

Cache-Control

1. Khái niệm.

- **HTTP Cache-Control** chứa các lệnh trong cả yêu cầu và phản hồi kiểm soát bộ nhớ đệm trong trình duyệt và bộ nhớ đệm được chia sẻ.

Ví dụ : CDN, Proxy..

- Trình duyệt (browser cache).
- Proxy cache (CDN, reverse Proxy, intermediate server).

Cú pháp :

Cache-Control : <directive>, <directive>, ...

- chỉ thị bộ nhớ đệm không phân biệt chữ hoa hay chữ thường. Tuy nhiên, nên sử dụng, chữ thường vì một số triển khai công nhận ra lệnh viết hoa.

- nhiều lệnh được phép và giải được phân tách bằng dấu phẩy

Ví dụ : Cache-Control : max-age=180, public

- một số chỉ thị đối số tùy chọn. khi một đối số cung cấp, nó được phân tách khỏi tên chỉ thị bằng 1 ký hiệu bằng (). Thông thường, đối số cho các lệnh là số nguyên do đó không được đặt trong các ký tự trích dẫn

Ví dụ : Cache-Control : max-age=12

2. Hoạt động.

Trong request header (Client → Server).

- client nói với server hoặc proxy cách nó muốn xử lý cache.

Ví dụ :

Cache-Control : no-cache

→ Client muốn bypass cache, yêu cầu bản gốc từ server.

Trong response header (Server → Client).

- Server ra luật cho client/proxy phải cache thế nào.

Ví dụ :

Cache-Control : max-age=3600, public

→ dữ liệu có thể cache trong 3600 giây (1h) và cache được ở mọi nơi (browser, CDN, proxy).

3. Các directive quan trọng.

Request	Response	Giải thích chi tiết
max-age=seconds	max-age=seconds	Client : chỉ chấp nhận cache nếu nó không quá x giây. Server : cho phép cache dữ liệu trong x giây kể từ lúc response.
max-stale[=seconds]	–	Client : sẵn sàng chấp nhận dữ liệu cũ (stale). Nếu có số giây → chấp nhận stale đến giới hạn đó.
min-fresh=seconds	–	Client : yêu cầu dữ liệu cache phải còn “tươi” ít nhất x giây. Nếu sắp hết hạn thì không dùng.
–	s-maxage=seconds	Server : giống max-age nhưng chỉ áp dụng cho shared cache (proxy, CDN). Browser vẫn dùng max-age.
no-cache	no-cache	Client : yêu cầu bỏ qua cache, phải xác minh lại với server. Server : cho phép cache, nhưng mỗi lần dùng lại phải xác minh với server (revalidation).
no-store	no-store	Không được lưu ở đâu cả (client hay proxy). Dùng cho dữ liệu nhạy cảm (token, banking).

no-transform	no-transform	Yêu cầu proxy/CDN không được “biến đổi” dữ liệu (VD: nén hình ảnh, đổi format).
only-if-cached	–	Client : chỉ lấy từ cache, không được gọi server. Nếu không có cache → trả về lỗi 504 .
–	must-revalidate	Cache (client/proxy) phải xác minh lại với server khi dữ liệu hết hạn, không được dùng stale.
–	proxy-revalidate	Giống must-revalidate nhưng áp dụng cho shared cache (proxy).
–	must-understand	Nếu cache không hiểu directive thì không được bỏ qua , phải xử lý theo chuẩn.
–	private	Chỉ cache ở client cá nhân (browser), không cache ở proxy/CDN .
–	public	Có thể cache ở mọi nơi (client, proxy, CDN).
–	immutable	Tài nguyên không bao giờ thay đổi. Browser có thể dùng cache mà không cần revalidate (thường dùng cho file có version hash).
–	stale-while-revalidate=seconds	Cho phép dùng dữ liệu cũ (stale) trong khi âm thầm fetch bản mới từ server.
stale-if-error=seconds	stale-if-error=seconds	Cho phép dùng dữ liệu stale nếu server trả về lỗi (5xx) hoặc không kết nối được, tối đa x giây.