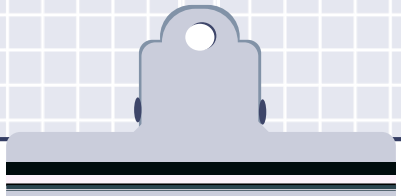


HTTP Caching

파일 최적화

20303 박순민





목차

01 Cache?
| HTTP Cache?

02 ETag

03 Cache-control

04 시연

05 느낀점

02 - I HTTP Cache

재사용 가능한 HTTP resource



HTML



.

.

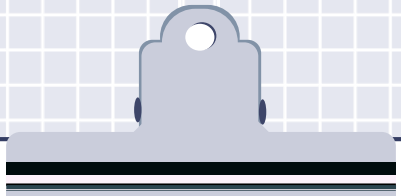
.

----->

임시 저장

google.com

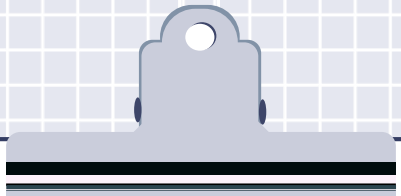




03 Etag

ETag: "xyz123" ← Strong Etag

ETag: W/"xyz123" ← Weak Etag



03 Etag

여러 가지 용도로 사용

1. 불필요한 데이터 전송 방지

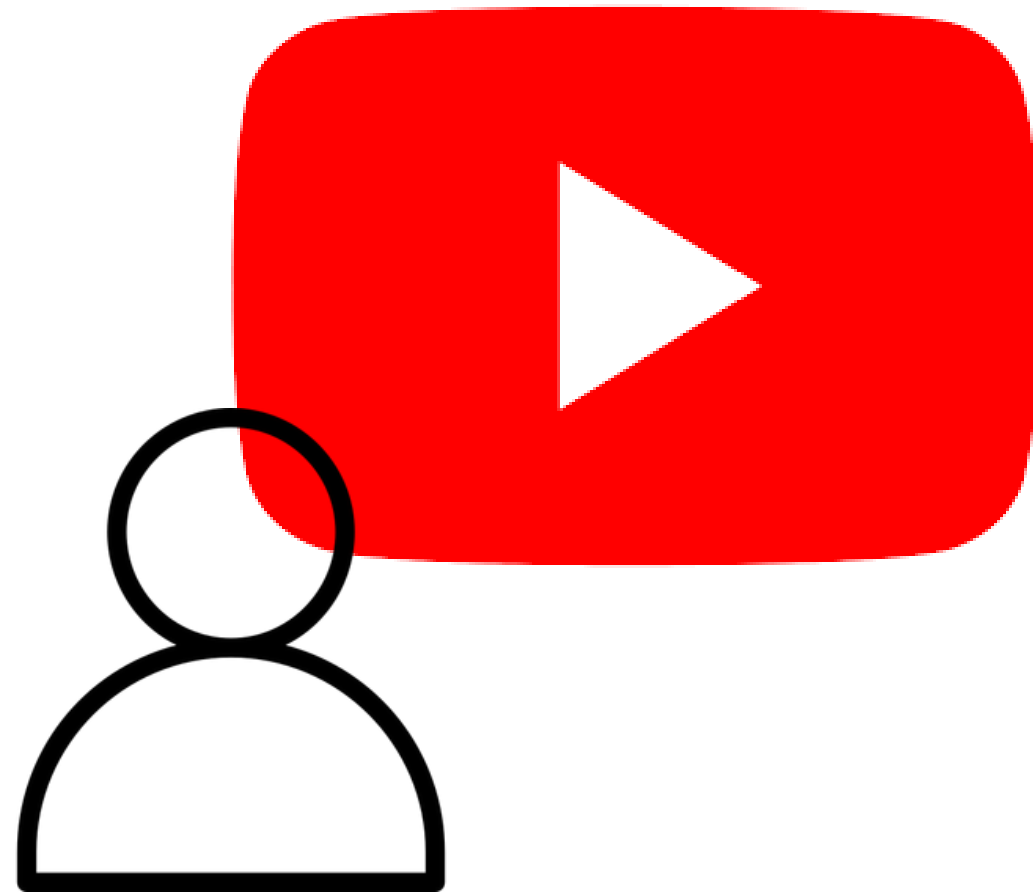
1. 충돌 방지

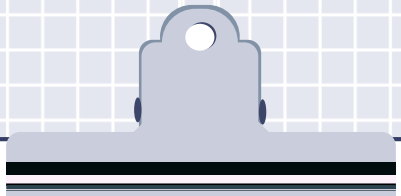
1. 대역폭 절감

03 Etag

Size : 100MB

Etag: "6a474321"

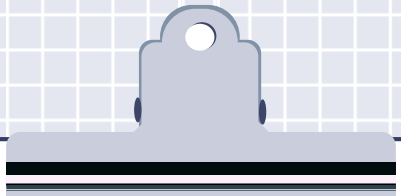




03 Etag

if-none-match

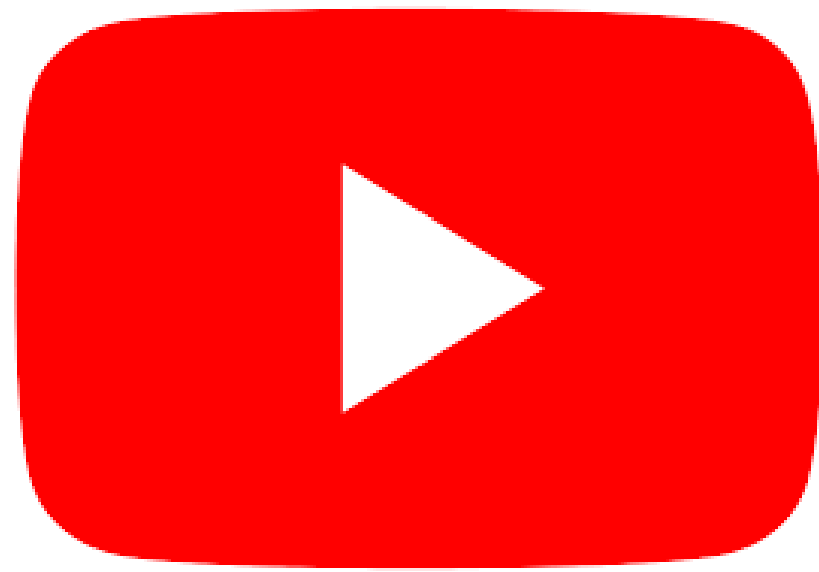
식별자를 통해 변화가 없으면
cache 된 것을 사용하는 프로세스



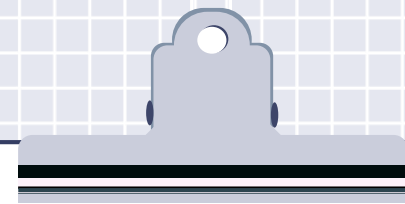
03 Etag

변화 X	if-none-match	변화 O
304 Not Modified	식별자를 통해 변화가 없으면	200 OK
cache 된 것을 사용하는	cache 된 것을 사용하는	새 파일
옛 파일 그대로 사용		+
		새 Etag 이름 내려 받음

03 Etag



영상을 바꾸지 않았다면?



그런데

