(클라이언트)

" 라우팅 "

라우터간 연결됨 (LAN, WAN은 나중에 설명)

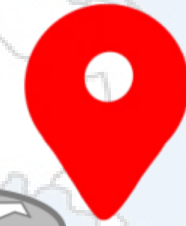
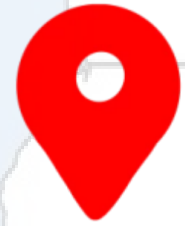
라우터 = 둘 이상의 서로 다른 네트워크를 연결하고, 목적지까지의 최적의 경로로 전송해주는 네트워크 장비



# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 라우팅, 라우터

(서버)

(클라이언트)

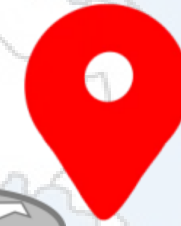
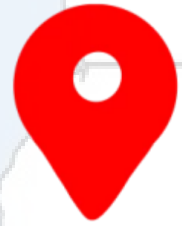


대륙간 떨어져있는데요?

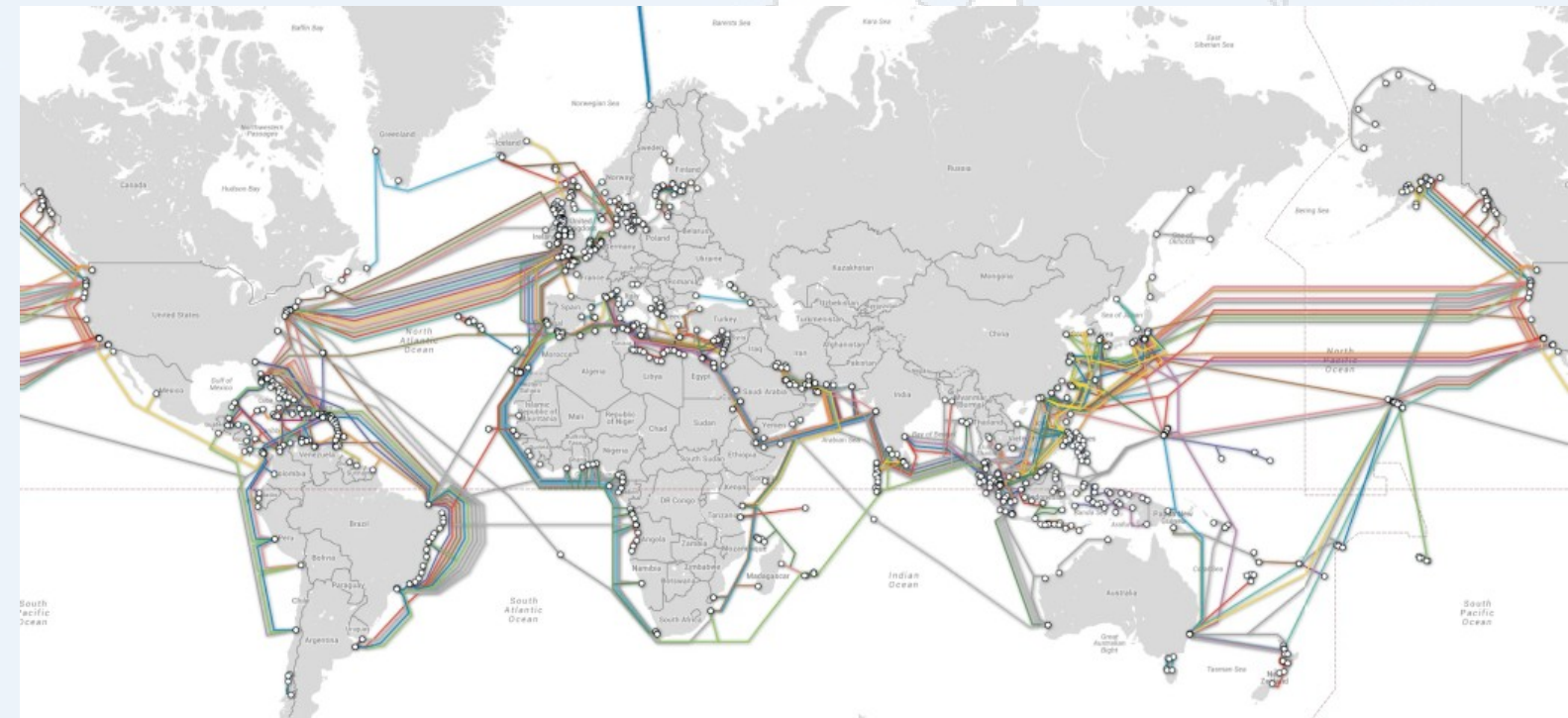
# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 라우팅, 라우터

(서버)

(클라이언트)



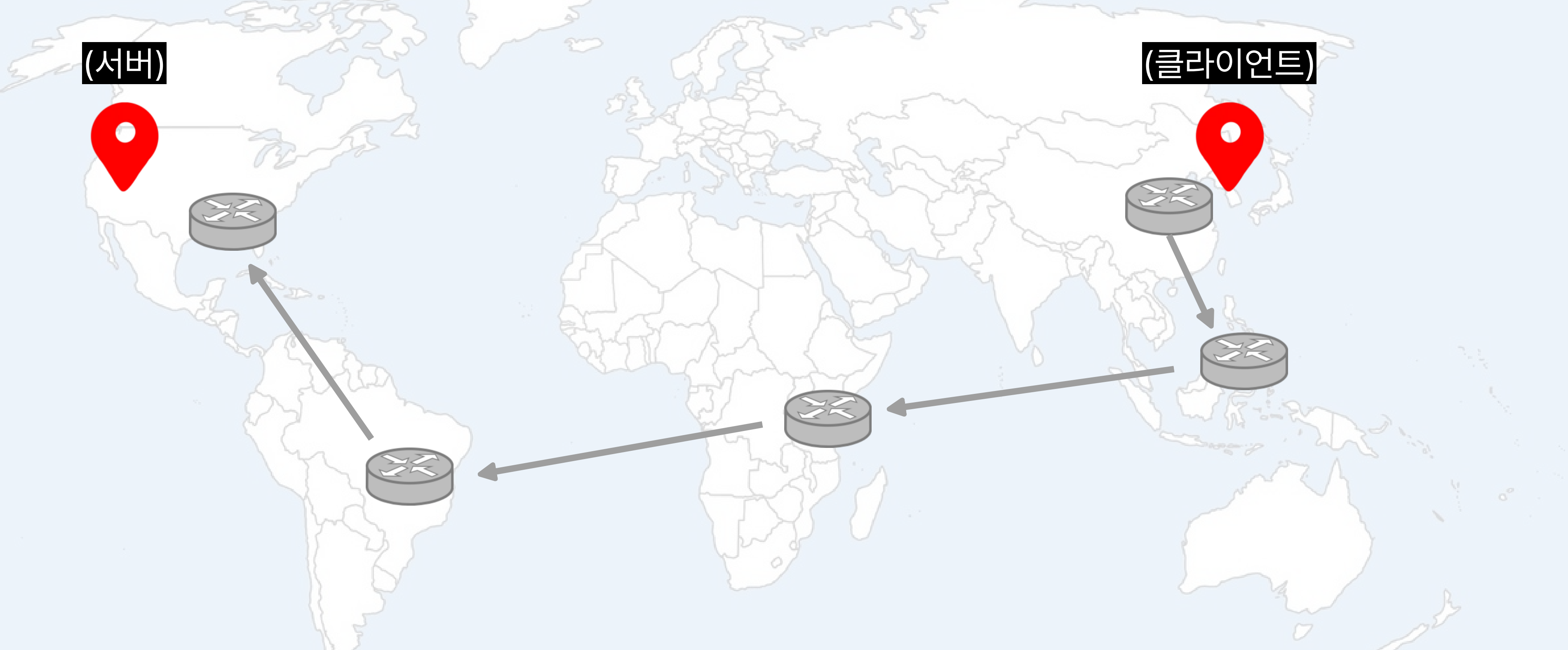
(해저 케이블)



# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 라우팅, 라우터

(서버)

(클라이언트)



물론 이때 LAN/WAN 공유기, 허브/스위치, 통신사 ISP 등등 매우 많은 개념이 있지만 지금은 생략..

# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 서버에서 처리

(앞서 최종 라우터에서 트래픽을 받음)

# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 서버에서 처리

(앞서 최종 라우터에서 트래픽을 받음)

**Web Server**

HTML, CSS, JS, 이미지 등 정적인 콘텐츠를 제공하는 서버  
(빠름)

**Web Application Server (WAS)**

# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 서버에서 처리

(앞서 최종 라우터에서 트래픽을 받음)

Web Server

동적 로직 처리, 데이터베이스 접근 등 동적 콘텐츠를 생성하는 애플리케이션

**Web Application Server (WAS)**

개발자가 서버를 개발한다는건 보통 이걸 다룸

# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 서버에서 처리

(앞서 최종 라우터에서 트래픽을 받음)

## **Web Server**

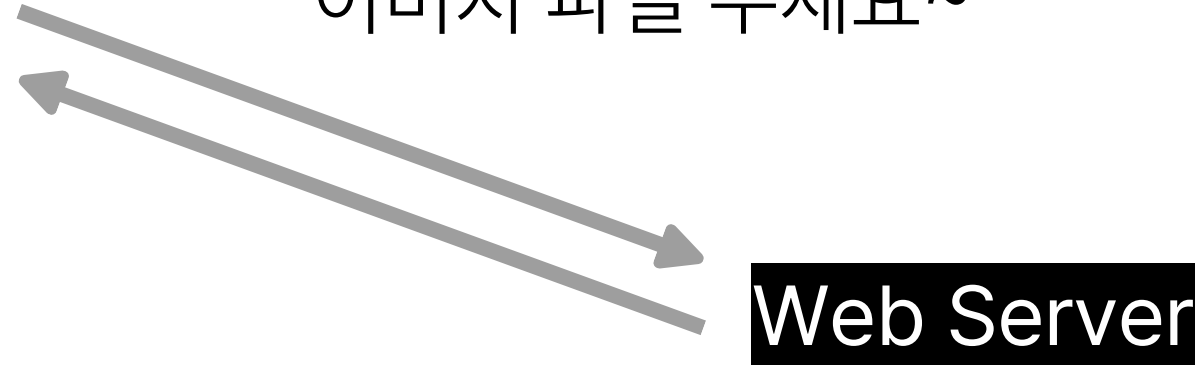
Nginx, Apache 등등

## **Web Application Server (WAS)**

Tomcat, JBoss 등등 + 수많은 개발 라이브러리/프레임워크 등등..

# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 서버에서 처리

이미지 파일 주세요~



내 선에서 해줄 수 있으니깐 바로 주겠음 〇〇

**Web Application Server (WAS)**

# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 서버에서 처리

게시글 목록 반환해주세요~

**Web Server**

WAS한테 받아왔으니 줄게요~

WAS야 클라이언트가 DB 접근해서 게시글 목록 달랜다

ㅇㅋ 목록 보내드릴

**Web Application Server (WAS)**



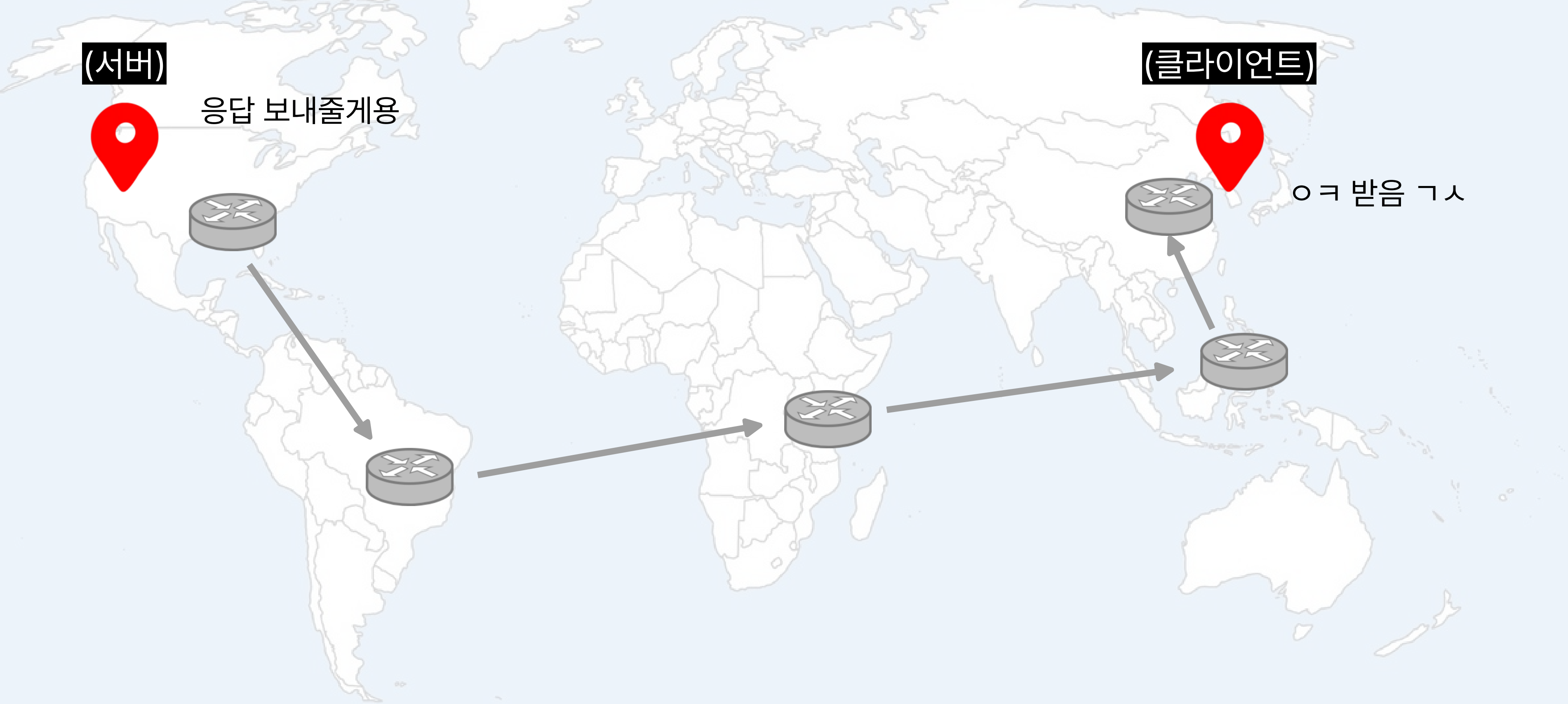
# 어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 라우팅, 라우터

(서버)

응답 보내줄게용

(클라이언트)

오key 받음 ㄱ사



## 프론트엔드와 백엔드

지금까지 웹에서의 통신 과정, 흐름을 살펴보았다면 이제 개발 분야로 넘어감

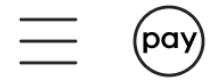
(프론트엔드/백엔드 개념은 IT 전반적으로 다양하게 사용되나, 여기선 웹 기준임)

## 프론트엔드와 백엔드 - Frontend

웹사이트나 앱에서 사용자가 눈으로 보고 상호작용하는 화면 (UI/UX)

# 프론트엔드와 백엔드 - Frontend

웹사이트나 앱에서 사용자가 눈으로 보고 상호작용하는 화면 (UI/UX)



N 검색어를 입력해 주세요.

서해수호의 날  
호국영웅들을 기억합니다

- 메일
- 카페
- 블로그
- 스토어
- 뉴스
- 증권
- 부동산
- 지도
- 웹툰
- 치지직

대웅제약 건강기능식품사업부

### 요즘 이게 대세라며? 1타 4피

대웅제약 에너지플라티늄

[오늘마감] 공식몰 초특가 ~68% >

대웅제약 건강기능식품

- 간 건강
- 혈압
- 식후혈당
- 콜레스테롤

Kim Jun Y... 님 | 네이버ID | 로그아웃 [→]

normal8781@naver.com

N+ 시작하기 | 쪽지 0 | 해피빈

83

블로그 | 메일 | 페이 | 카페 >

뉴스스탠드 · 언론사편집 / 엔터 / 스포츠 **LIVE** / 경제 / 쇼핑투데이

전체언론사 ▾ | 연합뉴스 · '이란발전소 공격 또 유예'...트럼프, 협상교착 속 확전도 부담

|          |             |                  |          |        |       |
|----------|-------------|------------------|----------|--------|-------|
| KBS NEWS | 스포츠서울       | 한겨레              | 일간스포츠    | 아시아투데이 | 프레시안  |
| BLOTER   | ZDNET Korea | The Korea Herald | 뉴스1      | MBC    | 아시아경제 |
| 한국경제TV   | 문화일보        | 스포츠조선            | 주간조선     | 산업일보   | metro |
| CNN NEWS | 이로운넷        | 식품지널 foodnews    | Ablenews | 헬스조선   | 한국금융  |

빅데이터/AI 솔루션은 슈어소프트테크

AI학습 데이터, 모델 품질 관리를 위한 솔루션!

슈어소프트테크

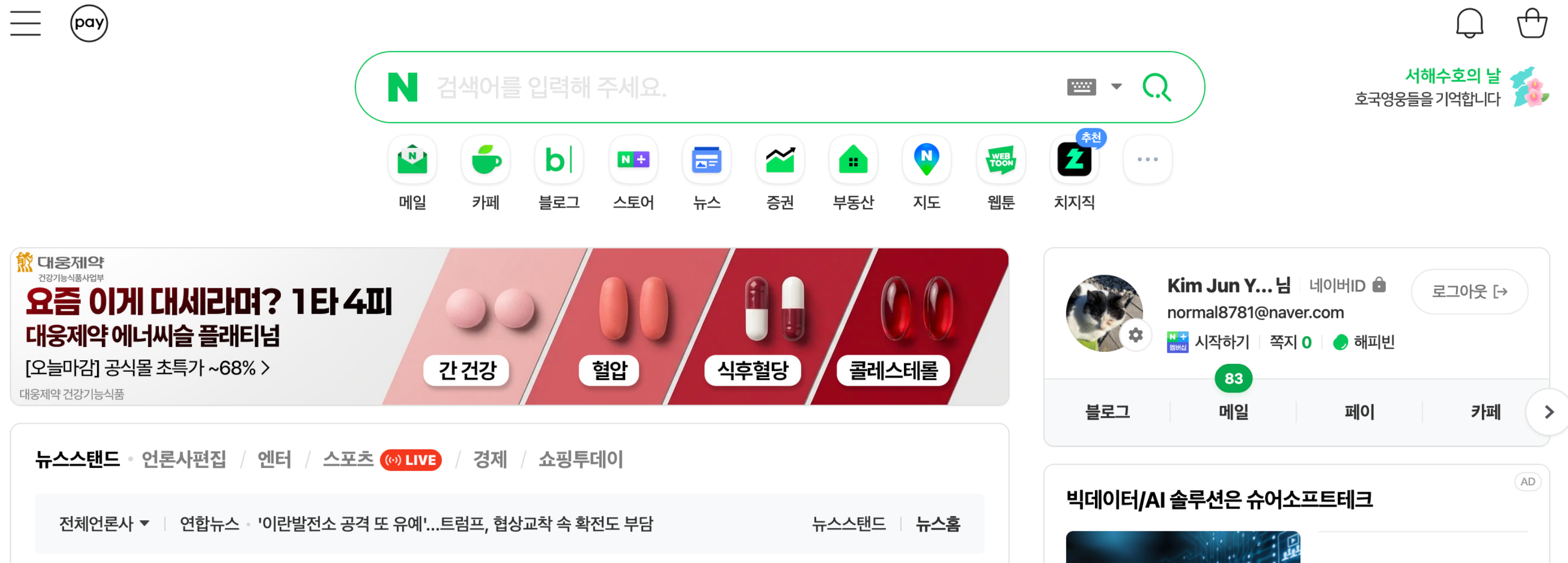
더 알아보기 >

기상특보 초미세먼지 매우나쁨

# 프론트엔드와 백엔드 - Frontend

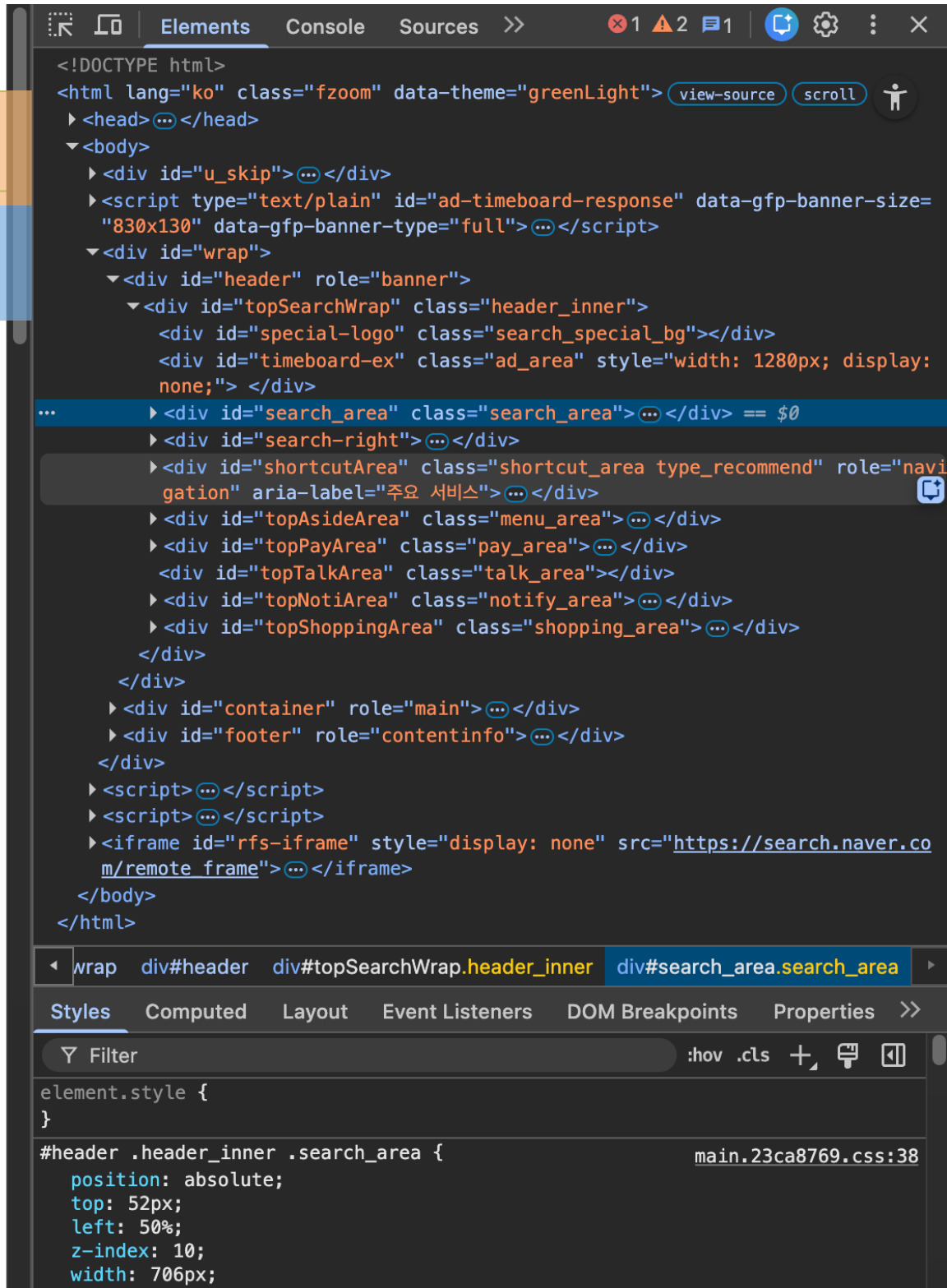
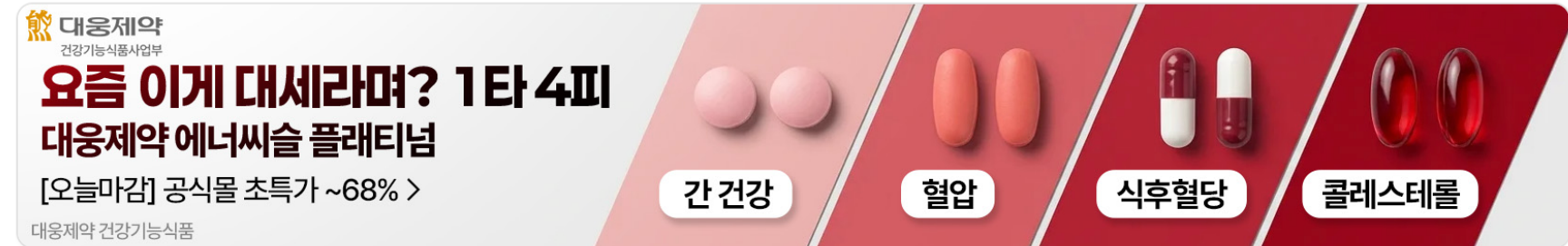
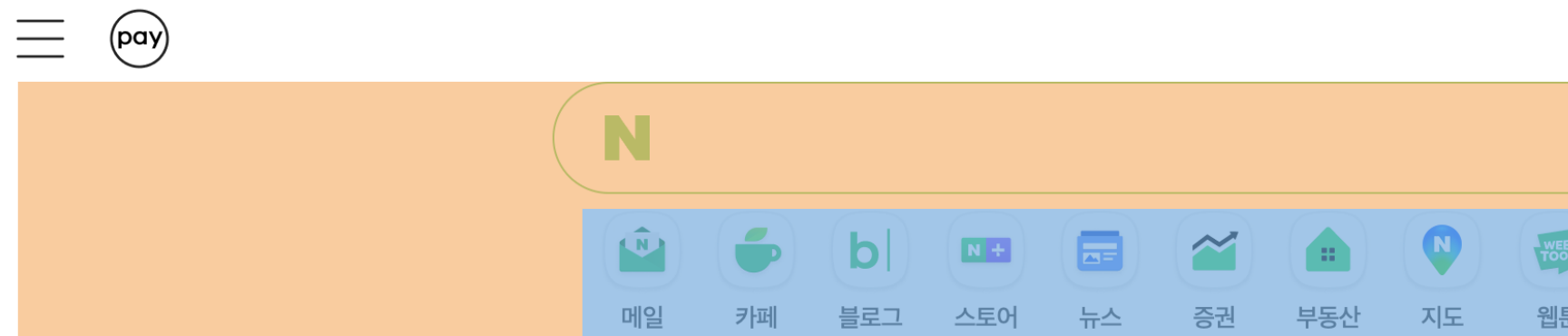
웹사이트나 앱에서 사용자가 눈으로 보고 상호작용하는 화면 (UI/UX)

기본적으로 HTML, CSS, JS로 구성됨. (React, VueJS 등도 있으나 결국엔 HTML 코드 덩어리)



프론트엔드 - 웹 페이지는 사실 코드 덩어리다.

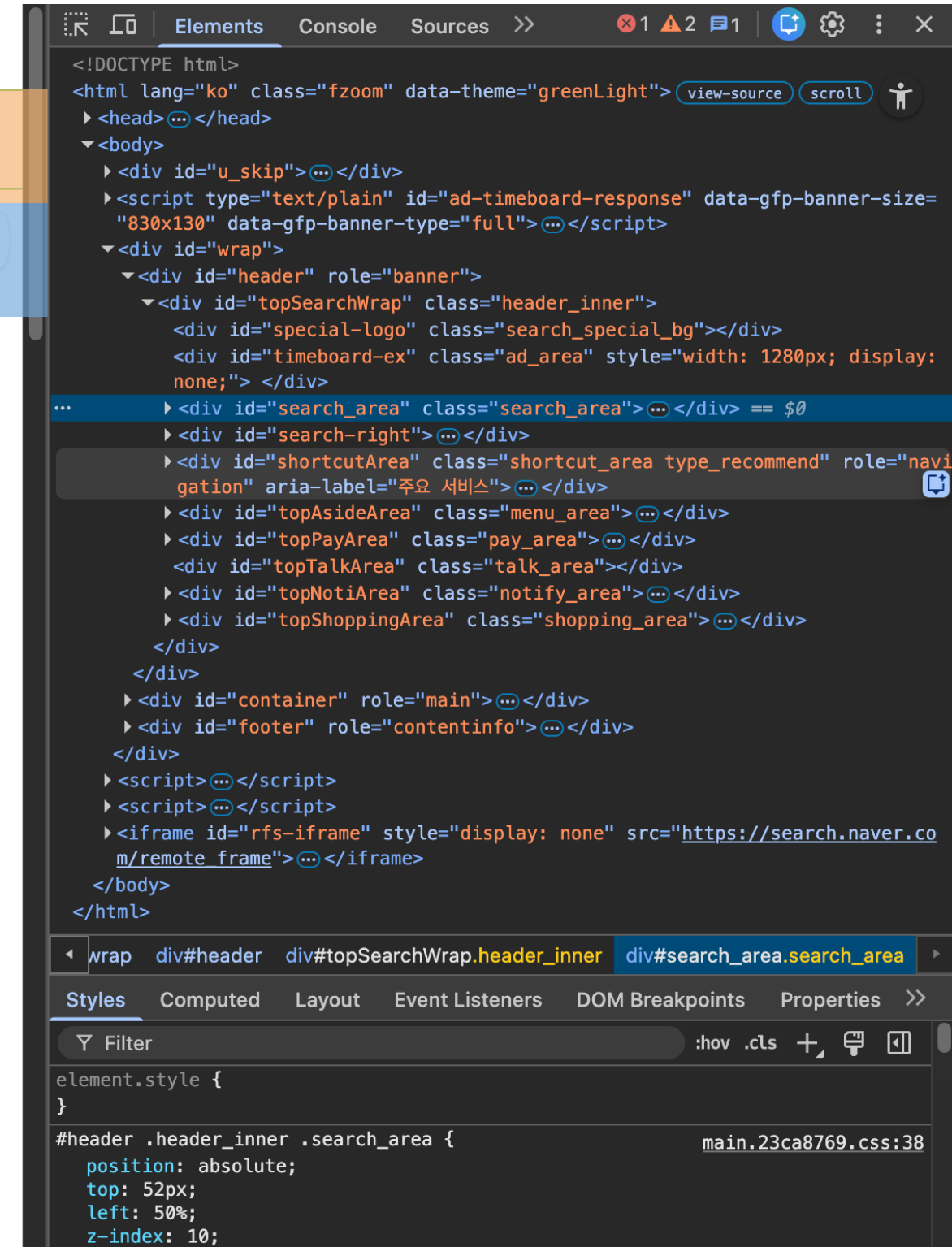
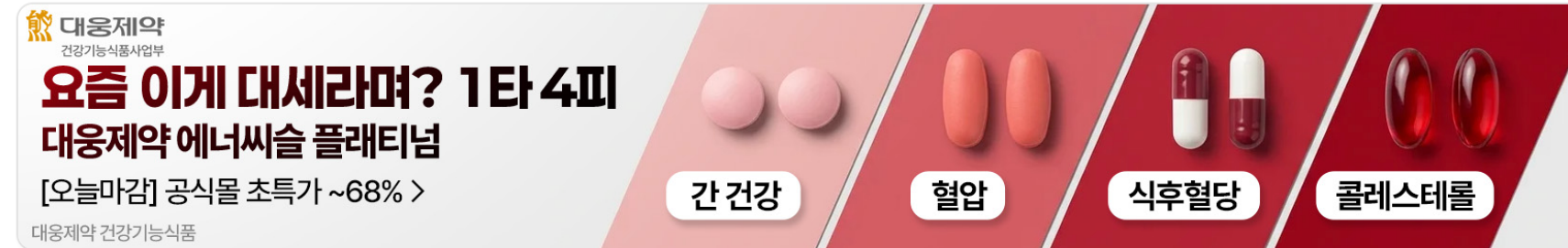
아무 웹 페이지에 접속하여 개발자 도구를 열어봅시다.



프론트엔드 - 웹 페이지는 사실 코드 덩어리다.

아무 웹 페이지에 접속하여 개발자 도구를 열어봅시다.

HTML(Hyper Text Markup Language)



프론트엔드 - 웹 페이지는 사실 코드 덩어리다.

아무 웹 페이지에 접속하여 개발자 도구를 열어봅시다.

## HTML(Hyper Text Markup Language)

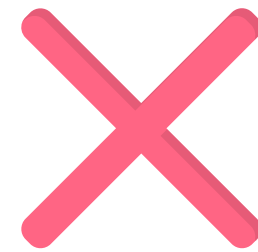
퀴즈) HTML은 프로그래밍 언어일까요?

프론트엔드 - 웹 페이지는 사실 코드 덩어리다.

아무 웹 페이지에 접속하여 개발자 도구를 열어봅시다.

## HTML(Hyper Text Markup Language)

퀴즈) HTML은 프로그래밍 언어일까요?

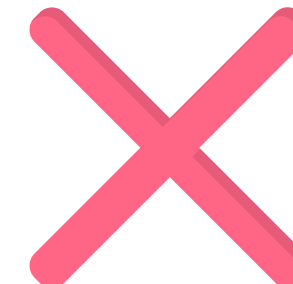


HTML은 디자인 요소를 만드는 "마크업" 언어임.

프로그래밍 언어는 컴퓨터에게 명령을 내리기 위한 언어

프론트엔드 - 웹 페이지는 사실 코드 덩어리다. HTML(Hyper Text Markup Language)

```
<!doctype html>
<html lang="en-US">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width" />
    <title>Planet data table</title>
    <link href="style.css" rel="stylesheet" type="text/css" />
  </head>
  <body>
    <h1>Planet data table</h1>
  </body>
</html>
```



Ps, Ai 처럼 디자인을 마우스로 만드는데 아닌 코드를 작성해야함

# 실습) 자신만의 HTML 페이지 만들어보기

1. VSCode 실행하기.

a. 반드시 VSCode만 사용해야 할까? (Feat. VSCode도 메모장일 뿐이다)

2. VSCode Live Server 익스텐션 설치하기

a. VSCode가 메모장보다 훨씬 강력한 도구가 된 이유.

3. 디렉토리 열기 및 HTML 문서 작성하기 (자기 소개 페이지 만들어보기)

4. !를 입력후 탭을 쳐서 나오는 내용은 무슨 의미일까?

a. 메타데이터의 중요성 (Feat. 검색 엔진 노출, SEO 등)

5. 주요 태그 설명 (h1...h6, p, span, img, input 등)

6. Live Server로 실행해보기

a. 퀴즈) Live Server로 실행하면 서버가 실행되는걸까?

b. 스스로 찾아보기) 다른 컴퓨터에서 내 페이지에 접속되게 해보기

# 프론트엔드만 있으면 무슨 문제가 생길까?

로그인하고 싶습니다. 프론트엔드에 로그인 메뉴가 있네?

네이버를 더 안전하고 편리하게 이용하세요

**NAVER** 로그인

[아이디 찾기](#) | [비밀번호 찾기](#) | [회원가입](#)

## 프론트엔드만 있으면 무슨 문제가 생길까?

로그인하고 싶습니다. 프론트엔드에 로그인 메뉴가 있네?



로그인 처리는 데이터베이스 접근, 세션/토큰 생성 등 내부 비즈니스 로직 실행이 필요함

## 프론트엔드만 있으면 무슨 문제가 생길까?

로그인하고 싶습니다. 프론트엔드에 로그인 메뉴가 있네?

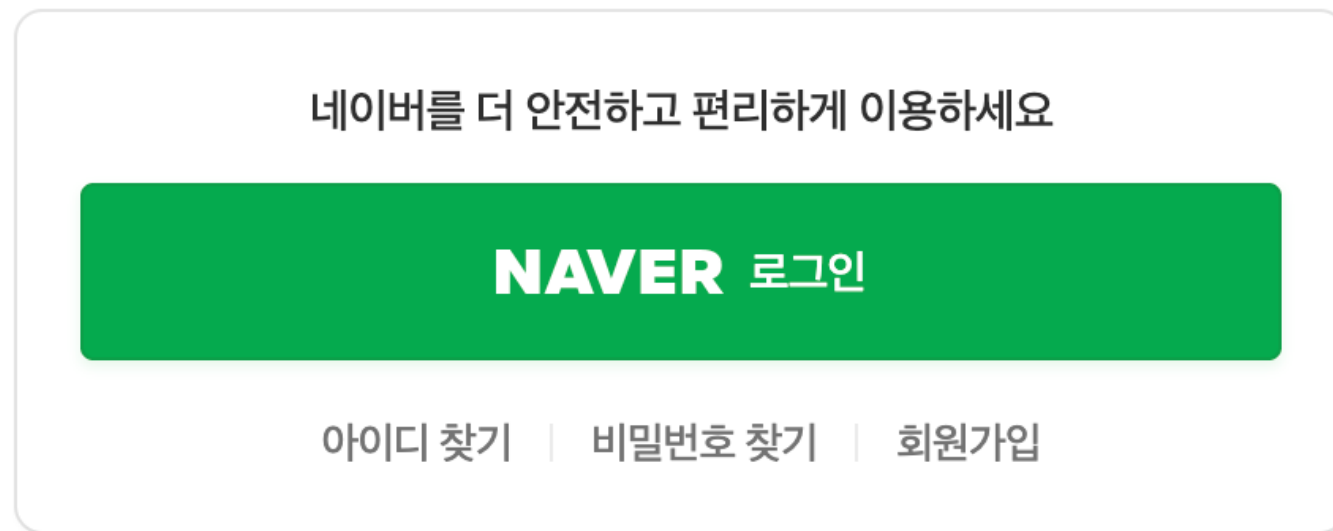


로그인 처리는 데이터베이스 접근, 세션/토큰 생성 등 내부 비즈니스 로직 실행이 필요함

" 프론트엔드는 그러한 작업을 처리하지 못함. 단순히 화면에 표시해주는 것. "

백엔드는 눈에 보이지 않는다.

Backend의 본질적인 뜻은 "눈에 보이지 않는" 이라는 의미를 가짐.



로그인 처리는 데이터베이스 접근, 세션/토큰 생성 등 내부 비즈니스 로직 실행이 필요함

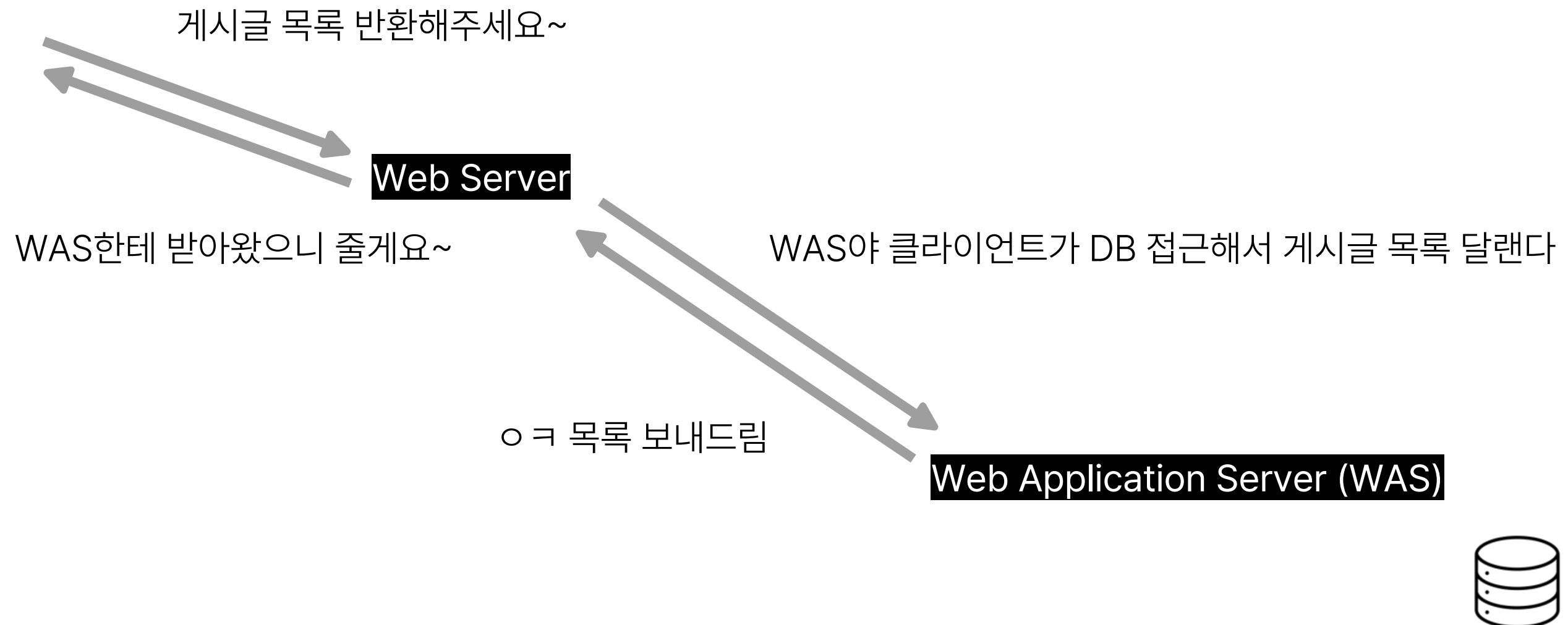
위와 같은 작업을 처리하는 것은 눈에 보이지 않는 백엔드의 영역임

백엔드 애플리케이션 = 그러한 작업을 처리하는 애플리케이션.

백엔드는 눈에 보이지 않는다.

아까 다뤘던 "서버"는 백엔드 애플리케이션을 실행해주는 공간임.  
개념적으로는 다른 개념이지만, 혼용해도 큰 문제는 없음.

어떻게 인터넷이 동작할까? (Feat. 네트워크 기초) - 서버에서 처리



퀴즈) 프론트엔드는 무조건 클라이언트, 백엔드는 무조건 서버일까?

퀴즈) 프론트엔드는 무조건 클라이언트, 백엔드는 무조건 서버일까?

정답: 아니다.

퀴즈) 프론트엔드는 무조건 클라이언트, 백엔드는 무조건 서버일까?

정답: 아니다.



