A HTTP-fejlécek (HTTP headers) az ügyfél (pl. böngésző) és a szerver közötti kommunikáció során használt metaadatokat tartalmazzák. Ezek az üzenetek részeként kerülnek továbbításra a kérések (request) és válaszok (response) során. Egy HTTP-fejléc mindig kulcs-érték párokból áll.

**Válaszoknál használt fejlécek  
 melyek a szerver által küldött információkat tartalmazzák**

**Content-Type**

A Content-Type HTTP fejléc azt jelzi, hogy a kérésben vagy válaszban található adat milyen típusú. Ez a fejléc segít a szervernek és a kliensnek abban, hogy megértsék, hogyan kell értelmezni az adatokat. Például, ha egy szerver JSON formátumban küld adatokat, akkor a Content-Type fejléc a application/json értékre lesz állítva.

**Példák:**

* Content-Type: text/html: Az adat HTML formátumban van.
* Content-Type: application/json: Az adat JSON formátumban van.
* Content-Type: text/plain: Az adat sima szöveg formátumban van.
* Content-Type: application/xml: Az adat XML formátumban van.

A Content-Type fejlécet akkor használják, amikor az adatokat a kliens küldi a szervernek (például egy POST kérésben), vagy amikor a szerver válaszol egy kérésre.

**Accept-Charset**

Az Accept-Charset fejléc arra utal, hogy a kliens milyen karakterkódolásokat képes elfogadni a válaszban. Ez lehetővé teszi a szerver számára, hogy a választ a kívánt karakterkódolásban küldje vissza. Ha a szerver nem képes a kívánt karakterkódolásban válaszolni, akkor az alapértelmezett karakterkódolást kell használnia, vagy hibát kell jeleznie.

**Példák:**

* Accept-Charset: utf-8: A kliens UTF-8 karakterkódolást vár.
* Accept-Charset: iso-8859-1: A kliens ISO-8859-1 karakterkódolást vár.
* Accept-Charset: \*: A kliens bármilyen karakterkódolást elfogad.

**Accept-Encoding**

Az Accept-Encoding fejléc arra utal, hogy a kliens milyen típusú tömörítést képes elfogadni a válaszban. Ez segít csökkenteni a hálózati sávszélességet, mivel a válaszok tömörítése kisebb adatforgalmat eredményezhet. A válaszban a szerver a kért tömörítés formátumot alkalmazza, ha lehetséges.

**Példák:**

* Accept-Encoding: gzip: A kliens képes elfogadni a gzip tömörítést.
* Accept-Encoding: deflate: A kliens képes elfogadni a deflate tömörítést.
* Accept-Encoding: br: A kliens képes elfogadni a Brotli tömörítést.
* Accept-Encoding: \*: A kliens bármilyen tömörítési formátumot elfogad.

**Cache-Control**

A Cache-Control fejléc a webes alkalmazásokban használt egyik legfontosabb fejléc, amely a válasz gyorsítótárazásának viselkedését szabályozza. Ezzel a fejléccel lehet megadni, hogy a válaszokat hogyan és milyen ideig tárolhatják a köztes proxyk és a böngészők, vagy hogy egyáltalán tárolhatók-e.

**Példák:**

* Cache-Control: no-cache: Az adatok nem tárolhatók gyorsítótárban.
* Cache-Control: no-store: A válasz nem tárolható el a gyorsítótárban.
* Cache-Control: public: A válasz nyilvános gyorsítótárazásra is alkalmas.
* Cache-Control: private: A válasz csak a felhasználó böngészőjében tárolható.
* Cache-Control: max-age=3600: A válasz legfeljebb 1 órán keresztül tárolható a gyorsítótárban.
* Cache-Control: must-revalidate: A gyorsítótárban lévő válasz csak akkor használható, ha az érvényessége nem járt le.

**Kéréseknél használt fejlécek  
melyek az ügyfél által küldött metaadatokat atartalmazzák**

* Host: A kért webhely címe (pl. Host: example.com).
* User-Agent: Az ügyfél azonosítása (pl. Mozilla/5.0 (Windows NT 10.0; Win64; x64))
* Accept: A támogatott válaszformátumok (pl. Accept: application/json).
* Authorization: Hitelesítési adatok küldése (pl. Authorization: Bearer <token>).
* Referer: Ahol a felhasználó az adott oldalt meglátogatta (Referer: https://google.com).
* Cookie: Az ügyfél által tárolt sütik továbbítása a szervernek.

**Általános fejlécek  
melyek mind kéréseknél, mind pedig válaszoknál előfordulhatnak**

* Connection: Kapcsolat típusa (keep-alive, close). Nyitva kell hagyni a kapcsolatot a kérés teljesítését követően, vagy le kell zárni.
* Date: A szerver válaszküldési ideje (Date: Tue, 18 Mar 2025 12:00:00 GMT).

**Különleges fejlécek**

CORS (Cross-Origin Resource Sharing): Engedélyezi vagy tiltja más domainekről érkező kéréseket.

* Access-Control-Allow-Origin: \*
* Access-Control-Allow-Methods: GET, POST