**HTTP-Hypertext Transfer protocol:**

HTTP is a top-level application protocol that exchanges information between a client computer and a local or remote web server.

**HTTP versions:**

* HTTP/0.9,
* HTTP/1.0,
* HTTP/1.1,
* HTTP/2.0, and
* HTTP/3.0.

**Difference between HTTP/1.1 and HTTP/2**

**HTTP/1.1**

* The first standardized version of HTTP, HTTP/1.1, was published in early 1997.
* In this process, a client sends a text-based request to a server by calling a method like GET or POST.
* In response, the server sends a resource like an HTML page back to the client.
* HTTP/1.1, keeps all requests and responses in plain text format.
* In HTTP/1.0, the client had to break and remake the TCP connection with every new request, a costly affair in terms of both time and resources (can be avoided by using persistent connections and pipelining).
* In HTTP/1.1, flow control relies on the underlying TCP connection.
* 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.
* HTTP/1.1 loads resources one after the other, so if one resource cannot be loaded, it blocks all the other resources behind it.

**HTTP/2**

* HTTP/2 developed in 2015 by Internet Engineering Task Force (IETF)with the intention of reducing web page load latency by using techniques such as **compression**, **multiplexing**, and **prioritization**.
* In this process also, a client sends a text-based request to a server by calling a method like GET or POST.
* HTTP/2 uses the binary framing layer to encapsulate all messages in binary format.
* In HTTP/2, the binary framing layer encodes requests/responses and cuts them up into smaller packets of information, greatly increasing the flexibility of data transfer.
* The interleaved requests and responses can run in parallel without blocking the messages behind them, a process called **multiplexing**.
* HTTP/2 multiplexes streams of data within a single TCP connection.
* HTTP/2 can split headers from their data, resulting in a header frame and a data frame. [HPACK](https://tools.ietf.org/html/draft-ietf-httpbis-header-compression-12) can then **compress** this header frame.
* HTTP/2 offers a feature called weighted **prioritization**. This allows developers to decide which page resources will load first, every time.
* HTTP/2 is much faster and more efficient than HTTP/1.1.
* HTTP/2 is **able to use** a single TCP connection to send multiple streams of data at once so that no one resource blocks any other resource.

# **Benefits of HTTP/2 over HTTP/1.1:**

* Less prone to errors.
* Lighter network footprint.
* Effective network resource utilization.
* Eliminating security concerns associated with the textual nature of HTTP1.x such as response splitting attacks.
* Enables other capabilities of the HTTP/2 including compression, multiplexing, prioritization, flow control and effective handling of TLS.
* Compact representation of commands for easier processing and implementation.
* Efficient and robust in terms of processing of data between client and server.
* Reduced network latency and improved throughput.