**Write a blog on Difference between HTTP1.1 vs HTTP2**

The **Hyper Text Transfer Protocol (HTTP)** is a data communications protocol.

**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. It's an application protocol that allows a client computer to exchange information with a web server. When a client visits a website, the web browser sends an HTTP request to the server, which is in the form of a text-based message.

**HTTP 2:** HTTP 2 is a newer version of the Hypertext Transfer Protocol (HTTP) that is faster and more efficient than HTTP 1.1. It was released in 2015 to improve the online experience by speeding up page loads and reducing round-trip time (RTT). It is a major revision to the World Wide Web's HTTP network protocol that aims to be more efficient and gentler on the internet

HTTP 2 is more effective than HTTP 1.1 in following ways:

**Prioritization**

HTTP 2 gives developers detailed control over prioritization, allowing them to decide which page resources load first. This can maximize page load speed. While, HTTP 1.1 doesn't have a built-in mechanism for prioritizing requests, so requests can be processed in any order, regardless of their importance or urgency

**Server push**

HTTP 2 enables "server push", which means the server proactively pushes resources to the client without the client needing to request them. This speeds up page load times. HTTP 1.1 doesn't support server push

**Multiplexing**

HTTP 2 uses multiplexing to allow multiple concurrent requests and responses over a single TCP connection. This improves overall efficiency. HTTP 1.1 loads resources one after the other, so if one resource cannot be loaded, it blocks all the other resources behind it

**Encoding**

HTTP 1.1 sends messages as plain text, and HTTP 2 encodes them into binary data and arranges them carefully. HTTP 2 uses binary encoding rather than text encoding, making it more efficient.

**Write a blog about objects and its internal representation in JavaScript**

**JAVASCRIPT:** JavaScript (JS) is a high-level, interpreted programming language that is one of the three core technologies of World Wide Web content production (along with HTML and CSS).

**OBJECTS:** Objects are the fundamental data type and are used to store and organize data. They are made up of properties and methods, which are used to access and manipulate the data within the object.

**Object literals:** Object literals are the most common way to create objects in JavaScript. They are created using curly braces ({}) and contain a list of key-value pairs.

**Object properties:** Object properties are the key-value pairs that make up an object. The keys are strings, and the values can be any valid JavaScript value, including other objects.

**Prototype properties:** All objects in JavaScript have a prototype property, which is another object. The prototype property is used to inherit properties and methods from other objects.

**Const:** The const keyword can be used to declare constant objects. Constant objects cannot have their properties changed.

**hasOwnProperty:** The hasOwnProperty() method can be used to check if an object has a specific property.

**Properties:** Object properties can be accessed using the dot notation (.).

**JavaScript properties:** JavaScript properties are the key-value pairs that make up an object. The keys are strings, and the values can be any valid JavaScript value, including other objects.

**Undefined:** The undefined value is used to indicate that a variable has not been assigned a value.