**DAY 1 TASK**

**1.Write a blog on the difference between HTTP/1.1 and HTTP/2?**

**Definition of HTTP:**

**HTTP is standing hypertext transfer protocol.**

**It is the basis for almost all web application.**

**HTTP is the method computer and servers use to request and send information.**

**For instance, when someone navigates to website.com on their laptop, their web browser sends a HTTP request to the website.com servers for the content that appears on the page.**

**Then, website server sends HTTP Reponses with the text, image and formatting that the browser display to the user.**

**It is used in client –server.**

**DIFFERENCE BETWEEN HTTP/1.1 AND HTTP/2**

|  |  |
| --- | --- |
| **HTTP/1.1** | **HTTP/2** |
| **1.The first useable version of HTTP was created in 1997.** | **The new version of HTTP called HTTP/2 was**  **Created in 2015.** |
| **2. HTTP 1.1 went through several stage of development, this version is still in use on web is called HTTP/1.1** | **HTTP/2 solves several problems that the creator of http/1.1 did not anticipate.** |
| **3. HTTP/1.1 is slow and less efficient than HTTP/2** | **HTTP/2 is much faster and more efficient than HTTP/1.1** |
| **4.HTTP 1.1 is slower in prioritizes content during the loading process.** | **HTTP/2 is faster in prioritizes content during the loading process.** |
| **5.multiplexing: It load resources one after the other.** | **It loads all resources at the once.** |
| **6. so if one resources cannot be loaded, it blocks all the other resources behind it.** | **HTTP/2 can use a single TCP connection to send multiple streams of data at once. So, no resources block any other resources.** |
| **7.Server push: HTTP/1.1 allows the server to push content to the** **client . once the first content reached and message from the client received, it will send remaining content.** | **HTTP/2.2allows the server to push content to before the client ask for it. the server also sends a message letting the client know what pushed context to expect.** |

|  |  |
| --- | --- |
| **8.Header compression: small files load more quickly than large ones. To speed up web performance, HTTP/1.1 compress HTTP message to make them smaller.** | **HTTP/2.2 use a more advanced compression method called HPACK that eliminates redundant information in HTTP header packet.** |
| **9.header packets is added up takes time resulting in slower loading.** | **Header packets is added up quickly resulting in faster loading.** |
| **10.It works on the textual format.** | **It works on the binary protocol.** |
| **11.There is head of line blocking that blocks all the request behind it until it doesn’t get its all resources.** | **It allows multiplexing so one TCP connection is required for multiple requests.** |
| **12. It uses request resources inlining for use getting multiple pages.** | **It uses PUSH frame by server that collects all multiple pages.** |
| **13.It compresses data by itself.** | **It uses HPACK for data compression.** |
| **14. It consumes more bandwidth.** | **It consumes less bandwidth.** |
| **15.It is secure and less reliable.** | **It is more secure and reliable compared to HTTP/2.** |

**2.WRITE A BLOG ABOUT OBJECT AND ITS INTERNAL REPRESENTATION IN JAVASCRIPT.**

**OBJECT: A object is a group of data that is stored as a series of name value pairs encapsulated in one entity.**

**If defined with keyword / always object**

* **BOOLEAN**
* **NUMBER**
* **STRINGS**
* **DATA**
* **REGULAR EXPRESSION**
* **ARRAYS**
* **FUNCTION**

**DIFFERENT WAYS OF CREATING AN OBJECT IN JAVASCRIPT**

1. **Using the object () construction:**

**Var d=new object ();**

**2. Using the object. Create () method:**

**Var d=object. Create (null);**

**3.Using the function construction:**

**Var object=function (name);**

**{**

**This.name=name;**

**}**

**Var c=new object (“hello”);**

**4.Using the function construction + prototypes:**

**Function my object ()**

**{ };**

**my object.prototypes.name=” hello”;**

**Var k=new my object ()**

**5.Using the singleton pattern:**

**Var 1=new function ();**

**{**

**This.name= “hello”;**

**}**

**INTERNAL REPRESENTATION:**

**A JavaScript object has properties associated with it. A property of the object can be explained as a variable that is attached to the object. Object properties are actually same as ordinary JavaScript variable, except for the attachment to object. The property of the object defines the characters of the object.**

**1.Let person= {};**

**Console.log(person);**

**Result: {}**

**2.let person= {**

**Name: “Nirmala”,**

**Age: 22,**

**Gender: "female”,**

**Address:” Chennai, Tamil Nadu, India”,**

**}**

**Console.log (person);**

**Result: name: Nirmala, age:22, gender: female, address: Chennai, Tamil Nadu, India**

**3.DOT NOTATION:**

**let person= {**

**Name: “Nirmala”,**

**Age: 22,**

**Gender: "female”,**

**Address:” Chennai, Tamil Nadu, India”,**

**}**

**Console.log(person.name);**

**Result:**

**Name: Nirmala**

**4.BRACKET NOTATION:**

**let person= {**

**Name: “Nirmala”,**

**Age: 22,**

**Gender: "female”,**

**Address:” Chennai, Tamil Nadu, India”,**

**}**

**Console.log (person. [Age]);**

**Result:**

**22**

**5.**

**let person= {**

**Name: “Nirmala”,**

**Age: 22,**

**Gender: "female”,**

**Address:” Chennai, Tamil Nadu, India”,**

**Sibling:**

**{**

**Brother:” Janarthanam”,**

**Sister: "Nandhini”,**

**}**

**Console.log (person. Sibling);**

**Result:**

**Brother: Janarthanam,**

**Sister: Nandhini**