Javascript

Date ___/___/_____

Saathi

High level, interpreted programming language
Runs on the client/browser as well as on server
It is programming language of browser
Used in building very fast server side and
full stack applications.
used in Mobile and Desktop Development.

<head>
<script src = "ja.js"></script> ya
</head>

<body>
</body>

(fast execution)

External

<script>
</script>

alert ("Welcome to World!"); <- do ya na do.
document.write ("Sanjali"); // white screen ko document

// variables

var a; var 10; var 10.10;
var a = "thak";

var a=10;
var b=20;
var c= a+b;
document.write (c);

a++ ++a

```
var d = a<b;        // false
```

== है कि वही है। sirk value ko check

=== Datatype ko check karta hai

```
switch (10) {
   case 10: document.write("its a 10");
            break;
```

```
var arrays = new Array;      or = new Array("tha", "H");
arrays[0] = "thapa";
arrays[1] = "H";                          or
document.write(arrays[1]);          ("thap", "H");
```

```
alpha . concat (numeric);
  ↓                ↓
array1          array 2
```

```
emp.sort();
```

```
sum //                                  → sum();

function sum(){                          sum(a, b)
      var a = 10;
      var b = 20;
      var c = a+b;
      docum. wr(c);
}
```

Saathi

Date ___/___/_____

// String

```
var str = "Javascript";
var answer : str.length;
                .charAt(4);
                .indexOf('s');
                .localeCompare("apa");
```

// Events

```
<button onClick = "myfun()" > </button>
```

(Click)

```
function myfun() {

}
```

onMouseOver , onmouseout , onmousedown , onsubmit

//eval 4+4 => 8

// window.location = "https. "

confirm ("Good Morn")

// prompt

var str = prompt ("Hi! My name is Vino", "");
alert ("Nice to meet you" + str + "!")
;

Math.floor (10.99); → 10
Math.log (0) → -∞ (,) → 0
Math.pow (4, 2) → 16
Math.random () → 0 से 1 के बीच Between

Document.write (Date())

// form validation

<form name = "myform" onsubmit = "return valid ()">
Username : <input type = "text" value = "">

id

Pass : <input type = "pass" value = "">

<input type = "submit" value = "submit">
</form>

function validation () {

Date ___/___/_____ Saathi

var cn = document.getElementById("user").value;

11

(1) function sum(a,b){ → let sum2 = (a,b) => {
return a+b; return a+b
} }

(2) function isPositive(number){ let isPositive = (number) =>
return number >= 0 number >= 0
}

(3) document.get('click', function(){
console.log('click')
})

document.add('click', () => console.log('click'))

... arr [] let kar value

const numbers = [1, 2, 3];
console.log(...numbers); 1 2 3

null v/s undefined

```
let name;
console.log(name);
```

```
let node = {
    parent: null
};
```

null: intentionally absent

undefined: uninitialized

null == undefined
false

#

```
const arr = [1, 2, 3, 4]  ------>  1, 4, 6, 8
                                   (new array)

function callback (element) {
    return element * 2;
}

const newArr = numbers.map (callback)
```

Higher Order function

are functions that accepts function as an argument or return function as a result. or both

```
function higherOrderFunc(func) {
    return function() {
        func()
    }
}
```

- reuse logic

Saathi

Date ___/___/_____

Reduce

Multiple values → Single value

sum of array of no.

```
const numbers = [1, 2, 3, 4];
const sum = numbers.reduce(callback);

const callback = (accumulator, currentItem) => {
    return accumulator, currentItem;
}
```

Immediately Invoked function expression

Anonymous function that is executed immediately after it is declared is known as IIFE

```
(function () {
    console.log("Hello World!");
})();
```

↳ new scope

↳ allows you to use await

```
(async function () {
    const res = await fetchData();
})();
```

Tagged Template literal

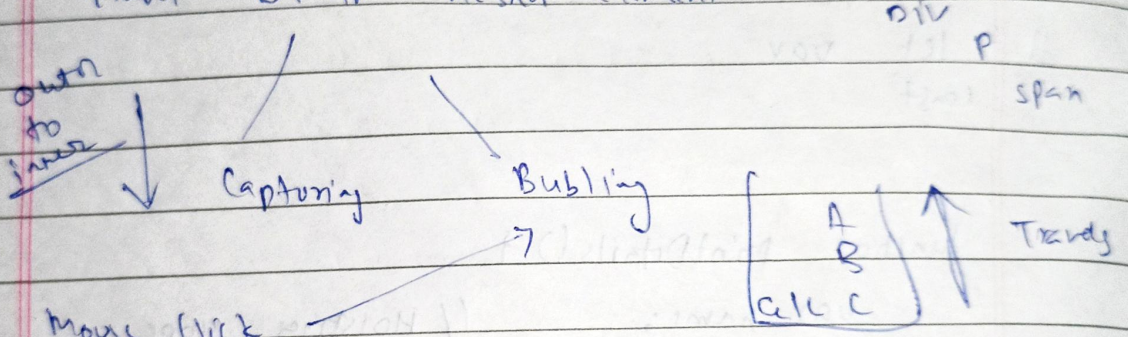
```
const color = 'green';
const element = 'span';

generate 'This is a ${color} ${element}';
```

↓

function before them

Event propagation

Travel within nested element



spread may nested element copy nhi hoy

Array Destructuring → extract value from array and assign them to individual variable using modern syntax

Date ___/___/_____

Saathi

```
const numbers = [10, 20, 30];

const [first, second, third] = numbers

console.log(first, third); // 10 20 30
```

Callback

is just a function passed to another function as an argument.

v/s

let var const

```
function printDetails() {
    var name;          // Hoisting scope
    if (true) {
        var name = "Jane";
        let age = 23;
    }
    console.log(name);  Jane
    console.log(age);   not defined
```

// let block scoped only accessible inside block.
// const are also block scope

Type → statically

Saathi

Date ___/___/_____

Java → Dynamically typed

Data types of variables are determined by the value they hold at runtime and can change throughout the program as we assign diff values to them

// function short (){
    return(
        true || console.log("Hello")
    );
}

console.log(short()); → True

// short ckt evaluation

// ECMA script ES6 → set of rules a lang sho behave

// passed by value in Javascript

// Javascript is interpreted language

Just in time compiler