Write a blog on Difference between HTTP1.1 vs HTTP2

## Both versions serve the same purpose of transferring data over the web, there are several key differences between them:

## What is HTTP/1.1?

* The first usable version of HTTP was created in 1997. Because it went through several stages of development, this first version of HTTP was called HTTP/1.1. This version is still in use on the web.
* In HTTP 1.1, multiple requests and responses were handled sequentially over a single TCP (Transmission Control Protocol) connection. This meant that if one resource took longer to load, it would block subsequent requests
* HTTP 1.1 used plain text for communication, which was human-readable but inefficient for parsing and processing.
* HTTP 1.1 used plaintext headers for each request and response, resulting in redundant data being sent with every communication.
* Programs like [gzip](https://www.gzip.org/) have long been used to compress the data sent in HTTP messages, especially to decrease the size of CSS and JavaScript files. Additionally, the use of cookies can sometimes make headers much larger, increasing the need for some kind of compression.

## What is HTTP/2?

* In 2015, a new version of HTTP/2 was created. 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.
* In HTTP 2, multiplexing is introduced, allowing multiple requests and responses to be sent and received concurrently over a single connection. This leads to more efficient and faster utilization of the network.
* HTTP 2, on the other hand, uses a binary protocol, which is more compact and easier to parse for both servers and clients.
* HTTP 2 introduces header compression, where the headers are compressed using a technique called HPACK. This reduces overhead and improves performance.
* HTTP 2 supports server push, a feature that allows the server to proactively send resources to the client before they are explicitly requested. This helps to reduce latency by minimizing the round trips required to fetch additional resources.
* HTTP 2 introduces the ability to assign weights and dependencies to different requests, allowing for more efficient resource allocation. This means that more important or critical resources can be prioritized and delivered first.
* HTTP 2 introduces the concepts of streams and frames. A stream is a bidirectional flow of bytes within an established connection, while a frame is the smallest unit of communication in HTTP 2. Streams can be independently multiplexed and prioritized, allowing for better resource management.
* Using HPACK and other compression methods, HTTP/2 provides one more feature that can reduce client-server latency.

## [Conclusion](https://www.digitalocean.com/community/tutorials/http-1-1-vs-http-2-what-s-the-difference#conclusion)

As you can see from this point-by-point analysis, HTTP/2 differs from HTTP/1.1 in many ways, with some features providing greater levels of control that can be used to better optimize web application performance and other features simply improving upon the previous protocol. Now that you have gained a high-level perspective on the variations between the two protocols, you can consider how such factors as multiplexing, stream prioritization, flow control, server push, and compression in HTTP/2 will affect the changing landscape of web development.