

Inline javascript

file = .js

External javascript

JS (Javascript)

JS is used for functioning.

Inline javascript <script>

INTRODUCTION OF JS

- * JS was invented by Brendan Eich in 1995.
 - * It was developed for Netscape 2.8 became the ECMAScript 262 standard in 1997.
 - * European Computer Manufacturers Association Ecma International (formally European Computer Manufacturers Association) is an organization that develops standards in computer & technology.
 - * ES1 to ES5 (1997 to 2009)
 - * After that in 2015 (major changes to follow the rules and regulations) this is called EcmaScript / ES 2015 / ES6.
 - * ES6 is standard for javascript after that every year new changes came ES7, ES8, ES10 Etc.
 - * JS is a light weight object oriented programming language.
 - * Use in form submit.
 - * In client side validation
 - * Pop up / events on click.
- Uses:
- * Client side execute / browsers (JS query, Reactjs, angular JS)
 - * Website Server Side (node.js, Express.js)
 - * Mobile Development (Hybrid app) (framework for mobile app react native, phone gap etc)
 - * Software development (Electron.js, Ex - vs code, framework etc)

Console.log() se print hogya console
pe.

Variable Rules:

- 1- Case Sensitive variable: Name same ho ~~letter~~
agl wo ^{1st} capital letter or dusra small letter me ho to
wo dono algi alg honge
i.e. Letter, letter
- 2- Koi bhi key word variable ka name
nhi alkhi skte.
- 3- Variable me ~~letter~~ letter, number, -, dollar
sign "\$" likh skte hoga.
- 4- Variable name can't be start with
digits (number) its first letter.
- 5- No space allowed.
 - As a good programmer,
 - your variable name should match with its contents
 - When you want 2 words form in variable name so first word start with small letter and 2nd word start with capital

VARIABLE Scope:

- 1- Block scope variable: if variable declare in block of code (in curly brackets {})
 - * It will alive only in block and will not be accessible after curly brackets
- 2- Global Scope variable: these variables used globally in whole program.

Comments in JS:

- Single line // ;
- multi line /* */

Print / Display in JS:

- On browser : window.document.write
 - (" ") ;
 - Sub-object → sdocument.write (" ") ;
- In Console : console.log (" ") ;
- 3- Pop up : window.alert (" ") ;

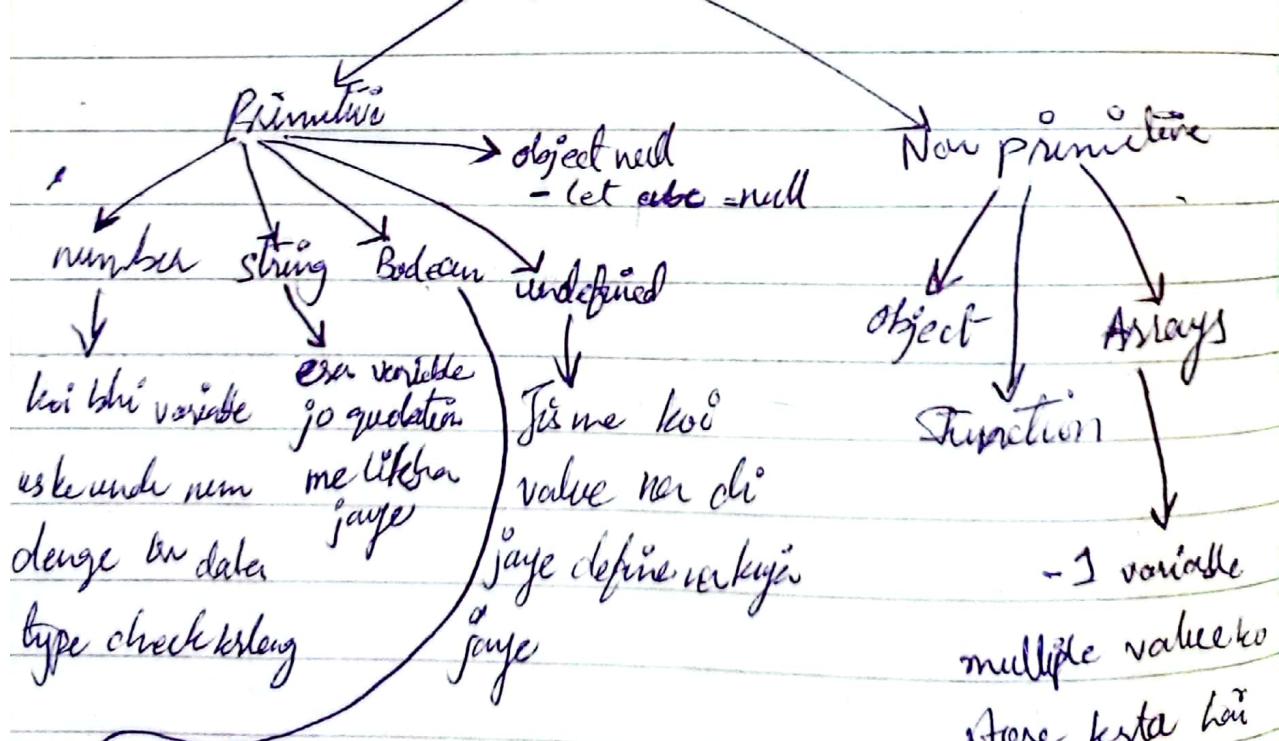
TAKING INPUT from Users in JS:

Prompt: In JS we use the prompt function to ask the user for input. As a parameter we input the text we want to display to the user. Once the user presses "OK" the input value is returned. We typically store user input in a variable so that we can use the information in our program.

```
let num =  
    prompt ("choose any number : 1-104)
```

* console.log (typeof)

Data types in JS



True or False
defined kar le
hain

* let info [5] [name] computer
index 1 index 2 index 3

~~Object~~. Obj → store multiple value in single variable
- Value written in curly brackets in pair with keys.

* ~~roll no~~

let student → {

roll no =

name = "A",

subj = "C",

}

let items {

name = "A", "

welcome()

price = ,

~~sum~~ let answer =

rating =

sum(4, 5);

available: true,

document.write

offer: 20,

(answer);

}

console.log(

functions

function name()

{ document.write("welcome"); }

}

function sum(a, b)

{ let c = a + b; }

return c;

OPERATORS IN JS:

1- Arithmetic Operator. Examples $a+b$, $4+5$

here $4, 5$ is operand & $+$ is operator.

$(+, -, \times, \div, \% \text{ modulus}, / \text{ remainder}, \text{Exponentiation})$

~~2 Ass~~

double operand

a^{++}

Increment, Decrement

Single operand

let $a = 5$

let $b = 7$

1- document.write($a+b$);

2- document.write($a + b$);

3- $a = a + b$

4- $a * b$

5- a / b

6- $a \% b$

7- $a ** b$

Unary operators: Increment

$a^{++};$

, document.write(a^{++})

document.write(a);

$b^{--};$

Decrement

document.write(b);

Pre decreament:

document.write(--a);

Pre increment:

document.write(++a);

2- Assignment Operators

=, +=, -=, *=, % =, ** =

For = :

a = 2

For += :

a += 4 / a = a + 4;

For -= :

a -= 4 / a = a - 4;

For *= :

a *= 4 / a = a * 4;

For % = :

a % = 4 / a = % 4;

For ** = :

a ** = 4 / a = a ** 4;

let $a = 8$; "a me jo bhi value hai
 $1 - a^t = 4$; usko is value me add
hogaega."

2- $a^t = 3$; "a me jo bhi value hai
usko is value se subtract
hogaega."

3- $a^t = 5$; "a me jo value hai usko
is value se multiply hogaega"

4- $a^t = 6$; "a me jo value hai usko
is value se divide hogaega"

5- $a\% = 7$; "a me jo value ho usko is se
divide kerte jo remainder
aasga 60."

6- $a^{t^n} = 9$; "a me jo power deni hoga
wo is se denge."

COMPARISON OPERATOR;

1. $=$ (equal to)

Just check
value

$==$ (equal to + same)
(data type)

2. $<$ (less than)

let age = 23;

if (age > 18)

document.write ()

let a = 5;

let b = 5;

if (a == b)

document.write ("yes")

}

3. $>$ (greater than)

Value is the
sth data type
bhi check
krega

LOGICAL OPERATOR:

logical AND $\&\&$ → condition 1 & condition 2

logical OR $||$ → any 1 bhi condition true ho to hogi

logical NOT !

let a = 5;

let b = 6;

if (a > $\&\&$ b > 5)

alert ("both conditions true")

alert ("your conditions false")

CONDITIONAL OPERATOR

- * if
- * else if
- * if else if

2- let a = 5;

let a = 6;

if (a > 7 || a > 9)

{ alert (" "); };

else {

alert (" "); };

3- let a = 5;

let a = 6;

if (! (a < b))

{ alert (" "); };

else {

alert (" "); };

{ }

book name: Smarter way to learn JS.

W3 school
mtn website

TERNARY OPERATOR: 1 line code

let age=15;

let result;

result = age > 18? "adult": "not adult"
~~document~~ alert(result);

Practice:

let num = 403

let result = num % 2 == 0? "Even":

"odd":

~~document~~.alert(result);

Marksheet

let roll no. = 56;

let name = " ";

let class = " ";

let emarks = 3 - - -

let total = emark + - - -

let per = total / 500 * 100;

let grade;

if (per <= 100 && per >= 89)

{ grade = "A+" ; }

else if (per <= 90 && per >= 79)

{ grade = "A" ; } - - - -

else { grade = "F" ; }

(Roll No. " , roll no);

document.write (roll no);

" (< b > name);

" (< b > < c > class);

" (< table border = 1 >),

" (< tr > < th > Subjects < / th > < th > Mark
obtained < / th > < th > Out of < / th >)

4. (< tr > < td > English < td > " , emarks , < /
" < td > 100 < / td > ");

"

"

"

copy paste

~~SWITCH~~ STATEMENT

let ~~reply~~ = prompt ("Do you want to continue?")

switch (~~reply~~)

case "y":

document.write ("continue");

break;

case "yes":

document.write ("continue");

break;

case "n":

document.write ("end");

break;

~~default:~~

~~document.write ("~~

case "no":

document.write ("end")

break;

~~default:~~

~~document.write ("wrong")~~

2 Feb 2021

STRING:

* String is a sequence of characters used to represent a text.

* It is a primitive data type.

* We can create string by using template literals and single & double quotation

- String in double quotations:

let var = " " ;

- String in Single quotations:

let var = ' ' ;

- String in template literal:

let var = ` ` ;

Template Literal:

- Feature in JS introduced with ES6.

Uses of template Literal:

- For next line

let var = `hello \n how are you`

- For space.

let var = `hello \t how are you`

- For print:

let str = `hello` ;

let num = 5 ;

document.write(`hello \${num}`)

PROPERTIES & METHODS OF STRING

→ String me letters count krtा है।
OR LENGTH : let str = "hello! how are you";
dt l = str.length;
document.write(l);
document.write(str[]); → For find the position of letter

For concat or join two string:

document.write(str1 + str2)

(str1 ~~+~~ str2)

for space b/w them

: newvar = str.concat(str1, str2, str3);

: document.write(newvar)

TRIM : → starting or ending se space remove करेगा string ki

document.write(str.trim());

TRIM console.log

TRIM START : → Just start se space remove करेगा

document.write(str.trimStart());

TRIM END : Just end se space remove करेगा

document.write(str.trimEnd());

console.log

- **To UPPERCASE** : → Poore sentence capital alphabet document.write(stu.toUpperCase()); me hoga.
- **To LOWER CASE** → Poore Sentence small alphabets document.write(stu.toLowerCase()); me hoga.
- **Replaces**
document.write(stu.replace("jo kya wanhai", "Jis se replace kina"));
(stu.replaceall " ")
poore tent ke ande wo word
jha jha hoga wo replace
hogaega.
- **INCLUDES** : → Poore sentence me wo word find kina.
document.write(stu.includes("Jo word", "find kina"));
* value true or false me aagega.
- **Slice** : Jo ~~us~~ letter ki position num denge us ko leaogen.
document.write(stu.slice());

template literal,

let roll no = 56;

let name = " ";

document.write('my roll no is \${roll no}
my name is \${name});

FOR CALCULATION:

let num1 = 78;

let num2 = 89;

document.write('the sum of \${num1} and
\${num2} is \${num1 + num2}.');

String manipulation:

* Per next line:

Sentence \n Sentence

* Per space:

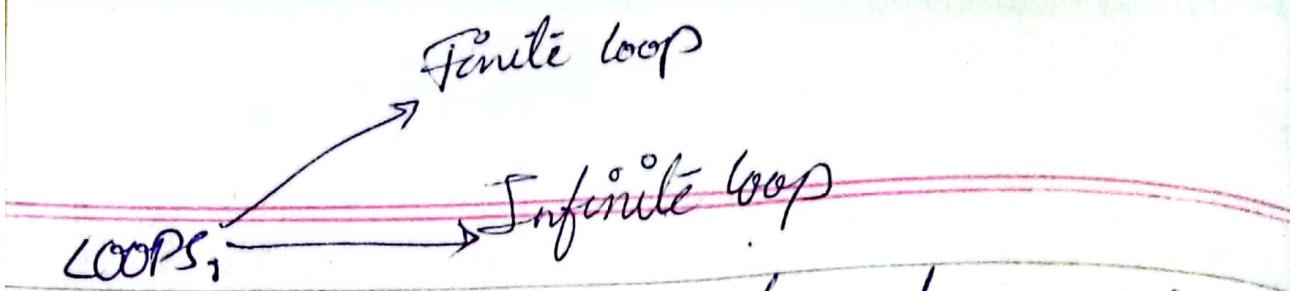
Sentence \t Sentence

* Per quotation:

Sentence " Sentence

Per Single /:

Sentence // Sentence



- to execute a piece of code again & again
- Finite loop (ending point)
- Infinite (not end) memory full / computer hang
↓
Ye are not known.

FOR Loop:

First step, Variable declare

Example, \rightarrow block scope variable use for counting.

(let $k = 1$; $k \leq 4$; $k++$)

Second step: Condition check. If the condition true block of code execute ho.

Third step: Updation

CODING:

{ for (let $k = 2$; $k \leq 6$; $k++$)

 document.write("Hello");
}

Jini baar

→ Print-karwani ho

let tab no = ~~parseInt~~ (prompt)
ParseInt
ager
table off no. of your choice
for ($i = 1 ; i <= 10 ; i++$)
long

document.write(\$`\$`{labno}
\$`\$`{i} = \$`{labno * i})
document.write(`
`)

For backward counting
for (let i = 1 ; i >= 10 ; i--)

ARRAY (Non-primitive data type)

- * store multiple value in single variable
- * values written in square brackets []
- * Values separated by comma.
- * Each position is called index
- * Each value call through index no. start with 0

CODING:

let arr = [$\$8$, "a", "a"];
document.write(arr[0], arr[2]);

2- FOR OR Loop:

For (let i = 0; i < arr.length; i++)

```
{  
    document.write(i);  
}
```

Properties & methods of Arrays

Function in JS:

- A JavaScript function is a block of code designed to perform a particular task.
- A JS function is executed when something invokes (calls it).

Function

definition

function fname()

{

 block of codes

}

Function invoke

(call)

fname()

Slice :

```
let array = [2, 4, 6, 8, 10, " ", 3];  
document.write(array.slice(index no.))
```

jo index no. kchange
wohi se aage se parne
print hoga.

Index-Of : To find the position of index.

Splice :

```
document.write(arr.splice(2, 1))
```

index \downarrow
index 2 \downarrow
per daye \downarrow 1 item \downarrow rep.
delete

1st value : index

2nd value : remove index

3rd value : replace item.

OBJECT:

- * Store multiple value in single variable.
- * Values written in curly brackets {} in pairs with keys.

Cooling:

```
const student =
```

```
{ name : " ",
```

```
    rollno. : " ",
```

```
    Subject : " ".
```

FOR IN LOOP

```
for (let key in student)
```

Var ↓ value of for in
 variable

```
document.write(key, student[key], "
```

FOR OF LOOP

```
let var = [1, 2, 3, 4, 5, 6]
```

```
for (let i of var)
```

```
{
```

```
if (i % 2 == 0)
```

```
{ document.write(i, "
```

```
}
```

```
}
```

While Loop:

```
let i = 1  
while (i <= 10)  
{  
    do current.write (i, "hello<br>");  
    i++;  
}
```

Do While loop:

```
let i = 11;
```

```
do  
{
```

```
    document.write (i, "hello<br>");  
} while (i <= 10);
```

```
let i = 1;
```

```
let sum = 0
```

```
do  
{
```

```
    document.write (i);
```

```
    sum = sum + i;
```

```
i++;
```

```
} while (i <= 10);
```

```
document.write (sum);
```

"sum of 1 to 10"

Properties & Methods of Array

array push : Array me item insert kr dega end me

```
let arr = [ , " ", " ", " " ]
```

```
arr.push(" ");
```

```
document.write(arr);
```

array unshift : array ke start me item insert kr dega.

```
arr.unshift(" ");
```

```
document.write(arr);
```

array shift : Remove a word from start.

```
arr.shift();
```

```
document.write(arr)
```

array pop : Remove item from end

```
arr.pop();
```

```
document.write(arr)
```

To String : Convert array into string

```
arr.toString();
```

```
document.write(arr)
```

Array concat: Join 2 or 3 array in new array. can't change original array.

```
let arr = [ , " ", " ", " ", " ", " " ];  
let arr2 = [ " ", " ", " ", " ", " ", " " ];
```

```
let arr3 = arr.concat(arr2);  
document.write(arr3);
```

-HTML pe function ko call kaise

- <button on click="abc"> click me/ button
function name

JavaScript pe function ko call kaise:

```
abc();  
function  
name
```

function definition:

function fname(p1, p2) → parameter

Coding:

abc(5,7);

let a = 5;

let b = 8;

let ans = mult(a, b);

function mult(p1, p2)

let ans = p1 * p2

return ans;

document.write(ans).

let a = 7;

let b = 5;

abc(a, b)

direct print:

let abc = mult(7, 4);

document.write(abc)

function call:

fname(arg1, arg2) → argument

Arrow function: (ES6) advanced.

const fname = () => { block of code }

Coding:

const mult = (p1, p2) => {

let ans = p1 * p2,

return ans;

let ans = func (")

")

~~let~~ function vowel(str)

let i = 0;

for (let charac of str)

{ if (charac == "a", charac == "e") —

document.write (charac, "
");

i++ ; , i = i + 1 ;

}

} document.write (i); }

Same work in arrow functions

let str = " ";

const vowel = (str) => {

let i = 0;

for (let charac of str)

if (charac == "a", charac == "e") —

{ document.write (charac, "
");

i++ ;

document.write (i);

}

vowel (str);