

JAVASCRIPT

Classmate

Introduction:

- JS was invented by Brendan Eich in 1995.
- It was developed for Netscape 2, and became the ECMA-262 Standard in 1997.
- European computer Manufacturers Association Ecma International (formally European computer manufacturers Association) is an organization that develops standards in computer and technology.
- ES1 to ES5 (1997 to 2009).
- After that in 2015 (major changes to follow the rules and regulations) this is called Ecma Script/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.

Class #2

→ Pop up / events on click

Uses -

- Client side execute (browsers.
(JS query, React JS, angular JS))
- Website Server side (node.js,
Express.js).
- Mobile development (hybrid APP)
frame work for mobile app react
native, phone gap etc).
- Software Development (Electron JS,
EU - Vscode, frame worker).

<head>

script defer src = "script.js" ></script>

we have qa

use script language in CSS

Class #1

* html

→ Script tag is used in Javascript

<head>

In head section memory
load ↴

'<title>.first class of JS </title>
<script src = "script.js"></script>

</head> ↴ b/w v/v ↴ ↓ by, we closing body
tag ↴ tag ↴

<body>

show pop up.

<script> alert (message?: any): void
alert ("Hello") .

</script>

→ HTML read line by line - 1st read
head section.

JS script.js

alert ("Hina");

Error shown in console window by
inspector

Class #2

Variable: → also called container

↳ space reserved in box

Index.html.

<head>

<title> Document </title>

</head>

<body> <h1> Variables </h1>

<script> src = "Variable and datatype.js" </script>

</body>

</>

Types of variables:

→ Var → before 2015 it was used.

→ let

→ Const] → 2016 start from:

Variable and data type - JS

Var a=6; → Old version.

Error ↗ (a) declare number optime.

Var a=5; → no response.

let a=5; One-line memory save.

let a=10; Save variable no declare

→ assign in let
operator.

Const pi = 3.142; → semi colon means
sentence end.

const abc; error.

Value should be given.

```
let name="Kiran";  
name="Sumit";  
console.log(name);
```

const abc = 84;

(a) notrowable with ~~to~~ ^{Save} keyword.

let a=5;

a=10; let a=5;

↗ var right.

const abc = 84; } error because it is
abc = 74; const abc

const abc = 84; } No error

Data types:

- number → `let age = 10; console.log("age");`
- String → `let fullNumber = "10"; console.log(fullNumber);`
- Boolean → `let isPass = true; console.log(isPass);`
- null → `let nullNo = null;`
- undefined
- Array
- Object
- functions.

Variable Rules:

→ Case sensitive

Capital letter → A AI

Small letter → a ai

Keywords:

→ let, var, not declare as variable

→ letter, variable, number and underscore
dollar sign should be write

let as - \$4

→ Variable will start from number

→ Variable start from dollar, alphabet

camel case start from small letter and in center we use capital letter

Class # 3

Variables

- Variables just like a container.
- Variable is used to store information.
- It reserve space in memory. & Its data can vary but memory location will always remains same.

Naming Variables In Java Script

Rules:

- Variable's name can't be any key word e.g alert, prompt etc.
- Variable is case ~~sensitive~~ sensitive, same name in capital and small and underscore. letters are different. e.g Name or name (both are 2 different variables).
- Variables can be consist of alphabet, numbers, dollar sign and underscore.
- Variable name can't be start with digit (number) in first letter.

→ no space allowed.

As a Good programmer:

→ Your variable name should match with its contents

→ When you want 2 words join in variable name. So first word start with small letter and 2nd word start with Capital letter.

e.g. fullName, roll Number etc.

Types of variables

Var → (used before EcmaScript. this type of variable can be declare again and again etc.). After ES6 in modern or advance Javascript these 2 keywords use for declaration variables.

let (its value can change any time in programming language and can declare and assign in 2 steps)

e.g. let name ; (declare)

name = "Hello" ; (assignment)

Const → It is used for constant value e.g. pi value) • Its value can't be changed. It value must be assigned at the time of declaration -

e.g const name = "Hina" ; (declare and assign in same sentence).

Variable Scope

1 Block Scope variable: If variable declare in block of codes (in curly bracket { })

If will alive only in block and will not be accessible after curly bracket

2 Global Scope Variable: These variables used globally in whole program.

~~Comments~~

Comments in JS.

Single line: // let name = "hina"; //

multi line: /* */

Print / Display in JS.

On Browser: window.document.write

In Console: ("hina");

~~Pop~~ In Console: console.log("hina");

Popup: window.alert("hina");

Taking Input from Users in JS.

→ Prompt: In Javascript, 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.

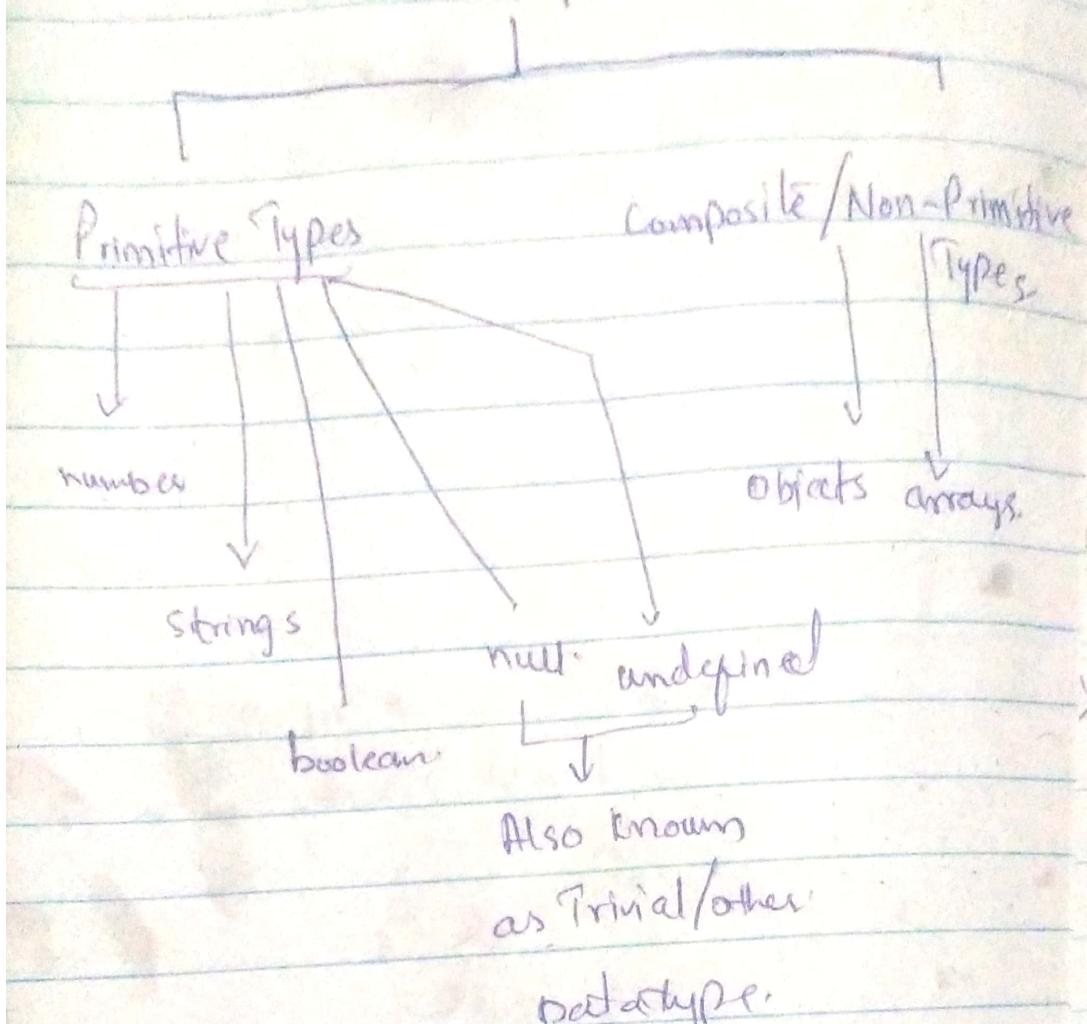
MONday
smarter way to learn.
in schools
monday

let answer = prompt ("Do you want to
Send Payment Y/N? : ");

In answer variable value will be stored
and you can print it.

Variables and Data types in JavaScript

Data types in JS.



Operators in JS

- Arithmetic Operators
- Assignment Operator
- Comparison Operators
- Logical Operators
- Conditional operators

function welcome()

{
document.write("Welcome");

}

function sum(a,b)

{
let c = a + b;

return c;

}

welcome();

~~sum(a,b);~~

let answer = sum(4,5);

document.write(answer);

xx

document.write(a,"xx",b,"=",a * * b,);

Arithmetical Operators

let a=9;

let b=2;

Addition.

document.write($(a+b)$); // 11

or

document.write(a,"+",b,"=",a+b);
a + b = 11
"11"

Subtraction.

document.write(a,"-",b,"=",a-b);

Multiplication

document.write(a,"*",b,"=",a*b);

Divide

document.write(a,"/",b,"=",a/b);

% (modulus)

document.write(a,"%",b,"=",a%b);

let a=5;

let b=3;

a++;

document.write(a);

b--;

OR

document.write(a++);

document.write(a);

~~b~~;

document.write(a--);

document.write(--a)

Assignment Operators

let a = 2;

document.write(a);

↓

2

id a = 2;

a = 4;

document.write(a); → 4.

a -= 3;

document.write(a); → ①

a * = 3;

document.write(a); → 6.

a /= 3;

document.write(a); → 6

a % = 3;

document.write(a);

a ** = 3;

document.write(a); → $2 \times 2 \times 2 = 8$

Ch 1 Class Comparison Operator

$= =$ (equal to) $\approx \approx$ (equal to + Same data type)

\neq (not equal to)

\neq (not equal to & data type)

$>$ greater \rightarrow true

$>=$ greater than

$<$ less than

\leq ..

Logical Operator

\rightarrow logical And &

\rightarrow logical OR ||

\rightarrow logical NOT !

Conditional Operator: Ternary Operator

If statement

If - else statement

If - else if statement

Condition? true

outputs: false output

e.g.,

* JS

if $a == b$

let $a = 2$; // number

let $b = "2"$; // string

$a == b$ true

$a !== b$ false

$a != b$ (a not equal to b) \rightarrow false.

$a != b$ (true). $(==)$ \rightarrow not check
data type.

// comparison operator

let $a = 5$; // number

let $b = "5"$; // String

if ($a == b$) \rightarrow b is a string
document.write("no") or ("hello")

}

let $a = 5$;

let $b = "5"$;

If ($a == b$) or ($a != b$).

1

document.write("hello")

no print

1

```
let age = 16; // number  
// let b = "5"; // string S = "5"  
if (age >= 18)
```

```
{  
    document.write("You can vote");  
}
```

```
else  
{
```

```
    document.write("You cannot vote");  
}
```

Print:

You can not vote.

// Comparison operator

```
let age=18; // number  
// let b='5'; // String s=5.  
if(age<18)  
{  
    document.write("you are child");  
}  
else if (age>40)  
{  
    document.write("you are old");  
}  
else  
{  
    document.write("you are young");  
}  
Print:  
You are young
```

Logical Operator:

let a=5;

let a=6;

if ($a > 2 \& \& b > 5$)

}

document.write

or

alert ("both condition are
true");

else {

 alert ("your's conditions are true");

}

if ($a > 7 \& \& b > 5$)

{

 document.write ("both condition are true");

else {

 document.write ("your's condition not true");

}

Prints

let a = 5;

let b = 6;

if (a > 7 || b > 9) // If 2 conditions

{

 alert ("hello");

}

else {

 alert ("welcome");

}

 or

if (! (a < b)) // False

{

 alert ("hello");

}

else {

 alert ("Welcome");

}

Print → welcome

if (! (a > b)) // true

Print → hello

Ternary Operator: (Logical Operator)

~~let age;~~

let age = 20;

let result; ^{variable}

result = age > 18 ? "adult" : "not adult"

alert(result);

Print adult

let age = 10;

let result;

result = age > 18 ? "adult" : "not adult";

alert(result);

Print

not adult

Class # 6

<head>

<title>Marksheet </title>

<head>

<body bgcolor = " " >

<script src = "marksheet.js" > </script>

</body>

</html>

MarkSheet : 38

```
let roll no = 36;  
let name = "Hina";  
let teacher name = "Mrs";  
let class = "8";  
let Emarks = prompt ("Enter Your English's  
marks");  
let Emark = 70;
```

```
let total = emark + Umarks + Pmarks + Cmarks  
+ imarks;
```

```
let per = total / 500 * 100; →
```

let grade;

document.write (roll no, total, per);

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

grade = "A+";
document.write (roll no, "Grade", grade);

document.write (total);

Mark Sheet J.S.

let roll no = 01;

let name = "Majid";

let class = "8th";

let t name = "Miss Nabeela";

let emarks = 98;

let umarks = 98;

let Mmarks = 100;

let Tmarks = 48;

let Pmarks = 48;

let Cmarks = 48;

let Smarks = 98;

let SImarks = 48;

let total = emarks + umarks + Mmarks

+ Tmarks + Pmarks + Cmarks + Smarks

+ SImarks;

let Perc = total / 600 * 100 ;

let grade;

if (Perc <= 100 & & Perc >= 89) || 90 - 100

{ grade = "A+" ; }

else if (Perc <= 90 & & Perc >= 79) || 80 - 89

{ grade = "A" ; }

else if (Perc <= 80 & & Perc >= 69) || 70 - 79

{ grade = "B" ; }

```
else if (per <= 70 && per >= 59) // 60 - 69  
    | grade = "B"; }  
else if (per <= 60 && per >= 49) // 50 - 59  
    | grade = "D"; }  
else | grade = "U"; }
```

// now create marksheet

```
document.write("Roll NO:", rollno);  
document.write("Name:", name);  
document.write("Class:", class);  
document.write("Teacher Name:", tname);  
document.write("<table border=1>");  
document.write("<tr><th> Subjects </th>  
<th> Out Of </th> <th> Marks Obtained </th>  
</tr>");  
document.write("<tr><td> English </td> <td> 100 </td>");  
document.write("mark, " <td> <td> 100 </td>");  
"
```

```
"  
document.write("<tr><td> Total </td> <td> ",  
    total, "</td> <td> 600 </td>");  
document.write("<tr><td> Percentage </td> <td> ",  
    per / total * 100, "%");  
document.write("<tr><td> Grade </td> <td> ",  
    grade, "%");
```

Class #7
Switch Statement ex.

* 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;

case "no":

document.write ("end");

break;

default:

{ document.write ("wrong input"); }

Arithmetic Operators:

+ (Addition)

- (Subtraction)

* (multiplication)

/ (Division)

% (modulus / remainder)

Exponentiation

• Increment

• Decrement

Unary Operators:

Post increment a++

Pre increment ++a

Post decrement a--

Pre decrement --a

String. (primitive datatype).

- 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 in single and double quotation.

elif is the python coding.

String Creation and manipulation

```
let str1 = "I am learning js"; // double  
let str2 = 'I am learning ss'; // single  
let str3 = `I am learning string template  
literal | (adjacent to \` key board.  
or is called back tick
```

escape the backslash of the bracket
use button to
Template literal: template
variable.

Template literals are a feature in
JavaScript that were introduced with
ES6. They give you a more flexible
and maintainable way of working
with strings in JavaScript

How to use template literal.

for next line \n
for tab (space) \t
for print \ in string
for write variable in string \$ \${variable}
name.)

for double quotation "hello!" hello'

Syntax:

- Normal string: document.write(" ");
- template literal: document.write(` \${ } `);

let str = "Hello! how are you?"
document.write(str);

let num = 5;
document.write(

let str = "Hello! how are you?";

let str2 = "I am learning";

let str3 = "world";

document.write(str + str2);

↓

Hello! how are you? I am learning

↓
(answer)

document.write(str + " " + str2);

↓ answer

Hello! how are you? I am learning

let newVar = str.concat(str2, str3); //

document.write(newVar); // concatenation.

Some String Properties and methods.

```
let str1 = "I am learning Js";
```

```
let str2 = "css";
```

```
let str3 = "html";
```

Position / index Start with 0 in string.

to find length str.length

to join strings

```
document.write(str1 + " " + str2)  
str1, " ", str2).
```

by concat()

```
let str4 = str.concat(str2)/str.
```

```
concat(str2, str3) document.write  
(str4)
```

```
let str = "Hello! how are you?"
```

let I = str.length;

```
document.write(I) or document.write(str.length)
```

↓
1a (answer):

```
document.write(str[0]); // O
```

↓
(w) → answer

```
document.write(str[0]); // U // index -  
↓  
such
```

- Trim
- Str.trim() // to remove space from start and end.
 - Str.trimStart() // → space remove from start and end.
 - Str.trimEnd() //
 - Str.toUpperCase() // Change in uppercase
 - Str.toLowerCase() // change in lowercase
 - Str.replace(" javascript", " html") //
 - Search word and replace(case sensitive)
 - Str.includes("is") // search word is / if not found return -1
 - Trim is used to remove space.

```
let str2 = "    I am learning    ";
```

```
document.write(str2);  
str.trim();  
console.log(str2.trim()); //remove  
space from start and ending.  
↓
```

```
    I am learning  
console.log(str2);  
↓
```

```
I am learning
```

let str = " Hellow "

let str = " HELLOW ";

document.write(str.toLowerCase());

↓

hellow

let str = " hellow ";

document.write(str.toUpperCase());

↓

HELLOW

let str = " Hellow ";

document.write(str.replace("Hellow",
"hi."));

↓

hi

document.write(str.includes("ell"));

↓

(true) → show

document.write(str.includes("er"));

↓

(false) → show

document.write(str.slice(1));

↓

Class # 8

```
if(reply == "y")
document.write("continue...");
```

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

Switch (reply)

```
{
```

1 Template Literal. ^(L1) back tick.

```
let rollno = 56;
```

~~let~~ str = `let name = "hina";`

```
document.write(`my roll no is ${rollno}
my name is ${name}`);
```

```
let num1 = 56;
```

```
let num2 = 66;
```

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

F12 → key → console

console.log(" My name is fatima \n my
subject is computer ");

My name is fatima
my subject is computer

console.log(" My name is fatima \t my
subject is computer ");

↓

My name is fatima My subject is computer

console.log(" My name is fatima \t my
subject is "computer");

↓

My name is fatima My subject is "computer"

console.log(" My students are fatima \t salma \t sara ");

↓

My students are fatima \ salma \ sara .

Trim

```
let str2 = " I am learning "
document.write(str2, "  
");
document.write(str2.length, "  
");
document.write(str2.trimStart());
```

I am learning.

48

I am learning.

loops

→ Go in circle after piece of code again
and again

→ finite loop and infinite loop

→ finite loop (ending point).

→ finite loop (not end) memory full /
computer hang

For loop

```
for (let i=1; i<=5; i++) {  
    document.write("Hello");
```

i is block scope variable. Use for
iteration / counting.

first step initialization 2nd condition
check if that condition true block
of code execute

3rd step update

for (int i = 1; i < 10; i++) {
 cout << "Hello world";
}

document-writé ("Hello");
} // ("Hello world");

hello

hello

hello

hello

hello

for (int i = 1; i < 10; i++)

document-writé (i);

1
2
3
4
5
6
7
8
9
10

Practice questions:

→ Print writing 10 to 1.

→ Print your name 20 times on screen.

→ Print table square number by

- taking user input.

Print Table

```
let tabno = prompt("Enter  
Table no of your choice");
```

```
for (i=1; i<=12; i++) {
```

```
i
```

```
document.write(' '); // 3x1
```

```
}
```

```
i
```

```
document.write('${tabno}x${i} =
```

```
 ${tabno*i}); // 3x1 = 3*1
```

```
} document.write("<br>");
```

Class #9

Arrays:

Hand:

<body>

<script src="array.js"></script>

for (i=20;

for (let i=20; i>=1; i--)

{

document.write (
, i);

}

↓

20

19

18

17

16

15

14

13

12

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 number start with 0
- e.g arr[0], arr[1].

Syntax:

```
let info = [5, "hina", "computer"];  
console.log(info);
```

Info = []; // To empty array

`let arr = [34, "Sadia Adil", "JavaScript"];
document.write(arr[0]);`

↓ ↗
34 ②

or

`document.write(arr[1]);`

↓
JavaScript

or

`document.write(arr.length);`

↓

3

or

`let i = arr.length - 1;`

`for (let i = 0; i <= 1; i++)`

{

`document.write(arr[i], "
");`

} ↓

34

Sadia Adil

JavaScript

~~Almond Bay Sod & Soil Co.~~

~~'longaberger's)~~ is
the part (below) is mutual
domestic uses (are) in

~~300, Sod & Soil, Inc., complete deal~~

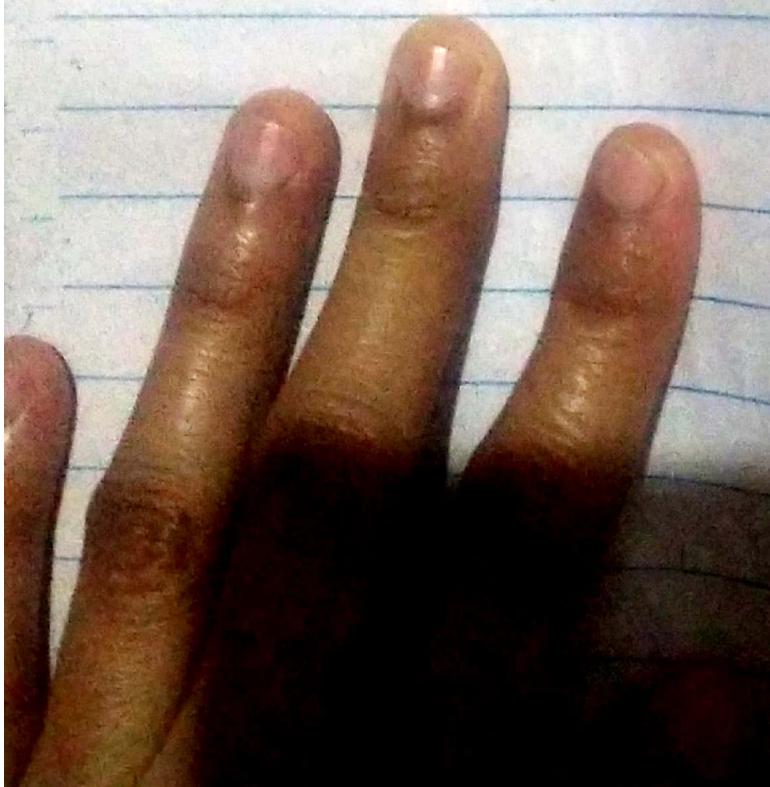
```
let arr = [34, "Sadia Adil", "Java",  
          "Computer"];
```

```
arr.push("html"); // insertion
```

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

f

34, Sadia Adil, Java, Computer, html



Properties and Methods in Array,

let book = ["Math", "English", "Urdu"] ;

- 1) book.length // length of array.
- 2) book.push("Chemistry"); → add word in array in the end.
- 3) book.unshift("Chemistry"); → add in start of array.
- 4) book.shift(); → remove a word from start.
- 5) book.pop(); → to remove word from last and return updated array.
- 6) book.toString() // convert array in string.
- 7) book.concat(book2) // join 2 or 3 array in one array don't change original array.
- 8) book.indexOf("Urdu"); → to find any words position in array.
- 9) book.slice(startIdx, endIdx);
don't change in original array // return a slice piece of array.

`book.splice(startIdx, delCount, next);`
Change original array // add, remove
and replace

e.g

`let arr = [1, 2, 3, 4, 5, 6, 7];
arr.splice(2, 2, 'S', 'G'); // index per
3 // 3 and 4 remove // replace
// them S and G.`

`arr.splice(3, 0, 'A'); // if you
don't want to delete any item //
3 index per just add A`

`arr.splice(2, 3); // if you don't want
add only delete items // 2 idx and
delete 3 items`

Class 10.

```
let array = [0, 1, 2, 3, 4, 5, 6, 7, 8];
```

```
document.write(array.splice(1, 2, 98, 99),  
"  
");
```

```
document.write(array);
```

array.splice(1, 2);

↓

0, 3, 4, 5, 6, 7, 8

↓

0, 88, 3, 4, 5, 6, 7, 8;

```
let array = [2, 3, 56, 78, 65, 23, 89];
```

```
console.log(array.splice(1, 4, 22));
```

↓

```
document.write(array);
```

2,

↓

22, 3, 56, 22, 65, 23, 89

Object:

- Store multiple value in single variable
- Values written in curly bracket {}
in pairs with keys
- Syntax -

```
let student = {  
    name: "hina",  
    rollno: 23,  
    Subj: Class: "computer"  
}
```

```
document.write (student);  
document.write (student.rollno);
```

Number always without question

Non Primitive Data Types

Array:

→ Store multiple value in single variable.

Variable:

→ values written in square brackets []

Syntax:

```
let info = [5, "hina", computer];
console.log(info);
```

Print:

```
document.write(info);
document.write(info[7]);
```

for in loop → Print the keys also.

for In loop

const Student =

{
name: "Fatima" , For In loop
roll no: 55 ,

subject: "Math"

{
heading (head).
for (let key in student)

{
documents.write (key) .

↓

name roll no subject

{
documents.write (key, "
")

{
name
↓

name

rollno

subject

{
documents.write (key, student [key], "
")

{
Y

name = Fatima

rollno = 55

subject = "Math"

document.write (student - subject);



"Math"

For Bf (arr p.

let arr = [1, 2, 3, 4, 5, 6];

for (let i of arr)

{

document.write(i);

}



1 2 3 4 5



document.write(i, "
");



1

2

3

4

5

6

For of loop:

let arr = [1, 2, 3, 4, 5, 6];

for (let i of arr) {

 1

 if (i % 2 == 0) {

 document.write(i, "br");

 2

 3

 ↓ Point.

 4

 5

 6

Q) Print odd numbers of this array:
[34, 45, 67, 98, 24, 12, 1, 5, 6 -]
using for loop.

Q) Create an object and print keys
and data with the help of for loop.

While and Do while loop

→ Print sum of 1 to 20 numbers by using
while loop and do while loop.

Let $i = 1$

While ($i \leq 10$)

{
document.write(i , " Hina
");
}

$i++$;

{
↓

1 Hina

2 Hina

3 Hina

4 Hina

10 i = 11

do

:

{

document.write(i , " Hina
");

$i++$;

} while ($i \leq 10$);

↓

false cancel this

(21 Hina
one time
execute)

let i = 1;

let sum = 0;

do

{

document.write(i);

sum = sum + i;

i++;

} while (i <= 10);

document.write(sum);

↓

1

2

3

4

5

6

7

8

9

10

55

let i = 1;

let sum = 0;

do

{

document.write(i, "
");

sum = sum + i;

i++;

} while (i <= 10);

document.write("sum of 1 to 10 = ", sum);

↓

sum of 1 to 10 = 55

function.html

<body>

<button onclick="abc()">Click me</button>

<script src="function.js"></script>

Function

definition (define)

function invoke

(call).

function fname()

• fname();

{

block of codes

argument
↑
parameter

fname(arg1, arg2)

}

parameter receive.

or

fname(2,3)

function fname(P_1, P_2)

{

document.write($P_1 + P_2$). let ans = sum()

}

- function sum(P_1, P_2)

console.log(a)

ans = $P_1 + P_2$;

return ans

Class II

functions i.e.

"hina".toUppercase();

document.write("hina".toUpperCase());

function abc()

document.write("hina");

]

↓

hina hina hina hina.

document.write("hina ");

"

"

↑

abc() ↓

hina hina hina hina.

$\text{abc}(\text{c}, \text{g});$

function abc(p₁, p₂) {

document.write(p₁ * p₂); // 35

}

↓ Show

35

document.write(p₁); → Not print

It is error

p₁ can not define

let a = 6

let b = 8

abc(a, b);

function abc(p₁, p₂) {

document.write(p₁ * p₂); // 48.

}

↓

48

// function define
function mult (P₁, P₂) {

let ans = P₁ * P₂;

return ans;

}

let a = 6;

let b = 8;

let ans = mult (a, b);

document . write (ans);

let abc = mult

1d abc mult (7, 4);

document . write (ans);

↓

28:

Arrow Function

↓ advanced

latest → ES6 me Agar ho

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

Const mult = (p1, p2) => {

let ans = p1 * p2;

return ans;

}

let a = 9;

let b = 8;

let ans = mult(a, b);

document.write(ans);

↓

72.

let a = 9;

let b = 8;

let c = 2;

let ans = mult(a, b, c);

document.write(ans);

$$\begin{array}{r} 72 \\ \times 2 \\ \hline 144 \end{array}$$

i → iteration
c → counting

(class #12 $\text{str} \rightarrow \text{string}$)

function is char work as a
value (as field function). array

function $\text{write}(\text{str})$ for loop use
in string -

let $i = 0;$

for (let char of str).

nested {

if ($\text{char} == "a" \text{ || char} == "e" \text{ || char} == "i"$)

block {

$\text{document.write}(\text{char}, "\backslash<br\>");$

$i = i + 1;$

or

$i++;$

}

$\text{document.write}(i);$

}

↓
Point:

a

e

i

o

u

l

loop (ie render)

loop use same

no nested

relative be.

$\text{||} \rightarrow \text{L}$

A row function
let str = "abe def gh ijk";
Vowels (string);
const vowels = (str) => {
 let i = 0;

 for (let char of str)
 {
 if (char == "a" || char == "e" || char == "i"
 || char == "o" || char == "u")
 {
 document.write (char, "
");
 i++;
 }
 }
}
Vowels (string);

function table(num)

```
{  
    for(let i=1; i<=10; i++)  
        document.write(` ${num} x ${i} = ${num*i}`)  
    }  
}  
table(6);
```

$$6 \times 1 = 6$$

$$6 \times 2 = 12$$

$$6 \times 3 = 18$$

$$6 \times 4 = 24$$

$$6 \times 5 = 30$$

$$6 \times 6 = 36$$

$$6 \times 7 = 42$$

$$6 \times 8 = 48$$

$$6 \times 9 = 54$$

$$6 \times 10 = 60$$

function table (num)

{

```
for (let i = 1; i <= 10; i++)
```

{

```
document.write(`${num} x ${i} = ${num * i});
```

```
document.write("<br>");
```

}

}

```
let tableNo = prompt("Enter table number");
```

```
table(tableNo);
```

↓

$$2 \times 1 = 2$$

$$2 \times 2 = 4$$

$$2 \times 3 = 6$$

$$2 \times 4 = 8$$

$$2 \times 5 = 10$$

$$2 \times 6 = 12$$

$$2 \times 7 = 14$$

$$2 \times 8 = 16$$

$$2 \times 9 = 18$$

$$2 \times 10 = 20$$

```
const student =
```

```
{ name: "h
```

```
roll no: "
```

```
Class: "
```

```
is Pass: "
```

```
}
```

```
for (let
```

```
{
```

```
document
```

```
}
```

```
name: "
```

```
roll no:
```

```
Class: "
```

```
is Pass: "
```

Const student =

{
name: "hina",

roll no: 33,

Class: 7,

is Pass: true

}

document.write(key, "
");

}

or

↓
document.write(key, "
student[key], "
");

name

roll no

class

is Pass

name = hina

roll no = 33

class = 7

is Pass = True.

Array

```
let city = ["Karachi", "Lahore",  
           "Islamabad", "Peshawar"];
```

```
document.write(city[0]);  
for (let val of city) {
```

```
}
```

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

```
let city = ["Karachi", "Lahore",  
           "Islamabad",];
```

```
document.write(city.shift());
```

```
↓  
Karachi (Print start item)
```

or

```
document.write(city.pop());
```

```
↓  
Print last item  
Islamabad.
```

```
document.write(city);
```

Web development
Sohail Gharib

CMS (Content Management System)

- Word Press
- Drupal
- Magento

→ Xamp control

Dashboard

→ local host / batch

→ Localhost start

CMS (Content Management System)

CMS is a software
browsers and allows
modify website.

- Word press
- Drupal
- Shopify
- Magento
- Joomla

document.write((it, "Mulla",
" (br>));

document.write((it, push ("faisalabad"),
" (br>));

let city = ["Karachi", "Lahore",
"Islamabad", "Peshawar");

: it, splice (2, 2, "Sialkot");

document.write((it, "(br>));

J

Karachi, Lahore, Sialkot.

2, 2 → item → p6, p7
↓ ↓ delete

if no deleted.

document.write((it, "(br>", it,
length));

J

Karachi, Lahore, Sialkot
Islamabad, Peshawar

let cit1 = ["Karachi", "Lahore", "Islamabad",
"Peshawar"] ;

let cit2 = ["Rawalpindi", "Khanewal",
"Bahawalpur"] ;

let cit3 = [] ;

Cit3 = Cit1.concat(Cit2) ;

document.write(Cit3, "
", (Cit3.length))



Karachi, Lahore, Sialkot, Islamabad,
Peshawar, Rawalpindi, Khanewal
Bahawalpur .

~~Cit3.slice(3, 6) ;~~

console.log(Cit3.slice(3, 6)) ;



Islamabad, Peshawar, Rawalpindi +

Civic

→ You are walking in a Park, enjoying the beautiful surroundings and you finish your snack. There is no trash bin nearby, but you notice an empty area. What to do?

Carry the wrapper until you find a ~~too~~ trash bin

→ You are at a busy intersection, and the traffic light turns red. However the pedestrian signal is still on Do Not Walk. What is the right thing to do?

- A) wait patiently on the side walk until the pedestrian signal allows you to cross

You're on crowded bus, and you notice an elderly person standing while you're sitting what should you do?

- A) Offer your seat to the elderly person.

Your community is organizing a cleanup day to keep the streets clean. What's your response?

- A) Volunteer your time to help clean up the neighbour.

You come across a public wall covered in graffiti. What is your reaction?

- A) Report it to the local authorities or community group for clean up.

You have recyclable items at home what do you do with them?

- A) Place them in a separate recycling bin for proper disposal.

You have to visit your relative's home - what do you do?

- A) Ask them to collect trash and take some away.

What is Civic Sense?

It refers to the awareness and consideration of one's responsibility and duties towards society. It involves behaviors and attitudes that contribute to the well-being of the community as a whole.

Why Civic Sense Is important?

These are the social ethics -

→ It ~~imposes~~ imposes every steps of life.

Importance of Civic Sense.

→ A society with a strong civic sense is more likely to be peaceful, clean, and cooperative.

→ It enhances the overall quality of life for everyone.

→ We need to be good listeners.

→ Should have廉俭勤儉.

How to promote civic sense ??

- We need to care Public property
 - Should be follow the laws.
 - find out the Solution → These are Optimistic people.
 - Should be proactive.
- We should learn from childhood -