**Assignment No**

**Title:** HTTP Header

**Aim:** Write a C++ program to read the HTTP header and analyze the parameters.

**Objective:**  To study how the client and server exchange the messages using HTTP header.

**Theory:** The Hypertext Transfer Protocol (HTTP), the Web's application-layer protocol, is at the heart of the Web. HTTP is based on the client-server architecture model and a stateless request/response protocol that operates by exchanging messages across a reliable TCP/IP connection.An HTTP "client" is a program (Web browser or any other client) that establishes a connection to a server for the purpose of sending one or more HTTP request messages. An HTTP "server" is a program (generally a web server like Apache Web Server or Internet Information Services IIS, etc.) that accepts connections in order to serve HTTP requests by sending HTTP response messages. Figure shows the interaction between client and server. Two versions of HTTP are HTTP1.0 and HTTP1.1. HTTP can use both non-persistent connections and persistent connections. The default mode of HTTP/1.1 uses persistent connections with pipelining.



**Figure:** HTTP request-response behavior

HTTP requests and HTTP responses use a generic message format of RFC 822 for transferring the required data.

**HTTP Request Message:** A typical HTTP request message is:

**GET** /somedir/page.html HTTP/1.1

**Connection:** close

**User-agent:** Mozilla/4.0

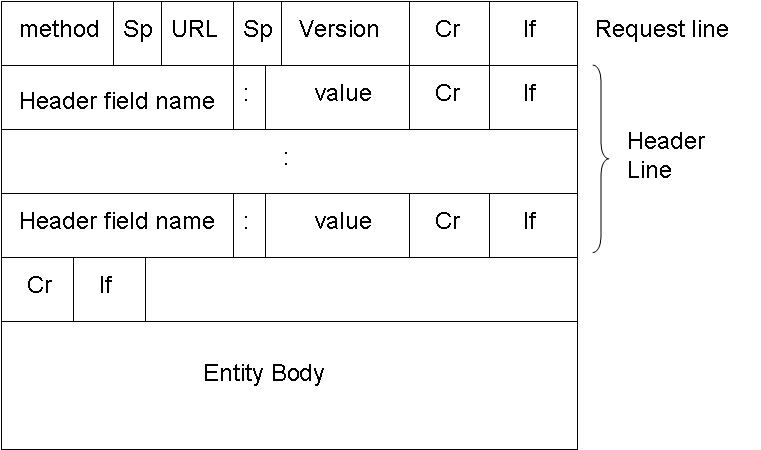
**Accept:** text/html, image/gif, image/jpeg

**Accept-language**: fr

(extra carriage return, line feed)

The first line of a HTTP request message is called the **request line**; the subsequent lines are called the **header lines**. The request line has three fields: the method field, the URL field, and the HTTP version field. The method field can take on several different values, including GET, POST, and HEAD. The great majority of HTTP request messages use the GET method. The GET method is used when the browser requests an object, with the requested object identified in the URL field. By including the Connection: close header line, the browser is telling the server that it doesn't want to use persistent connections; it wants the server to close the connection after sending the requested object. Thus the browser that generated this request message implements HTTP/1.1 but it doesn't want to bother with persistent connections. The User agent: header line specifies the user agent, i.e., the browser type that is making the request to the server. The Accept: header line tells the server the type of objects the browser is prepared to accept. In this case, the client is prepared to accept HTML text, a GIF image or a JPEG image. Finally, the Accept-language: header indicates that the user prefers to receive a French version of the object, if such an object exists on the server; otherwise, the server should send its default version.

The general format for a request message is shown in a following figure.



**HTTP Response Message:** A typical HTTP response message is:

HTTP/1.1 200 OK

Connection: close

Date: Thu, 06 Aug 1998 12:00:15 GMT

Server: Apache/1.3.0 (Unix)

Last-Modified: Mon, 22 Jun 1998 09:23:24 GMT

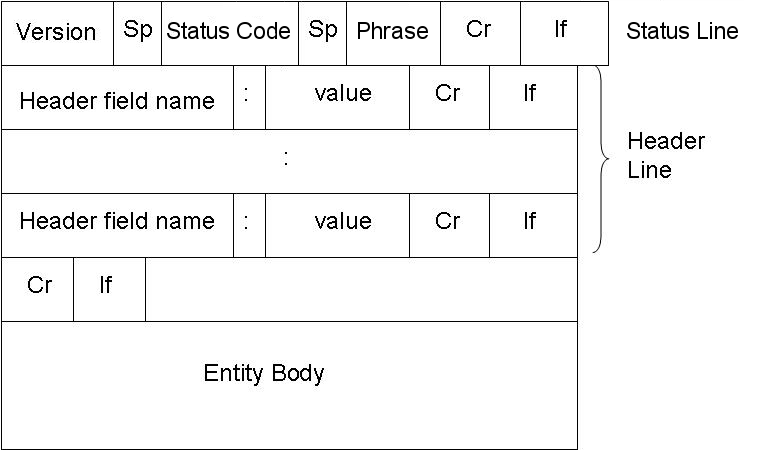
Content-Length: 6821

Content-Type: text/html

*datadatadatadatadata ...*

The Hyper Text Transfer Protocol status line has three fields: the protocol version field, a status code, and a corresponding status message. In this example, the status line indicates that the server is using HTTP/1.1 and that that everything is OK (i.e., the server has found, and is sending, the requested object). The server uses the Connection: close header line to tell the client that it is going to close the TCP connection after sending the message. The Date: header line indicates the time and date when the HTTP response was created and sent by the server. Note that this is not the time when the object was created or last modified; it is the time when the server retrieves the object from its file system, inserts the object into the response message and sends the response message. The Server: header lineindicates that the message was generated by an Apache Web server; it is analogous to the User-agent: header line in the HTTPrequest message. The Last-Modified: header line indicates the time and date when the object was created or last modified. The Content-Length: header line indicates the number of bytes in the object being sent. The Content-Type: header line indicates that the object in the entity body is HTML text.

The general format for a response message is shown in a following figure.



The status code and associated phrase indicate the result of the request. Some common status codes and associated phrases include:

**200 OK:** Request succeeded and the information is returned in the response.

**301 Moved Permanently:** Requested object has been permanently moved

**400 Bad Request:** A generic error code indicating that the request could not be understood by the server.

**404 Not Found:** The requested document does not exist on this server

**505 HTTP Version Not Supported:** The request HTTP protocol version is not supported by the server.

**Curl Libraries:** Curl is a tool and libcurl is a library for transferring data with URL syntax, supporting FTP, FTPS, HTTP.

**Algorithm**: Install curl package.

Install the curl Library.

Take the address of the website to be analysed from the user.

Compile the code using gccFilename.c–lcurl.

The output is the header and the body stored in the file.

**Note: write Mathematical model ,FAQs, Test Cases(Success case and failure cases while running the program).**

**FAQs:**

1. What is the difference between persistent and non persistent connection.
2. Explain various methods used in HTTP?
3. Compare HTTP and HTTPs