Whenever you visit a page on the web, your computer uses the **Hypertext Transfer Protocol (HTTP)** to download that page from another computer somewhere on the Internet.

**Step 1: Direct browser to URL**

#### Step 2: Browser looks up IP

The browser uses a DNS resolver to map the domain to an IP address:

**Step 3: Browser sends HTTP request**

Once the browser identifies the IP address of the computer hosting the requested URL, it sends an **HTTP request**.

An HTTP request:

GET /index.html HTTP/1.1

Host: www.example.com

The first word is the HTTP verb: "GET". There are other verbs for other actions on the web, like submitting form data ("POST").

The next part specifies the path: "/index.html". The host computer stores the content of the entire website, so the browser needs to be specific about which page to load.

The final part of the first line specifies the protocol and the version of the protocol: "HTTP/1.1".

The second line specifies the domain of the requested URL. That's helpful in case a host computer stores the content for multiple websites.

**Step 4: Host sends back HTTP response**

Once the host computer receives the HTTP request, it sends back a response with both the content and metadata about it.

The HTTP response starts similarly to the request:

HTTP/1.1 200 OK

The response begins with the protocol and version, "HTTP/1.1".

The next number is the very important **HTTP status code**, and in this case, it's 200. That code represents a successful retrieval of the document ("OK").

If the server failed to retrieve the document, the status codes provide more information, like if the failure was due to user error or server error. For example, the most well known status code is 404 ("File not found"). That happens whenever you visit a path on a server that doesn't correspond to any document. Since users have a habit of typing URLs incorrectly, 404s happen frequently, so websites often have fun 404 webpages.

The next part of an HTTP response are the **headers**. They give the browser additional details and help the browser to render the content.

These two headers are common to most requests:

Content-Type: text/html; charset=UTF-8

Content-Length: 208

The content-type tells the browser what type of document it's sending back. A common type on the web is "text/html", as all webpages are HTML text files. Other types are possible, like images ("image/png"), videos ("video/mpeg"), script ("application/javascript") and anything else you can load in your browser.

The content-length gives the length of the document in bytes, which helps the browser know how long a file will take to download.

Finally, the HTTP response writes out the actual document requested. This page is a simple HTML file:

**Step 5: The browser renders the response**

The browser now has all the information it needs to render the document requested.

### **HTTP and TCP/IP**

HTTP is a protocol that's built on top of the TCP/IP protocols.

Each HTTP request is inside an IP packet, and each HTTP response is inside another IP packet--or more typically, multiple packets, since the response data can be quite large.

There are many other protocols built on top of TCP/IP, like protocols for sending email (SMTP, POP, IMAP) and uploading files (FTP).

All of these protocols enable us to use the Internet to connect with other computers in useful ways, and to communicate and collaborate across wide distances.