

# JavaScript :-

- ⇒ JavaScript is a cross platform, object based scripting language invented specifically for use in web browsers to make website more dynamic and attractive.
- ⇒ A **Script** is a program or sequence of instructions that is interpreted or carried out by another program.
- ⇒ Some languages: VBScript, Javascript, Jscript and ECMA Script.
- ⇒ Scripts can be executed on client or the Server.

## Client - side Scripting

- Runs on the user's computer i.e. Browser interprets the Script.

1. Visible to user

My view - the  
html source.

## Server side Scripting

- Runs on the web server.

2. Not visible to user.

Note: ECMAScript is the official name  
of the language.

- o Javascript is mainly used as client side Scripting language.
- o it was originated with Netscape and invented by Brendan Eich.
- o it was initially called as Live Script.
- o Launched in 1995, after an agreement with sun microsystems, Live Scripting was replaced with javascript, to compete with popularity of java.

## Uses of Javascript:-

- ⇒ Java Scripting as programming language has many uses.
  - o Validating user input on client side
  - o for file handling and database connectivity on the server side.
  - o Can be used as alternative to applets.
  - o To implement programming on Client side web documents.
  - o Implement event driven programming
    - o Along with DOM, javascript can be used to develop DHTML.

## Why Study javascript?

- o Javascript is one of the 3 languages all web developers must learn.
- 1) HTML to define the content of web pages
- 2) CSS: to Specify the layout of web pages.
- 3) Javascript to Program the behavior of web pages.

## #What is Javascript can do?

- 1) Javascript can change HTML Content:
  - o getElementById() method changes the html element content.
  - o Javascript can accept both single and double quotes.

### Example:

```
document.getElementById("demo").  
innerHTML = "Hello javascript";
```

- 2) Javascript can change html attribute values.

### Example:

In the following example javascript  
the value of the `src` attribute  
of a `img` tag,

- `<button`
- `onclick = "document.getElementById('myimg').src = 'Pic-balloon.gif'">` Turn off the light
- `</button>`
- `<button`
- `onclick = "document.getElementById('myimg').src = 'Pic-balloon.gif'">` Turn on the light
- `</button>`

1) Javascript can hide HTML elements

by changing the display style.

Example:

`<button type="button" onclick = "document.getElementById('demo').style.display = 'none'"> click me! </button>`

15) Javascript can Show HTML elements:-

- Showing hidden HTML elements can also be done by changing the display style.

3) Javascript can change HTML styles (CSS):  
◦ Changing the style of an HTML element is a variant of changing on HTML attribute:

Example:

`<p id="demo" style="display:none">Hello Javascript! </p>`  
`<button type="button" onclick = "document.getElementById('demo').style.fontSize = '35px'"> Click me! </button>`

## JavaScript Where to:-

### <script> tag:

- ⇒ Java code is inserted between `<script>` and `</script>` tags.

#### Example:

```
<script>
```

```
document.getElementById("demo").innerHTML  
= "my first javascript";
```

```
</script>
```

- o Attribute `type` is not required. JavaScript is the default scripting language in HTML.

### JavaScript in <head> or <body>:-

- ⇒ Any number of scripts can be placed in an HTML document.

- ⇒ Scripts can be placed in the `<body>` or in the `<head>` section of an HTML page or in both.

### JavaScript in <head>:-

- o A javascript function is placed in the head section of an HTML page.
- o The function is invoked when a button is clicked.

#### Example:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<script>
```

```
function myFunction() {
```

```
document.getElementById("demo").innerHTML  
= "Paragraph changed";
```

```
}
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<button type="button"
```

```
onclick = "myFunction ()"> Try it! </button>
```

```
</body>
```

```
</html>
```

## JavaScript in body:-

- ⇒ A javascript function is placed in the `<body>` section of an HTML page.
- ⇒ The function is invoked when a button is clicked.

Example:

`<body>`

```

<button type="button"
        onclick="myfunction()>Try it
</button>

```

`<script>`

```

function myfunction(){
    document.getElementById("demo")
    .innerHTML = "Paragraph changed."
}

```

}

`</script>`

## External Javascript:-

- ⇒ Scripts can be placed in external files.

⇒ External Scripts are practical when the same code is used in many different web pages.

⇒ JavaScript files have the file extension `.js`.

- ⇒ To use an external Script put the name of the script file in the `(src)` attribute of a `<script>` tag:

Example:

`myscript.js`

```

function myFunction(){
    document.getElementById("demo").innerHTML =
    "Paragraph changed."
}

```

`<script src="myscript.js"></script>`

Note:

- o External Script reference can be placed in `<head>` or `<body>`.
- o The Script will behave as if it is located exactly where the `<script>` tag is located.

o External Scripts cannot contain `<script>` tags.

## Poss of external Javascript:-

- o It separates HTML and code.

- makes easier to read and maintain HTML and JavaScript.
- Cached JavaScript files can speed up page loads.
- To add several Script files to one page, use several Script tags.

Example:

```
<script src="myScript.js"></script>
```

```
<script src="myscript.js"></script>
```

### External references:-

- ⇒ An external Script can be referenced in 3 different ways:
  - With a full URL (a full web address)
  - With a file path (like /js/)
  - Without any path

Example: (1)

<script

```
src="https://www.w3schools.com/js/myScript.js">
```

</script>

(2)

```
<script src="/js/MyScript.js"></script>
```

(3) <script src="myScript.js"></script>

How to Create dynamic web pages?

There are two methods to create dynamic web pages:

⇒ Server based script  
⇒ Client side scripts

### Server Side Scripts:

- Server Side Scripts are stored and executed on servers.
- These scripts are used to provide user interaction.
- Data from the client machine is sent to the web server.
- Server Side Scripts process the request and send the required page to the client browser.

## Client-Side Scripts:

- ⇒ Client-side scripts are executed on client computer.
- ⇒ The browser supports the use of scripts.
- These scripts can be embedded in HTML document.
- ⇒ The browser interprets and executes the code embedded in the page.

## Javascript Display Possibilities

- Javascript can "display" data in different ways:
  - o Writing into an HTML element, using "innerHTML".
  - o Writing into the HTML output using "document.write()".
  - o Writing into an alert box, using "window.alert()".
  - o Writing into the browser's console, using "console.log()".

## Using innerHTML:

```
<body>
<script>
    document.getElementById("demo").innerHTML
        = "Hello";
</script>
```

⇒ It is a common way to display HTML data.

## Using document.write():

```
<body>
<script>
document.write(s+b);
</script>
```

- o Convenient to use this method for testing purpose.
- o Using this after an HTML document is loaded will delete all existing data.

```
<body>
<button type="button"
onlick = "document.write(s+b)">Try it</button>
```

## Using window.alert():

- o Alert box can be used to display data.

```
<script> optional keyword
window.alert(s+b);
</script>
```

## Using console.log():

Console.log() method is called in the browser to display data for debugging purpose.

```
<script>
console.log(s+b);
</script>
```

## JavaScript Statements:

⇒ A javascript program is a list of programming statements.

⇒ JavaScript statements are composed of:

- o Values
- o Operators
- o Keywords
- o Comments

⇒ multiple statements can be written and they are executed one by one.

⇒ Semicolon: Should be added at the end of executable statements.

# Javascript Syntax

## How to Create variables?

=> var, let, const keywords are used to declare variables.

=> Javascript is a case sensitive language.

=> Rules for declaring variables

Same like C++.

### Example:

var x;

var y;

let f;

### How to use variables?

x=5, y=6

let z=x+y;

## Javascript values

=> The javascript defines two types of values:

- Fixed values

- Variable values

- Fixed values are called literals.

- Variable values are called variables.

## Ways to declare variables

- Using var
- Using let
- Using const
- Using nothing

### Let:

=> Variables defined with let:

- Cannot be redeclared.
- must be declared before use.
- have block scope.

Block scope means they can be used just inside a {} block cannot be used outside the block.

Note: - But we can reassign values.

var keyword does not have block scope.

var variable can be redeclared.

=> with let, redeclaring a variable in the same block is not allowed.

var x=2; // Allow

let x=3; // Not Allow

let x=2 // Allow

let x=3 // Not allow

⇒ Redefining a variable with let in  
another block is allowed.

### let hoisting:

```
CarName = "Volvo";
var CarName;
let CarName;
```

### Const:

⇒ Variables defined with const:  
o cannot be redeclared.  
o Cannot be reassigned  
o have block scope.

⇒ must be assigned when they declared  
when to use:

- o A new array
- o A new object
- o A new function
- o A new RegExp

### Constant array:

o Elements can be changed of constant array.

```
const cars = ["Sed", "Volvo", "BMW"];
```

```
Cars[0] = "Toyota"
```

```
Cars.push("Audi")
```

⇒ But: reassign the whole array if it is not allowed.

### Constant objects:

- o Properties of objects can be changed.
- o But objects cannot be reassigned.

### Example:

```
const Car = { type: "Fiat",
model: "500", color: "white"};
Car.color = "red"; // Allow
```

## JavaScript Operators

### (i) Arithmetic Operators:

o +, -, \*, \*\*, /, %, ++, --

### (ii) Assignment Operators:

o =, +=, \*=, /=, %=, \*\*=

### (iii) Comparison operators:

⇒ ==, !=, >, <, >=, <=, ?

== (equal value & equal type)

!=

#### (iv) Javascript logical Operators:

- &&
- ||
- !

#### (v) Type Operators:

- typeof
- instanceof

#### (vi) Bitwise Operators:

&, |, ~, ^, <<, >>, >>>, <<<

#### (vii) String Operators:

- `<`, Compare two strings alphabetically
- `+`, Concatenate two strings
- `+=`
- `let z = "Hello" + s; // Allow`
- `let y = "S" + s; // Allow`

#### (ix) Bitwise assignment operators:

`&=`, `^=`, `||=`

#### (x) Logical assignment operators:

`&&=`, `|||=`, `??=`

## Javascript datatypes

Javascript has 8-different datatypes.

- 1) String
- 2) Number
- 3) BigInt
- 4) Boolean
- 5) undefined
- 6) Null
- 7) Symbol
- 8) Object

## Javascript objects

⇒ Javascript objects are written in curly braces {}.

⇒ Object properties are written as name: value pairs, separated by commas.

### Example:

```
const Person = { firstName: "John",
  lastName: "Doe", age: 50, eyeColor: "blue" };
```

## Undefined:

- A variable declared without a value, has the value undefined.

`let car;`

## Javascript Strings

- o Strings can be written in single or double quotes.

- o String.length:

  - o Built in property

Example:

```
let text = "Hello";
```

```
let length = text.length;
```

## Javascript Comments

=> Increase the readability.

=> // Single line comment

=> /\* and end with \*/ are multi-line comments.

Comments:

---

## Javascript Functions

Syntax:

```
function fun-name (parameters)
```

{

```
return P1 + P2;
```

}

## Function invocation

- o When an event occurs
- o When it is invoked from JS code
- o Automatically (self invoked)

## Javascript Arrays

Syntax:

```
Const array-name = [ items, items, ... ];
```

Syntax:

```
Const cars = new Array ("Isab",  
"volvo", "BMW");
```

# Javascript Conditional Statements

## if Statement:

Syntax:

```
if (Condition)
```

```
{  
    // block of code  
}
```

## if else:

Syntax:

```
if (Condition)
```

```
{  
    // block of code  
}
```

else

```
{
```

```
    // block of code  
}
```

## if-else-if:

```
if (Condition)
```

```
{
```

```
    // block of code  
}
```

else if (condition a)

```
{  
    // block of code  
}
```

else

```
{  
    // block of code  
}
```

## Switch Statement:

Syntax:

```
switch (expression)
```

```
{
```

case x:

```
    // code
```

```
break;
```

case y:

```
    // code
```

```
break;
```

default:

```
    // code of block  
}
```

# Javascript Loops

## for loop:

Syntax:

```
for (expression1; expression2; expression3)  
{  
    // block of code  
}
```

## While loops:

Syntax:

```
while (Condition)  
{  
    // block of code  
}
```

## do while loop:

```
do
```

```
    // block of code  
}
```

```
while (Condition);
```

## Past Paper Questions

### + Exercise:

#### (i) What can a variable store?

- o it is a named memory locations.
- o used to store program's input data and its computational results during execution.
- o value of variable can change during the execution of program.
- o However, the variable name cannot be changed.

#### (ii) By default? to which part of a document will javascript code output?

Are the operators && and and interchangeable?

- o No, && and and are not interchangeable.
- o They operate differently.
- o "&" it is bitwise AND operator
- o "&&" it is logical AND operator and work similar AND gate.
- o "&" compare each bit of numbers one by one.

Example:

$$a = 5 \& 6 \Rightarrow 14 \quad \begin{array}{r} 101 \\ 110 \\ \hline 101 \end{array}$$

$$a = 5 \&\& 6 \Rightarrow 1 \quad \begin{array}{r} 100 \\ 100 \\ \hline 100 \end{array}$$

why it is a good idea to explicitly declare properties within a class?

Q. How can you create a multi-dimensional array?

- o A multi-dimensional array is not present natively in javascript.
- o Hence, there is no direct way to create a multi-dimensional array in javascript.
- o However, we can define an array of elements to create a multi-dimensional array in such a way so that each element stored in array is also another array.

Ans (Ch#14) :-

Q. What characters are used to define a javascript variables name?

- ⇒ All javascript variables must be identified with unique names.
- ⇒ These unique names are called identifiers.

General rules:

- o Names can contain letters, digits, underscore & dollar signs.
- o Name can also begin with \$.
- o Reserved words are not allowed.
- o Names are case sensitive.

When would you use == the identity operator?

⇒ When we also compare values as well as their types.

⇒ When you wish to bypass PHP automatic type changing (operator type changing always casts to float)

# What are the two Simplest two forms of expressions?

⇒ The most basic forms of expressions are:

- o Literals or Constants
- o Variables.

# What is the best way to force your own operator precedence?

The best way to force your own operator precedence is to place parentheses around subexpressions to which you wish to give high precedence.

# What is the difference b/w unary, binary & tertiary operators?

# What is Operator associativity?

⇒ It refers to the direction of processing.

- o L-to-R
- o R-to-L

o It becomes important when we explicitly force precedence.

Why is a for loop more powerful than a while loop?

⇒ Loops using for statements are more powerful than while loop because they support two additional parameters to control the loop handling.

- ⇒ 1) An initialization expression
- 2) A condition expression
- 3) A modification expression

They are separated by semicolons.

Name three conditional statement types.

- o if statements
- o Switch statements
- o ? Operator.

# What is the purpose of with Statement?

o The with statement takes an object as its parameter.

o Using it, you specify an object once.

then for each statement within the blocks, what object is consumed.

Example:- changing floating point numbers into integers:-

$n = 3.1415$

`i = parseInt(n)`

`document.write(i)`

compound form:-  
`document.write(parseInt(3.1415))`

How are boolean values handled differently by PHP and Javascript?

$\Rightarrow$  In PHP TRUE represents the value 1 and FALSE represents now which can be thought of as "nothing" and is output as the empty string.

' '.

#What are explicit casting functions in javascript?

Change to type Function table

int, integer	<code>ParseInt()</code>
Bool, Boolean	<code>Boolean()</code>
Float, Double, Real	<code>ParseFloat()</code>
String	<code>String()</code>
Array	<code>Split()</code>

# How to validate form by javascript?

=> Javascript is used to validate the contents of the forms.

=> Form validation is a process of checking that proper information entered in the form before it is submitted.

## Common validation types

- 1) validating the forename
- 2) validating the surname
- 3) validating the username
- 4) validating the age
- 5) validating the email

### Validating forename:

```
function validateforename ( field ) {
```

```
    action ( field == "" ) ? "No forename entered"
```

```
}
```

```
function validateforename ( field ) {  
    if ( field == "" ) ? "No forename entered"  
        validate user name:  
    function myFun () {  
        var a = document.getElementById ("username").value;  
        if ( a == "" )  
            {  
                <script>  
                function validate ( form ) {  
                    fail = validateForename ( form . forename . value );  
                    fail += validateSurname ( form . surname . value );  
                    fail += validateUsername ( form . username . value );  
                    fail += validatePassword ( form . password . value );  
                    fail += validateAge ( form . age . value );  
                    fail += validateEmail ( form . email . value );  
                    if ( fail == "" )  
                        return true  
                    else
```

```

        {
            alert("fail")
            return false
        }

    }

function validateForename(field)
{
    if (field == "")? "No forename was entered.\n": ""

}

function validateSurname(field)
{
    if (field == "")? "No surname was entered.\n": ""

}

function validateUscn(field)
{
    if (field == "") {
        return "No username was entered.\n"
    }
    else if (field.length < 5) {
        return "User name must be at least 5 characters."
    }
    else if (field.length > 20) {
        return "Username must be less than 20 characters."
    }
}

```

```

else if (!/[a-zA-Z]/.test(field))
{
    return "Only a-z , A-Z , _ and allowed in usernames.\n"
}

}

function ValidatePassword(field)
{
    if (field == "") {
        return "No password was entered.\n"
    }
    else if (field.length < 6) {
        return "Password must be at least 6 characters.\n"
    }
    else if (field.length > 85) {
        return "Password length must be smaller than 85 characters."
    }
    // check if password contains at least one uppercase letter, one lower case letter and one number
    var UR = /[A-Z]/;
    var LR = /[a-z]/;
    var NR = /[0-9]/;
    if (!UR.test(field) || !LR.test(field) || !NRtest(field)) {
        return ("Password must contain at least one uppercase letter, one lower case letter, and one number.");
    }
}

```

```
function validateage(field){
```

```
    if (isNaN(field))
```

```
        return "No Age was entered.\n";
```

```
    else if (field < 18 || field > 110)
```

```
        return "Age must be b/w 18 and
```

```
        110.\n";
```

```
    return "";
```

```
}
```

```
function validateEmail(field){
```

```
    if (field == "")
```

```
        return "No Email was entered.\n";
```

```
    else if (field.indexOf('@') <= 0)
```

```
        return "Invalid @ Position";
```

```
    else if (! (field.indexOf(".")) > 0)
```

```
        return "Invalid position of .";
```

```
    else if (/^a-zA-Z0-9.@[ -]$/i.test(field))
```

```
        return "Invalid";
```

```
}
```

```
    if (field.charAt(field.length-1) != '.') && (field.charAt(field.length-1) != ',')
```

```
        document.getElementById("msg").innerHTML = "Please select age";
```

```
function validateMobileNo(field){
```

```
    if (field == "")
```

```
        return "Please fill mobile number";
```

```
    else if (isNaN(field))
```

```
        return "Enter only numeric value";
```

```
    else if (field < 10)
```

```
        return "Mobile number must be 10 digit";
```

```
    else if (field > 10)
```

```
        return "Mobile number must be 10 digit";
```

```
    else if ((field.charAt(0) != 9) && (field.charAt(0) != 8) && (field.charAt(0) != 7))
```

```
        return "Mobile number must start with 9
```

```
}
```

```
function validateRadioButton(field){
```

```
    var a = document.forms.Container;
```

```
    for (i=0; i < a.length; i++)
```

```
{
```

```
    if (a[i].checked == true)
```

```
        return true;
```

```
}
```

```
return "Please Select age";
```

```
}
```

```
</script>
```

```

function validatecheckbox()
{
    var checkbox = document.getElementById("agree");
    if (checkbox.checked == false)
        alert("Please agree to the terms and conditions");
    return false;
}

<label>
<input type="checkbox" id="agree" name="g" value="I agree to the terms and conditions" />
</label>

<button onclick="validate()">Submit</button>

function Radiobutton()
{
    var radios = document.getElementsByName("grades");
    var formvalid = false;
    for (var i = 0; i < radios.length; i++)
    {
        if (radios[i].checked)
        {
            formvalid = true;
            break;
        }
    }
}

```

- Ch #13
- ### Exercise Short Questions.
- Which tag is used to enclose javascript code?  
 Ans: The script tag is used to enclose javascript code  
 $\langle \text{script} \rangle$   
 $\text{//javacode}$   
 $\langle / \text{script} \rangle$
  - By default, to which part of document will javascript code output?  
 Ans: In which it resides.  
 If it is in the head, it will output in the head.  
 If it is in the body, it outputs to the body.
  - How can you include javascript code from another source in your document?  
 Ans: To include an external javascript file, we can use the script tag with the attribute src.

do by copying or pasting them.

⇒ The value for the src attribute should be the path to your javascript file.

⇒ This script tag should be included between the <head> tags in your HTML document.

Which javascript function is the equivalent of echo or print in PHP?

⇒ The "Console.log()" method in javascript is equivalent to the echo/print in PHP.

⇒ Console.log() allows to print text to the browser's console

⇒ `document.write()`, innerHTML attribute or `document.appendchild()` are used out to print/echo

How can you create a comment in javascript?

- // for single line

- /\* ... \*/ for multi-line

What is the javascript concatenation operator?

What javascript concatenation operator is +

Which keyword you can use within a javascript function to define a variable that has local scope?

- By preceding it with the var keyword upon first argument.

Give two cross-browsers methods to display the URI assigned to the link with an id of thislink;

⇒ `document.write(document.getElementById('thislink').href)`

⇒ `document.write(thislink.href)`

Which two javascript commands will make the browser load the previous page in its history array?

- `history.back()`

- `history.go(-1)`

What javascript command would you use to replace the current document with the main page at the oreilly.com website?

⇒ `document.location.href = 'http://oreilly.com'`

(Ch #15) :-

(1) Are javascript functions & variables names case-sensitive or insensitive?

⇒ Javascript variables and functions are case sensitive.

⇒ The variables Count, count, COUNT are all different.

How can you write a function that accepts and processes an unlimited number of parameters?

⇒ By Acces parameters through the argument array, which is a member of all the functions.

Name a way to write multiple values from a function?

⇒ To place them all inside an array and return the array.

When you're defining a class, what keyword do you use to refer to the current object?

⇒ "this" key word is used to refer to the current object.

Do all the methods of a class have to be defined within the class definition?

⇒ The methods of a class do not have to be defined within a class definition.

⇒ if the method is to be defined outside the constructor, the method name must be assigned to the this object within the class definition.

# What keyword is used to create an object?

=> New objects are created via new keyword.

# How can you make a property or method available to all objects in a class without replacing the property or method within the object?

=> By using prototype keyword to create a single instance, which is then passed by reference to all the objects in a class.

# How can you create a multi-dimensional array?

=> By placing all subarrays inside the main array.

# What syntax is used to create an associative array?  
value, with in the curly braces as  
the following:  
associarray =

{  
    "forename": "Paul",  
    "surname": "McCartney",  
    "group": "The Beatles"

# Write a statement to sort an array of numbers in descending numerical order?

numbers.sort(function(a,b){return b-a})

## Ch #16

1) What javascript method you can use to send a form for validation prior to submitting it?

=> By adding the javascript onsubmit attribute to the <form> tag.

=> Make sure that your function returns true, if the form is to be submitted, and false otherwise.

What javascript method is used to match a string against a regular expression?

⇒ By using the test method.

Write a regular expression to match any characters that are not in a word, as defined by regular syntax?

⇒ /[^\\w]/, /[^\\w]/, /[^a-zA-Z0-9],  
, and so on.

Write a regular expression to match either of the words for or fix?

/for|fix/

Write a regular expression to match any single word followed by any non-word characters?

⇒ /\w+\W/g

Write a regular expression to test whether the word for exist in the string The quick brown fox.

document.write( /for/.test ("The quick brown fox"))  
will output a php function to replace all occurrences of the word the in the The cow jumps over the moon with the word my?

\$s = preg\_replace("/the/i", "my", "The cow jumps over the moon");

What HTML attribute is used to precomplete form fields with a value?

A value attribute, which is placed within an input tag and takes the form value = "value".

Q: what is the purpose of javascript in web?

Q. How to display current date on paragraph using javascript?

⇒ By using the new Date() object

⇒ The date object supports numerous date methods like getFullYear(), getMonth(), etc.

Example:

```
const d = new Date();
```

Q. How can one include javascript in a web page?

⇒ By placing <script> & </script> tag.  
To add external javascript use the script tag with the attribute src: it specifies the path to your javascript file. Script tag should be included between the <head> tags.

Q. What is confirm box in javascript?

⇒ A confirm box is often used if you want the user to verify or accept something.

⇒ When a confirm box pops up, the user will have to click either "OK" or "Cancel" to proceed.

⇒ If the user clicks "OK" the box is true.

We used click "Cancel", the box  
is false.

### Syntax:

### confirm (message)

Example

(if test:

if (confirm ("Pssss a button")) == true  
text = "you pressed 'ok'."

else

text = "you cancelled"

⇒ confirm() is supported in all  
browsers

Q. what is the role of javascript  
in a web page?

⇒ The main purpose of HTML is to tell a  
browser how the document should appear.

⇒ Javascript brings HTML to life and makes  
web pages dynamic.

⇒ use javascript to change the contents of a  
web page after it has been rendered by a  
browser.

⇒ It is also used for advertising &  
entertainment purposes.

⇒ It programs the behavior of a web page.

identify the code:

```
document.write(<h1> welcome to  
javascript </h1>)
```

Q. what are the concatenation operators in javascript and PHP?

→ The concatenation operator in javascript is (+) that concatenates two or more strings.

→ The shorthand (+=) can also be used for concatenation in javascript.

⇒ The (+) operator is used in PHP for concatenation or also write (+=).

Q. which type of language is javascript

→ javascript is a lightweight interpreted programming language.

⇒ The web browser executes the javascript code in its original text form and runs the script from that.

⇒ It is a scripting language. it is a client side language.

to validate password?

```
<body>  
password <input type="password" id="Pw" value="">
```

```
<span id="msg"></span>
```

```
</body>  
function myfun()
```

```
{  
    document.  
    var a = getElementById("Pw").value;  
    if (a == "") {  
        document.getElementById("msg").innerHTML  
        = "Please fill password";  
    }  
}
```

# JavaScript Practice

## o Add HTML Tag in JavaScript:

```
<script>
```

```
document.write("Hello world <b></b>");
```

```
document.write("<i></i> Hello! <br/>");
```

```
</script>
```

## o Use of Variables:

```
<script>
```

```
var x = "Yahoo baba";
```

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

```
</script>
```

## Var vs Let vs Const

```
var n = 2;
```

```
let x = 2
```

```
const n = 2;
```

```
var n = 3;
```

```
let n = 3; //error
```

```
const n = 3; //error
```

```
n = 5;
```

```
x = 4
```

```
x = 3; //error
```

```
All possible
```

## Different type of data types:

```
var n = "Hello";
var n = 25;
var x = true;
Array - var x = ["HTML", "CSS", "JS"];
Object - var n = { first: "Jane", last: "Doe" };
```

### Example:

```
<script>
    var n = "Yahoo";
    document.write(n);
    document.write("<br>");
    document.write(typeof n);
</script>
```

## Addition of two numbers.

```
<script>
    var a = 80;
    var b = 100;
    var c = a+b;           // or document
    document.write(c);     .write(a+b);
</script>
```

## Google chrome Console:-

console.log() main purpose  
to display error or testing  
just not to display error.

```
console.log([1, 2, 3]);
console.log(2+x);
```

### (1) Alert Box:-

```

<script>
var a = 20;
var b = 30;
if (a > b)
{
    alert("a is greater");
}
else
{
    alert("b is greater");
}
</script>

```

### Example 2:

```

<script>
var a = 20;
alert("Value of A: " + a);
</script>

```

↑ used with events  
when anyone click website.

### (2) Confirm Box:-

```

<script>
var a = confirm("Do you like
our website?");
alert(a);
if (a)
    alert("Thanks");
else
    alert("Sorry");
</script>

```

### (3) Prompt dialog Box:

- it get some input from user.

### Example:

```

<script>
var x = prompt("Enter number");
document.write(x);
</script>

```

Q. Function that sum two variables.

```
<script>
```

```
my
```

```
function(a,b)
```

```
{
```

```
document.write(a+b);
```

```
}
```

```
my(8,2);
```

Q. Function with return statement.

```
<script>
```

```
function fullname(fname,lname)
```

```
{
```

```
var a = fname + " " + lname
```

```
return a;
```

```
}
```

```
var fn = fullname ("Himanshu")  
document.write(fn);
```

Q. Find even/odd numbers using loop?

```
<script>
```

```
for(var a=1;a<=10;a+1)
```

```
{
```

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

```
{
```

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

```
}
```

```
else
```

```
{
```

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

```
}
```

## How to create?

<script>

```
var arr = [10, 20, 30, 40];
```

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

```
document.write(arr[4]);
```

```
sum = 0;
```

```
for (var a = 0; a <= 4; a++)
```

```
{  
    document.write(arr[a] + "  
    sum = sum + arr[a];  
}
```

```
document.write("Sum is " + sum);
```

Method 2:- — Also called constructor

```
var a = new Array();
```

method

Example:

<script>

```
var arr = new Array(3)
```

```
arr[0] = 10;
```

```
arr[1] = "Harry";
```

```
arr[3] = "Gauri";
```

<script>

## Example 2: Input array values

<script>

```
var arr = new Array(3);
```

```
for (var g = 0; g < 3; g++)
```

```
{  
    arr[g] = prompt("Enter the value.");
```

```
}
```

</script>

## How to Create multi-dimensional arrays?

```
var a = [
```

```
    ["Harry", 18, "male", "B.com"],
```

```
    ["Sarah", 19, "female", "BCA"]
```

```
];
```

## How to delete element from array?

```
delete arr[1];
```

## How to Sort array?

<script>

```
var a = ["Hira", "Amin", "Noor"];
a.sort();
document.write(a);
</script>
```

## How to Reverse array?

a.reverse();

## Slice() & Splice()

↓

Used to make subarray from an existing array.

slice(start, end)

```
var a = [1, 2, 3, 5];
```

```
var b = a.slice(1, 3);
```

## Splice():-

To add value in array b/w elements.

optional

splice(index, howmany, "new value")  
Delete

## How to concatenate two arrays?

var a = [1, 2, 3];

var b = [3, 4, 5];

var c = a.concat(b);

document.write(c);

## Purpose of forEach() function

a. forEach(function)

function loop (value, index)

{

// statement

}

Javascript code to  
Count no of vowels.

function CountVowels(str){

// Convert the string to lowercase  
to handle both uppercase  
& lowercase case  
str = str.toLowerCase();

// Define an array of vowels

const vowels = ['a', 'e', 'i', 'o', 'u'];

let cv = 0;

for( let i=0 ; i < str.length ; i++ )

// check if the character is  
a vowel

if (vowels.includes(str[i]))

{

cv++;

}

return cv;

};

const str = prompt("Enter string :");

const nov = CountVowels(str);

console.log("Number of vowels :: ", nov);

Javascript code to Concat  
Q. the two strings

<script>

const str1 = prompt("Enter string 1");

const str2 = prompt("Enter and string 2");

// Concatenate

const result = str1 + " " + str2;

console.log(result);

// Using the concat()

const result2 = str1.concat(" ", str2);

console.log(result2);

</script>