Difference between HTTP1.1 vs HTTP2

HTTP stands for **Hyper Text Transfer Protocol** & it is used in **client-server communication.**

HTTP is the method computers and servers use to request and send information.

For example, when someone navigates to guvi.in on their laptop, their web browser sends an HTTP request to the server where the guvi website is hosted for the content that appears on the page. Then, that server send HTTP responses with the text, images, and formatting that the browser displays to the user.

There are several stages of development of HTTP. Here we are going see mainly about HTTP/1.1 & HTTP/2

HTTP/1.1 was created in **1997** & the new one is **HTTP/2** which was created in **2015**.

HTTP/2 solves several problems that the creators of HTTP/1.1 did not anticipate. In particular, **HTTP/2 is much faster and more efficient than HTTP/1.1. One of the ways in which HTTP/2 is faster is in how it prioritizes content during the loading process.**

How does prioritization in HTTP/2 affect performance?

In HTTP/2, developers have hands-on, detailed control over prioritization. This allows them to maximize perceived and actual page load speed to a degree that was not possible in HTTP/1.1.

HTTP/2 offers a feature called **weighted prioritization**. This allows developers to decide which page resources will load first, every time. In HTTP/2, when a client makes a request for a webpage, the server sends several streams of data to the client at once, instead of sending one thing after another. This method of data delivery is known as **multiplexing**. Developers can assign each of these data streams a different weighted value, and the value tells the client which data stream to render first.

Imagine that Alice wants to read a novel that her friend Bob wrote, but both Alice and Bob only communicate through the regular mail. Alice sends a letter to Bob and asks Bob to send her his novel. Bob decides to send the novel **HTTP/1.1-style**: He mails one chapter at a time, and he only mails the next chapter after receiving a reply letter from Alice confirming that she received the previous chapter. Using this method of content delivery, it takes Alice many weeks to read Bob's novel.

Now imagine that Bob decides to send Alice his novel **HTTP/2-style:** In this case, he sends each chapter of the novel separately (to stay within the postal service's size limits) but all at the same time. He also numbers each chapter: Chapter 1, Chapter 2, etc. Now, Alice receives the novel all at once and can assemble it in the correct order on her own time. If a chapter is missing, she may send a quick reply asking for that specific chapter, but otherwise the process is complete, and Alice can read the novel in just a few days.

In HTTP/2, data is sent all at once, much like Bob when he sends Alice multiple chapters at once. And just like Bob, developers get to number the chapters in HTTP/2. They can decide if

the text of a webpage loads first, or the CSS files, or the JavaScript, or whatever they feel is most important for the user experience.

Difference between HTTP/1.1 and HTTP/2 are:

HTTP/1.1	HTTP/2
It works on the textual format.	It works on the binary protocol.
There is head of line blocking that blocks all the requests behind it until it doesn't get its all resources.	It allows multiplexing so one TCP connection is required for multiple requests.
It uses requests resource Inlining for use getting multiple pages	It uses PUSH frame by server that collects all multiple pages
It compresses data by itself.	It uses HPACK for data compression.