

# 인증

- Authorization

: cli. 인증 정보를 serv.에 전달

- WWW-Authenticate

: resource 접근 시 필요한 인증 방법 정의

---

## Authorization

↳ cli. 인증 정보를 serv.에 전달

cf) auth 인증

ex) Authorization: Basic xxxx ~~~

---

## WWW-Authenticate

← 접근을 했는데 인증이 잘 안 됐을 때

↳ resource 접근 시 필요한 인증 방법 정의

- 401 Unauthorized res. 와 함께 사용

ex) WWW-Authenticate

: Newauth realm="apps", type=1,  
title



# 쿠키

- Set-Cookie

: serv.에서 cli.로 cookie 전달 (response)

- Cookie

: cli.가 serv.에서 받은 cookie 저장하고  
HTTP req.시 serv.로 전달

---

쿠키 미사용 ; 처음 welcome page 접근



---

쿠키 미사용 ; 로그인



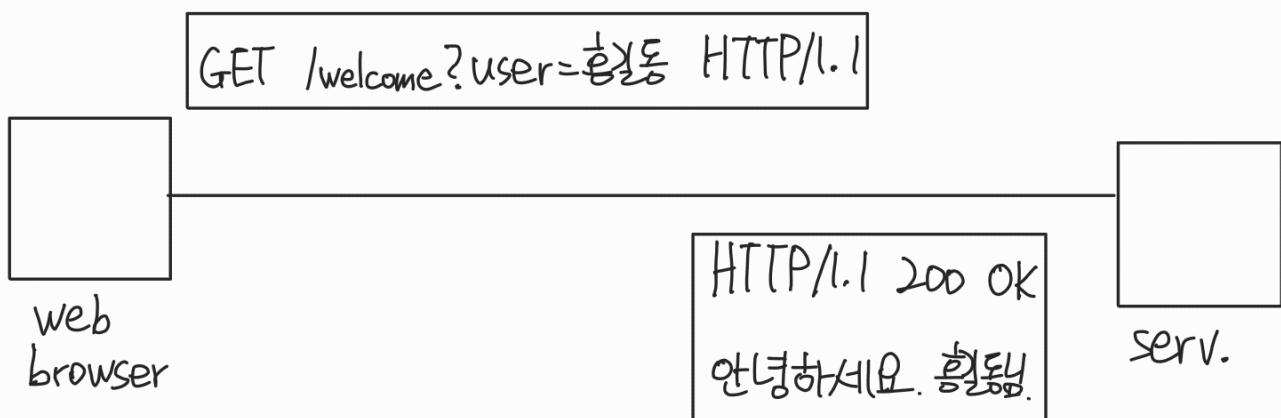
쿠키 미사용 ; 로그인 이후 welcome page 접근



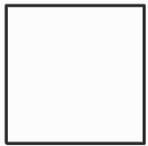
## Stateless (무상태)

- HTTP는 stateless protocol.
- cli. 와 serv. 가 req. 과 res. 주고 받으면  
⇒ 연결 끊어진다.
- cli. 가 다시 req. 시 serv. 는 이전 req. 기억 못 함!
- cli. 와 serv. 는 서로 state를 유지 X.

쿠키 미사용 ; (대안) 모든 req. 에 user 정보 포함



(대안) 모든 req. 과 link 에 user 정보 포함



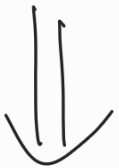
w.  
b.

GET /welcome?user=홍길동 HTTP/1.1

GET /board?user=홍길동 HTTP/1.1

GET /order?user=홍길동 HTTP/1.1

GET /xxx... ?user=홍길동 HTTP/1.1

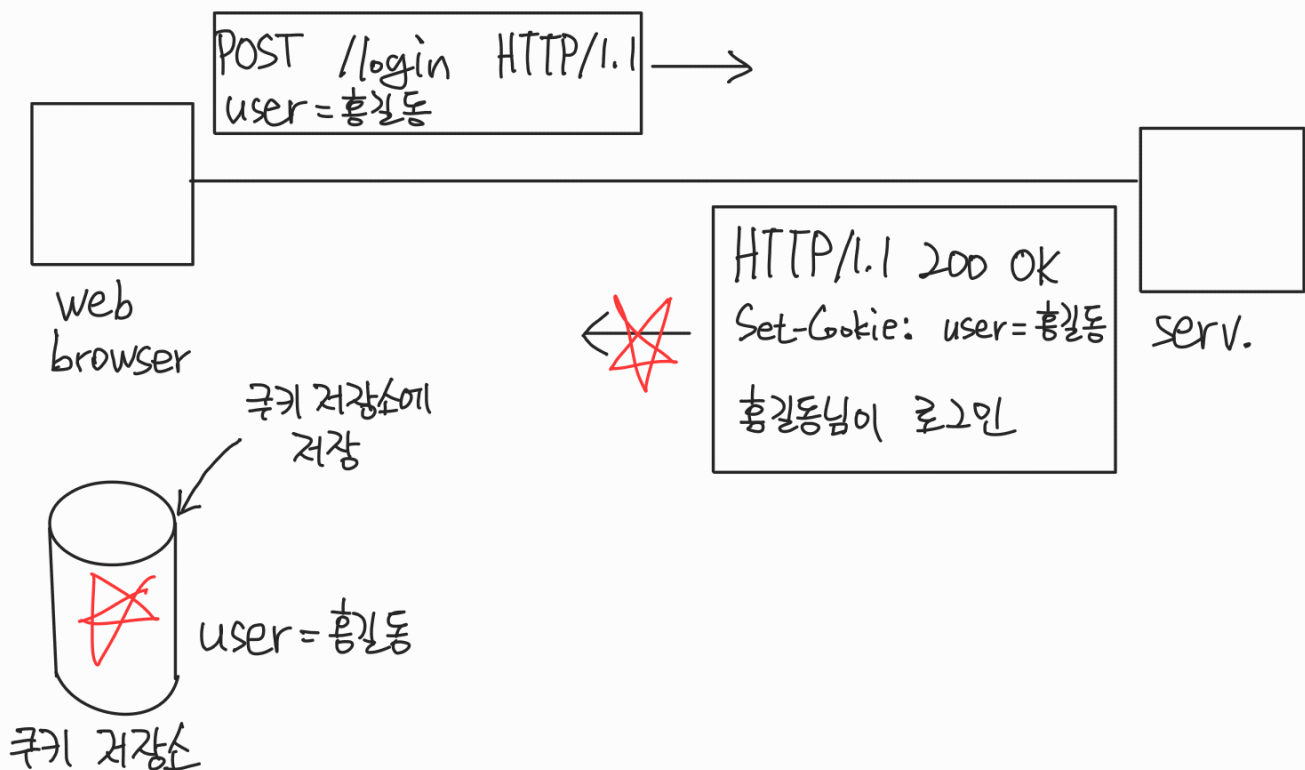


모든 req. 에 정보를 넘기는 문제

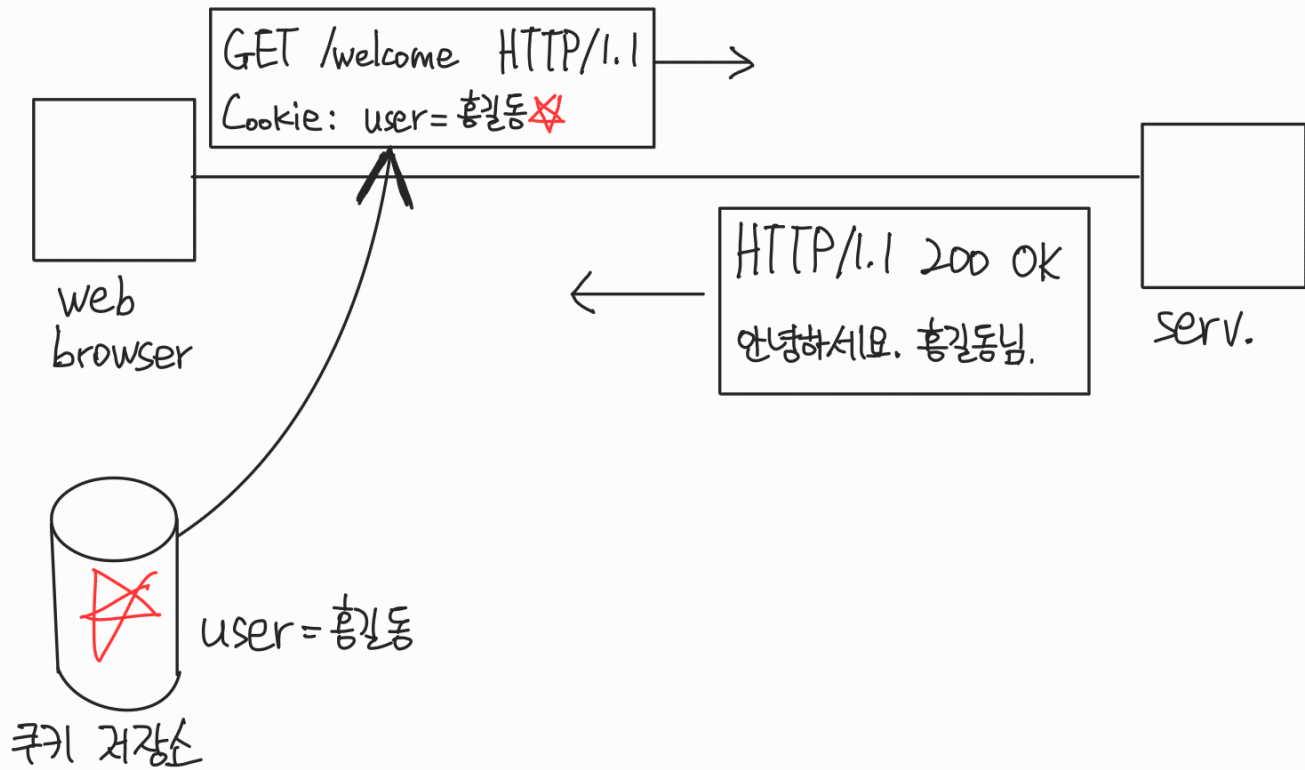
- 모든 req. 에 user 정보가 포함되도록 개발 필요
- browser 완전 종료 후 다시 여는 경우 ???

---

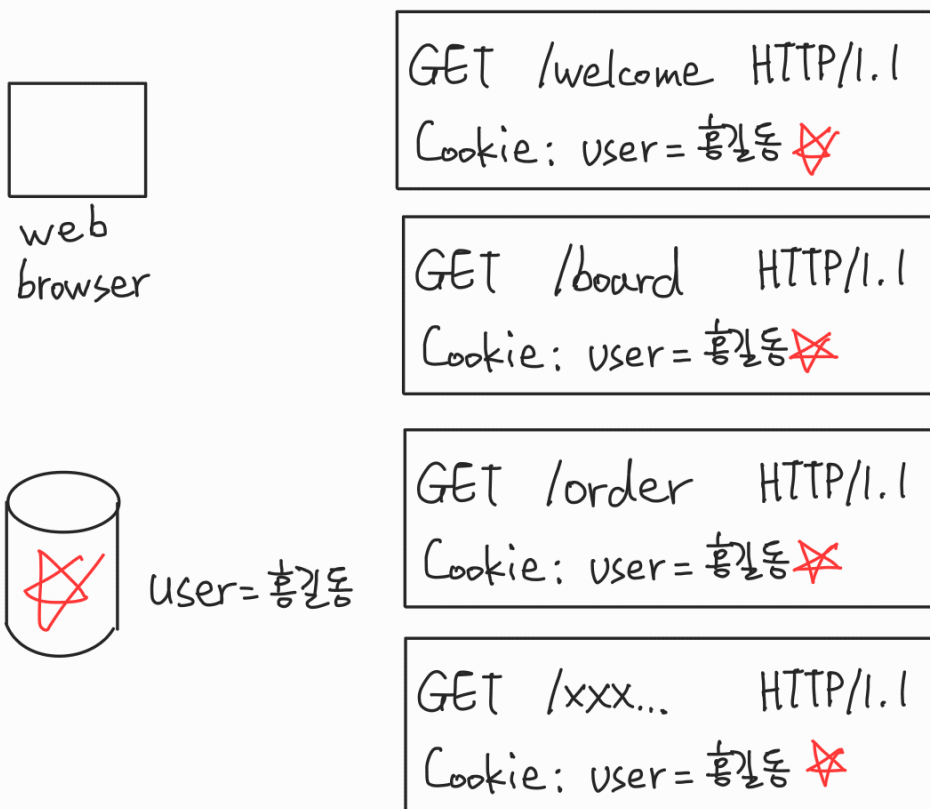
쿠키 ; 로그인



쿠키 ; 로그인 이후 welcome page 접근



모든 req.에 cookie 정보 자동 포함



# 쿠키

ex) Set-Cookie: sessionId=abcde1234; expires=Sat, 26-Dec-2020 ~ GMT;  
path=/; domain=.google.com; Secure

## · 사용처

- user login session 관리
- 광고 정보 tracking

## · cookie 정보는 항상 serv.에 전송됨.

- network traffic 추가 유발
- 최소한의 정보만 사용 (session id, 인증 token)
- if serv.에 전송 안되고 web browser 내부에 data 저장 가능하면?  
⇒ web storage (localStorage, sessionStorage)

★ 보안에 민감 data는 저장 X

쿠키 - 생명주기; expires, max-age

- Set-Cookie: expires=Sat, 26-Dec-2020 04:39:21 GMT

↳ 만료일이 되면 cookie 삭제

- Set-Cookie: max-age=3600 (3600초)

↳ 0이나 음수 지정 시 cookie 삭제

- session cookie

: 만료 날짜 생략 시, browser 종료 시까지만 유지

- 영속 쿠키

: 만료 날짜 입력 시, 해당 날짜까지 유지