

# How HTTP Works?

HTTP is an abbreviated form of Hyper Text Transfer Protocol..As we all know HTTP set the foundations for internet and WWW [world wide web]...I'll not go in much detail of its history, but I'll get straight to the topic of

## How HTTP works

HTTP is a request response protocol to communicate asynchronously between client and server. Mostly in HTTP a browser acts as a client and a web-server like Apache or IIS acts as server. The server which hosts the files (like html , audio , video files etc) responses to the client. Depending on the request a response contains the status of the request.

### What a Request Message contains?

A request can be a line such as :-

Code:

```
GET /index.php HTTP/1.1
```

Which request the file /index.php from the server ...

Headers can be like

Code:

```
Accept-Language: en
```

*Note : Every line of HTTP request should end with <cr> <lf> [carriage return and line feed]. And the empty line should contain only <cr><lf> no other data, not even white space.*

**In HTTP/1.1 all headers are optional except the Host...**

# Different types of HTTP requests

There are 9 types of HTTP requests/method.

## GET

This request is Used to get the Response header and the response body!!

eg :-

Code:

```
aneesh@aneesh-laptop:~$ telnet www.go4expert.com 80
Trying 174.133.80.67...
Connected to www.go4expert.com.
Escape character is '^]'.
GET / HTTP/1.1
Host : www.go4expert.com

HTTP/1.1 200 OK
Date: Sun, 30 Jan 2011 08:03:36 GMT
Server: Apache
X-Powered-By: PHP/5.2.10
Set-Cookie: expertlastvisit=1296374633; expires=Mon, 30-Jan-2012 08:03:53
GMT; path=/
Set-Cookie: expertlastactivity=0; expires=Mon, 30-Jan-2012 08:03:53 GMT;
path=/
Cache-Control: private
Pragma: private
X-UA-Compatible: IE=7
Set-Cookie: vbseo_loggedin=deleted; expires=Sat, 30-Jan-2010 08:03:52 GMT;
path=/
Vary: Accept-Encoding,User-Agent
Transfer-Encoding: chunked
Content-Type: text/html; charset=ISO-8859-1
```

```
14aec
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html dir="ltr" lang="en" xmlns="http://www.w3.org/1999/xhtml">
<head>

    <meta http-equiv="Cache-Control" content="no-cache" />
    <meta http-equiv="Pragma" content="no-cache" />
    <meta http-equiv="Expires" content="0" />
<title>Programming and SEO Forums </title>
```

Output truncated to save space

## HEAD

This request is used to get back The response header only (not the response body as returned by the GET request..)

eg :-

Code:

```
aneesh@aneesh-laptop:~$ telnet www.go4expert.com 80
Trying 174.133.80.67...
Connected to www.go4expert.com.
Escape character is '^]'.
HEAD / HTTP/1.1
Host : www.go4expert.com

HTTP/1.1 200 OK
Date: Sun, 30 Jan 2011 08:12:47 GMT
Server: Apache
X-Powered-By: PHP/5.2.10
Set-Cookie: expertlastvisit=1296375177; expires=Mon, 30-Jan-2012 08:12:57
GMT; path=/
Set-Cookie: expertlastactivity=0; expires=Mon, 30-Jan-2012 08:12:57 GMT;
path=/
```

```
Cache-Control: private
Pragma: private
X-UA-Compatible: IE=7
Set-Cookie: vbseo_loggedin=deleted; expires=Sat, 30-Jan-2010 08:12:56 GMT;
path=/
Vary: Accept-Encoding,User-Agent
Content-Type: text/html; charset=ISO-8859-1

Connection closed by foreign host.
aneesh@aneesh-laptop:~$
```

## **POST**

This request is used to submit data (eg : for to be used in HTML forms etc.)...

## **PUT**

Used for uploading resource

## **PATCH**

Is used to modify the resource

## **DELETE**

Used for deleting resource

## **TRACE**

simply echoes back the request sent by the client...  
This can be used for testing the servers and Checking weather the server is alive or not..

Code:

```
aneesh@aneesh-laptop:~$ telnet www.go4expert.com 80
Trying 174.133.80.67...
Connected to www.go4expert.com.
Escape character is '^]'.
TRACE / HTTP/1.1
Host : www.go4expert.com

HTTP/1.1 200 OK
Date: Sun, 30 Jan 2011 08:16:05 GMT
Server: Apache
Transfer-Encoding: chunked
Content-Type: message/http

2d
TRACE / HTTP/1.1
Host: www.go4expert.com

0

Connection closed by foreign host.
```

## **CONNECT**

Convert the It transparent TCP/IP tunnel , usually to facilitate SSL encrypted Communicated (HTTPS) through an unencrypted HTTP proxy.. This is usually used in the login / signup pages in which the data needs to be encrypted so as to protect it from sniffers etc etc...

## **OPTIONS**

This request returns back the HTTP options/methods supported by the server...

## **Summary**

Client side :-

Basically as A http client sends a HTTP request. The client actually establishes a

TCP connection to the port no on the host... (Mostly its 80 but can be change according to the setting of the server..)

Then the host send back the requested resource with a status code...

Like 202 , 404 (refer [http://en.wikipedia.org/wiki/List\\_of\\_HTTP\\_status\\_codes](http://en.wikipedia.org/wiki/List_of_HTTP_status_codes) to know more about the status codes)

A basic client request :-

Code:

```
GET / HTTP/1.1
Host : www.go4expert.com
```

Server side :-

A HTTP web server listens for requests on the port no specified (default : 80)...As the Server gets a request it returns back the requested resource with a status code...

A basic server output :-

I will try to explain each of them with an example using my [Own HTTP Server](#).

Code:

```
aneesh@aneesh-laptop:~$ sudo '~/Programming/C/Ani-srver/web server'
[sudo] password for aneesh:
Server is open for listening on port 80
Got connection from 0.0.0.0
Got connection from ::

Request by browser = GET / HTTP/1.1

REq=/index.html
```

```
Opening "/home/aneesh/aniroot/index.html"
```

```
200 OK!!!
```

```
Got connection from 0.0.0.0
```

```
Got connection from ::
```

```
Request by browser = GET /favicon.ico HTTP/1.1
```

```
REq=/favicon.ico
```

```
Opening "/home/aneesh/aniroot/favicon.ico"
```

```
404 File not found Error
```

```
Got connection from 0.0.0.0
```

```
Got connection from ::
```

```
Request by browser = GET /favicon.ico HTTP/1.1
```

```
REq=/favicon.ico
```

```
Opening "/home/aneesh/aniroot/favicon.ico"
```

```
404 File not found Error
```

**Source: <http://www.go4expert.com/articles/how-http-works-t24799/>**