Headers:

HTTP header fields are components of the header section of request and response messages in the HTTP Protocol. They define the operating parameters of an HTTP transaction.

```
//example of headers using postman


{
 'content-type': 'application/x-www-form-urlencoded',
 'user-agent': 'PostmanRuntime/7.22.0',
 accept: '*/*',
 'cache-control': 'no-cache',
 'postman-token': '35f27261-d4da-47d7-b540-9216d4fa694e',
 host: '127.0.0.1:3001',
 'accept-encoding': 'gzip, deflate, br',
 'content-length': '37',
 connection: 'keep-alive'
}
```

Headers:

The REST headers and parameters contain a wealth of information that can help you track down issues when you encounter them. HTTP Headers are an important part of the API request and response as they represent the meta-data associated with the API request and response. Headers carry information for:

1. Request and Response Body
2. Request Authorization
3. Response Caching
4. Response Cookies

Other than the above categories HTTP headers also carry a lot of other information around HTTP connection types, proxies etc. Most of these headers are for management of connections between

client, server and proxies and do not require explicit validation through testing.

Headers:

Headers are mostly classified as request headers and response headers, know the major request and response headers. You will have to set the request headers when you are sending the request for testing an API and you will have to set the assertion against the response headers to ensure that right headers are being returned.

The headers that you will encounter the most during API testing are the following, you may need to set values for these or set assertions against these headers to ensure that they convey the right information and everything works fine in the API:

Authorization: Carries credentials containing the authentication information of the client for the resource being requested.

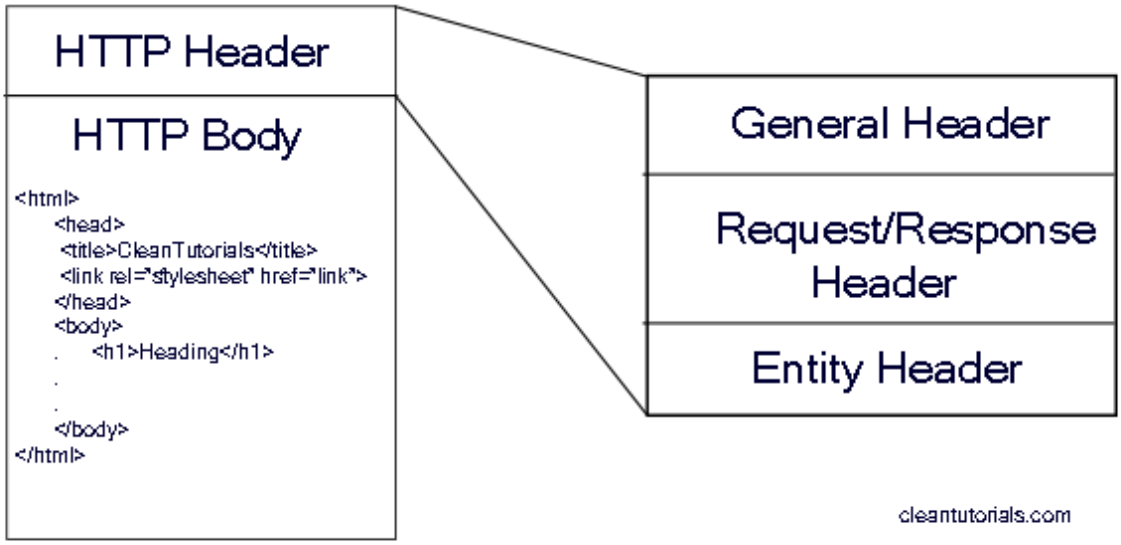
WWW-Authenticate: This is sent by the server if it needs a form of authentication before it can respond with the actual resource being requested. Often sent along with a response code of 401, which means 'unauthorized'.

Accept-Charset: This is a header which is set with the request and tells the server about which character sets are acceptable by the client.

Content-Type: Indicates the media type (text/html or text/JSON) of the response sent to the client by the server, this will help the client in processing the response body correctly.

Cache-Control: This is the cache policy defined by the server for this response, a cached response can be stored by the client and re-used till the time defined by the Cache-Control header.

Headers:



The HTTP Header contains information about the HTTP Body and the Request/Response. Information about the body is related to the content of the Body such as the length of the content inside the body.

The information about Request/Response is the general information about the request/response and it is not specific to the content of the body, example at what time the Request was made

The properties in header are specified as name-value pair which are separated from each other by a colon ':', (example name:value)

Headers:

Request Header: is present when you make a request to the server, it contains information about the request such as the URL that you have requested, the method(GET, POST, HEAD) ,the browser used to generate the request and other info.

Example

```
User-Agent:"Mozilla/5.0 (Windows NT 10.0; WOW64; rv:41.0) Gecko/20100101 Firefox/41.0
```

The term browser is also called user-Agent. So even a simple request to a page involves sending the information about your browser and the operating system you are using. You can see from the above Header field that we are using Windows 10 and Firefox 41.0 browser.

Response Header: is sent from the server after the user sends a request for a particular page or resource and it contains information such as the encoding used in the content, the server software that is used on the server machine to generate the response and other information.

Most of the sites usually hide their server information in order to make it hard for hackers to know which software is being used on the server.

Previous next