**Headers can be grouped according to their contexts:**

General header: Headers applying to both requests and responses but with no relation to the data eventually transmitted in the body.

Request header: Headers containing more information about the resource to be fetched or about the client itself.

Response header: Headers with additional information about the response, like its location or about the server itself (name and version etc).

**Authentication:**

WWW-Authenticate:

Defines the authentication method that should be used to gain access to a resource.

Authorization:

Contains the credentials to authenticate a user agent with a server.

Proxy-Authenticate:

Defines the authentication method that should be used to gain access to a resource behind a Proxy server.

Proxy-Authorization:

Contains the credentials to authenticate a user agent with a proxy server.

**Caching:**

Age:

The time in seconds the object has been in a proxy cache.

Cache-Control

Specifies directives for caching mechanisms in both requests and responses.

Expires:

The date/time after which the response is considered stale.

Pragma:

Implementation-specific header that may have various effects anywhere along the request-response chain. Used for backwards compatibility with HTTP/1.0 caches where the Cache-Control header is not yet present.

Warning:

A general warning field containing information about possible problems.

**Conditionals:**

Last-Modified:

It is a validator, the last modification date of the resource, used to compare several versions of the same resource. It is less accurate than ETag, but easier to calculate in some environments. Conditional requests using If-Modified-Since and If-Unmodified-Since use this value to change the behavior of the request.

ETag:

It is a validator, a unique string identifying the version of the resource. Conditional requests using If-Match and If-None-Match use this value to change the behavior of the request.

If-Match:

Makes the request conditional and applies the method only if the stored resource matches one of the given ETags.

If-None-Match:

Makes the request conditional and applies the method only if the stored resource doesn't match any of the given ETags. This is used to update caches (for safe requests), or to prevent to upload a new resource when one is already existing.

If-Modified-Since:

Makes the request conditional and expects the entity to be transmitted only if it has been modified after the given date. This is used to transmit data only when the cache is out of date.

If-Unmodified-Since:

Makes the request conditional and expects the entity to be transmitted only if it has not been modified after the given date. This is used to ensure the coherence of a new fragment of a specific range with previous ones, or to implement an optimistic concurrency control system when modifying existing documents.

**Connection management:**

Connection:

Controls whether the network connection stays open after the current transaction finishes.

Keep-Alive:

Controls how long a persistent connection should stay open.

**Content negotiation:**

Accept:

Informs the server about the types of data that can be sent back. It is MIME-type.

Accept-Charset:

Informs the server about which character set the client is able to understand.

Accept-Encoding:

Informs the server about the encoding algorithm, usually a compression algorithm, that can be used on the resource sent back.

Accept-Language:

Informs the server about the language the server is expected to send back. This is a hint and is not necessarily under the full control of the user: the server should always pay attention not to override an explicit user choice (like selecting a language in a drop down list).

**Cookies:**

Cookie:

Contains stored HTTP cookies previously sent by the server with the Set-Cookie header.

Set-Cookie:

Send cookies from the server to the user agent.

Cookie2:

Used to contain an HTTP cookie, previously sent by the server with the Set-Cookie2 header, but has been obsoleted by the specification. Use Cookie instead.

Set-Cookie2:

Used to send cookies from the server to the user agent, but has been obsoleted by the specification. Use Set-Cookie instead.

**CORS:**

Access-Control-Allow-Origin:

Indicates whether the response can be shared.

Access-Control-Allow-Credentials:

Indicates whether the response to the request can be exposed when the credentials flag is true.

Access-Control-Allow-Headers:

Used in response to a preflight request to indicate which HTTP headers can be used when making the actual request.

Access-Control-Allow-Methods:

Specifies the method or methods allowed when accessing the resource in response to a preflight request.

Access-Control-Expose-Headers:

Indicates which headers can be exposed as part of the response by listing their names.

Access-Control-Max-Age:

Indicates how long the results of a preflight request can be cached.

Access-Control-Request-Headers:

Used when issuing a preflight request to let the server know which HTTP headers will be used when the actual request is made.

Access-Control-Request-Method:

Used when issuing a preflight request to let the server know which HTTP method will be used when the actual request is made.

Origin:

Indicates where a fetch originates from.

Timing-Allow-Origin:

Specifies origins that are allowed to see values of attributes retrieved via features of the Resource Timing API, which would otherwise be reported as zero due to cross-origin restrictions.

**Downloads:**

Content-Disposition:

Is a response header if the resource transmitted should be displayed inline (default behavior when the header is not present), or it should be handled like a download and the browser should present a 'Save As' window.

**Message body information:**

Content-Length:

indicates the size of the entity-body, in decimal number of octets, sent to the recipient.

Content-Type:

Indicates the media type of the resource.

Content-Encoding:

Used to specify the compression algorithm.

Content-Language:

Describes the language(s) intended for the audience, so that it allows a user to differentiate according to the users' own preferred language.

Content-Location:

Indicates an alternate location for the returned data.

**Proxies:**

Forwarded:

Contains information from the client-facing side of proxy servers that is altered or lost when a proxy is involved in the path of the request.

X-Forwarded-For:

Identifies the originating IP addresses of a client connecting to a web server through an HTTP proxy or a load balancer.

X-Forwarded-Host:

Identifies the original host requested that a client used to connect to your proxy or load balancer.

X-Forwarded-Proto:

identifies the protocol (HTTP or HTTPS) that a client used to connect to your proxy or load balancer.

Via:

Added by proxies, both forward and reverse proxies, and can appear in the request headers and the response headers.

**Redirects:**

Location:

Indicates the URL to redirect a page to.

**Request context:**

From:

Contains an Internet email address for a human user who controls the requesting user agent.

Host:

Specifies the domain name of the server (for virtual hosting), and (optionally) the TCP port number on which the server is listening.

Referer:

The address of the previous web page from which a link to the currently requested page was followed.

Referrer-Policy:

Governs which referrer information sent in the Referer header should be included with requests made.

User-Agent:

Contains a characteristic string that allows the network protocol peers to identify the application type, operating system, software vendor or software version of the requesting software user agent. See also the Firefox user agent string reference.

**Range requests:**

Accept-Ranges:

Indicates if the server supports range requests and if so, in which unit the range can be expressed.

Range:

Indicates the part of a document that the server should return.

If-Range:

Creates a conditional range request that is only fulfilled if the given etag or date matches the remote resource. Used to prevent downloading two ranges from incompatible version of the resource.

Content-Range:

Indicates where in a full body message a partial message belongs.

**Security:**

Content-Security-Policy (CSP):

Controls resources the user agent is allowed to load for a given page.

Content-Security-Policy-Report-Only:

Allows web developers to experiment with policies by monitoring (but not enforcing) their effects. These violation reports consist of JSON documents sent via an HTTP POST request to the specified URI.

Expect-CT:

Allows sites to opt in to reporting and/or enforcement of Certificate Transparency requirements, which prevents the use of misissued certificates for that site from going unnoticed. When a site enables the Expect-CT header, they are requesting that Chrome check that any certificate for that site appears in public CT logs.

Public-Key-Pins (HPKP):

Associates a specific cryptographic public key with a certain web server to decrease the risk of MITM attacks with forged certificates.

Public-Key-Pins-Report-Only:

Sends reports to the report-uri specified in the header and does still allow clients to connect to the server even if the pinning is violated.

Strict-Transport-Security (HSTS):

Force communication using HTTPS instead of HTTP.

Upgrade-Insecure-Requests:

Sends a signal to the server expressing the client’s preference for an encrypted and authenticated response, and that it can successfully handle the upgrade-insecure-requests directive.

X-Content-Type-Options:

Disables MIME sniffing and forces browser to use the type given in Content-Type.

X-Frame-Options (XFO):

Indicates whether a browser should be allowed to render a page in a <frame>, <iframe> or <object>

X-XSS-Protection:

Enables cross-site scripting filtering.