

1. Difference between Browser JS and Node JS

Browser JS	Node JS
It uses Javascript as a programming language.	It also uses JavaScript as a programming language.
It may not support latest updates and releases.	It supports up to date versions and standards.
Browser needs to import the modules.	Node JS requires the modules which is available from the inbuilt libraries.
It is slow compared to Node JS	It is fast as compared to browser JS
It uses interpreter to convert source code to machine code.	It is executing the source code into machine code without using interpreter.
It has no inbuilt libraries.	It has inbuilt libraries such as express.

2. Watch and Summary 5 points:

HTML & CSS Parsing:

Parsing helps the browser to understand where to start and where it will end. The beginning step is called tokenization. This helps in HTML and CSS to send packets with tokens to the client.

DOM & CSSOM Tree

DOM is for HTML and CSSOM is for CSS. DOM also known as Document Object Model helps in making a tree format. This tree construction arranges which element comes first. Say for example : head and body tag comes inside html tag, so html will form a tree structure with two branches as head and body. This is same process followed by CSSOM which is meant by Cascading Style Sheet Object Model.

Rendering Tree

In this stage, DOM and CSSOM will form a tree together to mark and paint the display. All the basic elements will be rendered first.

Layout

Layout stage calculates the space that can be used like length, width, height and elements space consumptions.

Painting

This is the last stage. Here all the complete graphic works, aligning, background setups,etc everything will happen.

3. Explain the tags

```
console.log(typeof(1));
```

This is a number without decimals.

```
console.log(typeof(1.1));
```

This is a number with decimals.

```
console.log(typeof('1.1'));
```

This is a string due to quotation marks.

```
console.log(typeof(true));
```

This is a boolean value. When we compare two values , we get boolean output.

```
console.log(typeof(null));
```

Null is an object. When we specifically update the value as null, it will be displayed.

```
console.log(typeof(undefined));
```

This is undefined as the given value. This happens when the variable is not assigned with any values.

```
console.log(typeof([]));
```

This is an array. Array comes under object type.

```
console.log(typeof({}));
```

This is a block. Block comes under object type.

```
console.log(typeof(NaN));
```

This is Not a Number. This is a number type. When we perform an arithmetic operation between number and string, NaN will be shown.