

The **if** and **else** Statement

Syntax

```
if(condition)
{
    // block of code to be executed if the condition is true
}
else
{
    // block of code to be executed if the condition is false
}
```

The diagram illustrates the flow of execution for the if-else statement. The condition is evaluated. If it is true (T), the code block under the if-block is executed. If it is false (F), the code block under the else-block is executed.

```
1 <html>
2 <body>
3
4 <p>Example of if and else statement</p>
5
6 <button onclick="myFunction()">Check</button>
7
8 <h2 id="demo"></h2>
9
10 <script>
11 function myFunction() {
12     var a=2;
13
14     if (a % 2 == 0)
15     {
16         getdata = "Even Number";
17     } else
18     {
19         getdata = "Odd Number";
20     }
21 }
```

```
7 <h2 id="demo"></h2>
8
9
10<script>
11function myFunction() {
12    var a=2;
13
14    if (a % 2 == 0)
15    {
16        getdata = "Even Number";
17    } else
18    {
19        getdata = "Odd Number";
20    }
21    document.getElementById("demo").innerHTML = getdata;
22}
23</script>
24
25</body>
26</html>
```

Example of if and else statement

Check

Even Number



The **else if** Statement (Nested If)

Syntax

```
if (condition1)
{
    // block of code to be executed if condition1 is true
}
else if (condition2)
{
    // block of code to be executed if the condition1 is false and condition2 is true
}
else
{
    // block of code to be executed if the condition1 is false and condition2 is false
}
```

The diagram illustrates the control flow of the code. It starts with a 'T' above the first brace, indicating that if condition1 is true, the code block below it will be executed. If condition1 is false, it proceeds to the 'else if' section, indicated by a 'T' above the second brace. If condition2 is true, the code block below it will be executed. If both condition1 and condition2 are false, the code proceeds to the 'else' section, indicated by an 'F' above the third brace.

```
1 <html>
2 <body>
3
4 <p>Example of nested if statement</p>
5
6 <button onclick="myFunction()">Get Percentage</button>
7 I
8 <h2 id="demo"></h2>
9
10 <script>
11 var phy=76;
12 var che=88;
13 var maths=89;
14 var eng=66;
15 var hindi=70;
16 var total;
17 var percentage;
18 total=phy+che+maths+eng+hindi;
19 percentage=total/500*100;
20
```

```
16 var total;
17 var percentage;
18 total=phy+che+maths+eng+hindi;
19 percentage=total/500*100;
20
21 document.write("Total Number Is : " +total);
22 document.write("<br>");
23 document.write("Percentage Is : " +percentage);
24
25 function myFunction() {
26     if (percentage>=80)
27     {
28         getdata = "A Grade";
29     }
30     else if (percentage>=60)
31     {
32         getdata = "B Grade";
33     }
34     else if (percentage>=40)
35     {
36         getdata = "C Grade";
37     }
38 }
```

```
else if (percentage>=60)
{
    getdata = "B Grade";
}
else if (percentage>=40)
{
    getdata = "C Grade";
}
else
{
    getdata = "F Grade"
}
document.getElementById("demo").innerHTML = getdata;

```

</script>

</body>

</html>

Example of nested if statement

[Get Grade](#)

B Grade

Total Number Is : 389
Percentage Is : 77.8

The screenshot shows a Notepad++ window titled "C:\Users\Abhishek\Desktop\JavaScript\Control Statements\Nested_If.html - Notepad++". The code is a JavaScript function named "myFunction" that takes a percentage as input and outputs a grade based on nested if conditions. The code is as follows:

```
22 | document.write("<br>");  
23 | document.write("Percentage Is : " +percentage);  
24 |  
25 | function myFunction() {  
26 |     if (percentage>=80)  
27 |     {  
28 |         getdata = "A Grade";  
29 |     }  
30 |     else if (percentage>=60)  
31 |     {  
32 |         getdata = "B Grade";  
33 |     }  
34 |     else if (percentage>=40)  
35 |     {  
36 |         getdata = "C Grade";
```

A yellow circle highlights the opening brace of the first if block at line 27. The code is enclosed in a scrollable window.

JavaScript Switch Statement

Syntax

```
switch(expression)
{
    case x:
        //code which you want to execute
        break;
    case y:
        //code which you want to execute
        break;
    default:
        //code which you want to execute
}
```

```
1  <html>
2  <body>
3  <script>
4  var choice;
5  var no,output;
6  document.write("Press 1 for square<br>");
7  document.write("Press 2 for cube<br><br>");
8  choice=1;
9  document.write("Enter any number<br><br>");
10 no=3;
11 switch (choice) {
12   case 1:
13     output=no*no;
14     document.write("Square is : "+output);
15     break;
16   case 2:
17     output=no*no*no;
18     document.write("Cube is : "+output);
19     break;
```

```
8 choice=1;
9 document.write("Enter any number<br><br>");
10 no=3;
11 switch (choice) {
12     case 1:
13         output=no*no;
14         document.write("Square is : "+output);
15         break;
16     case 2:
17         output=no*no*no;
18         document.write("Cube is : "+output);
19         break;
20     default :
21         document.write("Wrong Input");
22 }
23 </script>
24 </body>
25 </html>
```

Press 1 for square

Press 2 for cube

Enter any number

Square is : 9

JavaScript Loops

The For Loop

For (initialization; condition; updation)
{
Body of the loop or statements;
}

```
1  <html>
2  <body>
3
4  <h1>JavaScript For Loop</h1>
5
6  <script>
7  var i;
8  for (i = 0; i <= 10; i++)
9  {
10    document.write("The number is " + i + "<br>");
11  }
12 </script>
13
14 </body>
15 </html>
```

JavaScript For Loop

The number is 0

The number is 1

The number is 2

The number is 3

The number is 4

The number is 5

The number is 6

The number is 7

The number is 8

The number is 9

The number is 10

The **While** Loop

Syntax

```
while (condition){  
    // write your code here  
}
```

The **Do While** Loop

Syntax

```
do{  
    // write your code here  
}  
while(condition);
```

```
1 <html>
2   <body>
3
4     <h2>JavaScript While Loop</h2>
5
6   <script>
7     var i=1;
8     while (i <= 10) {
9       document.write("Value : "+i);
10      i++;
11      document.write("<br>");
12    }
13  </script>
14
15  </body>
16 </html>
```

JavaScript While Loop

Value : 1

Value : 2

Value : 3

Value : 4

Value : 5

Value : 6

Value : 7

Value : 8

Value : 9

Value : 10

```
1 1<html>
2 2<body>
3
4 <h2>JavaScript Do While Loop</h2>
5
6 <script>
7 var no=10;
8 do {
9     document.write("<br>Number is |: "+no);
10    no--;
11 }while(no!=0);
12 </script>
13
14 </body>
15 </html>
```

JavaScript Do While Loop

Number is : 10
Number is : 9
Number is : 8
Number is : 7
Number is : 6
Number is : 5
Number is : 4
Number is : 3
Number is : 2
Number is : 1



```
1 <html>
2   <body>
3
4     <h1>JavaScript Strings Example 2</h1>
5
6     <p id="myID"></p>
7
8     <script>
9       document.getElementById("myID").innerHTML = "Learn " +
0       "JavaScript";
1     </script>
2
3   </body>
4 </html>
```

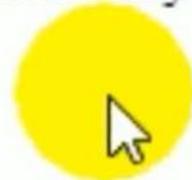
JavaScript Strings Example 2

Learn JavaScript

```
1 <html>
2 <body>
3
4 <h1>JavaScript Strings Example 3</h1>
5
6 <p id="myID"></p>
7
8 <script>
9 var s = 'It's okay';
10 document.getElementById("myID").innerHTML = s;
11 </script>
12
13 </body>
14 </html>
```

JavaScript Strings Example 3

It's okay



```
1 <html>
2 <body>
3
4 <h1>JavaScript Strings Example 4</h1> I
5 <h2>JavaScript Strings Methods</h2>
6
7 <p id="myID"></p>
8
9 <script>
10 var s = "JavaScript In Hindi By Micro Solution";
11
12 document.getElementById("myID").innerHTML = s.length;
13 |
```

```
<p id="myID"></p>

<script>
var s = "JavaScript In Hindi By Micro Solution";
I

//document.getElementById("myID").innerHTML = s.length;

var get = s.indexOf("Hindi");
document.getElementById("myID").innerHTML = get;

/*var get = s.search("Micro");
document.getElementById("myID").innerHTML = get;*/

/*var get = s.slice(10,18);
document.getElementById("myID").innerHTML = get;*/
```

JavaScript Strings Example 4

JavaScript Strings Methods

14

```
7 <p id="myID"></p>
8
9 <script>
10 var s = "JavaScript In Hindi By Micro Solution";
11
12 //document.getElementById("myID").innerHTML = s.length;
13
14 var get = s.indexOf("Hindi", 15);j
15 document.getElementById("myID").innerHTML = get;
16
17 /*var get = s.search("Micro");
18 document.getElementById("myID").innerHTML = get;*/
19
20 /*var get = s.slice(10,18);
21 document.getElementById("myID").innerHTML = get;*/
22
```

JavaScript Strings Example 4

JavaScript Strings Methods

-1

```
10 var s = "JavaScript In Hindi By Micro Solution";
11
12 //document.getElementById("myID").innerHTML = s.length;
13
14 /*var get = s.indexOf("Hindi");
15 document.getElementById("myID").innerHTML = get;*/
16
17 var get = s.search("Micro");
18 document.getElementById("myID").innerHTML = get;
19
20 /*var get = s.slice(10,18);
21 document.getElementById("myID").innerHTML = get;*/
22
23 /*var get = s.slice(19);
24 document.getElementById("myID").innerHTML = get;*/
```

JavaScript Strings Example 4

JavaScript Strings Methods

23

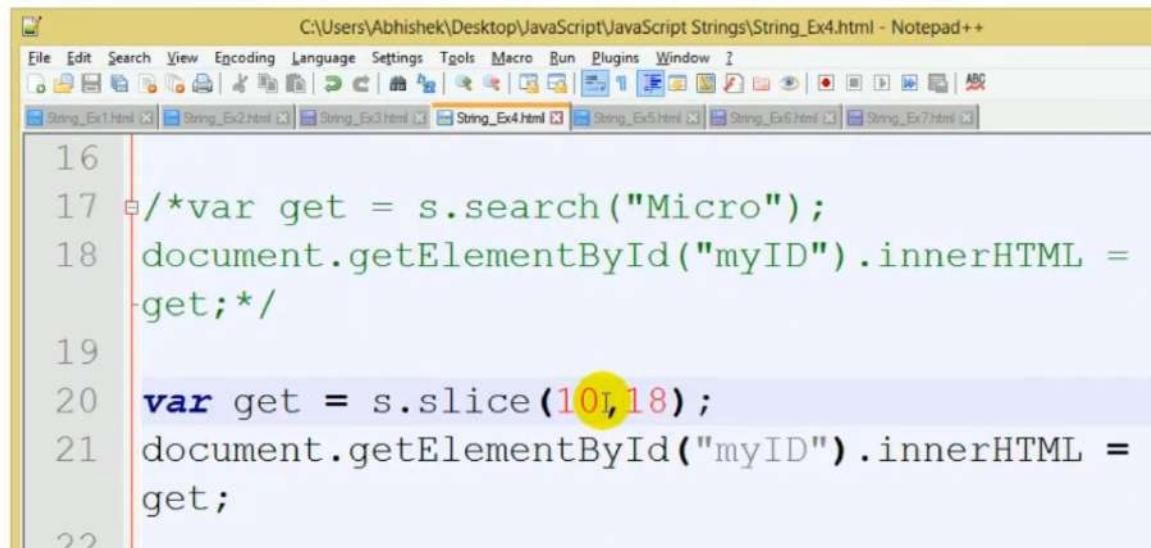


```
13
14 /*var get = s.indexOf("Hindi");
15 document.getElementById("myID").innerHTML = get;*/
16
17 /*var get = s.search("Micro");
18 document.getElementById("myID").innerHTML = get;*/
19
20 var get = s.slice(10,18);
21 document.getElementById("myID").innerHTML = get;
22
23 /*var get = s.slice(19);
24 document.getElementById("myID").innerHTML = get;*/
25
26 </script>
27
28 </body>
```

JavaScript Strings Example 4

JavaScript Strings Methods

In Hind



The screenshot shows a Notepad++ window with the title bar "C:\Users\Abhishek\Desktop\JavaScript\JavaScript Strings\String_Ex4.html - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, Window, and Help. Below the menu is a toolbar with various icons. The main editor area contains the following JavaScript code:

```
16
17 /*var get = s.search("Micro");
18 document.getElementById("myID").innerHTML =
19 -get;*/
20 var get = s.slice(10,18);
21 document.getElementById("myID").innerHTML =
22 get;
```

```
15 -document.getElementById("myID").innerHTML = get;/*
16
17 /*var get = s.search("Micro");
18 -document.getElementById("myID").innerHTML = get;/*
19
20 /*var get = s.slice(10,20);
21 -document.getElementById("myID").innerHTML = get;/*
22
23 var get = s.slice(19);
24 document.getElementById("myID").innerHTML = get;
25
26 </script>
27
28 </body>
29 </html>
30
```

JavaScript Strings Example 4

JavaScript Strings Methods

By Micro Solution



```
1  <html>
2
3  <body>
4
5  <h1>JavaScript Strings Example 5</h1>
6  <h2>JavaScript Strings Methods</h2>
7
8  <p>I am replacing "Microsolution" with "Abhishek" in the
example below:</p>
9
10 <button onclick="myFunction()">Click On This Button</button>
11
12 <p id="myID">JavaScript By Microsolution</p>
13
14 <script>
15  function myFunction() {
```

```
10 <button onclick="myFunction()">Click On This Button</button>
11
12 <p id="myID">JavaScript By Microsolution</p>
13
14 <script>
15 function myFunction() {
16     var s = document.getElementById("myID").innerHTML;
17     var get = s.replace("Microsolution", "Abhishek");
18     document.getElementById("myID").innerHTML = get;
19 }
20 </script>
21
22 </body>
23 </html>
```

JavaScript Strings Example 5

JavaScript Strings Methods

I am replacing "Microsolution" with "Abhishek" in the example below:

Click On This Button

JavaScript By Microsolution



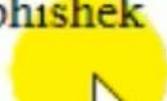
JavaScript Strings Example 5

JavaScript Strings Methods

I am replacing "Microsolution" with "Abhishek" in the example below:

Click On This Button

JavaScript By Abhishek



```
1 <html>
2
3 <body>
4
5 <h1>JavaScript Strings Example 7</h1>
6 <h2>The String concat() Method</h2>
7
8 <button onclick="myFunction()">Click On This Button</button>
9
10 <p id="myID"></p>
11
12 <script>
13
14 function myFunction()
15 {
16     var a1 = "Welcome To ";
```

```
10 <p id="myID"></p>
11
12 <script>
13
14 function myFunction()
15 {
16     var a1 = "Welcome To ";
17     var a2 = "By Micro Solution";
18     var a3 = a1.concat("JavaScript ",a2);
19     document.getElementById("myID").innerHTML = a3;
20 }
21
22 </script>
23
24 </body>
25 </html>
```

JavaScript Strings Example 7

The String concat() Method

Click On This Button

Welcome To JavaScript By Micro Solution



Different ways to print in JavaScript

Using document.write()

Using innerHTML

Using alert()

Using console.log()

Using console.warn()

Using console.error()

```
<script>
    // console.log(2+8);
    // console.log("Welcome to javascript");
    console.warn("You can write any Warning message");
    console.error("You can write any Error message");
</script>
```

Operators in JavaScript

Arithmetic Operators (+, -, *, /, %, **, ++, --)

Assignment Operators (=) or +=, -=, *=, /=

Comparison Operators (>, <, >=, <=, ==, !=, ?)

Logical Operators (&&, ||, !)

Data Types in JavaScript

Numbers, Strings, Booleans, Array, Objects & More.

JavaScript PDF.pdf

File | C:/Users/Abhishek/Downloads/JavaScript%20PDF.pdf

Draw Highlight Erase

```
// Array datatype

let student_name = ["Raj", "Kamal", "Rohan", "Sonu"];
console.log(student_name[2]);

// Object datatype

let person = {
  firstName : "Raj",
  lastName : "Singh",
  age : 40,
```



```
// Array datatype
```

```
let student_name = ["Raj", "Kamal", "Rohan", "Sonu"];  
console.log(student_name[2]);
```

```
// Object datatype
```

```
let person = {  
    firstName : "Raj",  
    lastName : "Singh",  
    age : 40,  
    mobile : "1001001001"  
};
```



```
console.log("Person name is " + person.firstName + " & " +  
"Mobile no. is " + person.mobile);
```

Conditional Statements

If you want to perform different actions based on different conditions then you can use conditional statements.

if, else, else if and switch statements

Functions in JavaScript

A function is a block of code that is designed to perform a particular task.

Built-in functions - `log()`, `write()`, `alert()`, `pow()`, `date()`

User defined functions - These are created by us.

```
<script>
    //Built-in functions
    let a = Date();
    let b = Math.pow(2, 3);
    console.log(a);
    console.log(b);
```

JavaScript Events

Here is a list of some common HTML events:

Event

onclick

onmouseover

onmouseout

onkeydown

onload



Description

The user clicks an HTML element

The user moves the mouse over an HTML element

The user moves the mouse away from an HTML element

The user presses a keyboard key

The browser has finished loading the page

127.0.0.1:5500 says

Welcome to JavaScript

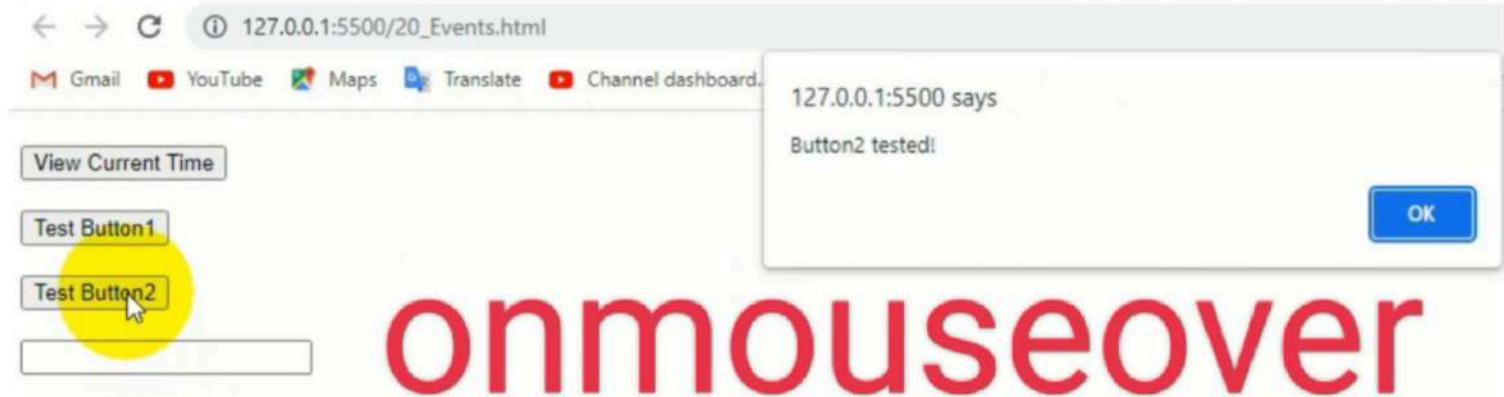


onload

OK

```
</head>
<body onload="test4()">
```

```
<button onmouseover="test2()">Test Button2</button>
<br><br>
```



Gmail YouTube Maps Translate Channel dashboard

[View Current Time](#)

[Test Button1](#)

[Test Button2](#)

127.0.0.1:5500 says

You pressed a key inside the input field

OK

onkeydown

```
<br><br>
<input type="text" onkeydown="test3()"></input>
```

```
//Template Literals
let price = 10;
let tax = 0.30;

let total = `Total bill is: ${price + tax}`;
console.log(total);
```

```
// let programming = ["Java", "C", "C++", "Python"];
// console.log(programming);

// Creating Array Method 2 (by using new keyword)
// const students = new Array("Sunil", "Raj", "Mark");
// console.log(students);

// You can also access full array by using ID.
let softwares = ["VLC", "Notepad", "VSCode"];
document.getElementById("demo").innerHTML = softwares;

// Accessing Array Elements
// let student = students[1];
// console.log(student);
```

```
29
30 // Changing an Array Element
31 // students[1] = "Komal";
32 // console.log(students);
33
34 // Arrays are special kinds of objects.
35 // Because of this, you can store different types of values in them
36 const employee = {firstName:"Suresh", lastName:"Singh", age:42};
37 document.getElementById("demo").innerHTML = employee.firstName;
38
39 // Length Property
40 // let programming = ["Java", "C", "C++", "Python"];
41 // let length = programming.length;
42 // console.log(length);
43
```

```
// Access array elements by using loop
let students = ["Sunil", "Raj", "Mark"];
let len = students.length;

for (let i = 0; i < len; i++) {
  console.log(students[i]);
}
```

```
// 1. Converting Arrays to Strings
// let students = ["Sunil", "Raj", "Mark"];
// let myString = students.toString();
// console.log(myString);

// 2. Converting Arrays to Strings (Join())
let students = ["Sunil", "Raj", "Mark"];
let myString = students.join("_");
console.log(myString);
```

```
// 3. pop() Method - removes the last element from an array
let softwares = ["VLC", "Notepad", "VSCode"];
document.getElementById("demo1").innerHTML = softwares;
softwares.pop();
document.getElementById("demo2").innerHTML = softwares;

// 4. push() Method - adds a new element to an array (but a
// let softwares = ["VLC", "Notepad", "VSCode"];
// document.getElementById("demo1").innerHTML = softwares;
// softwares.push("Photoshop");
// document.getElementById("demo2").innerHTML = softwares;
```

```
// 5. shift() Method - removes the first array element and
// "shifts" all other elements to a lower index.
let softwares: string[]
let softwares = ["VLC", "Notepad", "VSCode"];
document.getElementById("demo1").innerHTML = softwares;
softwares.shift();
document.getElementById("demo2").innerHTML = softwares;

// 6. unshift() Method - adds a new element to an array at
// let softwares = ["VLC", "Notepad", "VSCode"];
// document.getElementById("demo1").innerHTML = softwares;
// softwares.unshift("Photoshop","Tally");
// document.getElementById("demo2").innerHTML = softwares;
```

```
// 7. concat() Method - creates a new array by concatenation
```

```
let students = ["Sunil", "Raj", "Mark"];
let marks = [300, 400, 350];
let data = marks.concat(students);
console.log(data);
```

```
script>
```

JavaScript Date Objects

Date objects are created with the new Date()
constructor.



Example

```
<script>  
    let a = new Date();  
    console.log(a);  
</script>
```

Date Methods

you can use get and set the year, month, day, hours, minutes, seconds, and milliseconds of date object, using either local time or UTC (universal, or GMT) time.

Example

```
<center><div id = "clock" onload="currentTime()"></div></center>

<script>
    function currentTime() {
        let date = new Date();
        let hour = date.getHours();
        let min = date.getMinutes();
        let sec = date.getSeconds();
        let session = "AM";
        I

        if(hour > 12){
            session = "PM";
        }
    }

```

```
hour = (hour < 10) ? "0" + hour : hour;
min = (min < 10) ? "0" + min : min;
sec = (sec < 10) ? "0" + sec : sec ;

let time = hour + ":" + min + ":" + sec + " " + session;

document.getElementById("clock").innerText = time;
let t = setTimeout(function(){ currentTime() }, 1000);

currentTime();

```

```
<style>
#clock {
    background-color: lightgrey;
    width: 300px;
    border: 5px solid #rgb(0, 83, 0);
    padding: 15px;
    margin: 15px;
    font-size: 50px;
    font-family: Arial, Helvetica, sans-serif;
}
</style>
```

The HTML DOM (Document Object Model)

When you open your web page in a browser, the browser creates a document object model of the page.

With the object model, JavaScript gets all the power it needs to create dynamic HTML.

- You can access HTML elements.
- You can show your HTML dynamically.
- By using JavaScript you can change all the HTML elements & HTML attributes in the page. You can also change CSS styles in the page.
- By using JavaScript you can remove existing HTML elements and attributes.
- By using JavaScript you can add new HTML elements and attributes.
- By using JavaScript you can react to all existing HTML events in the page.

```
<title>HTML DOM EXAMPLE 1</title>
</head>
<body>
    <h1 id = "demo"></h1>

    <script>
        document.getElementById("demo").innerHTML="Welcome to JavaScript"
    </script>
```

Welcome to JavaScript



```
Elements Console Sources Network Performance
top Filter Default levels No iss
> document
<document>
> document.getElementById('demo').style.border = '3px solid blue';
<3px solid blue>
> document.getElementById('demo').style.background = 'blue';
<blue>
> document.getElementById('demo').style.background = 'yellow';
<yellow>
> |
```

```
<body>
  <h1 id = "demo"></h1>

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

  <script src="js/DOM_methods.js"></script>
</body>
</html>
```

html element change to dom in java

```
1 let eleId = document.getElementById("demo");
2
3 eleId.innerHTML = "Welcome to DOM";
  I
```

output

Welcome to DOM

```
</head>
<style> html code before
.c1 {
    background: blueviolet;
    width: 300px;
    border: 5px solid green;
    padding: 5px;
    margin: 5px;
}
</style>
```

```
<body>
    <h1 id = "demo"></h1>
    <div class="c1">
        <h1>This is heading.</h1>
    </div>

    <div class="c1">
        <p>This is paragraph.</p>
    </div>
```

Welcome to DOM

This is heading.

This is paragraph.

```
ML_DOM_EXAMPLE_2.html JS DOM_Methods.js •  
DOM_Methods.js > ...  
1 let eleId = document.getElementById("demo");  
2  
3 eleId.innerHTML = "Welcome to DOM";  
4  
5 eleId.style.background = "yellow";  
6  
7 let eleClass = document.getElementsByClassName("demo");  
8  
9 eleClass[0].style.background = "yellow";  
10  
11 eleClass[1].style.background = "yellow"; I
```

javascript code after changing in HTML CODE

Welcome to DOM

This is heading.

This is paragraph.

class add in javascript

```
5_HTML_DOM_EXAMPLE_2.html • JS DOM_Methods.js •  
JS DOM_Methods.js > ...  
3 // eleId.innerHTML = "Welcome to DOM";  
4  
5 // eleId.style.background = "yellow";  
6  
7 // let eleClass = document.getElementsByClassName("c1");  
8  
9 // eleClass[0].style.background = "yellow";  
10  
11 // eleClass[0].style.border = "10px solid black";  
12  
13 let eleClass = document.getElementsByClassName("c1");  
14 eleClass[0].classList.add("bgchange");
```



Welcome to JavaScript

This is heading.

This is paragraph.

default output

```
HTML_DOM_EXAMPLE_2.html   JS DOM_Methods.js X
DOM_Methods.js > ...
6
7 // let eleClass = document.getElementsByClassName("c1");
8
9 // eleClass[0].style.background = "yellow";
10
11 // eleClass[0].style.border = "10px solid black";
12
13 let eleClass = document.getElementsByClassName("c1");
14 eleClass[0].classList.add("bgchange");
15
16 eleClass[0].classList.remove("bgchange");|
```

Welcome to JavaScript

This is heading.

This is paragraph.

```
dit Selection View Go Run Terminal Help • DOM_Methods.js - JavaScript - Visual Studio Code  
6_HTML_DOM_EXAMPLE_2.html JS DOM_Methods.js ●  
JS DOM_Methodsjs > ...  
15  
16 // eleClass[0].classList.remove("bgchange");  
17  
18 let tg = document.getElementsByTagName("div");  
19  
20 console.log(tg);  
21  
22 makeElement = document.createElement("h1");  
23 makeElement.innerText="My New Heading";  
24  
25 tg[0].appendChild(makeElement);
```

← → C ① 127.0.0.1:5500/26_HTML_DOM_EXAMPLE_2.html

YouTube Translate Channel dashboard... Login - Hostinger I... Home - Canva Speedtest by Ookla... WhatsApp (206) JavaScript Tutorials Time Table

Welcome to JavaScript

This is heading.

My New Heading

This is paragraph.

Elements Console Sources

top Filter

HTMLCollection(2) [div.c1, div.c1]

Question for you

Q.8 You have to create an HTML DOM, in this DOM you have to show your name and favorite programmin language name now you have to apply text color, text size and border on it by using JavaScript.

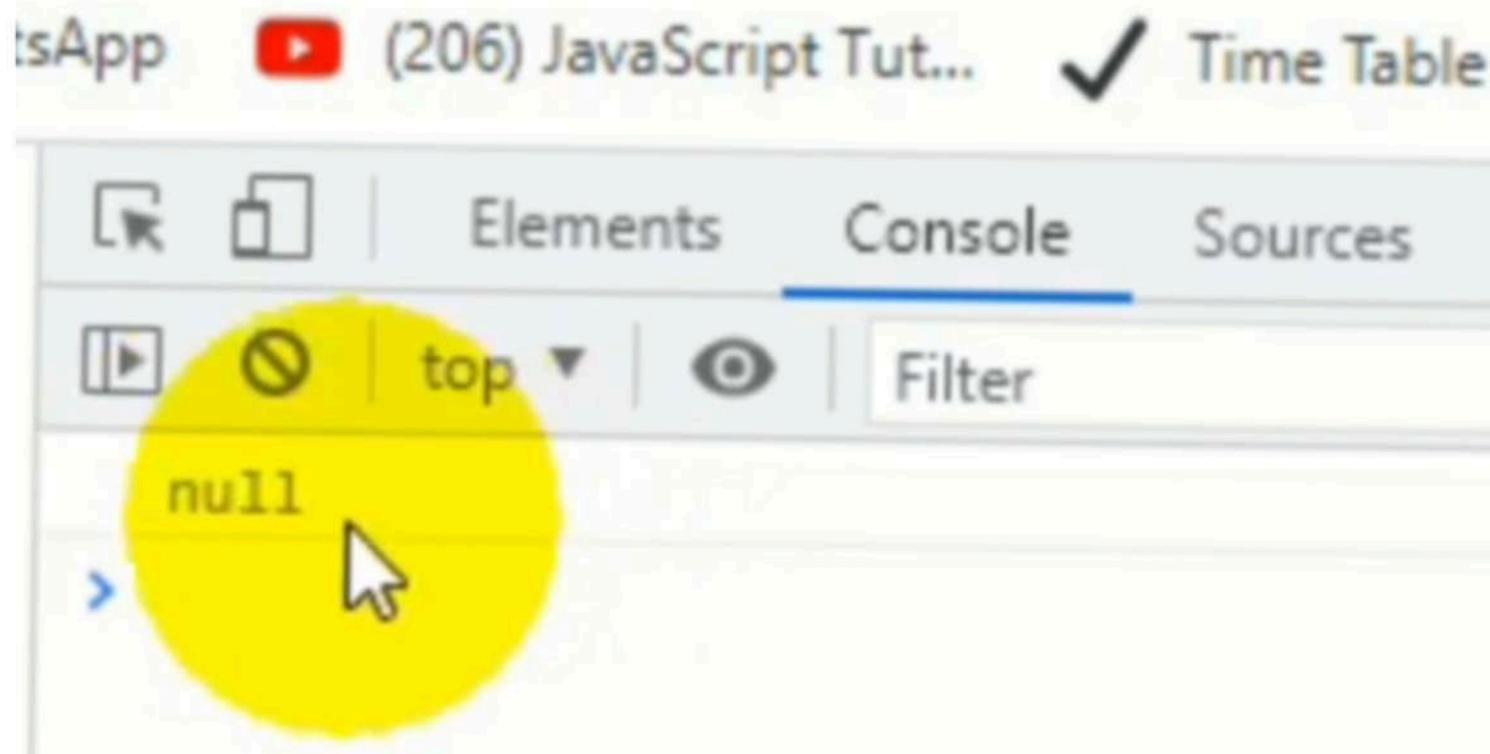
JavaScript Regular Expressions (RegExp)

In JavaScript, a Regular Expression (RegEx) is an object that describes a sequence of characters used for defining a search pattern.

RegEx can be used for text search and text replace operations.

Syntax : /pattern/modifiers;

```
<body>
  <h1 id = "demo"></h1>
  <script>
    // Pattern Matching
    let pattern = /[a-z]/i;
    let pass = "12345";
    console.log(pattern.exec(pass));
```

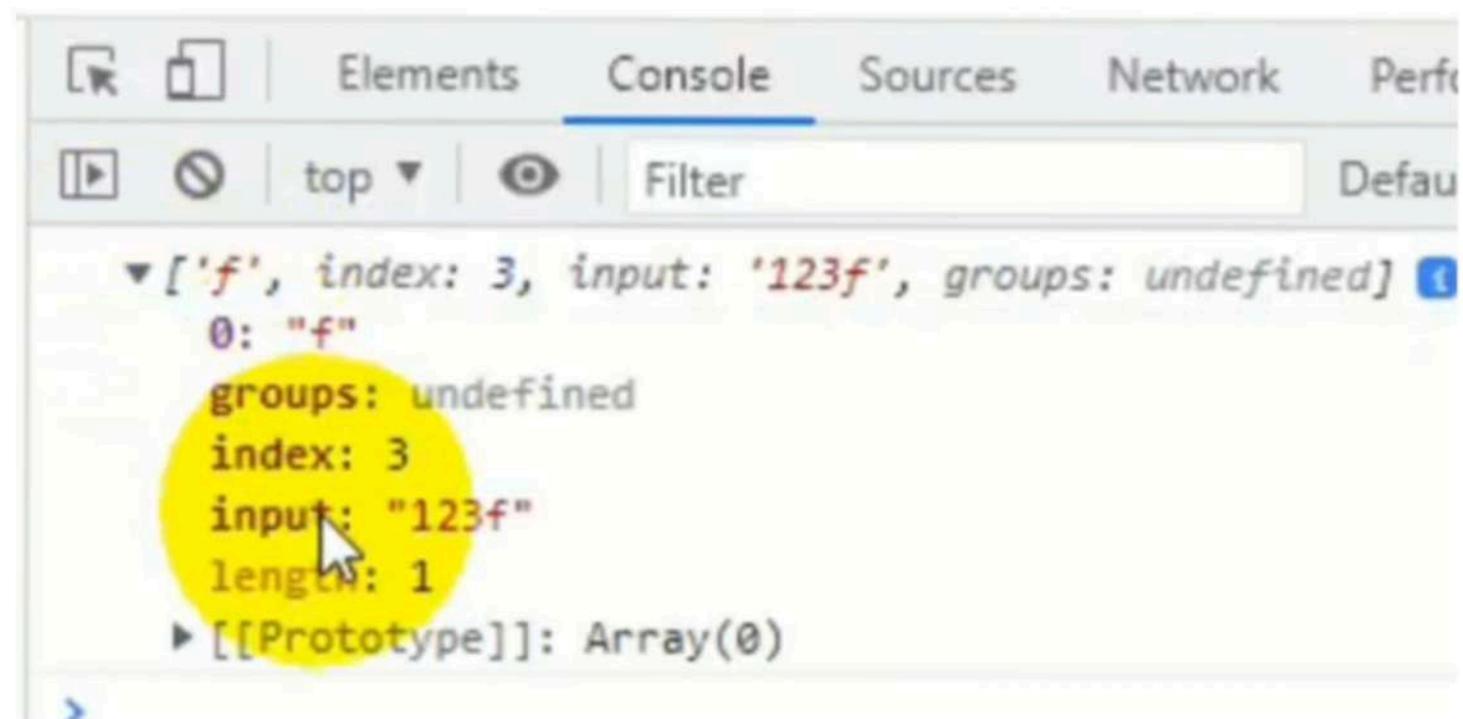


```
8 </head>
9 <body>
10 <h1 id = "demo"></h1>
11 <script>
12 // Pattern Matching
13 let pattern = /[a-z]/i;
14 let pass = "f";
15 console.log(pattern.exec(pass));
16
```

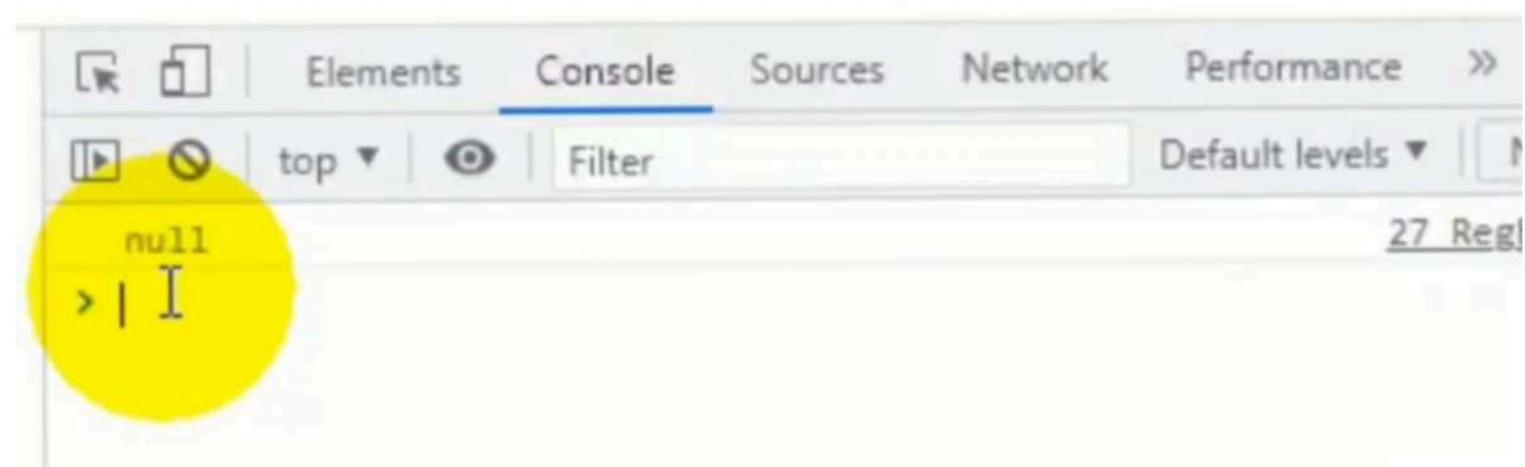
The screenshot shows the developer tools interface with the 'Console' tab selected. The console output displays the result of running a regular expression pattern against the string 'f'. A yellow circle highlights the 'groups' property of the returned object, which is shown as 'undefined'.

```
▼ ['f', index: 0, input: 'f', groups: undefined]
  0: "f"
  groups: undefined
  index: 0
  input: "f"
  length: 1
▶ [[Prototype]]: Array(0)
```

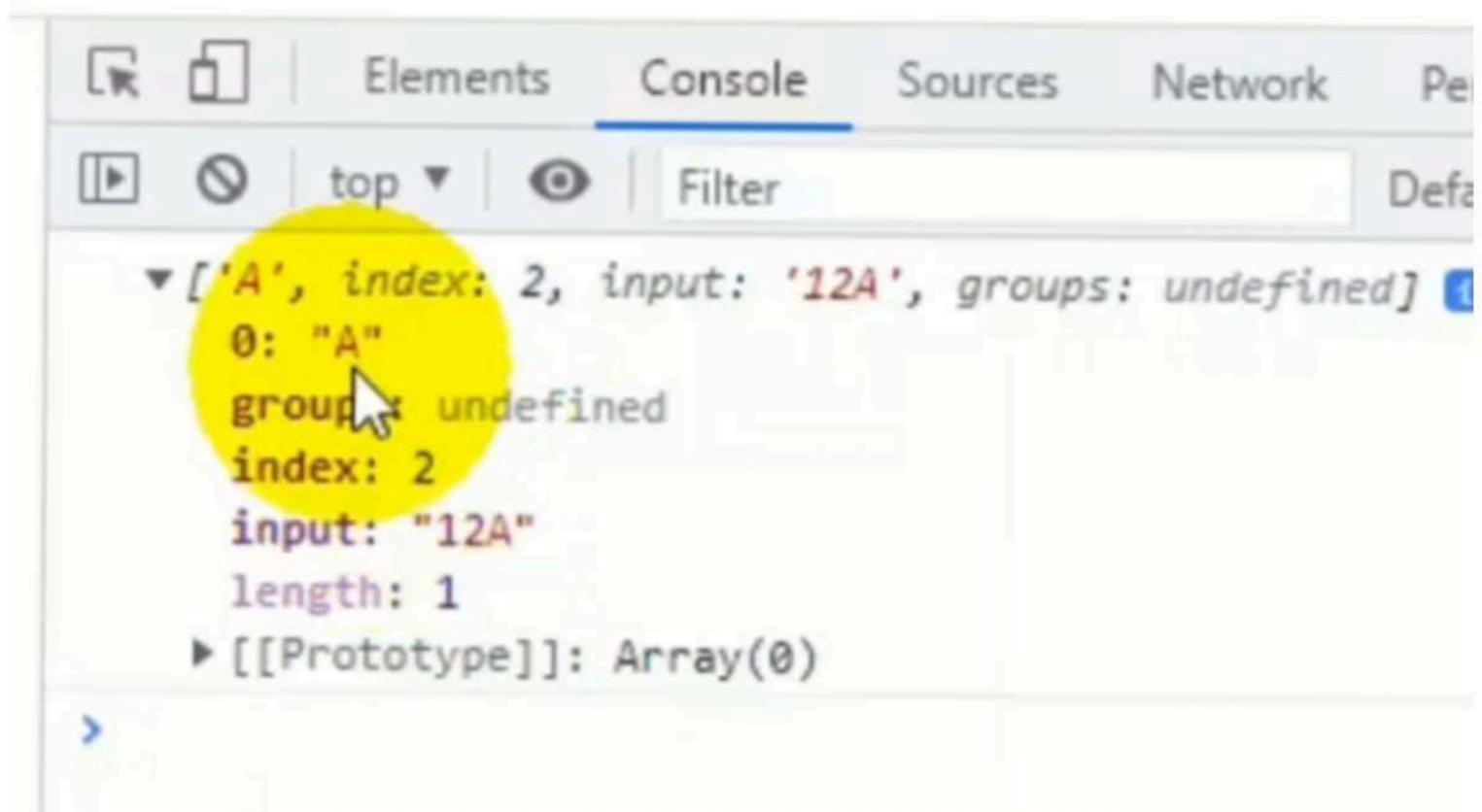
```
/head>
body>
<h1 id = "demo"></h1>
<script>
    // Pattern Matching
    let pattern = /[a-z]/i;
    let pass = "123f";
    console.log(pattern.exec(pass));
```



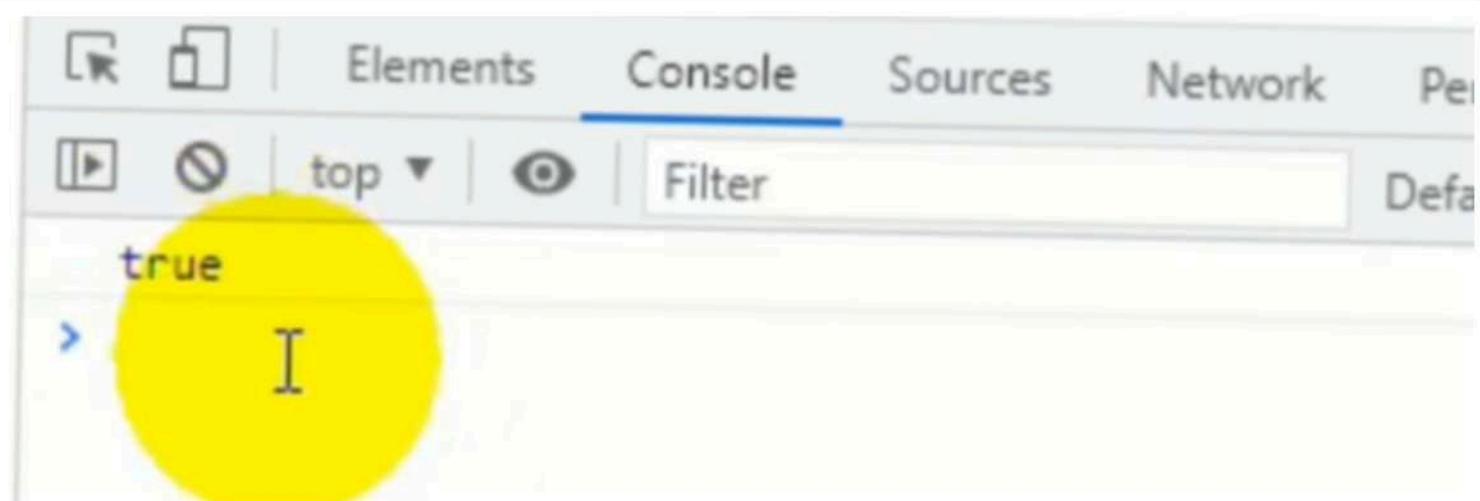
```
</head>
<body>
  <h1 id = "demo"></h1>
  <script>
    // Pattern Matching
    let pattern = /[a-z]/;
    let pass = "12A";   I
    console.log(pattern.exec(pass));
```



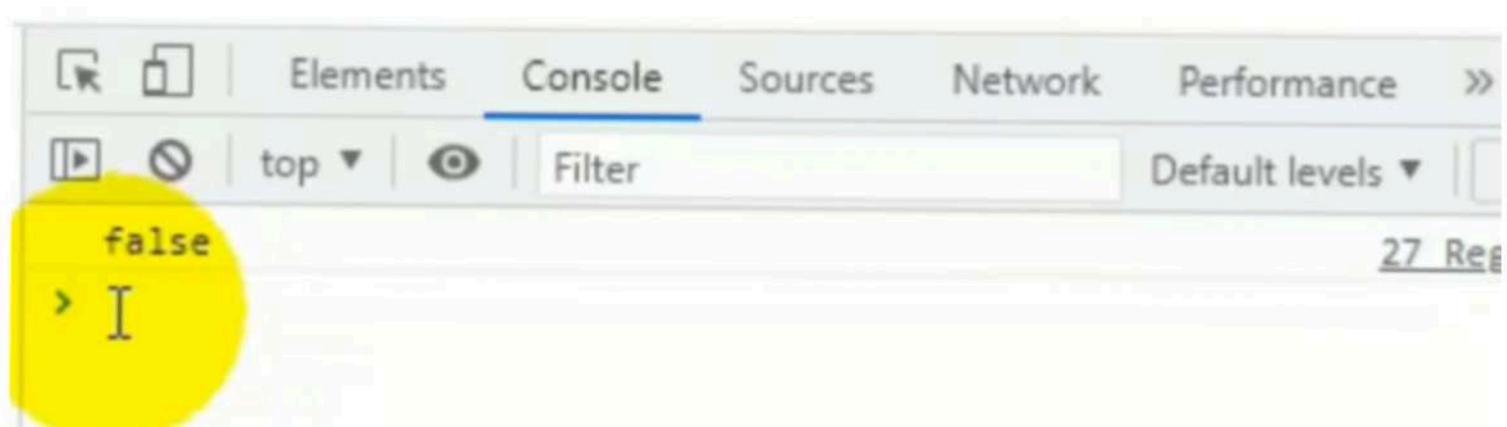
```
</head>
<body>
    <h1 id = "demo"></h1>
    <script>
        // Pattern Matching
        let pattern = /[a-z]/i;
        let pass = "12A";
        console.log(pattern.exec(pass));
```



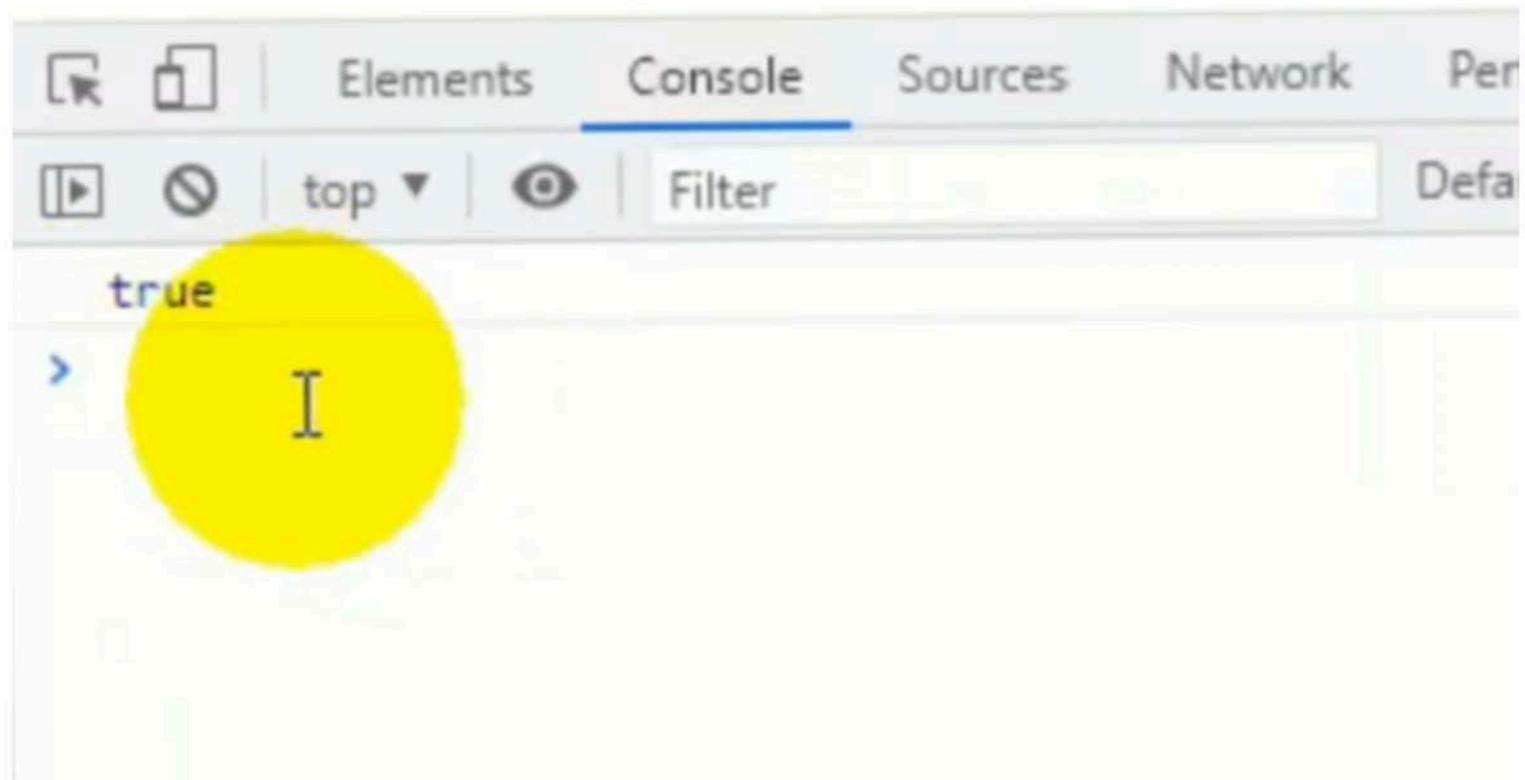
```
let pattern = /[a-z][0-9]/g;  
let pass = "a12345";  
//console.log(pattern.exec(pass));  
console.log(pattern.test(pass));
```



```
let pattern = /[a-z][0-9]/g;  
let pass = "z12345";  
//console.log(pattern.exec(pass));  
console.log(pattern.test(pass)); I
```



```
let pattern = /[a-z][0-9]/ig;  
let pass = "Z12345";  
//console.log(pattern.exec(pass));  
console.log(pattern.test(pass));
```



first assess in match text pattern

```
let text = "JavaScript is a popular programming language. Let's start JavaScript."  
let result = text.match(/JavaScript/);  
document.getElementById("demo").innerHTML = result;
```

```
vaScript is a popular programming language. Let's start JavaScript."  
ext.match(/JavaScript/);  
ementById("demo").innerHTML = result;
```

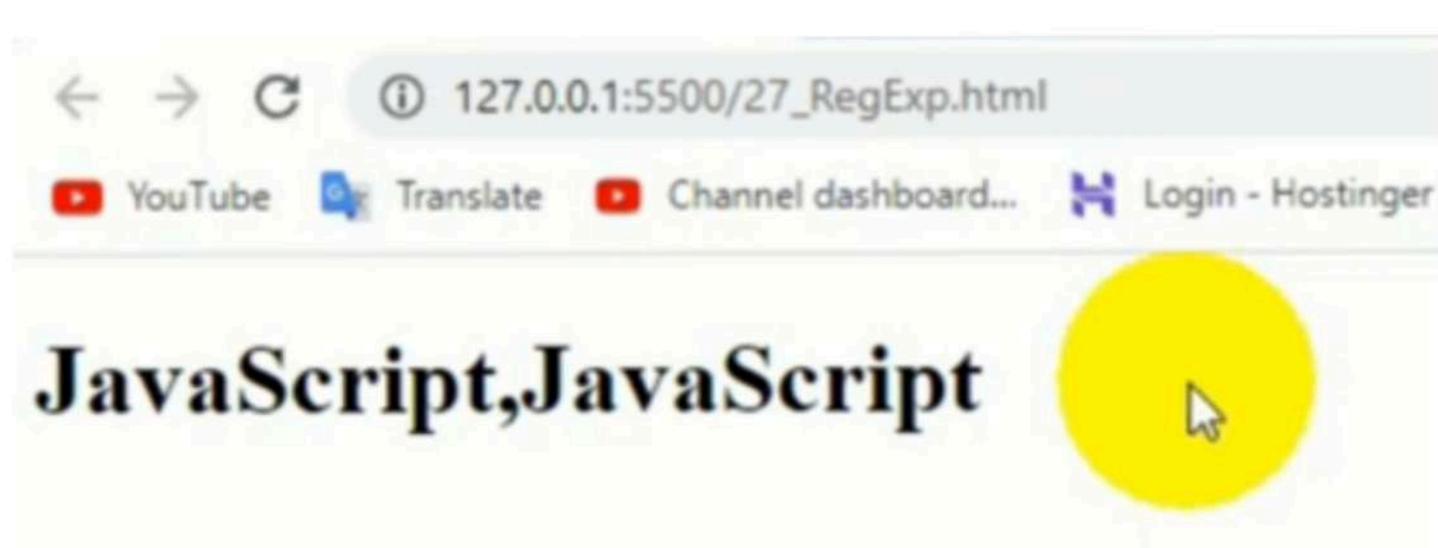
← → ⌂ ⓘ 127.0.0.1:5500/27_RegExp.html

 YouTube  Translate  Channel dashboard...

JavaScript

asscess all match text

```
let text = "JavaScript is a popular programming language. Let's :  
let result = text.match(/JavaScript/g);           I  
document.getElementById("demo").innerHTML = result;
```



search and replace in pattern

```
// String search() With a Regular Expression
let text = "Welcome to Microsolution!";
let t = text.search(/microsolution/i);
document.getElementById("demo").innerHTML = t;
```

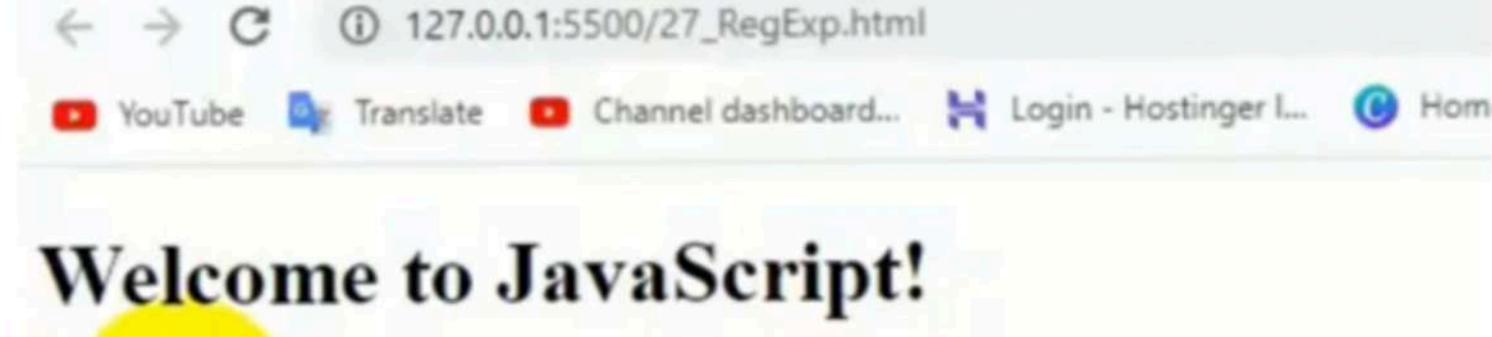


← → ⌂ ⓘ 127.0.0.1:5500/27_RegExp.html

YouTube Translate Channel dashboard... Login - Hostinger ... Home - Canva Speedte

11

```
// String replace() With a Regular Expression
let text = "Welcome to Microsolution!";
let t = text.replace(/Microsolution!/i,"JavaScript!");
document.getElementById("demo").innerHTML = t;
```



← → ⌂ ⓘ 127.0.0.1:5500/27_RegExp.html

YouTube Translate Channel dashboard... Login - Hostinger ... Home

Welcome to JavaScript!

11

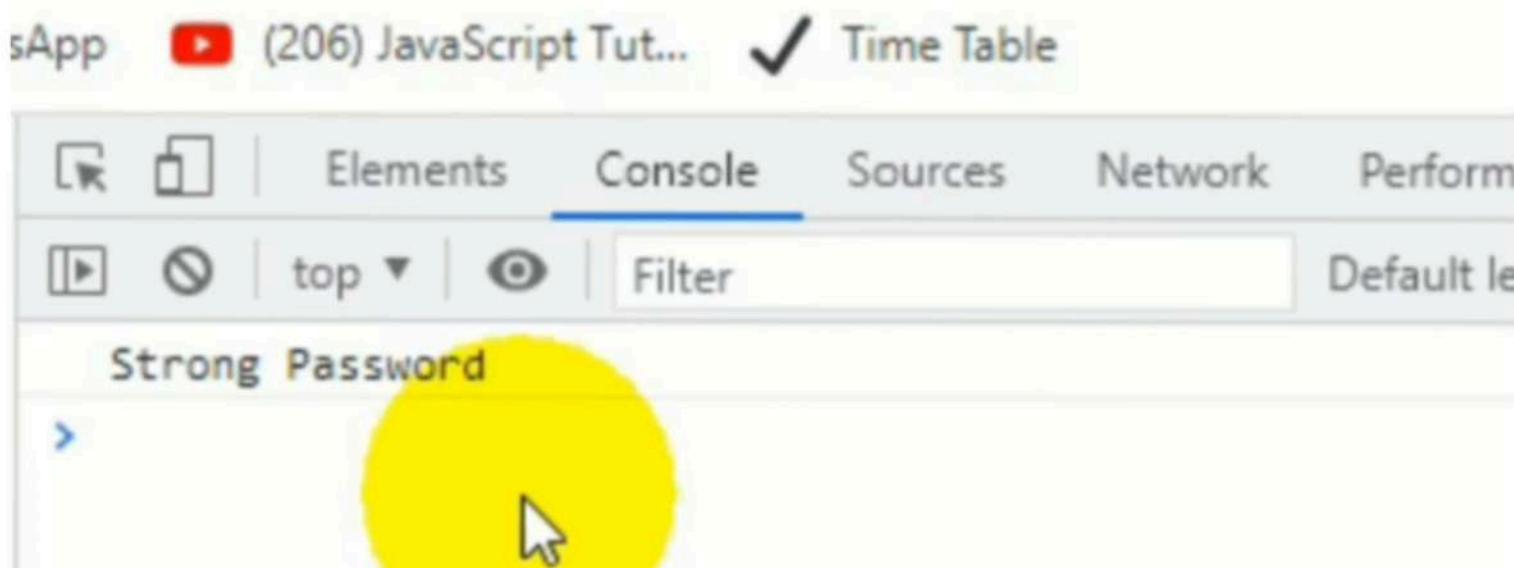
strong password pattern

```
// javascript regex password pattern
let pattern = /^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[@#$%^&*])(?=.{8,})/g;
let pass = "Test@12345";
let t = pattern.test(pass);
if(t==true){
    console.log("Strong Password");
} else {
    console.log("Weak Password");
}
```

```
< password pattern
=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[@#$%^&*])(?=.{8,})/g;
?345";
;t(pass);

:rong Password");

Weak Password");
```



All Validation From

The screenshot shows a web browser window with the following details:

- Tab Bar:** (132) JavaScript Tutorial in Hindi, JavaScript Validation, X, +
- Address Bar:** 127.0.0.1:5500/28_Validation.html
- Toolbar:** YouTube, Translate, Channel dashboard..., Login - Hostinger I..., Home

The page content consists of a form with the following fields:

- Name:
- Email:
- Password:
- Confirm Password:

A **SUBMIT** button is located to the left of the password fields.

HTML CODE

```
<body>
  <form name="myform" action="https://www.youtube.com/playlist?list=PL">
    Name: <input type="text" name="name"><br><br>
    Email: <input type="text" name="email"><br><br>
    Password: <input type="password" name="password1" /><br><br>
    Confirm Password: <input type="password" name="password2"/><br><br>
    <input type="submit" value="SUBMIT">
  </form>
</body>
</html>
```

```
9 <script>
10     function validate_form(){
11         // can't be empty validation
12         let name = document.myform.name.value;
13
14         if (name==null || name==""){
15             alert("Name can't be blank");
16             return false;
17         }
18
19         // email validation
20         let x=document.myform.email.value;
21         let atposition=x.indexOf("@");
```

javascript code

```
19         // email validation
20         let x=document.myform.email.value;
21         let atposition=x.indexOf("@");
22         let dotposition=x.lastIndexOf(".");
23
24         if (atposition<1 || dotposition<atposition+2 || dotposition+2>=x.length){
25             alert("Please enter a valid e-mail address");
26             return false;
27         }
```

```
20 ment.myform.email.value;
21 tition=x.indexOf("@");
22 ition=x.lastIndexOf(".");
23
24 tition<1 || dotposition<atposition+2 || dotposition+2>=x.length){
25 ease enter a valid e-mail address");
26 se;
27
```

Retype password validation in javascript

```
29 // Retype Password Validation
30 let firstpassword = document.myform.password1.value;
31 let secondpassword = document.myform.password2.value;
32
33 if(firstpassword == secondpassword){
34     return true;
35 }
36 else{
37     alert("password must be same!");
38     return false;    I
39 }
```

strong and weak password validation

```
41 // javascript regex password pattern
42 // strong password validation
43 let pattern = /^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[@#$\%^\&\*])/
44 let pass = document.myform.password1.value;
45 let t = pattern.test(pass);
46 if(t==true){
47     return true;
48 } else {
49     alert("Weak Password");
50     return false;
51 }
52 }
```

YouTube (132) JavaScript Tutorial in Hindi × JavaScript Validation × +

← → C ⓘ 127.0.0.1:5500/28_Validation.html

YouTube Translate Channel dashboard... Login - Hostinger I... Home - Canva Speedtest by Ookla...

Name:

Email:

Password:

Confirm Password:

SUBMIT

youtube.com/playlist?list=PLnSDvcENZlwDOiSN1owHqcZUF4jL0xUp-

YouTube Translate Channel dashboard... Login - Hostinger IN Home - Canva Speedtest by Ookla... WhatsApp (206) JavaScript Tutorials Time Table

YouTube IN

Search

SORT

#1 JavaScript Tutorial in Hindi - IDE Software Installing [Part - 1]
Micro Solution

#2 JavaScript Tutorial in Hindi - Outputs, Variables, Data types [Part - 2]
Micro Solution

#3 JavaScript Tutorial in Hindi - Operators [Part - 3]
Micro Solution

JavaScript Tutorial in Hindi

PLAY ALL

13 videos • 1,189 views • Updated 3 days ago

Public

HTML CODE

```
55  
56  
57 iSN1owHqcZUF4jL0xUp-" method="post" onsubmit="return validate_form()" >  
58
```

javascript code changing code

```
52  
33     if(firstpassword == secondpassword){  
34         I  
35         // javascript regex password pattern  
36         // strong password validation  
37         let pattern = /^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[@#$%^&*])  
38         let pass = document.myform.password1.value;  
39         let t = pattern.test(pass);  
40         if(t==true){  
41             return true;  
42         } else {  
43             alert("Weak Password");  
44             return false;  
45         }  
46     }
```

YouTube Channel Validation

Name: Abhishek

Email: indmicrosolution@gmail.com

Password: *****

Confirm Password: *****

SUBMIT



Submit click after action youtube channel playe list open

youtube.com/playlist?list=PLnSDvcENZlwDoiSN1owHqcZUF4jL0xUp-

YouTube Translate Channel dashboard... Login - Hostinger I... Home - Canva Speedtest by Ookla... WhatsApp (206) JavaScript Tutorials Time Table

Search

JavaScript Tutorial in Hindi

Introduction & Installation VS Code

PLAY ALL

JavaScript Tutorial in Hindi

13 videos • 1,189 views • Updated 3 days ago

Public

SORT

- JavaScript Tutorial in Hindi - IDE Software Installing [Part - 1]
- JavaScript Tutorial in Hindi - Outputs, Variables, Data types [Part - 2]
- JavaScript Tutorial in Hindi - Operators [Part - 3]

Full Code

28_Validation.html > html > script > validate_form

```
9 <script>
10     function validate_form(){
11         // can't be empty validation
12         let name = document.myform.name.value;
13
14         if (name==null || name==""){
15             alert("Name can't be blank");
16             return false;
17         }
18
19         // email validation
20         let x=document.myform.email.value;
21         let atposition=x.indexOf("@");
22         let dotposition=x.lastIndexOf(".");
23
24         if (atposition<1 || dotposition<atposition+2 || dotposition+2>=x.length){
25             alert("Please enter a valid e-mail address");
26             return false;
27         }
28
```

```
29
30         // Retype Password Validation
31         let firstpassword = document.myform.password1.value;
32         let secondpassword = document.myform.password2.value;
33
34         if(firstpassword == secondpassword){
35             // javascript regex password pattern
36             // strong password validation
37             let pattern = /^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!@#$%^&*])(?=.{8,})/g;
38             let pass = document.myform.password1.value;
39             let t = pattern.test(pass);
40             if(t==true){
41                 return true;
42             } else {
43                 alert("Weak Password");
44                 return false;
45             }
46         }
47     else{
```

```
42     } else {
43         alert("Weak Password");
44         return false;
45     }
46 }
47 else{
48     alert("password must be same!");
49     return false;
50 }
51
52
53 }
54
55
56 </script>
57 <body>
```

I

```
55
56 </script>
57 <body>
58     <form name="myform" action="https://www.youtube.com/playlist?list=PLnSDvcENZlwDOI
59     Name: <input type="text" name="name"><br><br>
60     Email: <input type="text" name="email"><br><br>
61     Password: <input type="password" name="password1" /><br><br>
62     Confirm Password: <input type="password" name="password2"/><br><br>
63     <input type="submit" value="SUBMIT">
64
65 </body>
66 </html>
```

```
55
56
57
58 <form name="myform" action="https://www.youtube.com/playlist?list=PLnSDvcENZlwDOI
59     Name: <input type="text" name="name"><br><br>
60     Email: <input type="text" name="email"><br><br>
61     Password: <input type="password" name="password1" /><br><br>
62     Confirm Password: <input type="password" name="password2"/><br><br>
63     <input type="submit" value="SUBMIT">
64
65 </body>
66 </html>
```

Exception Handling

What is Exception?

Exception is an abnormal condition in a program.

Exception Handling

In programming, exception handling is a process for handling the abnormal statements in the code.

There are following statements that handle if any exception occurs:

try...catch statements

throw statements

try...catch...finally statements.

try and catch exception handling

```
<script>  
    aleart("Welcome");
```

The screenshot shows the browser's developer tools with the 'Console' tab selected. It displays two error messages:

- Uncaught ReferenceError: aleart is not defined at 29 Exception Handling.html:11
- Failed to load resource: the server responded with a status of 404 (Not Found) favicon.ico:1

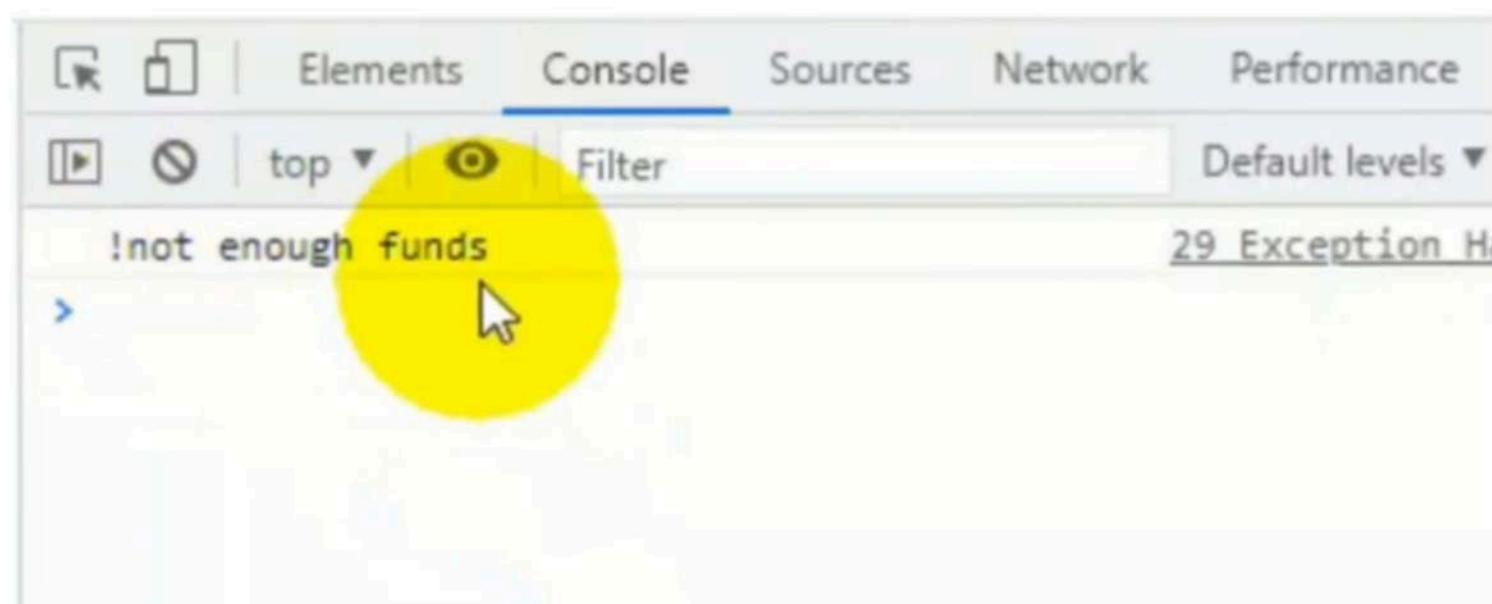
Below the errors, it says "Live reload enabled." and has a timestamp "29 Exception Handling.html:76".

```
<body>  
    <script>  
        // try and catch example  
        try {  
            alertt("Welcome!");  
        }  
        catch(err) {  
            console.log("Please check your syntax!");
```

The screenshot shows the browser's developer tools with the 'Console' tab selected. It displays a single log message: "Please check your syntax!" with a timestamp "29 Exception Handling.html:16".

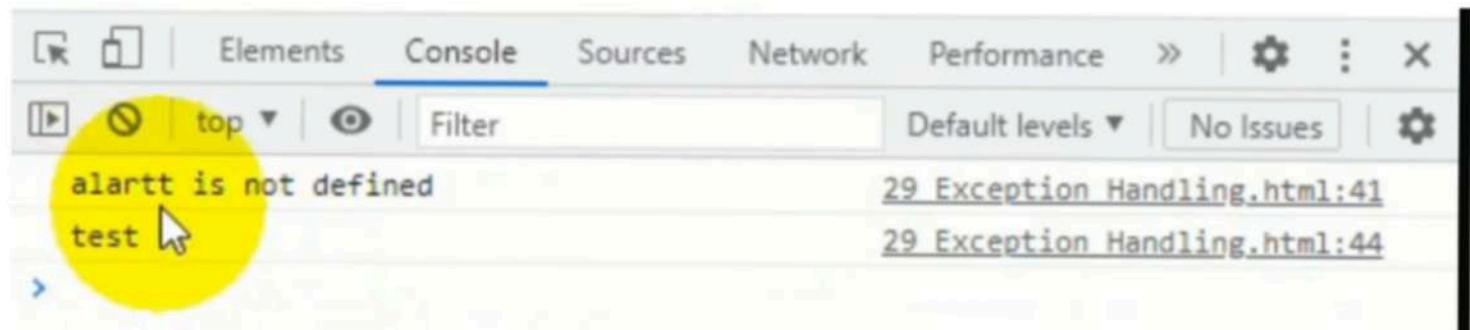
```
19 // throw example
20 let saving = 5000;
21 let withdrawal = 7000;
22 try{
23     if(saving > withdrawal){
24         saving -= withdrawal;
25         console.log("your current balance is : "+saving);
26     }
27     else {
28         throw "not enough funds";
29     }
30 }
31 catch(err){
```

```
else {
    throw "not enough funds";
}
}
catch(err){
    console.log(err);
}
```



finally exception handling

```
34
35      // finally example
36      try {
37          alartt("Welcome!");
38      }
39      catch(err) {
40          let a = err.message
41          console.log(a);
42      }
43      finally{
44          console.log("test");
45      }
46 <script>
```



Loan EMI Calculator Project in JS

LOAN EMI CALCULATOR

Loan Amount :

Interest Rate :

Years to Pay :

Calculate

Loan EMI : **99.01**

Total Interest Payable : **940.60**

Total Payment : **5940.60**

```
8 </head>
9 <style>
10    input[type=text], select {
11        width: 50%;
12        padding: 12px 20px;
13        margin: 8px 0;
14        display: inline-block;
15        border: 1px solid #ccc;
16        border-radius: 4px;
17        box-sizing: border-box;
18    }
```

```
20   label {
21     font-size: 30px;
22     font-family: Arial, Helvetica, sans-serif;
23   }
24
25   I
26   input[type=submit] {
27     width: 15%;
28     background-color: #4184ff;
29     color: white;
30     padding: 14px 20px;
31     margin: 8px 0;
32     border: none;
33     border-radius: 4px;
34     cursor: pointer;
35     font-size: 30px;
```

```
34     font-size: 30px;
35     font-family: Arial, Helvetica, sans-serif;
36   }
37
38   input[type=submit]:hover {
39     background-color: #45a072;
40   }
41
42   div {
43     border-radius: 5px;
44     background-color: #f2f2f2;
45     padding: 20px;
46   }
47 </style>
```

```
49   <h1>LOAN EMI CALCULATOR</h1>
50
51   <div>
52     <form onsubmit="return false">
53       <label for="LoanAmount">Loan Amount :</label>
54       <input type="text" id="amount">
55       <br>
56       <label for="InterestRate">Interest Rate :</label>
57       <input type="text" id="rate">
58       <br>
59       <label for="Years">Years to Pay :</label>
60       <input type="text" id="years">
61       <br>
62       <input type="submit" value="Calculate" onClick="calcu]
63   </form>
```

```
61         <br>
62         <input type="submit" value="Calculate" onClick="calcul
63             </form>
64         </div>
65         <label for="output" id="demo"> </label>
66     <script>
```

```
62 onClick="calculate(amount.value,rate.value,years.value);"/>
63
```

```
66 <script>
67     function calculate(amount,rate,years){
68         var p=parseInt(amount);
69         var r=parseFloat(rate);
70         var n=parseInt(years);
71
72         var monthly_interest = r/12/100;
73
74         var months = n * 12;
75
76         // P x R x (1+R)^N / [(1+R)^N-1] This is Loan EMI Form
77
78         var x = Math.pow(1 + monthly_interest, months);
79         var monthly = (p * x * monthly_interest) / (x - 1);
80             I
81         var emi = monthly.toFixed(2);
82         var totalPayment = (emi * months).toFixed(2);
83         var totalInterest = (totalPayment - p).toFixed(2);
84
85         document.getElementById("demo").innerHTML="Loan EMI :
```

```
80
81         var emi = monthly.toFixed(2);
82         var totalPayment = (emi * months).toFixed(2);
83         var totalInterest = (totalPayment - p).toFixed(2);
84
85         document.getElementById("demo").innerHTML="Loan EMI : " + "<
86     }
87 </script>
```

```
83 2);  
84  
85 | EMI : " + "<b>" + emi + "</b>" + "<br>" + "Total Interest Payable : "  
86  
87
```

```
83  
84  
85 total Interest Payable : " + "<b>" + totalInterest + "</b>" + "<br>" + "Total Payment : "  
86  
87
```

```
83  
84  
85 : " + "<b>" + totalInterest + "</b>" + "<br>" + "Total Payment : "  
86  
87
```

```
83  
84  
85 r>" + "Total Payment : " + "<b>" + totalPayment + "</b>";  
86  
87
```

Fuel Calculator Project in JS

← → C ⓘ 127.0.0.1:5500/31_Fuel_Calculator.html

YouTube Translate Channel dashboard... Login - Hostinger I... Home - Canva Speedtest by Ookla... WhatsApp (206) JavaScript Tutorials Time Table

FUEL CALCULATOR

Enter Total Kilometer :

Enter Vehicle Milage :

Enter Fuel Price (₹) :

Calculate

Fuel needed : **6.98 Liters**
Fuel cost : **837.60 ₹**

```
38     input[type=submit]:hover {  
39         background-color: #45a072;  
40     }  
41  
42     div {  
43         border-radius: 5px;  
44         background-color: #f3eea8;  
45         padding: 20px;  
46     }  
47 </style>
```

```
51     <div>  
52         <form onsubmit="return false">  
53             <label for="km">Enter Total Kilometer :</label>  
54             <input type="text" id="totalkm">  
55             <br>  
56             <label for="milage">Enter Vehicle Milage :</label>  
57             <input type="text" id="milage">  
58             <br>  
59             <label for="fuelprice">Enter Fuel Price (₹) :</label>  
60             <input type="text" id="fuel_price">  
61             <br>  
62             <input type="submit" value="Calculate" onClick="calcul  
63             <br>  
64         </form>  
65     </div>
```

```
62 ' onClick="calculate(totalkm.value,milage.value,fuel_price.value);"
62 e(totalkm.value,milage.value,fuel_price.value);"/>
62
```

```
66      <label for="output" id="demo"> </label>
67      <script>
68          function calculate(a,b,c){
69              let totalkm=parseInt(a);
70              let milage=parseInt(b);
71              let fuel_price=parseFloat(c);
72
73              let Total_Cost = ( ( totalkm / milage ) * fuel_price).toFixed(2);
74              let total_fuel = totalkm/milage.toFixed(2);
75              document.getElementById("demo").innerHTML="Fuel needed : " + <b>" + total_fuel + "</b>" + "Litters" + "<br>" + "Fuel cost : " + <b>" + Total_Cost + " ₹" + "</b>" ;
76
77      </script>
```

```
73 Fixed(2);
74
75   " + "<b>" + total_fuel + " Litters" + "</b>" + "<br>" + "Fuel c
```

```
73 ( ( totalkm / milage ) * fuel_price).toFixed(2);
74 totalkm/milage.toFixed(2);
75 ntById("demo").innerHTML="Fuel needed : " + "<b>" + total_fuel + "</b>" + "Litters" + "<br>" + "Fuel cost : " + <b>" + Total_Cost + " ₹" + "</b>" ;
```

```
73
74
75 " + "Fuel cost : " + "<b>" + Total_Cost + " ₹" + "</b>" ;
```

```
69     let totalkm=parseInt(a);
70     let milage=parseInt(b);
71     let fuel_price=parseFloat(c);
72
73     let Total_Cost = ( ( totalkm / milage ) * fuel_price).toFixed(2);
74     let total_fuel = totalkm/milage.toFixed(2);
75     document.getElementById("demo").innerHTML="Fuel needed is "+total_fuel;
76   }
77 </script>  I
78 </body>
79 </html>
```

Question for you

Q.10 Create a age calculator project.

Email - indmicrosolution@gmail.com