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

The Hyper Text Transfer Protocol (HTTP) is an application layer protocol designed to transfer information between network protocol stack. HTTP is the foundation of web, and used to load webpages using hypertext links.

|  |  |  |
| --- | --- | --- |
|  | **HTTP/1.1** | **HTTP/2.0** |
| Release Year | 1997 | 2015 |
| Key Features | It supports connection reuse  i.e. for every TCP connection there could be multiple requests and responses, and pipelining where the client can request several resources from the server at once. However, pipelining was implement due to issues such as head-of-line blocking and was not a feasible solution. | Uses multiplexing, where over a single TCP connection resources to be delivered are interleaved and arrive at the client almost at the same time. It is done using streams which can be prioritized, can have dependencies and individual flow control. It also provides a feature called server push that allows the server to send data that the client will need but has not yet requested. |
| Status Code | Introduces a warning header field to carry additional information about the status of a message. Can define 24 status codes, error reporting is quicker and more efficient. | Underlying semantics of HTTP such as headers, status codes remains the same. |
| Authentication Mechanism | It is relatively secure since it uses digest authentication, NTLM authentication. | Security concerns from previous versions will continue to be seen in HTTP/2. However, it is better equipped to deal with them due to new TLS features like connection error of type inadequate Security. |
| Caching | Expands on the caching support by using additional headers like cache control, conditional headers like if match and by using entity tags. | HTTP/2 does not change much in terms of caching. With the server push feature if the client finds the resources are already present in the cache, it can cancel the pushed stream. |
| Web Traffic | HTTP/1.1 provides faster delivery of web pages and reduces web traffic as compared to HTTP/1. However, TCP starts slowly and with domain sharding (resources can be downloaded simultaneously by using multiple domains), connection reuse and pipelining there is an increased risk of network congestion. | HTTP/2 utilizes multiplexing and server push to effectively reduce the page load time by a greater margin along with being less sensitive to network delays. |

1. Write a blog about objects and its internal representation in Javascript ?

Object in Javascript are used to store keyed collection of various data more complex entities

An object can be created with{..} with an optional list of properties. A property is a “key:value” pair, where key is a string(also called as property name)and value can be anything.

An empty object can be created using one of the two methods

**Let user=new Object( );** // object constructor syntax

**Let user={ };** // object literal syntax

We can immediately put some properties into {…}as “key:value” pairs:

**Let user={**

**Name:”Rajesh”,**

**Age:25**

**};**

A property has a key (also known as “name” or “identifier”)before the colon ”:” and avalue to the right of it.

**In the user object, there are two properties:**

1. The first property has the name **“ name ”** and the value **“Rajesh**”.

2. The second one has the name “**age”** and the value **30**.

**Property values are accessible using the dot notation:**

// get property values of the object:

**Alert( user.name ):** //Rajesh

**Alert( user.age ):** // 30

The value can be any type.Let’s add a Boolean one:

**User.isAdmin = true;**

To remove a property, we can use the delete operator:

**Delete user.age;**

We can also use multiword property names, but then they must be quoted:

**Let user = {**

**Name:”Rajesh”,**

**Age:25,**

**“Likes birds”:true // multiword property name must be quoted**

**};**

For multiword properties, the dot access doesn’t work:

**// this would give a syntax error**

**User.likes birds = true**

Javascript doesn’t understand that. It thinks that we address user.likes, and then gives a syntax error when comes across unexpected birds.

The dot requires the key to be a valid variable identifier. That implies: contains no spaces, doesn’t include special characters($ and\_ are allowed).

There’s an alternative “square bracket notation ” that works with any string:

**Let user ={ };**

**// set**

**user[“likes birds”]=true;**

**// get**

**Alert(user[“likes birds”]); //true**

**// delete**

**Delete user[“likes birds”];**