**DAY: 2**

**Question no:1**

|  |  |
| --- | --- |
| **Java Script** | **Node JS** |
| * Java script is a programming language. It running in any web browser with proper browser engine. | * It is a Java Script runtime environment. And an interpreter with some specific useful libraries. |
| * Java script is capable to add HTML and play with the DOM | * It doesn’t have capability to add HTML tags. |
| * Java script is used in front-end development | * Node Js is used in server-side development. |
| * it is basically used on the client-side activity for a web application, like possible attribute validation or refereshing the page in specific inverval or provides some dynamic change in web pages without refreshing. | * It is mostly used on the server-side. for accessing or performing any non-blocking operation of any operating system like creating or executing shell script or running any back-end job. |
| * Some of the javascript framework are RamdaJS, typedJS..etc | * Some of the Nodejs module are lodash, express..etc, these modules are to be imported from npm. |
| * It is the upgraded version of ECMA script that uses Chrome’s V8 engine written in C++. | * Nodejs is written in C, C++ and Javascript. |

Questions No:2

*How the browser actually render a website*

**Summary:**

Introduction:

Browser can render a website by, initially parsing the html and also parsing the css then it construct the render tree by creating the DOM tree. After the dom tree it make a layout of web page, and to get a visual output painting takes place.

the following steps are explained process of rendering:

* Parsing.
* Render tree.
* Layout and Painting.

**Parsing:**

Parse the HTML :

The first step The Parse html and parse css are combined to form render tree or frame tree depending on which browser. And also break down the HTML into TOKENS this process is called Tokenizer. A bunch of characters in a text file does not do the browser engine a lot of good. Without this tokenization process, It is forgiving nature and also isn’t straight forward.

**Process of Parsing:**

Script execution

Tree constructionn

DOM tree

Tokenizer

Parse CSS:

By parsing the css it build a CSSOM. The css object model is a map of all css selectors and relevant properties.it also used for build a render tree.

**Render tree or frame tree:**

The render tree is combination of document object model and css object model ( DOM+CSSOM). And represents everything that will be render onto page. This is actual representation of what will show on screen. Render tree is actually multiple tree, there are like four of them,

* Rendering objects
* Render style.
* Render layers
* Line boxes

**Layout and Painting:**

Now, after complete the render tree browser know what to render, but not where to render it. Therefore layout of the page must be calculated.

**Calculating Visual Properties:**

* Combine all styles
* Default, external, style elements & inline.
* Complexity around matching rules for each elements.
* Style computation.

**Recursive process:**

* traverse the render tree.
* Node position and size.
* Layout its children.

**Painting setup:**

* Will take the layed out render tree.
* Creates a layers.
* Incremental process.
* Builds up over 12 phases .

**Painting :**

* Produces a bitmap from each layer.
* Bitmap is uploaded to the GPU as a texture.
* Composite the texture into the final image to render to the screen.