HTTP Requests

Understand the basics of how your web browser communicates with the internet.

BACKGROUND:

This page is generated by a web of HTML, CSS, and Javascript, sent to you by Codecademy via the internet. The internet is made up of a bunch of resources hosted on different servers. The term "resource" corresponds to any entity on the web, including HTML files, stylesheets, images, videos, and scripts. To access content on the internet, your browser must ask these servers for the resources it wants, and then display these resources to you. This protocol of requests and responses enables you view *this* page in your browser.

This article focuses on one fundamental part of how the internet functions: HTTP.

WHAT IS HTTP?

HTTP stands for Hypertext Transfer Protocol and is used to structure requests and responses over the internet. HTTP requires data to be transferred from one point to another over the network.

The transfer of resources happens using TCP (Transmission Control Protocol). In viewing this webpage, TCP manages the channels between your browser and the server (in this case, codecademy.com). TCP is used to manage many types of internet connections in which one computer or device wants to send something to another. HTTP is the command language that the devices on both sides of the connection must follow in order to communicate.

HTTP & TCP: HOW IT WORKS

When you type an address such as www.codecademy.com into your browser, you are commanding it to open a TCP channel to the server that responds to that URL (or Uniform Resource Locator, which you can read more about on Wikipedia). A URL is like your home address or phone number because it describes how to reach you.

In this situation, your computer, which is making the request, is called the client. The URL you are requesting is the address that belongs to the server.

Once the TCP connection is established, the client sends a HTTP *GET* request to the server to retrieve the webpage it should display. After the server has sent the response, it closes the TCP connection. If you open the website in your browser again, or if your browser automatically requests something from the server, a new connection is opened which follows the same process described above. GET requests are one kind of HTTP method a client can call. You can learn more about the other common ones (*POST*, *PUT* and *DELETE*) in this article.

Let's explore an example of how GET requests (the most common type of request) are used to help your computer (the client) access resources on the web.

Suppose you want to check out the latest course offerings from http://codecademy.com. After you type the URL into your browser, your browser will extract the `http` part and recognize that it is the name of the network protocol to use. Then, it takes the domain name from the URL, in this case "codecademy.com", and asks the internet Domain Name Server to return an Internet Protocol (IP) address.

Now the client knows the destination's IP address. It then opens a connection to the server at that address, using the `http` protocol as specified. It will initiate a GET request to the server which contains the IP address of the host and optionally a data payload. The GET request contains the following text:

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

This identifies the type of request, the path on www.codecademy.com (in this case, "/") and the protocol "HTTP/1.1." HTTP/1.1 is a revision of the first HTTP, which is now called HTTP/1.0. In HTTP/1.0, every resource request requires a separate connection to the server. HTTP/1.1 uses one connection more than once, so that additional content (like images or stylesheets) is retrieved even after the page has been retrieved. As a result, requests using HTTP/1.1 have less delay than those using HTTP/1.0.

The second line of the request contains the address of the server which is `"www.codecademy.com"`. There may be additional lines as well depending on what data your browser chooses to send.

If the server is able to locate the path requested, the server might respond with the header:

```
HTTP/1.1 200 OK
Content-Type: text/html
```

This header is followed by the content requested, which in this case is the information needed to render www.codecademy.com.

The first line of the header, `HTTP/1.1 200 OK`, is confirmation that the server understands the protocol that the client wants to communicate with (`HTTP/1.1`), and an HTTP status code signifying that the resource was found on the server. The third line, `Content-Type: text/html`, shows the type of content that it will be sending to the client.

If the server is not able to locate the path requested by the client, it will respond with the header:

```
HTTP/1.1 404 NOT FOUND
```

In this case, the server identifies that it understands the HTTP protocol, but the `404 NOT FOUND` status code signifies that the specific piece of content requested was not found. This might happen if the content was moved or if you typed in the URL path incorrectly or if the page was

removed. You can read more about the 404 status code, commonly called a 404 error, [here](#).

AN ANALOGY:

It can be tricky to understand how HTTP functions because it's difficult to examine what your browser is actually doing. (And perhaps also because we explained it using acronyms that may be new to you.) Let's review what we learned by using an analogy that could be more familiar to you.

Imagine the internet is a town. You are a client and your address determines where you can be reached. Businesses in town, such as Codecademy.com, serve requests that are sent to them. The other houses are filled with other clients like you that are making requests and expecting responses from these businesses in town. This town also has a crazy fast mail service, an army of mail delivery staff that can travel on trains that move at the speed of light.

Suppose you want to read the morning newspaper. In order to retrieve it, you write down what you need in a language called HTTP and ask your local mail delivery staff agent to retrieve it from a specific business. The mail delivery person agrees and builds a railroad track (connection) between your house and the business nearly instantly, and rides the train car labeled "TCP" to the address of the business you provided.

Upon arriving at the business, she asks the first of several free employees ready to fulfill the request. The employee searches for the page of the newspaper that you requested but cannot find it and communicates that back to the mail delivery person.

The mail delivery person returns on the light speed train, ripping up the tracks on the way back, and tells you that there was a problem "404 Not Found." After you check the spelling of what you had written, you realize that you misspelled the newspaper title. You correct it and provide the corrected title to the mail delivery person.

This time the mail delivery person is able to retrieve it from the business. You can now read your newspaper in peace until you decide you want to

read the next page, at which point, you would make another request and give it to the mail delivery person.

WHAT IS HTTPS?

Since your HTTP request can be read by anyone at certain network junctures, it might not be a good idea to deliver information such as your credit card or password using this protocol. Fortunately, many servers support HTTPS, short for HTTP Secure, which allows you to encrypt data that you send and receive. You can read more about HTTPS on Wikipedia.

HTTPS is important to use when passing sensitive or personal information to and from websites. However, it is up to the businesses maintaining the servers to set it up. In order to support HTTPS, the business must apply for a certificate from a Certificate Authority.