CODING NOMADS

9) HTTP & Web Services Lesson

HTTP: The Protocol Powering the Web

7 min to complete · By Ryan Desmond, Jared Larsen

Contents

The last section in this course mentioned that this course focuses only on RESTful services. This means that you can safely ignore how UDP, TCP, etc. work and really dig deep into HTTP: the protocol (or "language") upon which the internet and RESTful web services are built.

This lesson contains a general explanation of what HTTP is and how it works.

What Does HTTP Stand For?

HTTP stands for HyperText Transfer Protocol.

What is HTTP?

HTTP and other protocols are sets of rules or standards that are formulated ways of doing things online. HTTP lets systems communicate over the internet.

You've seen it often (even though you may not have noticed it). Take a look at the URL next time you're Googling something. Notice how it begins with either **http:// or https://** (we'll go into the difference between these in a few lessons). These preceding characters tell your browser to use the HTTP protocol when making a request to the server.

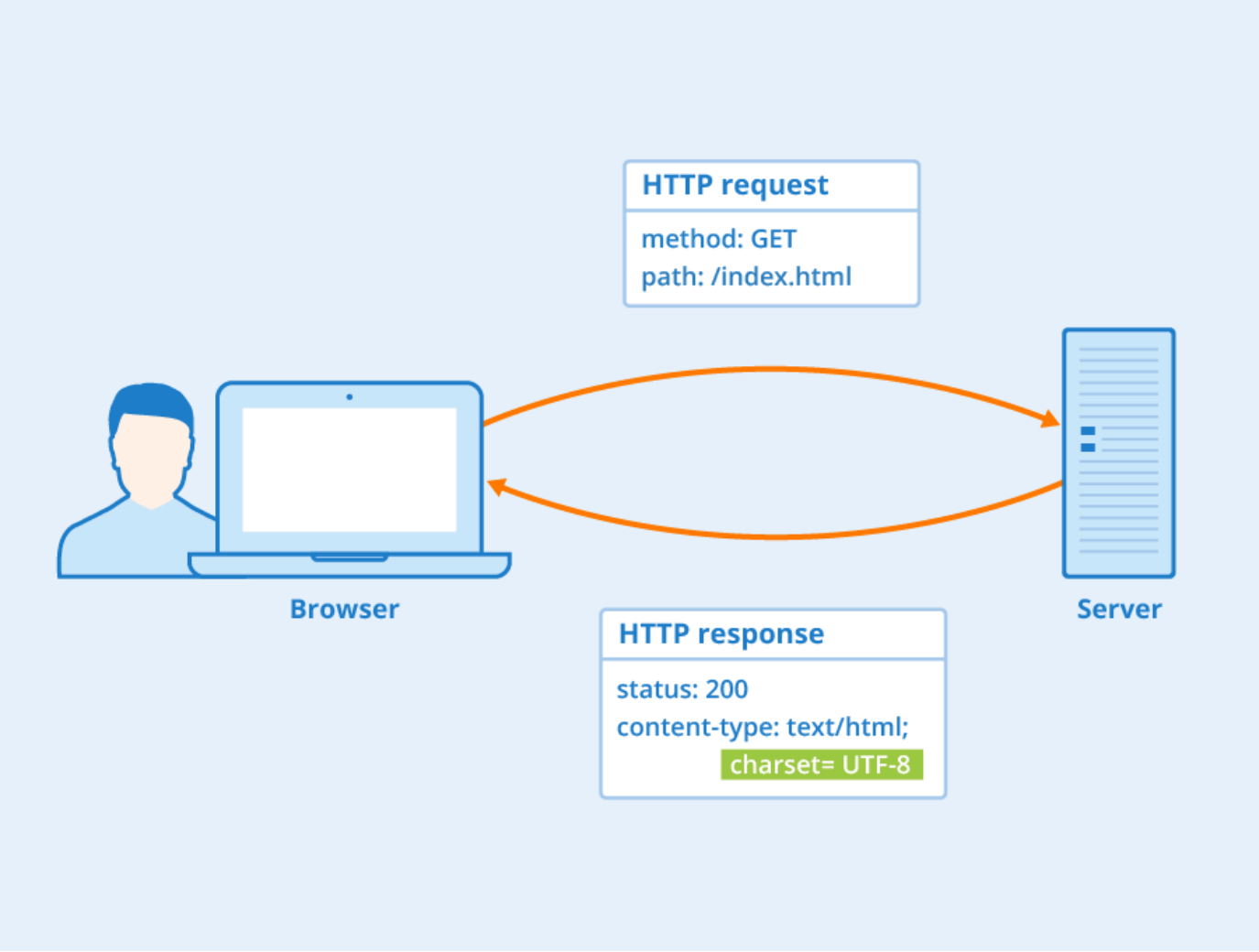
While HTTP is the standard protocol for web browsers, underlying internet communication can rely on other protocols like TCP or UDP. However, sticking with a single protocol for most communication increases interoperability and simplifies deployment to the web. So, with almost universal usage, **HTTP can be said to be the bedrock protocol of the internet.**

Let's clarify what HTTP's role is with an analogy: If you consider the English language a protocol for verbal communication, but before one could interact and make use of existing English systems, they would have to learn English. The same is true for the internet; before you can communicate, you must learn the protocol. Otherwise, it's just a bunch of incomprehensible shouting back and forth. Fortunately, learning the HTTP protocol is much simpler than learning English.

How does HTTP work?

You now have a general idea of what HTTP is, but what is the actual set of instructions or procedures that this protocol lays out? The interactions between two systems over HTTP are commonly called the **HTTP Request/Response Cycle**.

Here is a general overview, with more detail coming in subsequent lessons.



1. **An HTTP Request is made using a URL:** The client enters a URL (Uniform Resource Locator) in a browser and requests the resource.
2. **DNS and NAT operations are performed:** Behind the scenes, some translating needs to happen before we can actually make the request for the page. This is because URLs are usually in human speak. Computers don't know what **codingnomads.com** means they only understand IP addresses (e.g. 192.168.1.1). So a DNS (Domain Name Service) lookup and NAT (Network Address Translation) are performed to find the correct IP address associated with **codingnomads.com**.
3. **The request is received and is fulfilled:** The server receives the request and responds with the requested resource in the form of an HTTP Response.

A Note on HTTP State

In software development, "state" generally refers to the conditions or attributes of a system or application at any specific point in time. It represents the current situation based on previous interaction or input. When two systems have "state," it means both

systems share some information about their interactions and each maintains its own state based on the information exchanged during those interactions.

With that in mind, it is important to note that HTTP is "stateless". This means every HTTP request is entirely independent of all other requests (they have no knowledge of previous or future requests). You can see this by simply refreshing a page; the new request does not know anything about the previous request. There is no knowledge or communication between server and client outside of the independent HTTP requests.

Tip: While each HTTP request is independent, state can be somewhat simulated using cookies and other techniques that enable additional continuity between server and client. This is an advanced topic, no need to dig in deeper for now.

Summary: What is HTTP?

- HTTP is a standard for systems to communicate over the internet.
- HTTP stands for HyperText Transfer Protocol.
- HTTP lays out instructions for a request/response cycle.
- HTTP is "stateless." Meaning every HTTP request is entirely independent of all other requests.

Learn more

Previous Next → HTTP Requests: Methods, Headers, and Bodies

Discord