→ Javascript is the world's mo[…]
programming language[for web-pages].

→ Javascript is the programming language of the web. It is used for create client side dynamic pages.

→ Javascript is easy to learn.

→ <!DOCTYPE html>
<html>
<body>
<h2> First Javascript </h2>
<button type= "button" onclick ="document.getElement ById ('demo').
innerHTML= Date ()">
Click me to display Date and Time </button>
<p id="demo" > </p>
</body>
</html>

→ It is object-based scripting language.

→ It is not a compiled language

→ It is translator [i.e Javascript translator embedded in browser is responsible for translating.

→ Although Java & Javascript not have any connection, but name was suggested, when Java was gaining popularity in the market.

My First Javascript

Click me to display Date and Time

→ <!DOCTYPE html>
<html>
<body>
<h2> What can Javascript Do? </h2>
<p id= "demo" > Javascript can change HTML content. </P>
<button type= "button" onclick= " document.get Element By Id ('demo'). innerHTML= "Hello Javascript ! ""> Click Me ! </button>
</body>
</html>

→ Javascript follows syntax of and structure of C-programming. Thus it is a structured programming language.

What can Javascript Do?

Hello Javascript

Applications:-

→ client-side validations

→ Displaying pop-up windows and dialog boxes

<!DOCTYPE html>
<html>
<body>
<h2) My First Javascript </h2>
<button type="button"
onclick="document.getElementById('demo').
innerHTML=Date()">
Click me to display Date and Time</button>

</body>
</html>

My First Javascript

Click me to display Date and Time

Fri Dec 29 2023 07:14:31

Javascript can change HTML attribute values:-

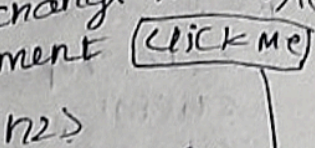
<!DOCTYPE html>
<html>
<body>
<h2) What can Javascript Do? </h2>
 Javascript can change HTML attribute values
 In this case Javascript changes the value of the src attribute of an image
<button onclick="document.getElementById('myImage').src='pic_bulbon.gif'">Turn on the Light </button>
<img id="myImage" src="pic_bulboff.gif"
style="width:100px">
<button onclick="document.getElementById('myImage').src='pic_bulboff.gif'">Turn off the Light </button>

Javascript Code:-
→ within body tag
→ within head tag
→ External Javascript.

Disadvantages of External Javascript:-
→ The stealer may download the coder's code using the url of the js file.
→ If two js files are dependent on one another, then a failure in one file, may affect the execution of other dependent file.
→ The web browser need to do additional http request.

<html>
<body>
<h2> What can Javascript do? </h2>
 Javascript can change the style of an HTML element.
<button type="button"
onclick="document.getElementById('demo').
style.fontSize='35px'"> Click Me! </button>
</body>
</html>

o/p:- Javascript can change of an HTML element



If we click on this font-size will change to 35px.

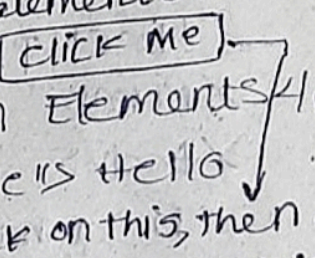
→ Javascript can show HTML elements:-

showing hidden HTML elements can also be done by changing the display style:-

<!DOCTYPE html>
<html>
<body>
<h2> What can Javascript do</h2>
 Javascript can show hidden Elements
 Hello Javascript!
<button type="button"
onclick="document.getElementById('demo').
style.display='block'"> Click Me! </button>
</body>
</html>

o/p:→ What can Javascript do?

Javascript can hide HTML elements.



If we click on this, then o/p be, it will'nt display anything.

The <script> tag:-

Javascript is inserted between <script> and </script>

<script>
document.getElementById("demo").innerHTML="My First Javascript";
</script>

function is a block of JavaScript code, that can be executed when called for.
→ For example, a function can be called when an event occurs, like when the user clicks a button.
→ JavaScript in <head> or <body>
In this example, a JavaScript function is placed in the <head> section of an HTML page. or in both.
→ The function is invoked when a button is clicked.

```
</script>
</head>
<body>
<h2> Demo Javascript in Head </h2>
<p id="demo"> A paragraph </p>
<button type="button"
onclick="myFunction()"> Try it
</button>
</body>
</html>
```

→ It is better to use script in body section.

→ document.write() function is used to display dynamic content through JavaScript.

JavScript in body:-

```
<!DOCTYPE in html>
<html>
<body>
<h2> Demo Javascript in Body</h2>
<p id="demo"> A paragraph</p>
<button type="button" onclick="myFuntion()">
try it </button>
<script>
function myFunction(){
document.getElementById("demo").innerHTML
="paragraph changed";
}
</script>
</body>
</html>
```

```
document.write("Rupa");
document.write("Devi");
```

o/p:- Rupa Devi

Demo Javascript in Body

A paragraph

→ Writing into an HTML element, using innerHTML.
→ Writing into the HTML output using document.write().
→ Writing into an alert box, using window.alert()
→ Writing into the browser console, using console.log() → this advanced one, than other three.

Using innerHTML:-

→ To access an HTML element, JavaScript can use the document.getElementById(id) method.
→ the id attribute defines the HTML element. The innerHTML property defines the HTML content:-

```
<p id="demo"></p>
<script>
document.getElementById("demo").innerHTML
        =5+6;
</script>
```

Using document.write():-

for testing purpose, it is convenient to use document.write():

```
<script>
document.write(5+6);
</script>
```

→ direct o/p it will get

→ Using document.write() after an HTML document is loaded, will delete all existing HTML.

for this direct it not get by clicking the button,

```
<button type="button" onclick="document.write(5+6)">Try it</button>
```

→ the document.write() method should only be used for testing → html will delete and all the

scope object. this means that variables, properties, and methods by default belong to window object.

```
<script>
    alert(5+6);
</script>
```

→ console.log()

typeof:-

```
let x=5;
let y="Hello";
let z=true;
console.log(typeof x); → number
console.log(typeof y); → string
console.log(typeof z); → b
```

For debugging purposes, you can call the console.log() method in the browser to display data.

```
<script> console.log(5+6);
</script>
```

Javascript print:-

→ Javascript does not have any object or print methods.

→ You cannot access output devices from Javascript.

→ The only exception is that you can call the window.print() method in the browser to print the content of the current window.

```
<button onclick="window.print()">print this page</button>
```

Javascript programs:-

→ A computer program is a list of instructions to be executed by a computer.

→ In a programming language these programming instructions are called statements.

→ Javascript program has a list of programming statements.

→ Javascript programs composed of

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

→ Most Javascript programs contain many Javascript statements.
→ semicolons separate Javascript elements.

```
let a, b, c, C;
a=5;
b=6;
C=a+b;
```

Javascript white space:-

Javascript ignores multiple spaces. You can add white space to your script to make it more readable.

```
let person= "Hege";
let person= "Hege";
```

Javascript Line Length and Line Breaks:-

For best readability, programmers often like to avoid code lines longer than 80-characters.
→ If a Javascript statement does not fit the one line, the best place to break it is after an operator.

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

Javascript code Blocks:-

Javascript statements can be grouped together in code blocks, inside curly brackets {...}. The purpose of code blocks is to define statements to be executed together.

```
function myFunction() {
  document.getElementById("demo").
  innerHTML="Hello Dolly";
```

Javascript keywords:-

var → Declares a variable
let → Declare a block variable
const → Declares a block constant
if → marks a block of statements to be executed on a condition.
switch → marks a block of statements to be executed in different cases.
for → marks a block of statements to be executed in a loop
function → Declares a function
returns → Exists a function
try → Implements error handling to a block statements.

→ Javascript keywords are reserved words. Reserved words cannot be used as names for variables.

Javascript syntax:-

Set of rules, how javascript programs are constructed

var x;
let y;

→ to declare javascript global variables function, you need to use window object.

window.value=90;

function m() {
window.value=100;
}
function n() {
alert(window.value);
}

Javascript values:-

2 types of values:-
→ Fixed [literals]
→ variable [variables].

Javascript Literals:-

<script>
document.getElementById("demo").innerHTML=10.50;
</script>

o/p:-10.5

when you declare a variable outside the function, it is added in window object.

var value=50;