19/Nov/2018

JS.01 →

1) Environment :- → Browser
→ Node.Js

② Javascript Engines :-

③ Variables → DeepDive → Var, let, Const

④ DataTypes → Number, String, undefines, Boolean, Null

Primitive DataType

→ Array, function, object → reference DataType

⑤ Strings :- methods → String manipulation

Browsers :-

→ Js Introduction   Index.Html
→ intro.js
→ index-Html.

HTML → Hypertext Makeup Language (make up)
(skeleton)

method ① Now How to ron JS in Browses:-

```
<body>

    <suipt src='intro.js'>

    </suipt>

<Ibody>
```

intro.js
↳
Console.log('Hello from the browses')

method ②

```
<body>
  <soript>

Console.log('Hello from the Browser through HTML');

  </script>
<Ibody>
```

JavaScript Eniginep

Chrome ↓ V8 Engine

fire fox ↓ Spider monkey

Inter E. ↓ Chakra.

Node.JS ↳(Buid in V8 engine)

Rayn Dan ↓ local System

V8 ⟶ local System ↓ Node.JS.

Intro.JS

Console.log ("Hello frome Node");

↳ Terminal node intro.JS

" Variable" declaration with let var and Const

```
var a;      // declare first the variable is initialize with undefined
Console.log(a) → o/p undefined.
a=20;
Console.log(a) → 20.
```

// Java Script is a Synchronous Language
means
- Run line by line
- Single threaded langauge.

Javascript is a Dynamic Language
means

```
var a;
a=20
a = 'Hello'
```

} Dynamic behaviour.

(reasing any data type)

// Semi Colon not Compulsory.

Problems in var keywords

Problem 1 first problem with var
- Redecalration

```
var b=2
Console.log(b)
b="I am a String"
Console.log(b)
```

Solution

```
let b=2
Console.log(b)
let b= 'I am a string'
Console.log(b) → (error)
```

(Identifier 'b' has already been declared)

// let keyword does not allows us to redeclare a variable.

// Loops and if else

```
let flag= true;
var num= 13
for(let i=2 ; i <= num ; i++){
    if (num%i==0){
        flag =false
        break
    }
```

```
if (flag){
    console.log('num is prime', num)
  }else{
{       console.log('num is not prime' )
        }
```

Problem ②:- 2nd Problem with var - Scoping
- 1- function scoping
- 2.- block scoping

```
if (10%2 ==0)
 {
    var c = 2
    console.log(c)
  }
    console.log(c)
```

Both work (It become more dangerous) (Var is function Scope)

```
if (10%2 ==0){
        let c=2
          console.log(c)
       }
       console.log(c)
```

Now error (Let is block scope)

Var keyword का use कभी नहीं करेंगे
only use for Interview Preparation.

Const

Const a
a= 'Lollo'
(Not allowed)

Const c = 2

Strings

c = 'Lollo'
errors
// Type errors.
// not reassign.

let a=2
a='hello'
(block scope)

Page No.
Date

{ Let a=2
a=3 // allowed reassigning
Let a=2
let a=3 // not allowed
redeclaring }

Strings

let str = 'pepcoders'
Console.log (str)

String methods :-

1. // .length
Console.log(str.length)

2. Extracting a part of string
   - A → slice (start, end) → return sliced array (from starting to ending -1)
   - B → substr

A) let slice.Str = str.slice( ) (start , end)
= str.slice (3, 7) // last index is excluded
Console.log (slice.Str) → code

B) Substr method

substr (start , length)

let substring = str. substr(3, 4) → o/p Code
Console.log(substring)

It expect starting and the length of the word.

Date

③ Replace method

```
let sayHello = 'Hello Mukesh'
    Console.log(sayHello) ——→ o/p → Hello Mukesh

let sayBye = sayHello.replace('Hello', 'Bye')
    Console.log (sayBye) → o/p → Bye Mukesh
```

expect the word to be replaced as first argument and the replacement argument as Second argument.

④ toUppercase and tolowerCase

```
let text1 = 'Hello world'
let text2 = text1.toUpperCase()
  Console.log (text2) → HEllO WORLD

let text3 = text1.toLowerCase()
  Console.log (text3) → helloworld.
```

⑤

// Concatenation method

```
let firstStr = 'Hello'
let secondStr = 'world'
                                              ('  ', secondStr.)
let ConcatenatedStr = firstStr.Concat(SecondStr)
  Console.log(ConcatenatedStr) → Helloward
                                     Hello word
```

⑥ Trim Method (remove space)

```
let text = '   Hello world   '
let trimmedText = text.trim()
Console.log(trimmedText) → o/p → Hello world
```