1. 1. **Multiplexing: The Art of Concurrent Requests**
   - **HTTP/1.1:** This version processes requests sequentially, leading to a phenomenon known as "head-of-line blocking." If a resource is slow to load or if there's a delay in the server's response, it can hinder the loading of other resources.
   - **HTTP/2:** Multiplexing is a game-changer. It allows multiple requests and responses to be transmitted concurrently over a single TCP connection. This means that even if one resource experiences latency, it doesn't hold up the loading of others. It significantly enhances the efficiency of data transfer.
2. **Header Compression: Trimming the Fat**
   - **HTTP/1.1:** Every request and response involves the transmission of headers, which can be substantial. In HTTP/1.1, these headers are not compressed, leading to increased data overhead.
   - **HTTP/2:** With header compression, HTTP/2 minimizes redundant information, reducing the amount of data that needs to be transmitted. This results in faster load times and more efficient use of network resources.
3. **Binary Protocol: The Language of Efficiency**
   - **HTTP/1.1:** This version relies on plain text for communication, which is human-readable but can be inefficient in terms of parsing and processing.
   - **HTTP/2:** Switching to a binary protocol, HTTP/2 uses a more compact and efficient format for data transmission. While not as readable to humans, this binary format is faster and requires less bandwidth.
4. **Server Push: Anticipating User Needs**
   - **HTTP/1.1:** Resources are fetched one at a time based on explicit client requests. The server has limited ability to push resources to the client without a specific request.
   - **HTTP/2:** Server push allows the server to proactively send resources to the client before they are requested. This can significantly reduce latency and improve page load times by anticipating the user's needs.
5. **Connection Handling: Keeping it Alive**
   - **HTTP/1.1:** Each request/response cycle typically requires a new connection to be established, incurring the overhead of multiple connections.
   - **HTTP/2:** Persistent connections are the norm, and a single connection can be reused for multiple requests and responses. This minimizes latency and resource consumption, especially beneficial for mobile and high-latency networks.

1. 2. **Properties: Key-Value Pairs**
   - At the core of every JavaScript object lies a collection of properties, each consisting of a key-value pair. These properties define the characteristics or attributes of the object.
2. **Methods: Functions as Values**
   - Objects in JavaScript can also contain methods, which are essentially functions associated with the object. These methods can perform actions or compute values related to the object.

Under the Hood: Object's Internal Representation

1. **Object Prototype Chain**
   - In JavaScript, objects are linked to a prototype object. This forms a chain known as the prototype chain. When a property or method is accessed on an object, JavaScript searches for it in the object itself and then in its prototype, continuing up the chain until it's found or the chain ends.
2. **Memory Allocation and Object Storage**
   - JavaScript engines allocate memory for objects dynamically. The internal representation of objects includes not only the actual data (properties and values) but also additional information for the engine to manage the object efficiently.
   - **Object Instantiation: Constructors and Classes**

1. **Constructor Functions**
   - Constructor functions are a way to create multiple instances of objects with a shared structure. They are invoked using the `new` keyword and allow for the definition of object properties and method
2. **ES6 Classes**
   - With the introduction of ES6, JavaScript now supports class syntax for object creation. Classes provide a cleaner and more structured way to define object blueprints.

4. 1. IP Address (Internet Protocol Address):

An IP address is a numerical label assigned to each device connected to a computer network that uses the Internet Protocol for communication. IP addresses serve two main purposes: host or network interface identification and location addressing. There are two types of IP addresses:

- **IPv4 (Internet Protocol version 4):** This is the most widely used IP address format and consists of four sets of numbers separated by dots, such as 192.168.1.1.
- **IPv6 (Internet Protocol version 6):** With the depletion of IPv4 addresses, IPv6 was introduced to provide a vastly expanded address space. IPv6 addresses are much longer and are expressed in hexadecimal, such as 2001:0db8:85a3:0000:0000:8a2e:0370:7334.

IP addresses play a critical role in routing data across the Internet, allowing devices to communicate with each other.

2. Port:

A port is a virtual endpoint for communication in a computer's operating system. Ports are used to uniquely identify different services running on a single device. In networking, a port number is associated with each endpoint, and it helps direct network traffic to the correct application or service. Ports are divided into three ranges:

- **Well-known ports (0-1023):** These are reserved for system services and commonly used applications.
- **Registered ports (1024-49151):** These are assigned by the Internet Assigned Numbers Authority (IANA) to specific services.
- **Dynamic or private ports (49152-65535):** These are used for ephemeral purposes, like temporary connections.

For example, HTTP typically uses port 80, while HTTPS uses port 443.

3. HTTP Methods (Hypertext Transfer Protocol Methods):

HTTP methods, also known as HTTP verbs, define the actions that can be performed on a resource. They are an essential part of the Hypertext Transfer Protocol, which is the foundation of data communication on the World Wide Web. Common HTTP methods include:

- **GET:** Retrieve data from a specified resource.
- **POST:** Submit data to be processed to a specified resource.
- **PUT:** Update a resource or create a new resource if it doesn't exist.
- **DELETE:** Remove a specified resource.
- **PATCH:** Apply partial modifications to a resource.

These methods allow for a variety of interactions between clients (e.g., web browsers) and servers, forming the basis of dynamic web applications.

4. MAC Address (Media Access Control Address):

A MAC address is a unique identifier assigned to network interfaces for communications within a network segment. It is also known as the hardware address or physical address. MAC addresses are assigned by the manufacturer and are embedded in the network interface card (NIC) or network adapter.

A MAC address is a 48-bit address typically represented in hexadecimal format and is used at the data link layer of the OSI model to control the flow of data between devices within the same network. Unlike IP addresses, MAC addresses are generally fixed and don't change when a device moves to a different network.

In summary, these fundamental elements of networking, including IP addresses, ports, HTTP methods, and MAC addresses, play crucial roles in enabling communication and data transfer across the Internet and local networks. Understanding these concepts is essential for anyone involved in network administration, web development, or general IT.