

HTTP

Network

CONTENTS

01
HTTP ?

02
HTTP의 특징

01

HTTP?

01. CONTENTS

HTTP

Hyper Text Transfer Protocol

HTML과 같은 Hyper Text를 빠르게 교환하기 위한 프로토콜의 일종

현재는, HTTP 프로토콜에 모든 데이터를 넣어서 전송한다!

01. CONTENTS

HTTP 역사

HTTP/0.9 1991년 : GET 메서드만 지원, HTTP 헤더 X

HTTP/1.0 1996년 : 메서드, 헤더 추가

HTTP/1.1 1997년 : 가장 많이 사용하는 버전

HTTP/2 2015년 : 성능개선

HTTP/3 진행중 : TCP 대신에 UDP를 사용, 성능 개선

01. CONTENTS

HTTP

 m.naver.com	302	http/1.1
---	-----	----------

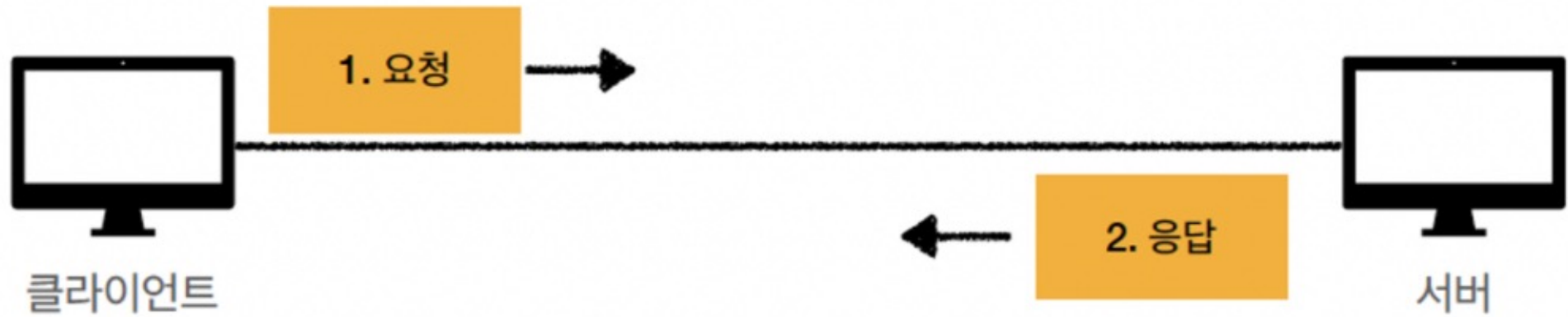
대기업인 네이버도 1.1버전을 쓴다 !

02

HTTP 의 특징

02. CONTENTS

클라이언트 서버 구조



02. CONTENTS

Stateful : 상태유지

고객 : 이 노트북 얼마가요?

점원 : 100만원 입니다

고객 : 2개 구매하겠습니다

점원 : 200만원 입니다. 신용카드, 현금 중에 어떤걸로 구매하시겠어요?

고객 : 신용카드로 구매하겠습니다.

점원 : 200만원 결제 완료되었습니다.

02. CONTENTS

Stateful : 상태유지

If 점원이 중간에 바뀔 경우

고객 : 이 노트북 얼마가요?

점원A : 100만원 입니다

고객 : 2개 구매하겠습니다

점원B : 무엇을 2개 구매하시겠어요?

고객 : 신용카드로 구매하겠습니다.

점원C : 무슨 제품을 몇개 신용카드로 구매하시겠어요?

02. CONTENTS

Stateless : 무상태

고객 : 이 노트북 얼마가요?

점원 : 100만원 입니다

고객 : 노트북 2개 구매하겠습니다

점원 : 노트북 2개는 200만원 입니다.

신용카드, 현금중에 어떤걸로 구매하시겠어요?

고객 : 노트북 2개를 신용카드로 구매하겠습니다.

점원 : 200만원 결제 완료되었습니다.

02. CONTENTS

Stateless : 무상태

If 점원이 중간에 바뀔 경우

고객 : 이 노트북 얼마가요?

점원A : 100만원 입니다

고객 : 노트북 2개 구매하겠습니다

점원B : 노트북 2개는 200만원 입니다.

신용카드, 현금중에 어떤걸로 구매하시겠어요?

고객 : 노트북 2개를 신용카드로 구매하겠습니다.

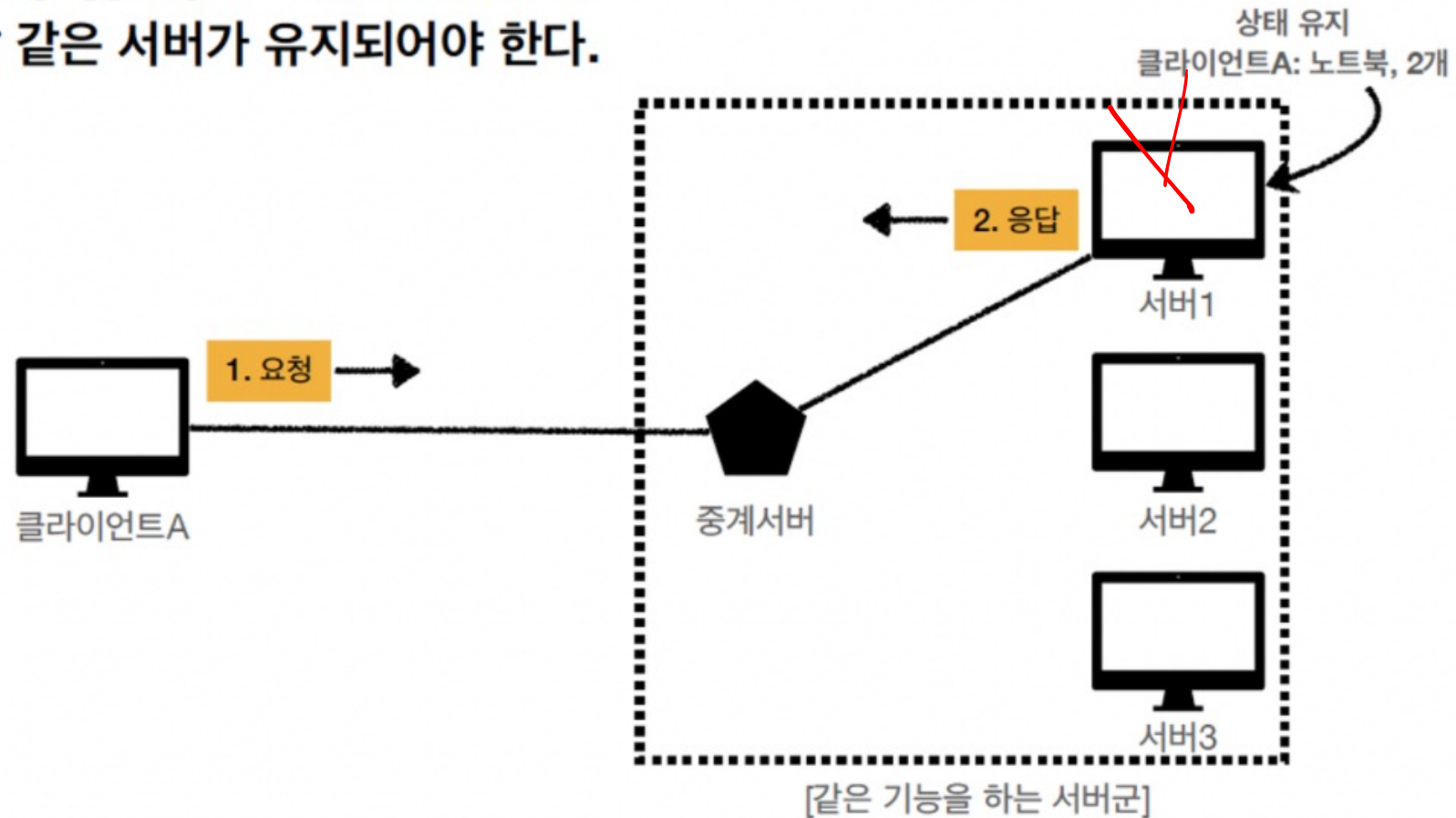
점원C : 200만원 결제 완료되었습니다.

02. CONTENTS

Stateful vs Stateless

상태 유지 - Stateful

항상 같은 서버가 유지되어야 한다.

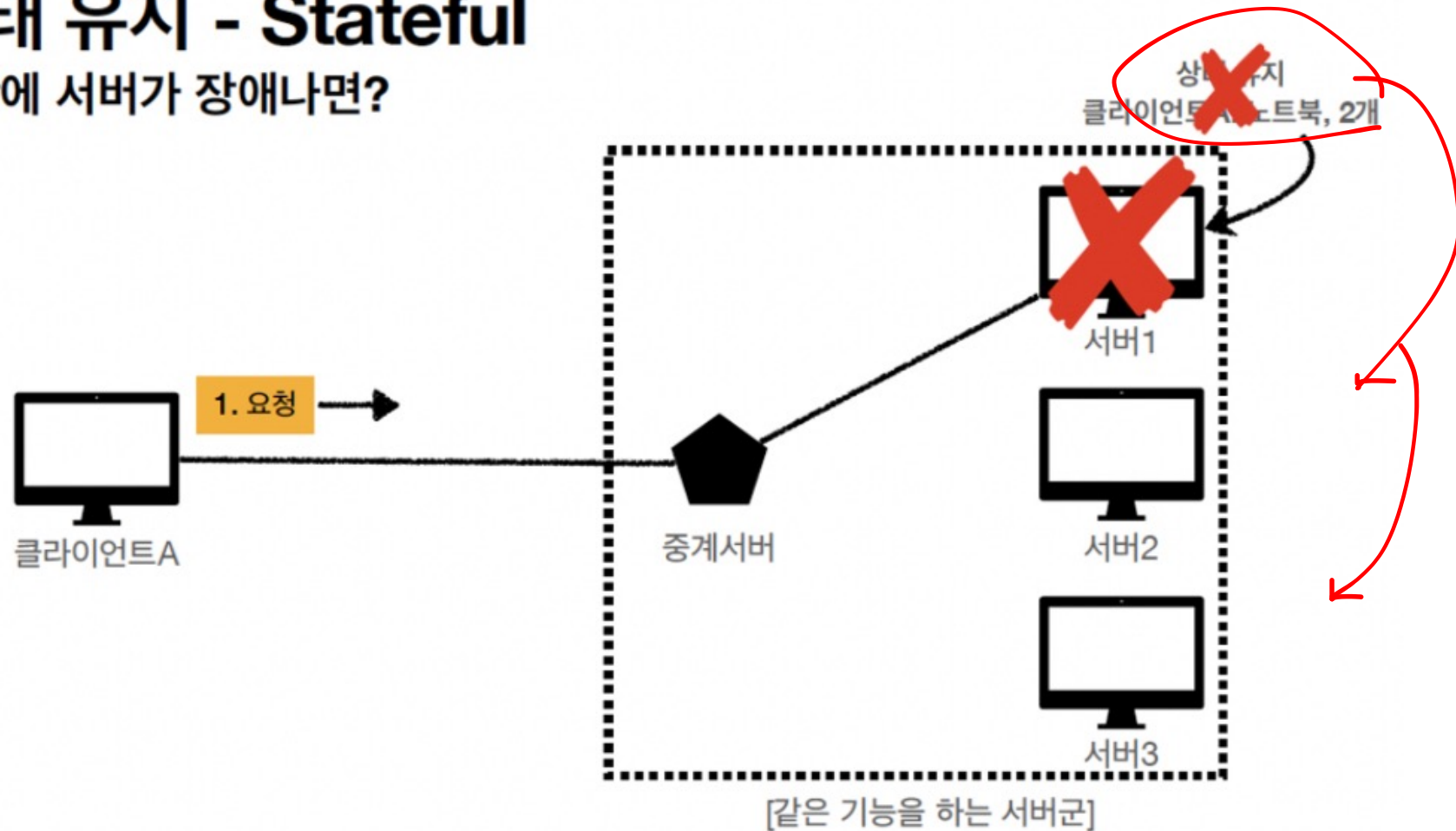


02. CONTENTS

Stateful vs Stateless

상태 유지 - Stateful

중간에 서버가 장애나면?

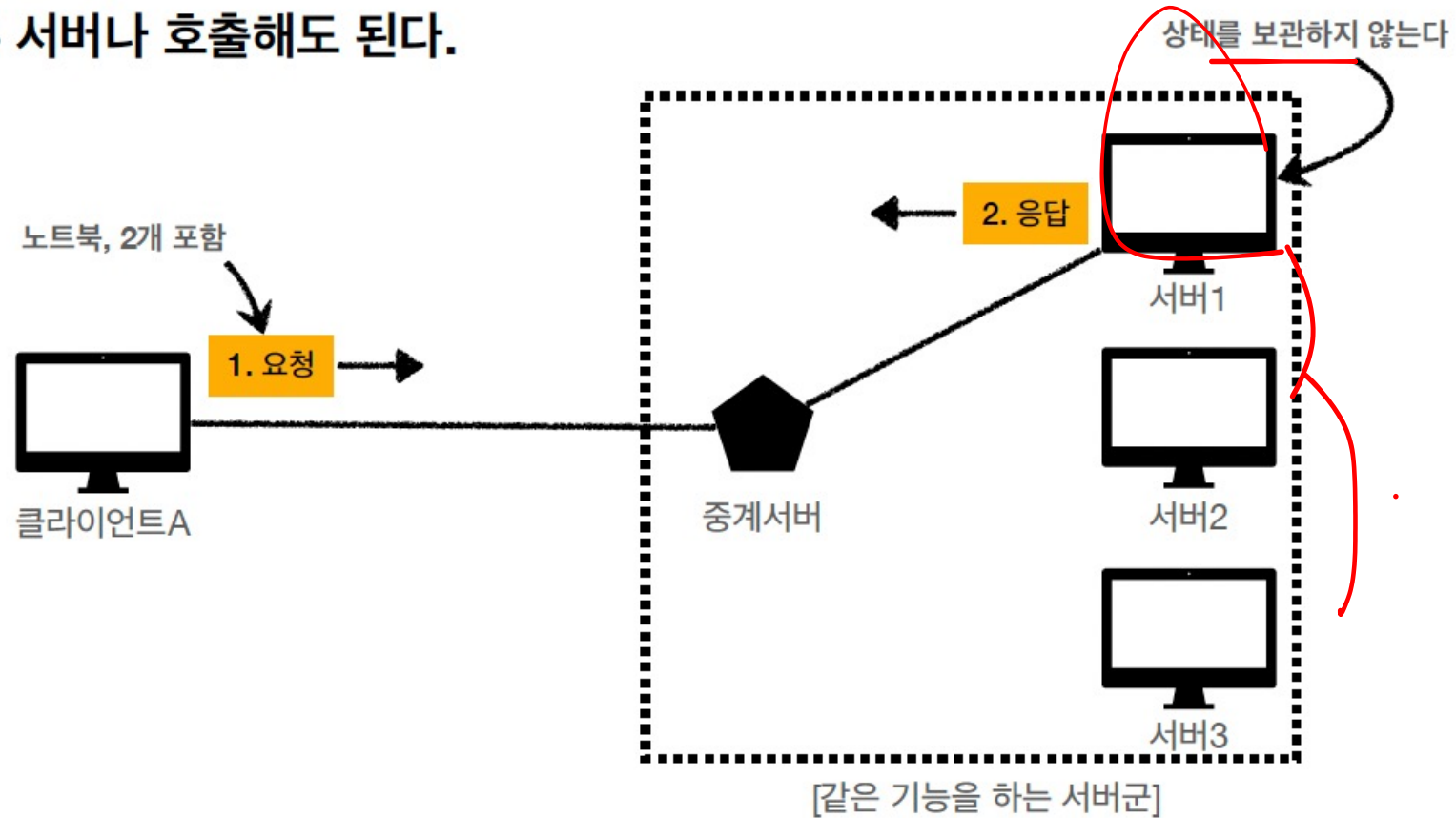


02. CONTENTS

Stateful vs Stateless

무상태 - Stateless

아무 서버나 호출해도 된다.

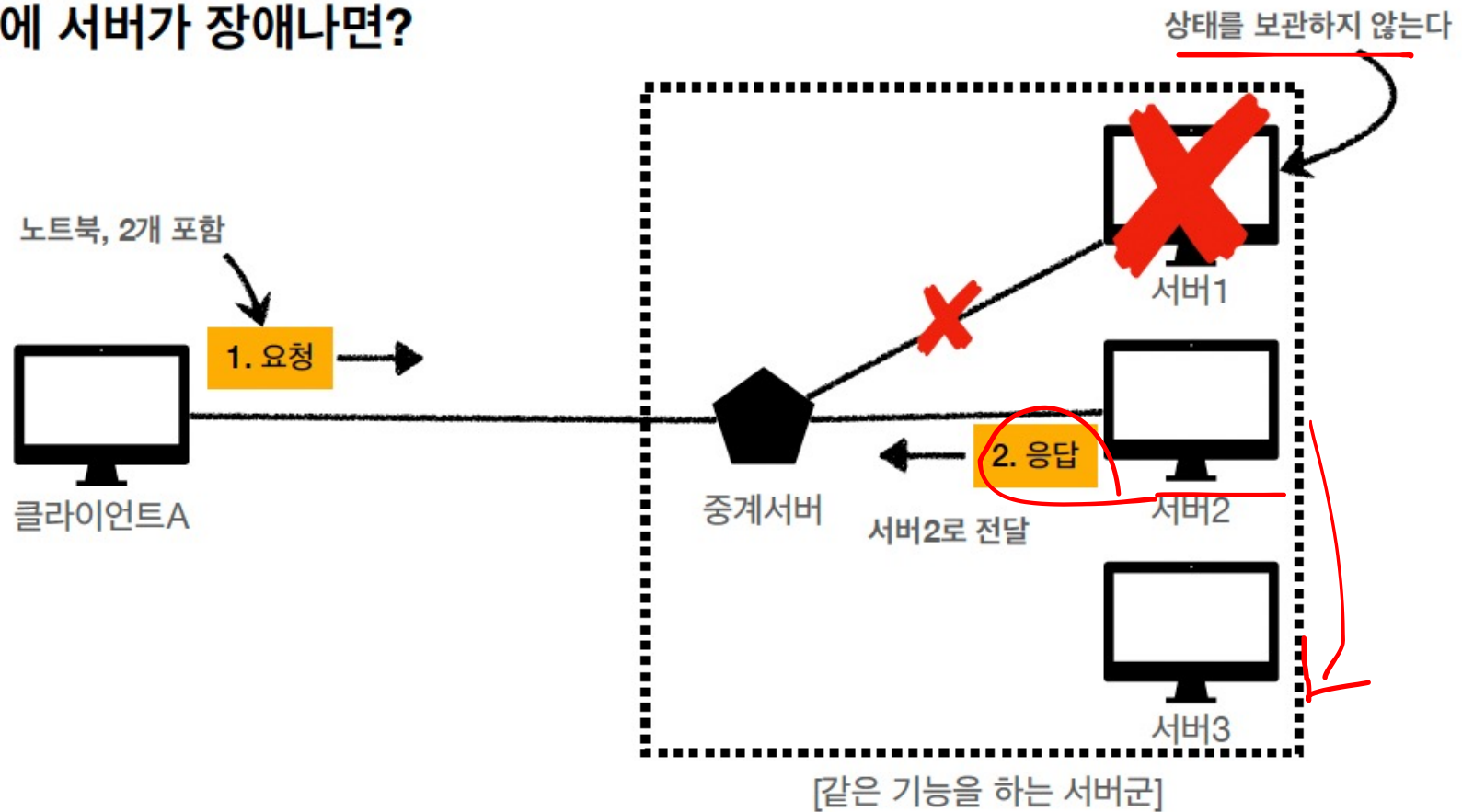


02. CONTENTS

Stateful vs Stateless

무상태 - Stateless

중간에 서버가 장애나면?



02. CONTENTS

Stateless 한계점

- 로그인 없이 단순한 서비스 소개의 경우 무상태로 설계하기 용이함
- 로그인과 같은 상태유지를 반드시 해야하는 경우,
로그인 브라우저에 있는 쿠키와 서버의 세션을 조합해서 상태를 유지함
- 전송해야하는 데이터양이 비교적 많다 -> 문맥을 모르기 때문

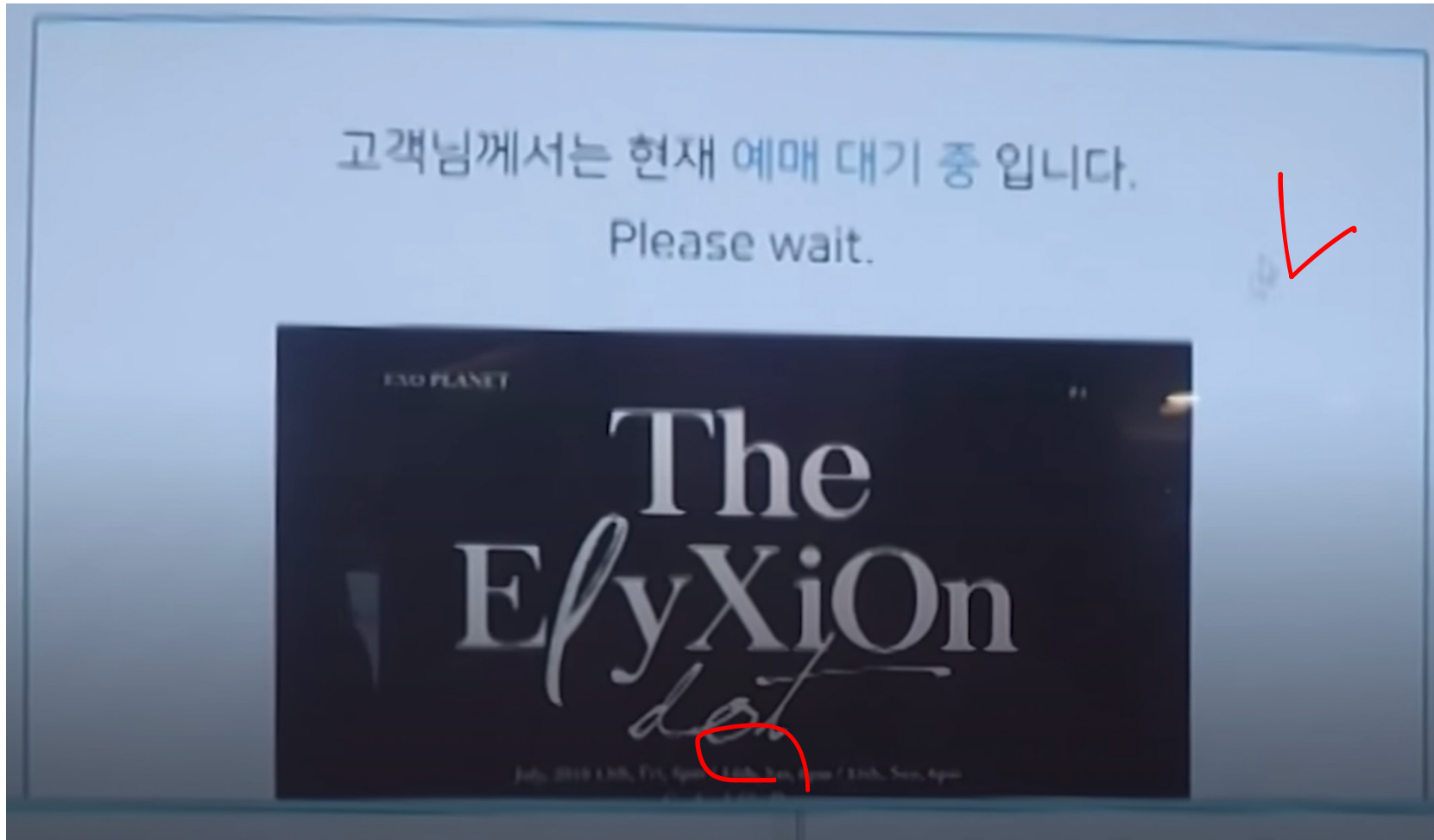
02. CONTENTS

서버개발자들이 어려워하는 업무

- 같은 시간에 딱 맞추어 발생하는 대용량 트래픽
- 예시) 콘서트 티켓팅, 선착순 이벤트, 수강신청
- 이런 상황은 수만명이 동시에 요청을 한다!

02. CONTENTS

서버 개발자들이 어려워하는 업무



02. CONTENTS

비연결성

HTTP는 기본적으로 연결을 유지하지 않는다.
즉, 응답을 돌려받으면 연결을 종료한다.

연결을 최소한으로 유지하기 때문에 자원을 효율적으로 사용함

02. CONTENTS

비연결성 한계

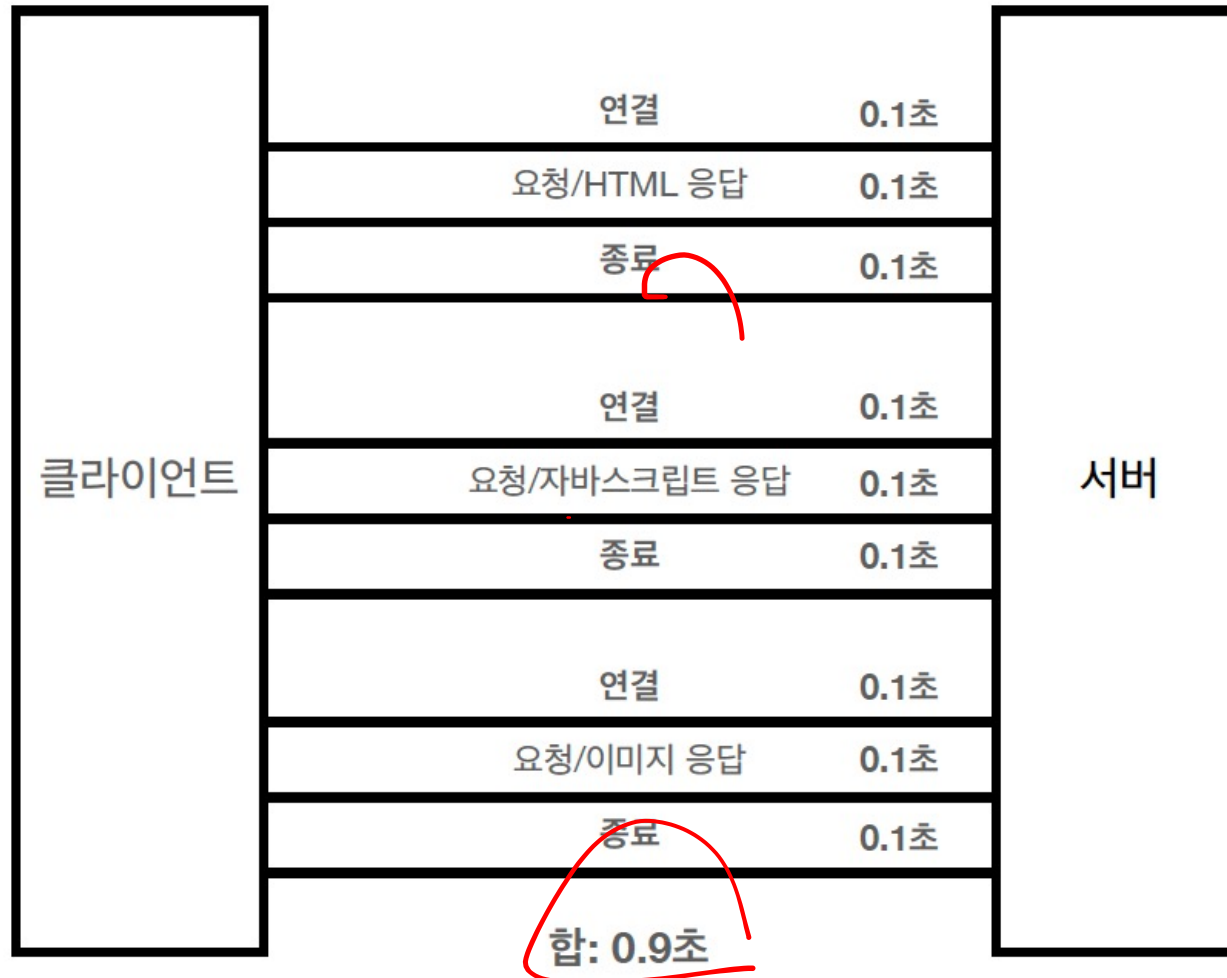
TCP/IP 연결을 새로 맺을때마다 - 3 way handshake 시간이 발생

단순 HTML이 아닌 다양한 자바스크립트등의 수많은 자원을 다운로드 하기 때문에 하나의 자원을 다운로드 하고 연결을 끊는 방식은 적절X

-> 위의 두가지 문제를 HTTP 지속 연결로 해결함

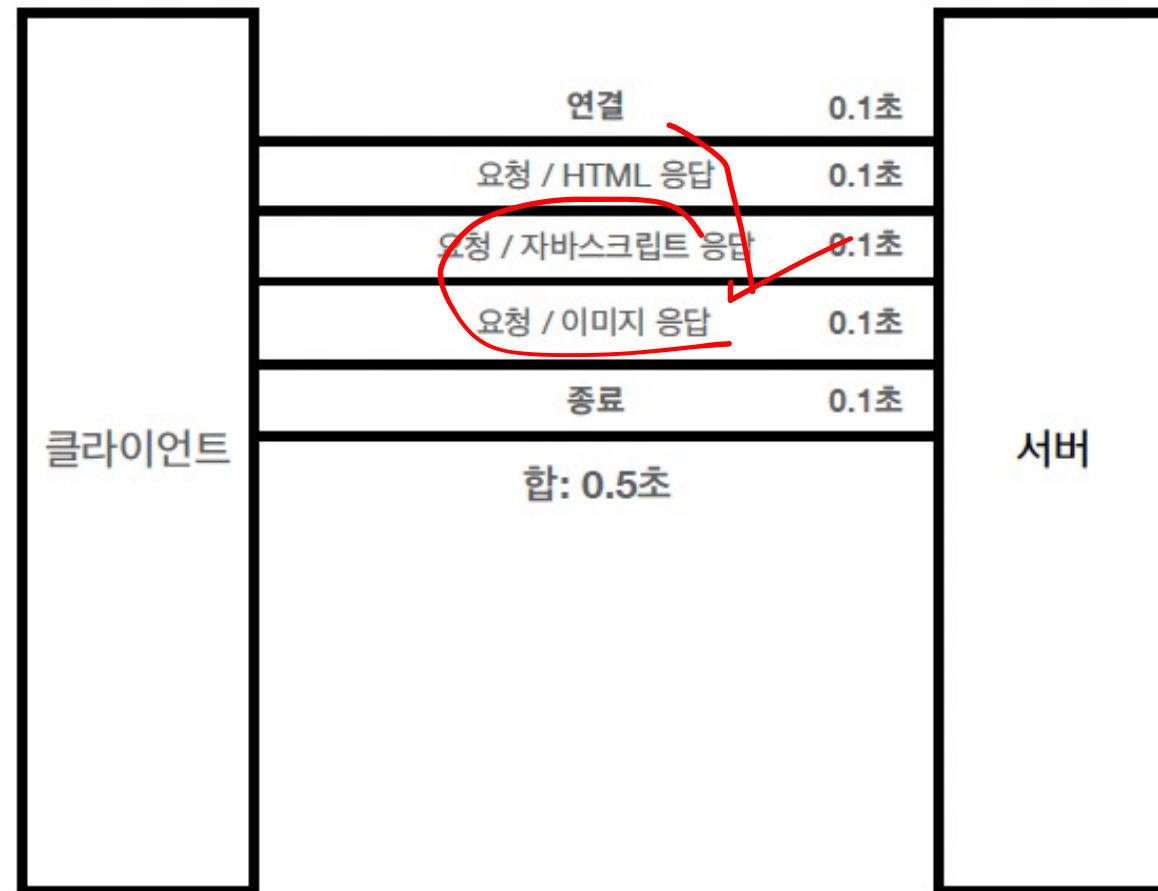
02. CONTENTS

비연결성



02. CONTENTS

비연결성



02. CONTENTS

QUIZ

1. HTTP 특징 중 상태유지와 무상태에 대해 설명하고 둘의 차이점에 대해 설명해주세요.
2. 무상태의 한계와 무상태로 상태유지를 어떻게 구현하는지 설명해주세요
3. 비연결성이 무엇인지 말하고, 비연결성의 한계점을 어떻게 극복했는지 설명하세요



Thank you

감사합니다

