Java Script

JavaScript is a scripting language which is for

1. To perform validation at client side.
2. To Generate dynamic html
3. To Apply CSS AT run time
4. To Make Ajax Call
5. To perform event at run time
6. To Consume REST services using JSON

etc

How to work with JavaScript

If we want to work with JavaScript we have two ways

1. Internal JavaScript: Internal JavaScript means we can write script tag in html page under head tag or we use JavaScript within a same page called as Internal JavaScript.

```
 <script type=’text/JavaScript’>
           Write here your logics
</script>
```

Example: Now we want to create program to display good morning message in html page using java script.
if we want to show the anything on program output we have document object and its one method name as write() this method can display output on web page.

Syntax: document.write(“good morning”);

Example:

```
<html>
 <head>
   <title>i am java script</title>
    <script type='text/javascript'>
             document.write("good morning");
           </script>
  </head>
  <body>

  </body>
</html>
```

2. External JavaScript: External JavaScript means we write separate file with extension .js and we use this file in html page.

Steps to work with external JS

1. Create separate file with extension .js
demo.js

```
document.write("good morning");
```

2. Use script tag in html page
Syntax: <script type=’text/javascript’ src=”filename.js”></script>
Example: <script type=’text/javascript’ src=”demo.js”></script>

Example: test.html

```
<html>
 <head>
   <title>i am java script</title>
    <script type='text/javascript' src="demo.js"></script>
  </head>
  <body>
  </body>
</html>
```

How to declare variable in java script

JavaScript is dynamic type language

There are two types’ languages
1. Static type language: static type language means those languages required to mention the data type at the time of variable declaration called as static type language.

Example: C, C++, JAVA etc these are the static type languages.
int a=100;
Here we must be required to mention int at the time of variable declaration.

2. Dynamic type language: Dynamic type language means that language not required specifying data type at the time of variable declaration and variable can change its type according value assign in it called as dynamic type languages**.**
Example: python, PHP, JS etc
a=100; it is variable declaration in java script here we not required to specify data type so it is dynamic type language.

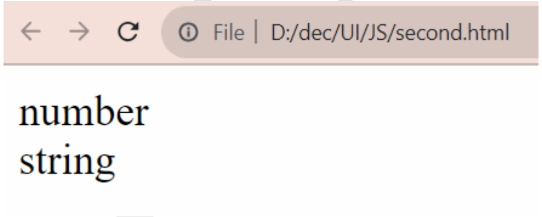
If we want to declare variable in JavaScript we have three keywords for declare variable in java script

1. let 2. const 3. var

Syntax for declare variable using var
Syntax: var variablename=value;
Example: var a=100;

if we think about above code variable a work as number and if we initialize value a=5.4 then a work as float, a="good" it work as string etc means here a has no specific type and variable a can modify its type according value assign in it.

Example:



```
number
string
```

```
<html>
 <head>
  <title>i am java script</title>
   <script type='text/javascript'>
            var a=100;
```

```
        document.write(typeof(a));
         a="good";
        document.write("<br>"+typeof(a));
       </script>
 </head>
 <body>
 </body>
</html>
```

Example: we want to declare two variables using a java script and calculate its addition.

```
<html>
 <head>
  <title>i am java script</title>
   <script type='text/javascript'>
         var a,b,c;
         a=100;
         b=200;
         c=a+b;
         document.write("Addition is  "+c);
       </script>
 </head>
 <body>

 </body>
</html>
```

Example: Declare variable using a let keyword and calculate its addition

```
<html>
 <head>
  <title>i am java script</title>
   <script type='text/javascript'>
         let a,b,c;
         a=100;
         b=200;
         c=a+b;
```

```
                document.write("Addition is  "+c);
             </script>
  </head>
  <body>

  </body>
</html>
```

Example: Declare addition of two values using const keyword.

```
             <html>
              <head>
                <title>i am java script</title>
                          <script type='text/javascript'>

                            const a=100;
                             const b=200;
                             const c=a+b;
                             document.write("Addition is  "+c);
                          </script>
               </head>
               <body>

               </body>
             </html>
```

Q. what is diff between let ,const and var keyword in JavaScript?

var	**let**	**Const**

Redeclaration of variable is allowed
 	
var a=100;
var a=200;
Note: allowed variable declarations
	Redeclaration of variable not allowed
	
	let a=100;
	let a=200;
	Note: Generate error at run time a is already declare
		Must be assign value at the time of variable declaration and reassigning not allowed
		
		const a=100;
		const a=200;
		Note: Generate error at run time

var keyword use function level scope	let keyword is used for use block level scope	const keyword is used for use block level scope

means if we define function and if function block and if we declare variable within block using var keyword then variable can access outside of block.
 	
Example:
function show(){
//block inside of function
{
var a=100;
}
document.write("A is "+a);
}
show();//call function
	means if we declare variable within a block then variable cannot access outside of block within a same function or outside of function also.
	
	Example
	function show(){
	{ let a=100;
	}
	document.write("A is "+a);
	}
	Note: Generate error a is not defined because we declare variable a within block and let has block level scope so we cannot access it outside of block.
		Means if we declare variable within a block then variable cannot access outside of block within a same function or outside of function also.
		Example
		function show(){
		{ let a=100;
		}
		document.write("A is "+a);
		}
		Note: Generate error a is not defined because we declare variable a within block and let has block level scope so we cannot access it outside of block.

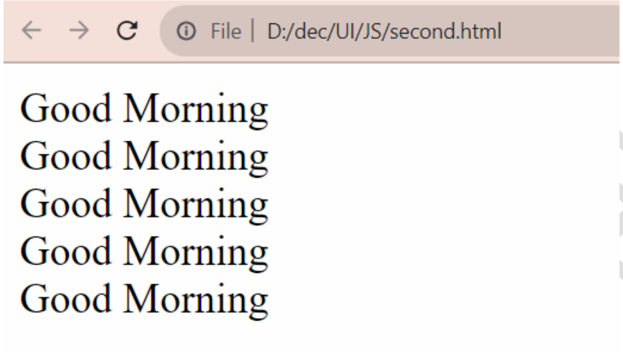
Using var we can declare variable without initialization
Example:
var a;
document.write(a);
	Using let we can declare variable without initialization
	Example:
	var a;
	document.write(a);
		Using const we can not declare variable without initialization
		Example:
		const a;
		document.write(a);

Default value is undefined	Default value is undefined	Generate error missing initializer

Decision making statement using JavaScript

```
<html>
 <head>
   <title>i am java script</title>
          <script type='text/javascript'>
                              let a=100;
                              let b=200;
                              if(a>b){
                               document.write("A is Greater");
                              }
                              else{
                                document.write("B is Greater");
                              }

                    </script>
              </head>
              <body>

              </body>
             </html>
```

How to use loop using JavaScript



File | D:/dec/UI/JS/second.html

```
Good Morning
Good Morning
Good Morning
Good Morning
Good Morning
```

Source code

```
             <html>
              <head>
                <title>i am java script</title>
                    <script type='text/javascript'>
                              for(var i=1;i<=5;i++)
                              { document.write("Good Morning<br>");
                              }

                    </script>
              </head>
              <body>

              </body>
             </html>
```

Interview Question

Q1. What is java script and why use it?
Q2. How many types of JavaScript?
Q3. What is dynamic language?
Q4. What is diff between dynamic type and static type language?
Q5. Why java script is most popular language?
Q6. How to declare variable in JavaScript and explain its ways?
Q7. What is diff between let, const and var keyword?

Data Type in JavaScript

There are 8 data types in JavaScript
1. String 2. Number 3. Bigint 4. Undefined 5. Null 6. symbol 7. boolean 8. Object

JavaScript provide some object type data type

1. Object
2. array
3. Date

Array in JavaScript

Q. what is array?
Array is a collection of similar type of data

How to declare array in java script
If we want to declare array in java script we have two ways

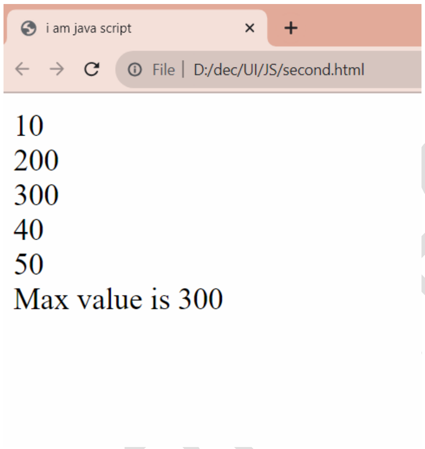
1. Use Array Subscript
2. Use Array class.

Use Array Subscript
let variablename = [values];
Example: let a = [10, 20, 30, 40, 50];

Example:

```
<html>
 <head>
  <title>i am java script</title>
          <script type='text/javascript'>
                         let a=[10,20,30,40,50];
                        for(let i=0; i<a.length;i++)
                        {
                          document.write(a[i]+"<br>");
                        }
          </script>
  </head>
  <body>

  </body>
</html>
```

Example: WAP to store 5 values in array and find maximum value from array.



```
10
200
300
40
50
Max value is 300
```

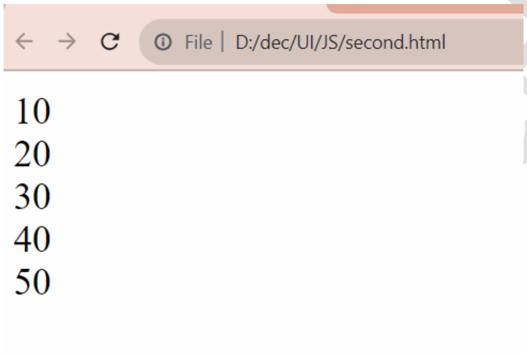
Source code

```
<html>
 <head>
  <title>i am java script</title>
          <script type='text/javascript'>
                         let a=[10,200,300,40,50];
                        for(let i=0; i<a.length;i++)
                        {
                          document.write(a[i]+"<br>");
```

```
              }
              let max=a[0];
              for(var i=0; i<a.length;i++){
                if(a[i]>max){
                      max=a[i];
                     }
              }
              document.write("Max value is  "+max);
       </script>
 </head>
 <body>

 </body>
</html>
```

You can create array in java script by using Array class just we required to create its object.

let variablename = new Array();



```
10
20
30
40
50
```

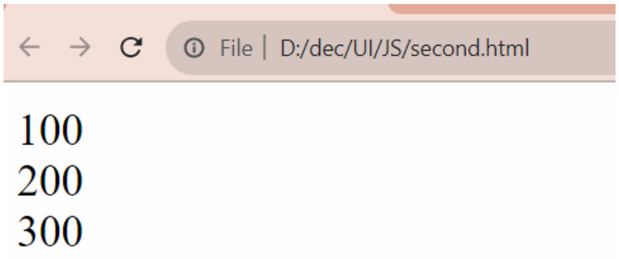
Source code

```
<html>
 <head>
  <title>i am java script</title>
       <script type='text/javascript'>
              let a= new Array(10,20,30,40,50);
              for(var i=0; i<a.length;i++)
              {
               document.write(a[i]+"<br>");
```

```
                        }
                </script>
          </head>
          <body>
          </body>
         </html>
```

JavaScript provide some method to us which help us to perform operation on array.

push(value): this is used for add data in array

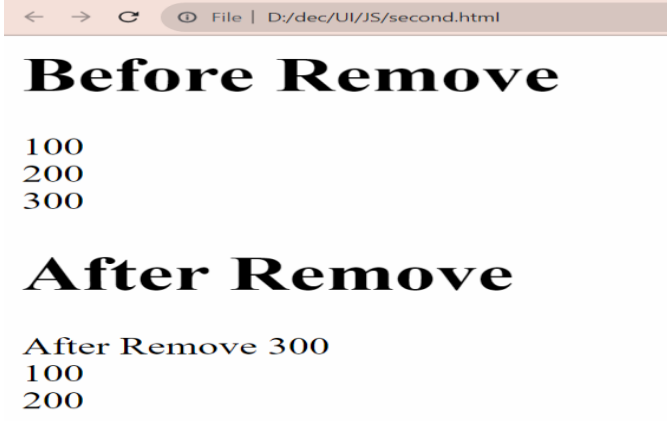


```
100
200
300
```

Source code

```
         <html>
          <head>
           <title>i am java script</title>
                <script type='text/javascript'>
                        let a= new Array();
                        a.push(100);
                        a.push(200);
                        a.push(300);
                        for(var i=0; i<a.length;i++)
                        {
                          document.write(a[i]+"<br>");
                        }
                </script>
          </head>
          <body>
          </body>
         </html>
```

pop(): this method is used for remove element from array using java script.



Before Remove

100
200
300

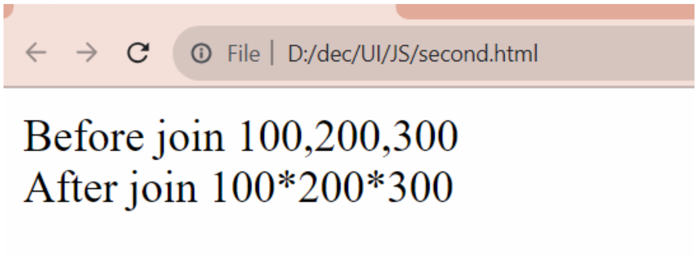
After Remove

After Remove 300
100
200

Source code

```
<html>
 <head>
   <title>i am java script</title>
          <script type='text/javascript'>
                      let a= new Array();
                      a.push(100);
                      a.push(200);
                      a.push(300);
                      document.write("<h1>Before Remove</h1>");
                      for(var i=0; i<a.length;i++) { document.write(a[i]+"<br>");
                      }
                      document.write("<h1>After Remove</h1>");
                      let value=a.pop();
                      document.write("After Remove "+value+"<br>");
                      for(var i=0; i<a.length;i++)
                      { document.write(a[i]+"<br>");
                      }
          </script>
  </head>
  <body>
  </body>
</html>
```

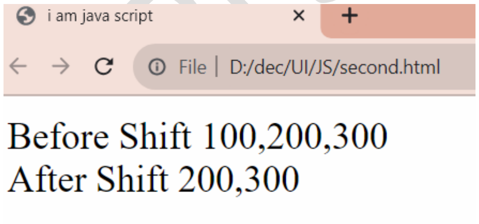
join(): this method is used for join all array elements using a specified character and create new array



```
Before join 100,200,300
After join 100*200*300
```

```
<html>
 <head>
  <title>i am java script</title>
         <script type='text/javascript'>
                  let a= new Array();
                  a.push(100);
                  a.push(200);
                  a.push(300);
                  document.write("Before join "+a);
                  let b= a.join("*");
                  document.write("<br>After join  "+b);

         </script>
  </head>
  <body>

  </body>
</html>
```

shift(): this method can remove the first element from array and shift remaining element at left hand side or at lower side.



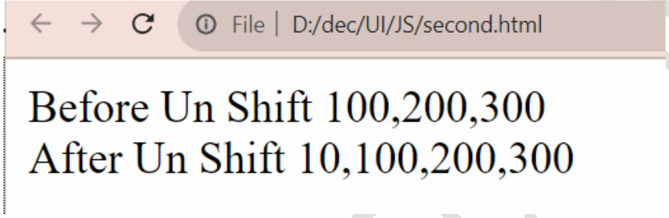
```
Before Shift 100,200,300
After Shift 200,300
```

Source code

```
<html>
 <head>
  <title>i am java script</title>
```

```
        <script type='text/javascript'>
                let a= new Array();
                a.push(100);
                a.push(200);
                a.push(300);
                document.write("Before Shift "+a);
                a.shift();
                document.write("<br>After Shift "+a);

        </script>
     </head>
     <body>
     </body>
    </html>
```

unshift(): this method can add new element at 0th position in array and shirt remaining element at right hand side.

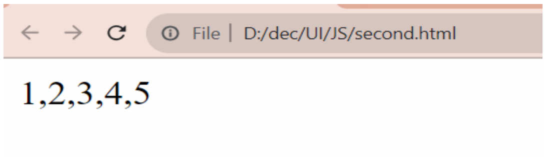


```
Before Un Shift 100,200,300
After Un Shift 10,100,200,300
```

Source code

```
    <html>
     <head>
      <title>i am java script</title>
        <script type='text/javascript'>
                let a= new Array();
                a.push(100);
                a.push(200);
                a.push(300);
                document.write("Before Un Shift "+a);
                a.unshift(10);
                document.write("<br>After Un Shift "+a);
               </script>
     </head>
     <body>
     </body>
    </html>
```

sort(): this method is used for sort   array values

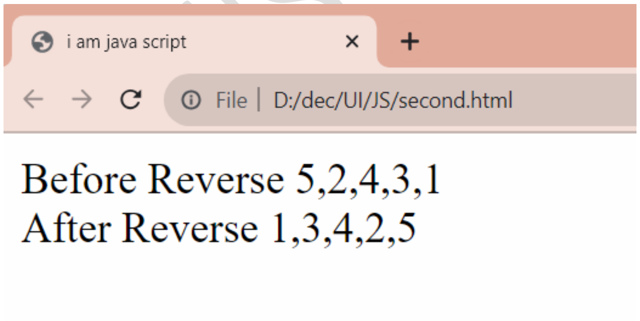


```
1,2,3,4,5
```

Source code

```
    <html>
     <head>
      <title>i am java script</title>
            <script type='text/javascript'>
                      let a=[5,2,4,3,1];
                             a.sort();
                             document.write(a);

            </script>
     </head>
     <body>
     </body>
    </html>
```

reverse():this method can reverse element of array.



```
Before Reverse 5,2,4,3,1
After Reverse 1,3,4,2,5
```

```
    <html>
     <head>
```

```
  <title>i am java script</title>
          <script type='text/javascript'>
                        let a=[5,2,4,3,1];
                                document.write("Before Reverse  "+a);
                                  a.reverse();
                                  document.write("<br>After Reverse  "+a);

          </script>
 </head>
 <body>

 </body>
</html>
```

concat(): this method is used combine or merge two arrays.
Syntax: let thirarray=concat(firstarray,secondarray);

let variable=a.indexOf(value): this method can search value in array and return its index and if value not found return -1

Example:

```
<html>
 <head>
  <title>i am java script</title>
          <script type='text/javascript'>
                        let a=[10,20,30];
                                let b=[40,50,60];
                                let c=a.concat(b); //merge two array
                                document.write(c);
                                let index=a.indexOf(20);
                                 if(index!=-1)
                                 { document.write("<br>element found");
                                 }
                                 else{
                                   document.write("<br>Element not found");
                                 }
```

```
              </script>
        </head>
        <body>
        </body>
       </html>
```

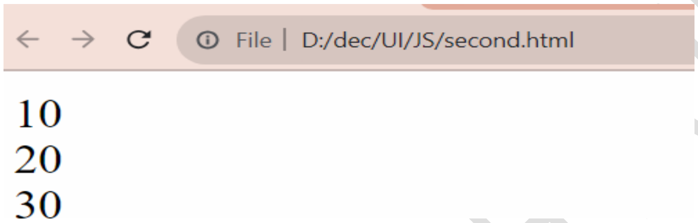
Iteration of Array in JavaScript

____

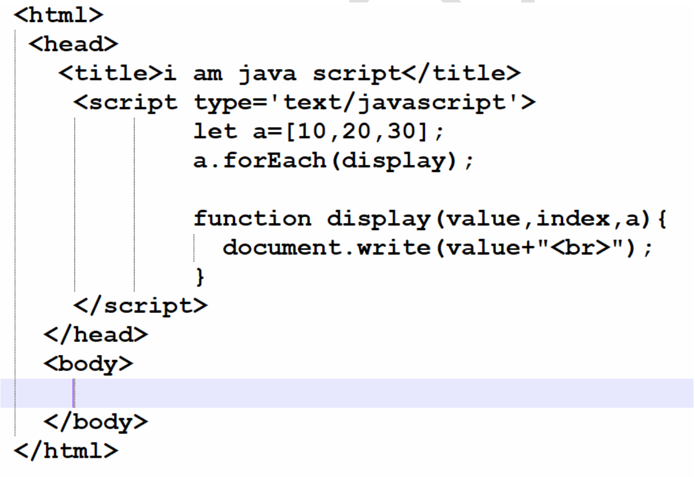
if we want to perform iteration on array using JavaScript we have forEach() method

Syntax:

arrayvariable.forEach(function(value,index,array)): if we use forEach method then we have to pass function in it and pass three variable in function for each internally travel array and display values.



```
10
20
30
```



```
<html>
 <head>
   <title>i am java script</title>
    <script type='text/javascript'>
            let a=[10,20,30];
            a.forEach(display);

            function display(value,index,a){
              document.write(value+"<br>");
            }
    </script>
  </head>
  <body>

  </body>
</html>
```

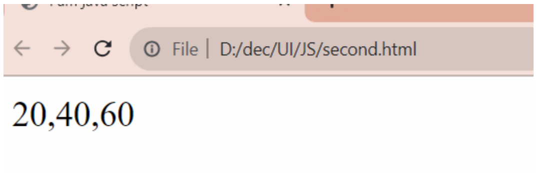
Source code

```
       <html>
        <head>
```

```
          <title>i am java script</title>
                 <script type='text/javascript'>
                          let a=[10,20,30];
                         a.forEach((value,index,a)=>document.write(value+"<br>"));
                 </script>
         </head>
         <body>
         </body>
        </html>
```

map(): this method create a new array by performing a function on each array element
Syntax: map(function);

Example: we want to multiply every element of array by 2

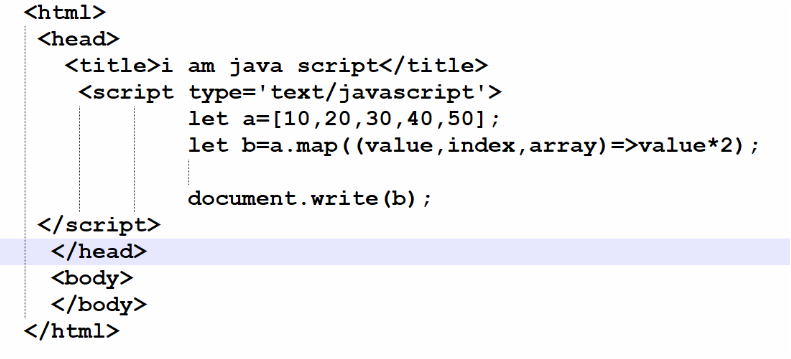


```
20,40,60
```

Source code

```
        <html>
         <head>
          <title>i am java script</title>
                 <script type='text/javascript'>
                            let a=[10,20,30];
                            let b=a.map(mul);
                                  function mul(value,index,array)
                                  { return value*2;
                                  }
                                  document.write(b);
         </script>
         </head>
         <body>
         </body>
        </html>
```

Shortcut of above code



```
<html>
 <head>
   <title>i am java script</title>
    <script type='text/javascript'>
            let a=[10,20,30,40,50];
            let b=a.map((value,index,array)=>value*2);

            document.write(b);
 </script>
  </head>
  <body>
  </body>
</html>
```

How to use function in JavaScript

---

if we want to use function in java script we have function keyword

Syntax: function functionname(datatype variablename)

```
       {  write here your logics
       }
```

Example: we want to declare function which accept two values as parameter and calculate its addition.

```
<html>
 <head>
  <title>demo js function</title>
  <script type='text/javascript'>
    function add(x,y)
         {
          return x+y;
         }

         let result=add(100,200);
         document.write("<h1>Addition is  "+result+"</h1>");
  </script>
 </head>
 <body>

 </body>
</html>
```

Example: WAP to print the five times good morning using a recursion in java script.

```
<html>
 <head>
   <title>demo js function</title>
   <script type='text/javascript'>
      let count=0;
      function show()
              {
                 if(count!=5){
                           document.write("<h1>good morning</h1>");
                                    ++count;
                                    show();
                          }
              }
               show();

   </script>
 </head>
 <body>
 </body>
</html>
```

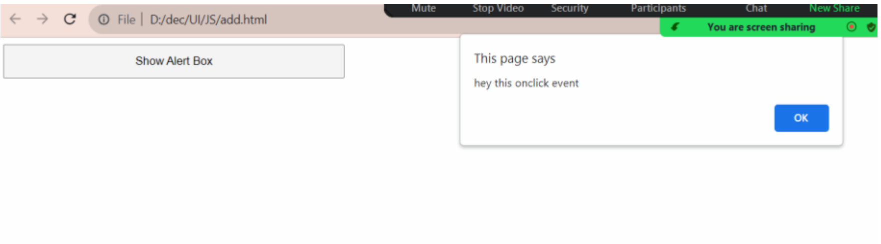
Event Handling in JS

If we think about HTML so HTML provides some inbuilt event to us we can use that event with html element and can call java script function on that event.

Types of events in JS

1. onclick 2. ommouseover 3. onmouseleave 4. onload 5. onblur 6. onkeyp 7. onkeydown 8.onkeypress etc

Example: we want to design web page with one button and when user click on button then we want to display alert() message on it and show good morning message in alert box.



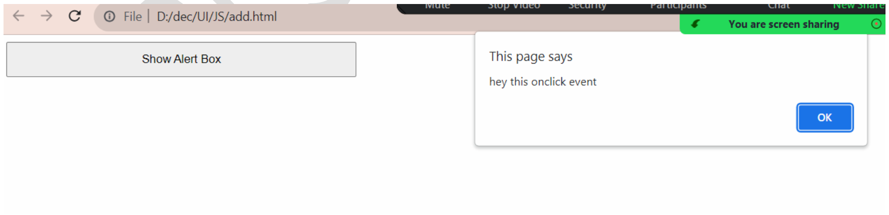
This page says

hey this onclick event

Source code

```
<html>
 <head>
   <title>demo js function</title>
   <script type='text/javascript'>
      function show(){
                 alert("hey this onclick event");
               }
   </script>
 </head>
 <body>

  <input type="button" name="s" value="Show Alert Box"
  style='width:400px;height:40px;' onclick="show()"/>

 </body>
</html>
```

onmouseover: this event fire when user enter mouse on html element or html control.



This page says

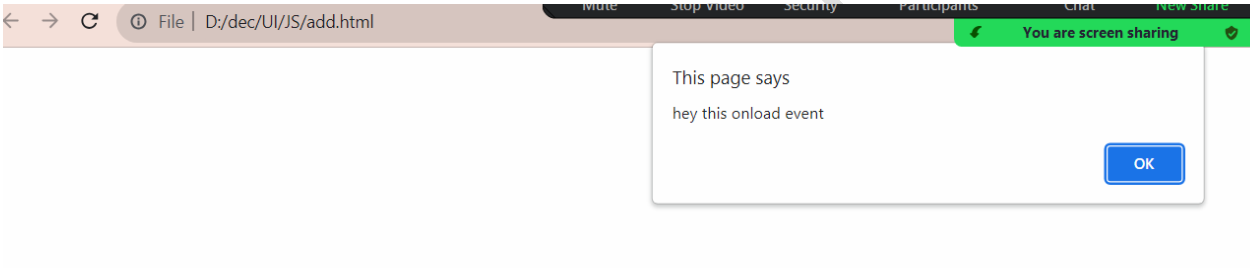
hey this onclick event

Source code

```
<html>
 <head>
   <title>demo js function</title>
   <script type='text/javascript'>
```

```
      function show(){
                 alert("hey this onclick event");
              }
   </script>
 </head>
 <body>

  <input type="button" name="s" value="Show Alert Box"
  style='width:400px;height:40px;' onmouseover="show()"/>

 </body>
</html>
```

onload: this event execute or fire when we refresh page or when we load page.



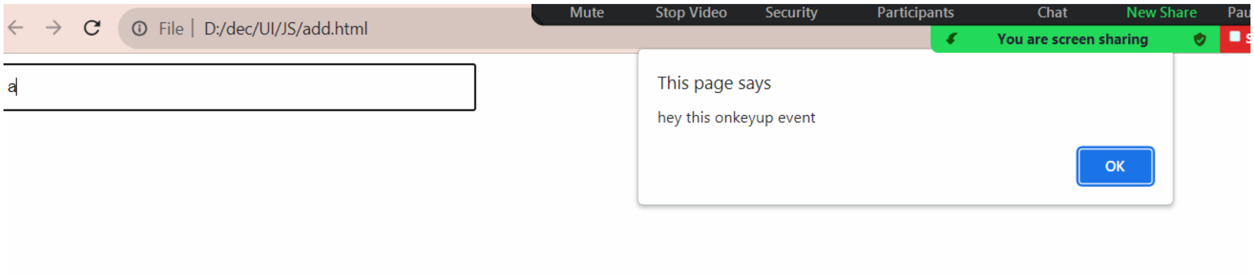
This page says

hey this onload event

Source code

```
<html>
 <head>
   <title>demo js function</title>
   <script type='text/javascript'>
      function show(){
                 alert("hey this onclick event");
              }
   </script>
 </head>
 <body onload="show()">
  <input type="button" name="s" value="Show Alert Box"
  style='width:400px;height:40px;' onmouseover="show()"/>
 </body>
</html>
```

onkeyup: this event fire when user press key in textbox and leave it.

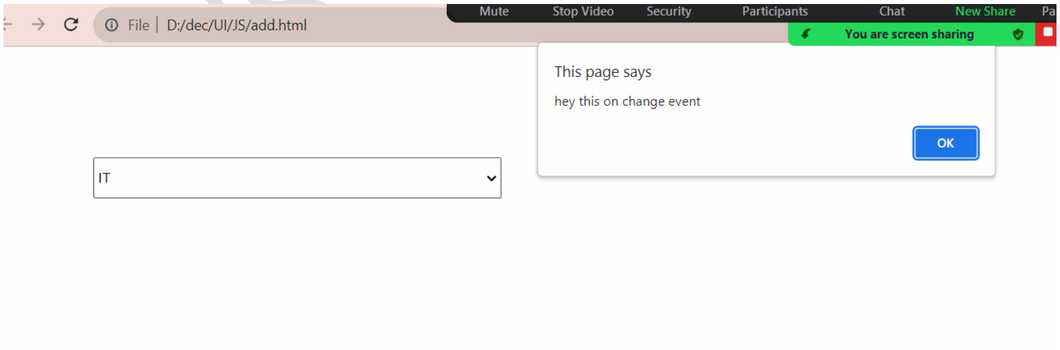


This page says

hey this onkeyup event

```
<html>
 <head>
   <title>demo js function</title>
   <script type='text/javascript'>
      function show(){
                 alert("hey this onkeyup event");
               }
   </script>
 </head>
 <body >

  <input type="text" name="s" value=" "
  style='width:400px;height:40px;' onkeyup="show()"/>
 </body>
</html>
```

onchange: this event normally we use on dropdown list means when user select a value or change value from a dropdown then onchange event get executed.



This page says

hey this on change event

Source code

```
<html>
 <head>
```

```
  <title>demo js function</title>
  <script type='text/javascript'>
    function show(){
            alert("hey this on change event");
           }
  </script>

  <style>
    select{
             margin-left:100px;
                    margin-top:100px;
                    width:400px;
                    height:40px;
           }
  </style>
 </head>
 <body >

 <select onchange="show()">
   <option>CSE</option>
          <option>IT</option>
          <option>MECH</option>
          <option>ETC</option>
 </select>

 </body>
</html>
```

DOM: DOM stands for document object model

Q. Why use DOM or what is use of DOM?

---

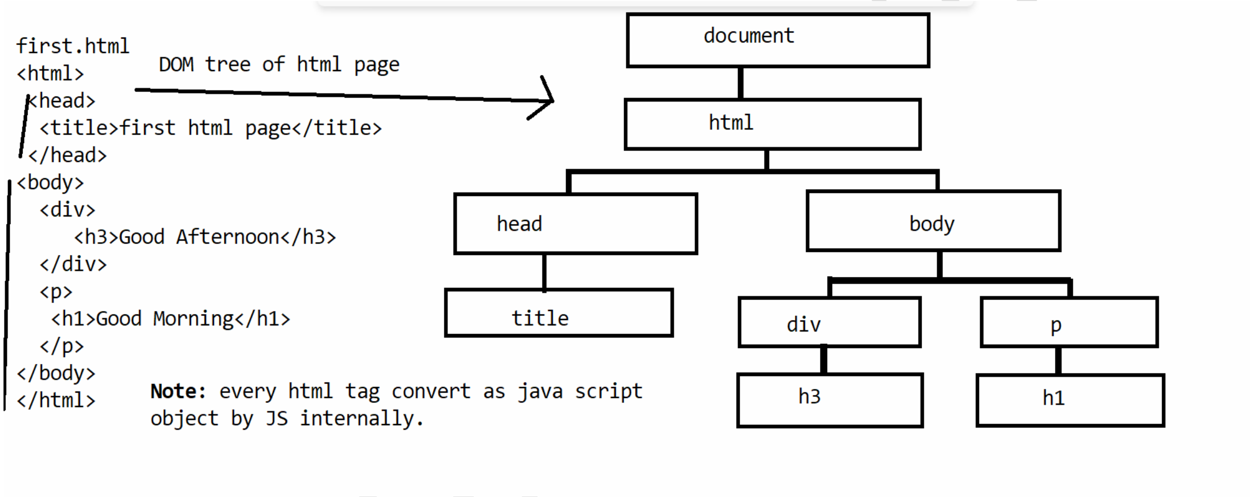
1. DOM is used for generate HTML element at run time by using a JavaScript.
2. DOM is used for apply run time CSS on HTML page
3. DOM helps us to remove html element at run time

4. DOM is used for update html element values at run time.
etc

If we want to work with DOM practically by using JavaScript we have document object and document is parent of all java script objects and if we any html element in the case of JavaScript it is internally consider as object.

If we think about html page using a JavaScript it is internally create its DOM Tree.

DOM Tree for html page shown in diagram.



```
first.html
<html>
 <head>
  <title>first html page</title>
 </head>
<body>
  <div>
     <h3>Good Afternoon</h3>
  </div>
  <p>
   <h1>Good Morning</h1>
  </p>
</body>
</html>
```

Note: every html tag convert as java script object by JS internally.

if we want to work with DOM in JavaScript we have some inbuilt methods provided by document object to us

Methods of document object

1. write(string): this method is used for display content on web page using java script.

Syntax:

document. write("good morning");

2. getElementById(): this method is used for access html element in java script by using its id.

Syntax:

let variablename = document.getElementById("idname") means this method access html element in java script using its id and create its object internally in java script

3. getElementByName(): this method can access html element by using name attribute
Syntax:
let variablename=document.getElementByName("html element name")

4. getElementByClassNames(): this method can access html element by using its class
Syntax: let variable=document.getElementByClassName("classname");

5. createElement(): this method is used for create new html element using javascript.
Syntax:
let variable =document.createElement("html element name");

6. append(): this method is used for add new html element under body tag or some other html element created by JavaScript at run time.

Syntax: parentobject.append(childobject)

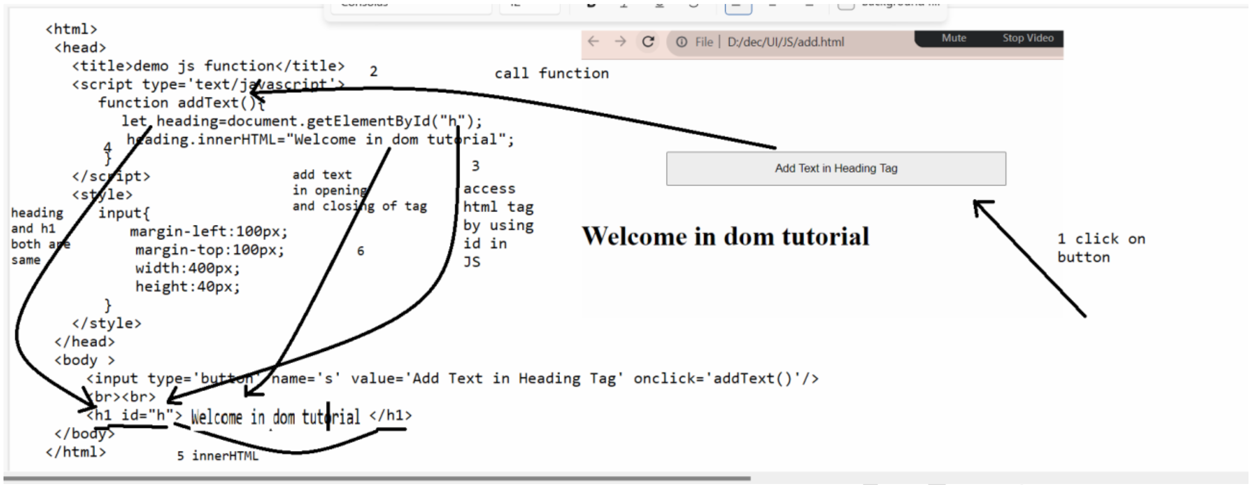
Properties of HTML element

innerHTML="text": this property help us to add new text in html tag or fetch html tag data in java script at run time.

value : this property help us to access html input control values or add value at run time in input contro

Object of document:
sytle : this object help us apply CSS at run time on html element using java script.

Example using DOM
we want to design on button and heading tag on web page and when user click on button then at run time we want to add message in heading i.e welcome in dom tutorial.



```
<html>
 <head>
   <title>demo js function</title>
   <script type='text/javascript'>
      function addText(){
        let heading=document.getElementById("h");
         heading.innerHTML="Welcome in dom tutorial";
       }
   </script>
   <style>
      input{
         margin-left:100px;
          margin-top:100px;
          width:400px;
          height:40px;
      }
   </style>
 </head>
 <body >
    <input type='button' name='s' value='Add Text in Heading Tag' onclick='addText()'/>
    <br><br>
    <h1 id="h"> Welcome in dom tutorial </h1>
 </body>
</html>
```

File | D:/dec/UI/JS/add.html
Mute
Stop Video
2
call function
4
3
access html tag by using id in JS
add text in opening and closing of tag
heading and h1 both are same
6
5 innerHTML
Add Text in Heading Tag
Welcome in dom tutorial
1 click on button

Source code

```
<html>
 <head>
   <title>demo js function</title>
   <script type='text/javascript'>
      function addText(){
                 let heading=document.getElementById("h");
                          heading.innerHTML="Welcome in dom tutorial";



            }
   </script>

   <style>
      input{
                margin-left:100px;
                        margin-top:100px;
                        width:400px;
                        height:40px;
            }
   </style>
 </head>
 <body >
```

```
        <input type='button' name='s' value='Add Text in Heading Tag' onclick='addText()'/>
        <br><br>
        <h1 id="h">              </h1>

 </body>
</html>
```

Example: WAP JS program create one button and generate paragraph tag at run time as well as apply run time CSS on paragraph

Example: WAP to create Array in JS and Remove duplicate element from array without using any inbuilt function

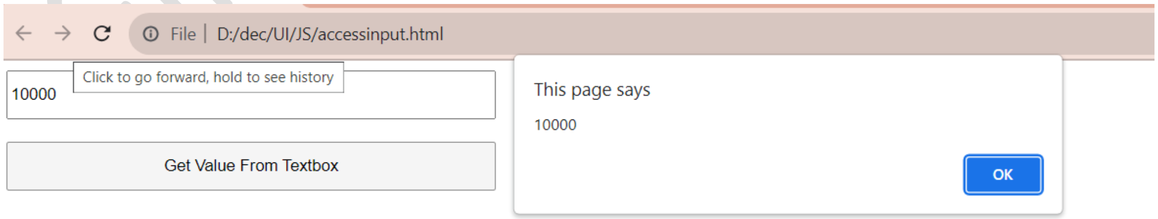
Example: WAP to create array in JS and insert value on specified index in array without using any inbuilt function

Example: WAP to create two array in java script and merge them in third array without using concat() function

How to access input control values using DOM in JavaScript

If we want to access input control value using DOM we have value attribute

Example: we want to create one text box and one button on web page and when user clicks on web page then we want to display textbox value in alert box.

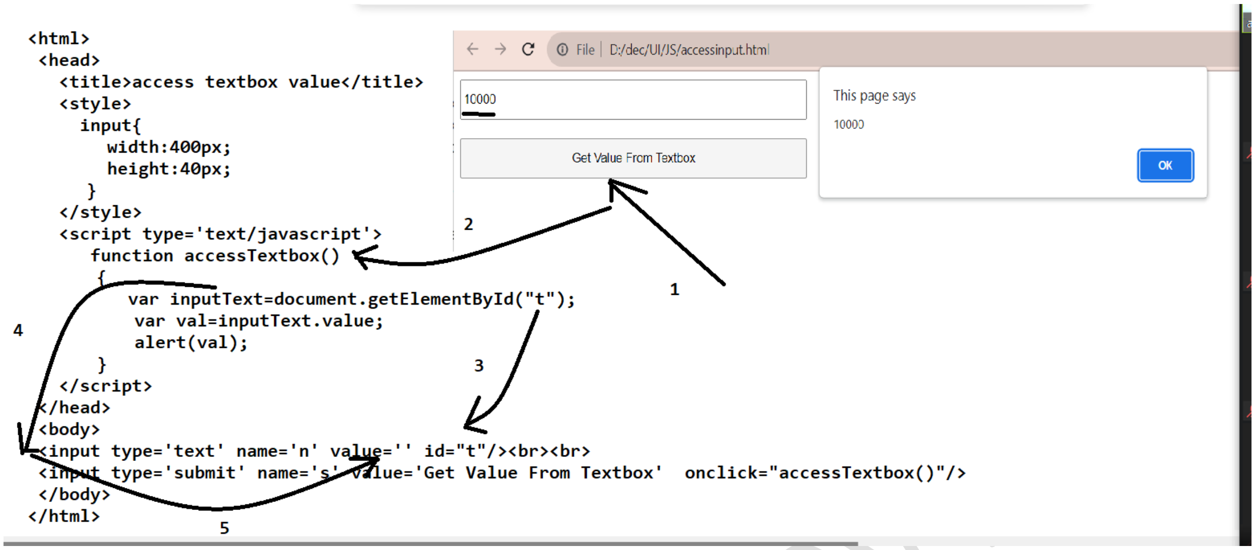


File | D:/dec/UI/JS/accessinput.html

10000

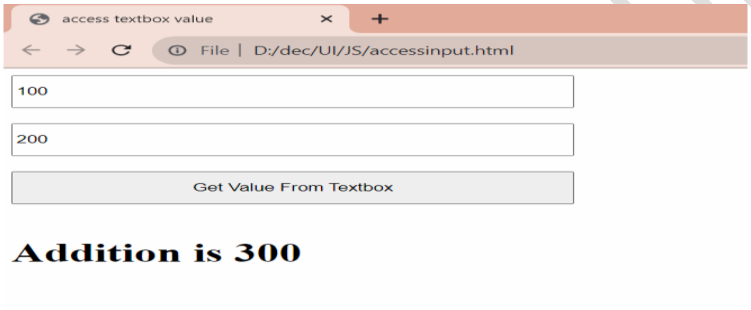
This page says

10000



```
<html>
 <head>
   <title>access textbox value</title>
   <style>
     input{
        width:400px;
        height:40px;
     }
   </style>
   <script type='text/javascript'>
      function accessTextbox()
      {
         var inputText=document.getElementById("t");
          var val=inputText.value;
          alert(val);
      }
   </script>
 </head>
 <body>
 <input type='text' name='n' value='' id="t"/><br><br>
 <input type='submit' name='s' value='Get Value From Textbox'  onclick="accessTextbox()"/>
 </body>
</html>
```

Example: we want to create web page with two textboxes and one button and when we input value in two different textboxes and when user click on button then we want to calculate addition of two values.

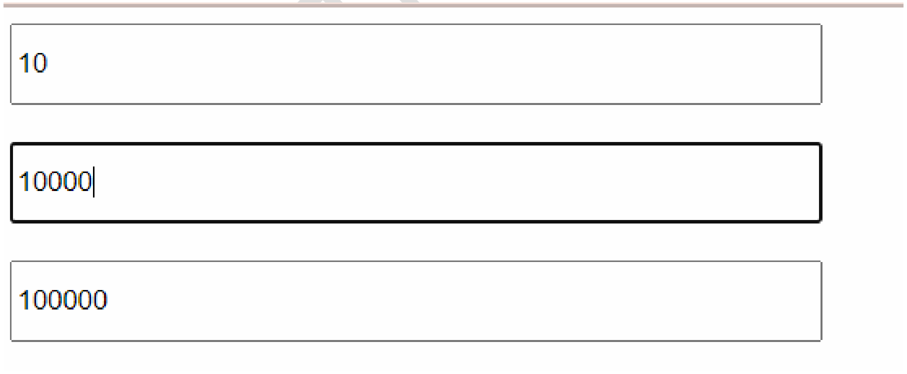


Addition is 300

Source code

```
<html>
 <head>
   <title>access textbox value</title>
   <style>
     input{
           width:400px;
           height:40px;
          }
   </style>
   <script type='text/javascript'>
     function accessTextbox()
          {
                var inputText=document.getElementById("first");
                        var val=inputText.value;
                        var inputText1=document.getElementById("second");
                        var val1=inputText1.value;
                        var result=parseInt(val)+parseInt(val1);
                        document.getElementById("h").innerHTML="Addition is "+result;
              }
  </script>
 </head>
 <body>
 <input type='text' name='n' value='' id='first' /><br><br>
 <input type='text' name='n' value='' id='second' /><br><br>

 <input type='submit' name='s' value='Get Value From Textbox'
 onclick="accessTextbox()"/>
  <br><br>
 <h1 id="h"></h1>
 </body>
</html>
```

Example: we want to create web page with three textboxes first textbox is used for input quantity and second textbox is used for input rate and calculate its total and show in third textbox.



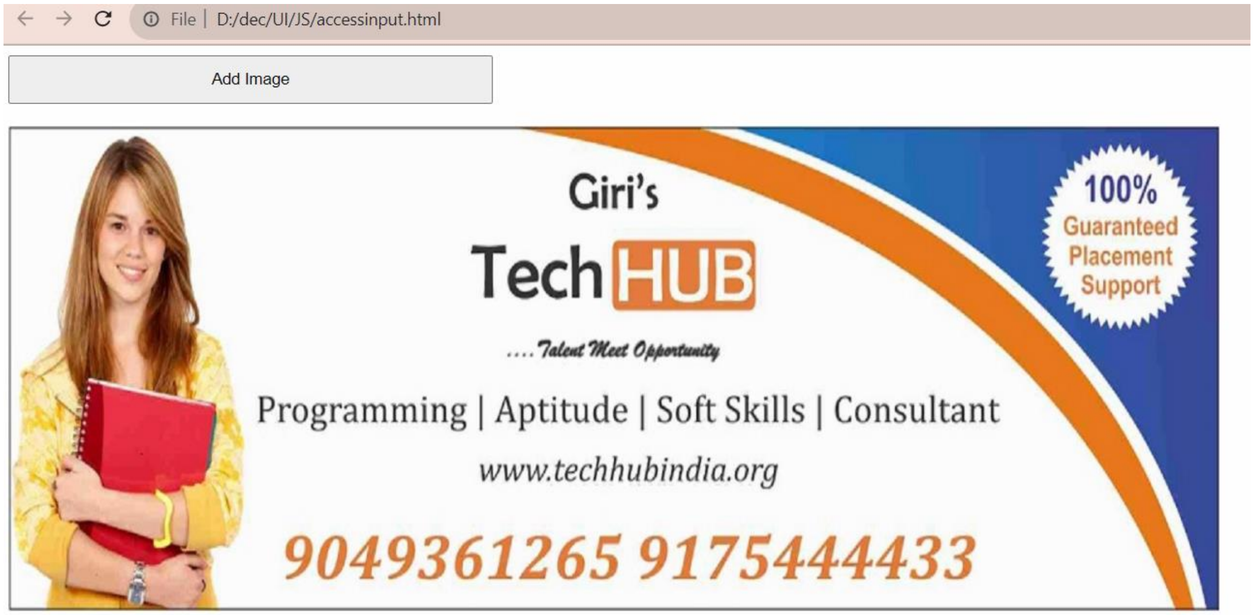
10

10000

100000

```
<html>
 <head>
   <title>access textbox value</title>
   <style>
   input{
           width:400px;
           height:40px;
          }
  </style>
  <script type='text/javascript'>
    function accessTextbox()
          {
            var val=document.getElementById("first").value;
                   var val1=document.getElementById("second").value;
                   var result=parseInt(val)*parseInt(val1);
      document.getElementById("r").value=""+result;

}
  </script>
</head>
<body>
<input type='text' name='n' value='' id='first' /><br><br>
<input type='text' name='n1' value='' id='second' onkeyup="accessTextbox()"/><br><br>
<input type='text' name='n2' value='' id='r' /><br><br>


 <br><br>
<h1 id="h"></h1>
</body>
</html>
```

Example: create one button and when user click on button then add image at run time on web page.


File | D:/dec/UI/JS/accessinput.html
Add Image
Giri's
Tech HUB
....Talent Meet Opportunity
Programming | Aptitude | Soft Skills | Consultant
www.techhubindia.org
9049361265 9175444433
100% Guaranteed Placement Support

Source code

```
<html>
 <head>
   <title>access textbox value</title>
   <style>
     input{
               width:400px;
               height:40px;
             }
   </style>
   <script type='text/javascript'>
      function applyImage()
             {
               let image=document.getElementById("image");
                        image.src="banner.jpg";
                        image.width="1000";
      }
   </script>
 </head>
 <body>
  <input type='button' name='b' value='Add Image' onclick="applyImage()" />
  <br><br>
  <img src='' width='400px' height='400px' alt='image not found' id='image'/>
  </body>
</html>
```

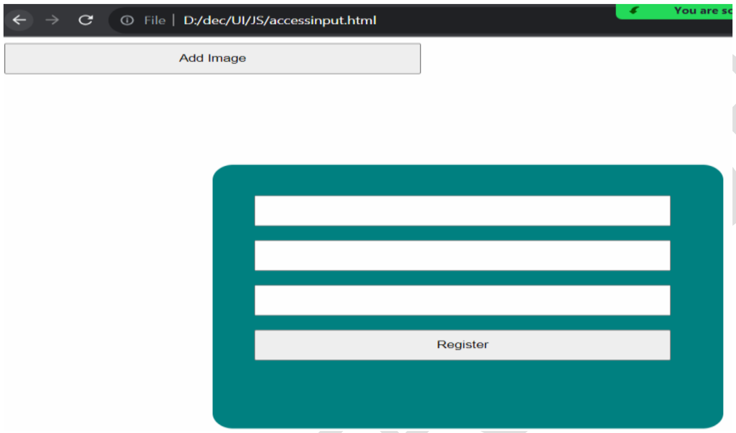
How to create HTML form using DOM in JavaScript

Now we want to design registration page using JavaScript.

We want to give one button on web page and when user clicks on button then we want to display registration form on web page.

If we want to generate element at run time using JavaScript we have createElement() method of JS.

as well as here we want to use appen() method for add element under another age.



Source code

```
      <html>
       <head>
         <title>access textbox value</title>
         <style>
               input{
                 width:400px;
                 height:40px;
               }
               div{
                 width:32%;
                 height:38vh;
               }
```

```
        </style>
        <script type='text/javascript'>
                function generateForm()
                {
                 let parent=document.getElementById("d");
                        parent.style.backgroundColor="teal";
                        parent.style.padding="40px";
                        parent.style.position="relative";
                        parent.style.left="200px";
                        parent.style.top="100px";
                        parent.style.borderRadius="20px";

                        let nameTextBox=document.createElement("input");
                         nameTextBox.setAttribute("type","text");
                         nameTextBox.setAttribute("name","name");
                           parent.append(nameTextBox);



                         let br=document.createElement("br");
                         parent.append(br);
                          br=document.createElement("br");
                          parent.append(br);

                         let emailTextBox=document.createElement("input");
                         emailTextBox.setAttribute("type","text");
                         emailTextBox.setAttribute("name","name");
                           parent.append(emailTextBox);

                          br=document.createElement("br");
                         parent.append(br);
                          br=document.createElement("br");
                          parent.append(br);

                         let contactTextBox=document.createElement("input");
                         contactTextBox.setAttribute("type","text");
```

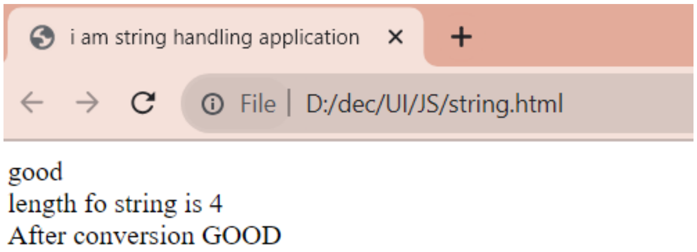
```
                          contactTextBox.setAttribute("name","name");
                            parent.append(contactTextBox);
                          br=document.createElement("br");

                          parent.append(br);
                           br=document.createElement("br");
                            parent.append(br);

                            let btn=document.createElement("input");
                            btn.setAttribute("type","button");
                            btn.setAttribute("value","Register");

                                     parent.append(btn);
                          br=document.createElement("br");

                          parent.append(br);
                           br=document.createElement("br");
                            parent.append(br);
              }
   </script>
 </head>
 <body>
  <input type='button' name='b' value='Add Image' onclick="generateForm ()" />
            <br><br>
   <div id="d">

   </div>
  </body>
</html>
```

String Handling and Regular Expression with Validations

______

String is   immutable class in java script

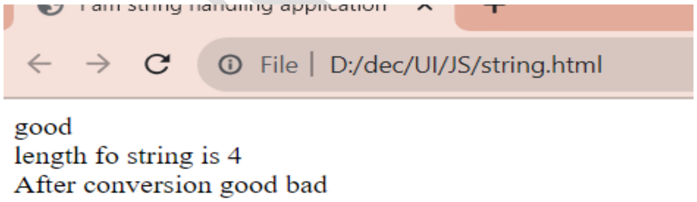
Immutable means once we initialize value never change later called as immutable.


i am string handling application
File | D:/dec/UI/JS/string.html

```
good
length fo string is 4
After conversion GOOD
```

```
<html>
 <head>
   <title>i am string handling application</title>
   <script type='text/javascript'>
       let s="good";
              document.write(s);
              let l=s.length ;
              document.write("<br>length fo string is  "+l);
              let s1=s.toUpperCase();
              document.write("<br>After conversion  "+s1);
   </script>
 </head>
 <body>

 </body>
</html>
```

Example:


i am string handling application
File | D:/dec/UI/JS/string.html

```
good
length fo string is 4
After conversion good bad
```

Source code

```
<html>
 <head>
   <title>i am string handling application</title>
```

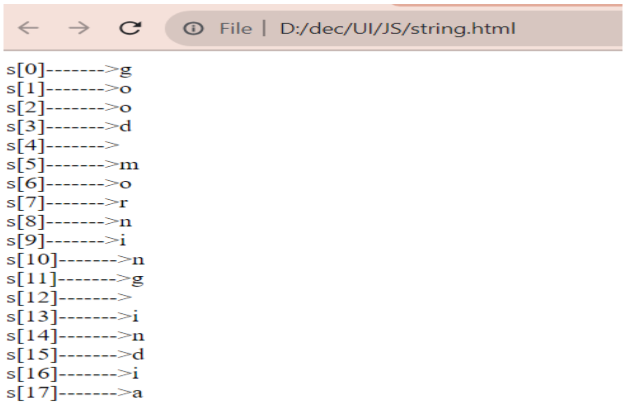
```
  <script type='text/javascript'>
      let s="good";
              let s1=" bad";
              document.write(s);
              let l=s.length ;
              document.write("<br>length fo string is  "+l);
               let s2=s.concat(s1);;
              document.write("<br>After conversion  "+s2);
  </script>
 </head>
 <body>
 </body>
</html>
```

trim(): remove white spaces at beginning or ending of string

```
<html>
 <head>
  <title>i am string handling application</title>
  <script type='text/javascript'>
      let s="            good";
              let s1=s.trim();
              console.log(s1);

  </script>
 </head>
 <body>

 </body>
</html>
```

Example:

charAt(int index): this method is used for return character from string using its index.

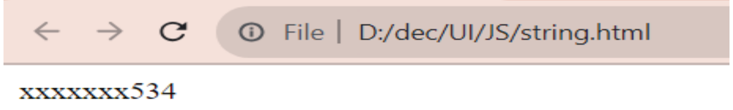


```
s[0]------->g
s[1]------->o
s[2]------->o
s[3]------->d
s[4]------->
s[5]------->m
s[6]------->o
s[7]------->r
s[8]------->n
s[9]------->i
s[10]------->n
s[11]------->g
s[12]------->
s[13]------->i
s[14]------->n
s[15]------->d
s[16]------->i
s[17]------->a
```

Source code

```
<html>
 <head>
   <title>i am string handling application</title>
   <script type='text/javascript'>
       let s="good morning india";
               for(var i=0; i<s.length; i++)
               {
                 document.write("s["+i+"]------->"+s.charAt(i)+"<br>");
               }
   </script>
 </head>
 <body>
 </body>
</html>
```

padStart(): adding some data at starting of the string



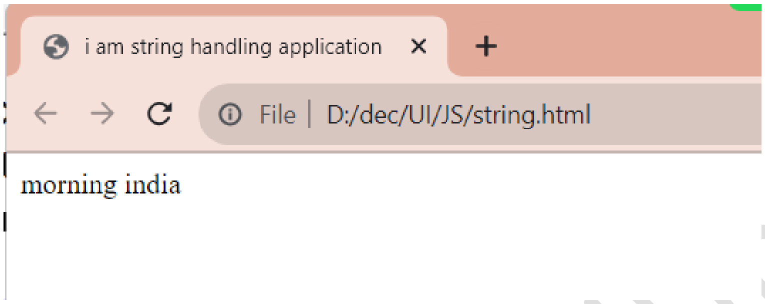
```
xxxxxxx534
```

```
<html>
 <head>
   <title>i am string handling application</title>
   <script type='text/javascript'>
                     let s="534";
```

```
                    let s1=s.padStart(10,'x');
                    document.write(s1);
  </script>
 </head>
 <body>
 </body>
</html>
```

slice(): it is used for splice or separate string using a some specified index.

Syntax: slice(index): split the string using a some specified index.

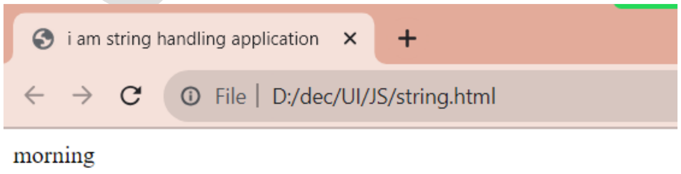


morning india

Source code

```
<html>
 <head>
  <title>i am string handling application</title>
  <script type='text/javascript'>
      let s="good morning india";
                    let s1=s.slice(5);
                    document.write(s1);
  </script>
 </head>
 <body>
 </body>
</html>
```

Example of slice()



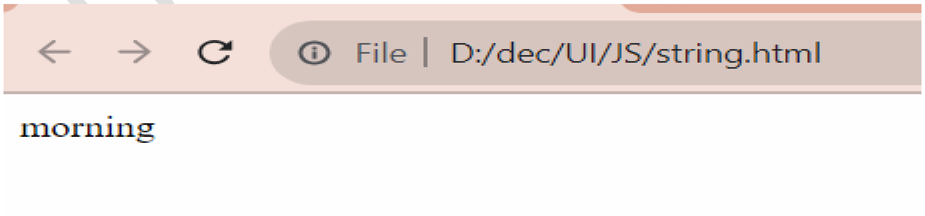
morning

```
<html>
 <head>
  <title>i am string handling application</title>
  <script type='text/javascript'>
      let s="good morning india";
                   let s1=s.slice(5,12);
                   document.write(s1);
  </script>
 </head>
 <body>

 </body>
</html>
```

charCodeAt(index): this method is used for return ascii code using a some specified index.

```
<html>
 <head>
  <title>i am string handling application</title>
  <script type='text/javascript'>
      let s="good morning india";
                   let ascii=s.charCodeAt(0);
                   document.write(ascii);
  </script>
 </head>
 <body>

 </body>
</html>
```

substr(): this method is used for perform substring operation with a string.



```
morning
```

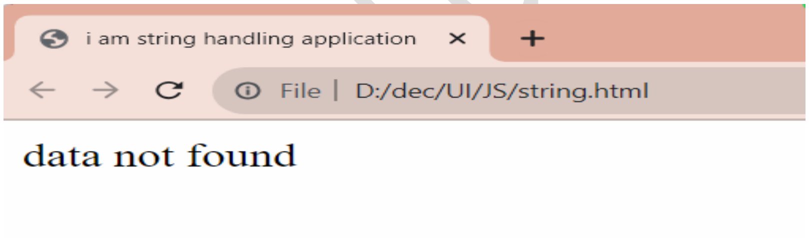
substr(startindex): extract data from index to complete length.

Syntax: substr(startindex,lengthofcharacter):

```
<html>
 <head>
   <title>i am string handling application</title>
   <script type='text/javascript'>
        let s="good morning india";
                    let s1=s.substr(5,7);
                    document.write(s1);
   </script>
 </head>
 <body>

 </body>
</html>
```

String searching method :

indexOf(data): this method is used for search data from a string and if data found return its index
otherwise return -1


i am string handling application
File | D:/dec/UI/JS/string.html

data not found

Source code

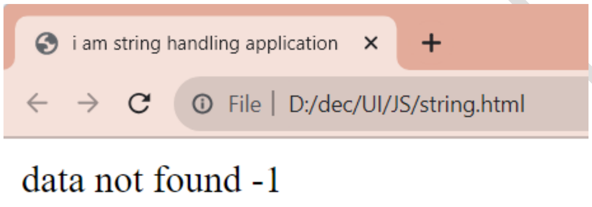
```
<html>
 <head>
   <title>i am string handling application</title>
   <script type='text/javascript'>
        let s="good morning india";
                    let index=s.indexOf("rrorning");
                     if(index!=-1)
                    { document.write("Data found");
                    }
                    else{
                      document.write("data  not found");
                    }
  </script>
 </head>
 <body>
 </body>
</html>
```

search(): this method can search data using a string and return its index and data found return -1 when data not found.

search method can work with string as well as can work regular expression but indexof not work with regular expression

let index=stringobject.search(data);

Example:

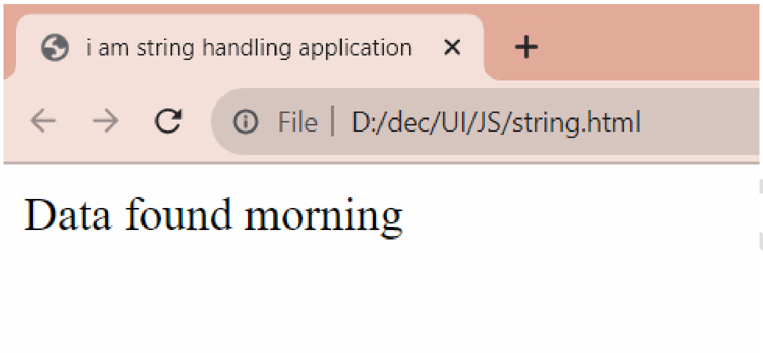

i am string handling application
File | D:/dec/UI/JS/string.html

data not found -1

Source code

```
<html>
 <head>
  <title>i am string handling application</title>
  <script type='text/javascript'>
      let s="good morning india";
                   let index=s.search("mosfarning");
                    if(index!=-1)
                    { document.write("Data found "+index);
                    }
                    else{
                         document.write("data  not found "+index);
                        }
  </script>
 </head>
 <body>

 </body>
</html>
```

match: match method is used for search data from a string and if data found return data otherwise return null.


i am string handling application
File | D:/dec/UI/JS/string.html

Data found morning

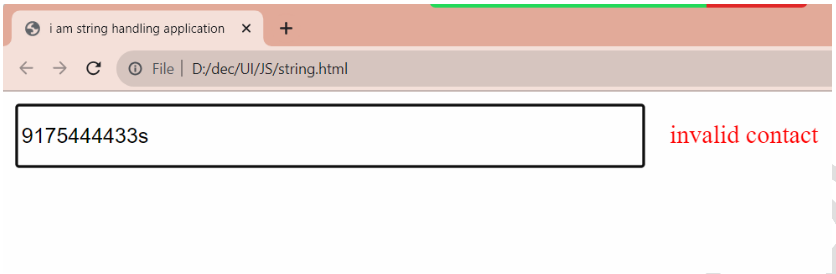
Source code

```
<html>
 <head>
  <title>i am string handling application</title>
  <script type='text/javascript'>
      let s="good morning india";
                        let index=s.match("morning");
                        if(index!=null)
                        { document.write("Data found "+index);
                        }
                        else{
                          document.write("data  not found "+index);
                        }
  </script>
 </head>
 <body>
 </body>
```

</html>

Note: using a string handling we can perform validation using a Javascript

Example: we want to create program to create textbox and input value in it but textbox should accept
only numbers not alphabet or special symbol means we can perform mobile number validation means number length should 10 and allow only digit in textbox


i am string handling application
File | D:/dec/UI/JS/string.html

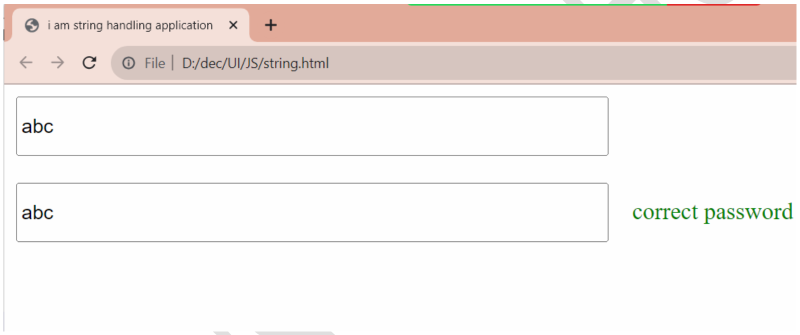
9175444433s

invalid contact

Source code

```
<html>
 <head>
   <title>i am string handling application</title>
   <script type='text/javascript'>

                function validContact(str)
                {   let flag=false;
                  for(var i=0; i<str.length;i++)
                        {
                          if(!((str.charCodeAt(i)>=48&&str.charCodeAt(i)<=57)&&
str.length==10))
                                { flag=true;
                                 break;
                                }
                        }
                        if(flag){
                  document.getElementById("s").innerHTML="invalid contact";
                  document.getElementById("s").style.color="red";
                  }
                 else{
                        document.getElementById("s").innerHTML="";
                       }
                     }

   </script>
 </head>
 <body>
   <input type='text' name='name' id='c' value='' style='width:400px;height:40px;'
   onkeyup="validContact(this.value)"/> &nbsp;&nbsp;
   <span id="s"></span>
 </body>
</html>
```

Example: WAP JS Program to create two textbox first password and second is confirm password and when user input in password textbox and confirm password textbox then if password and confirm password then show message to use password is correct otherwise password is incorrect


i am string handling application
File | D:/dec/UI/JS/string.html

abc

abc

correct password

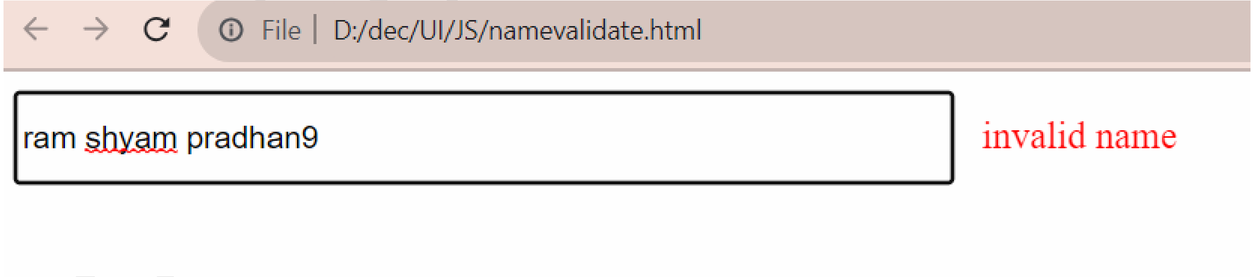
```
<html>
 <head>
   <title>i am string handling application</title>
   <script type='text/javascript'>

                     function validContact(str)
                     {    let str1=document.getElementById("c").value;
                        if(str===str1)
                        {
                          document.getElementById("s").innerHTML="correct password";
                                 document.getElementById("s").style.color="green";
```

```
                    }
                    else{
                      document.getElementById("s").innerHTML="In correct password";
                            document.getElementById("s").style.color="red";

                    }

                  }

  </script>
 </head>
 <body>
  <input type='text' name='name' id='c' value='' style='width:400px;height:40px;'
   /> &nbsp;&nbsp;<br><br>
   <input type='text' name='name' id='c1' value='' style='width:400px;height:40px;'
  onkeyup="validContact(this.value)"/> &nbsp;&nbsp;
  <span id="s"></span>
 </body>
</html>
```

Name Validation

---

1. Not allow special symbol except space
2. digit not allowed in name validation



ram shyam pradhan9

invalid name

Source code

```
<html>
 <head>
 <title>name validation</title>
<style>
 input{
    width:400px;
    height:40px;
   }
</style>
<script type='text/javascript'>
   function validateName(str)
   {  let len=str.length;
      let flag=false;
      for(var i=0; i<len;i++)
      { let ascii=str.charCodeAt(i);
        if(!((ascii>=65 && ascii<=90) ||(ascii>=97&&ascii<=122)||ascii==32)){
            flag=true;
            break;
        }
      }
      if(flag)
      {document.getElementById("s").innerHTML="invalid name";
       document.getElementById("s").style.color="red";
      }
     else{
document.getElementById("s").innerHTML="";
       document.getElementById("s").style.color="red";
      }
   }
</script>
  </head>
<body>
<input type='text' name='name'  value='' onkeyup="validateName(this.value)"/>
 &nbsp;&nbsp;<span id="s"></span>
</body>
</html>
```

Email validation

____

1) Must be present @

2) @ should not occur more than one time

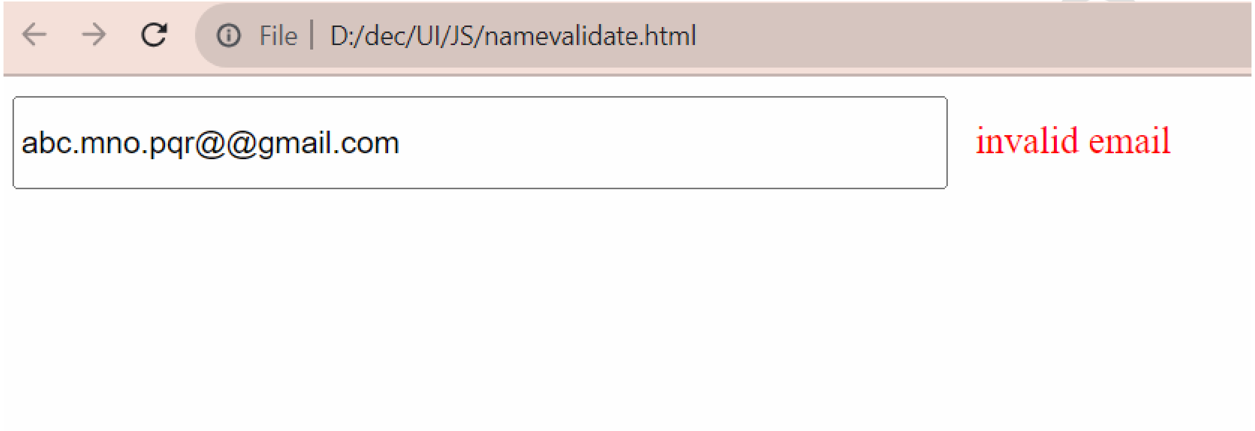
3) Minimum single. Must be present

4) After dot there should 3 or 2 letters at end of email

5) There is possibility dot may be present anywhere in email

6) Capital letter should not allow in email

7) @ should not come at first position



abc.mno.pqr@@gmail.com

invalid email

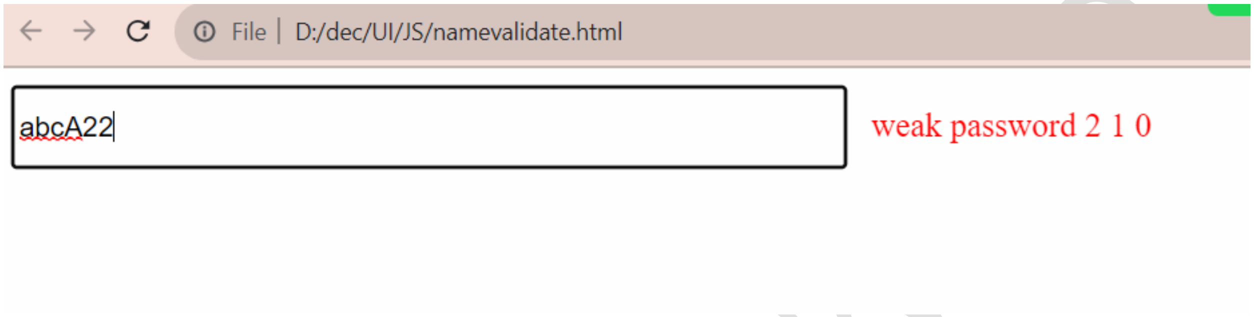
Source code

```
<html>
 <head>
 <title>name validation</title>
<style>
 input{
  width:400px;
  height:40px;
  }
</style>
<script type='text/javascript'>

  function validateEmail(str)
  {  let count=0;
   let span=document.getElementById("s");
   let index=str.indexOf("@");
   if(index<=0)
   {
    span.innerHTML="invalid email";
    span.style.color="red";
```

```
     }
    else{
       for(var i=0;i<str.length;i++)
       {   if(str.charAt(i)==='@')
           {  ++count;
           }
       }
      let newstring=str.substring(index,str.length);
       let index1=newstring.indexOf(".");

       if(index1==-1 || ((index1!=newstring.length-4) &&(index1!=newstring.length-3)))
         {span.innerHTML="invalid email";
          span.style.color="red";

         }
      else if(count>1)
      {span.innerHTML="invalid email  ";
       span.style.color="red";
      }
     else{

           span.innerHTML="";
       span.style.color="red";
      }

    }

  }
</script>
  </head>
<body>
<input type='text' name='name'  value='' onkeyup="validateEmail(this.value)"/>
 &nbsp;&nbsp;<span id="s"></span>
</body>
</html>
```

Strong password validation

1. Minimum password contain 8 letters
2. Minimum single special Symbol Required
3. Minimum 2 digit required
4. Minimum single character must be capital.



abcA22

weak password 2 1 0

```
<html>
  <head>
  <title>name validation</title>
<style>
  input{
    width:400px;
    height:40px;
   }
</style>
<script type='text/javascript'>
   function strongPasswordValidation(str)
   { let dc=0,cl=0,sp=0;
     let span=document.getElementById("s");
     for(var i=0; i<str.length;i++)
     {
         if(str.charCodeAt(i)>=48 && str.charCodeAt(i)<=57)
              {++dc;
              }
              if(str.charCodeAt(i)>=65 && str.charCodeAt(i)<=90)
              {  ++cl;
              }
```

```
            if(!((str.charCodeAt(i)>=48 && str.charCodeAt(i)<=57) ||(str.charCodeAt(i)>=65 &&
str.charCodeAt(i)<=90) || (str.charCodeAt(i)>=97 && str.charCodeAt(i)<=122)))
        {++sp;
        }
    }

    if(str.length<8 || dc<2  || cl<1 ||sp<1 )
    {  span.innerHTML="weak password "+ dc +"  "+ cl +"   "+sp;
            span.style.color="red";
    }
    else{
           span.innerHTML="";
            span.style.color="red";
           }
  }
</script>
  </head>
<body>
<input type='text' name='name'  value='' onkeyup="strongPasswordValidation(this.value)"/>
 &nbsp;&nbsp;<span id="s"></span>
</body>
</html>
```

AJAX

Q. what is AJAX?

AJAX is asynchronous java script and XML and it is not a language it is technology which work using java script or its dependent framework.

It developed by Google in 2005

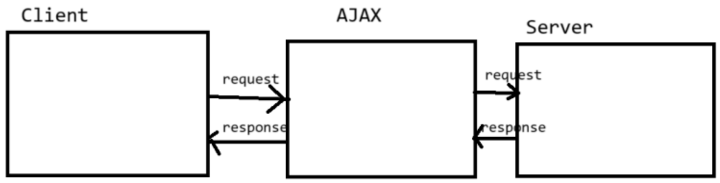
Q. why use AJAX in project and what is goal of AJAX?

---

1. Update a web page without reloading the page means performs partial updation.
2. Request data from a server – after the page has loaded.
3. Receive data from a server after the page has loaded.
4. Send data to server in background.

Note: when we send request to server to get response from a server without AJAX then client send direct request to server and get response server there is possibility direct communication with a server is responsible for refresh or reload the complete web age client side so it may hamper the perform of application when we have request to send large number of request to server as well as get response from a server.

When we use AJAX we have AJAX engine between client and server when client send request to AJAX engine and AJAX engine send request in background to server and server process on request send by AJAX engine and send response to AJAX engine then AJAX engine update the response on specified portion on web page without reloading a client page shown in following diagram.



How to implement AJAX Practically

1. Design a Function in JavaScript

```
 function callAjax(){

 }
```

2. Create object of XMLHttpRequest class

XMLHttpRequest is inbuilt class from java script which is used for accept request from client send to server and get response from a response and update on client page means we can say this class work as AJAX Engine practically.

```
function callAjax(){
  let xhttp=new XMLHttpRequest();

 }
```

3. Call its onreadystatechange method

Normally we use this method to cross verify request send by client to server properly or not as well as get response from a server and update on web page where we want.

If we want to identity request send by client to server as well as identity response send by client and its status then we have some fixed constant values provided XMLHttpRequest object.

Constant	Values
readyState	Holds the status of the XMLHttpRequest. 0: request not initialized 1: server connection established 2: request received 3: processing request 4: request finished and response is ready
responseText	Return response data send by server in the form string
Status	Returns the status-number of a request 200: "OK" 403: "Forbidden" 404: "Not Found"

Note: normally when we use onreadystatechange method of XMLHttpRequest then we cross verify request finished and server response status.

```
function callAjax(){
  let xhttp=new XMLHttpRequest();
  xhttp.onreadystatechange=function(){
        if(this.readyState==4 && this.status==200)
            {  request send send successfully and response get property from a server
            }
 }
 }
```

3) Call the open() method of XMLHttpRequest object

open() method is a member of XMLHttpRequest and this method decide how to send request to server as well as where request should be send and status of the request.

Syntax: open("GET/POST","URL",true|false);

GET and POST decide how send request to server via AJAX
URL here we need to specify destination URL page where we want to send request
if we set true as third parameter then you request send to asynchronously in background otherwise send synchronously

```
function callAjax(){
   let xhttp=new XMLHttpRequest();
   xhttp.onreadystatechange=function(){
         if(this.readyState==4 && this.status==200)
             {  request send send successfully and response get property from a server
             }
  }
  xhttp.open("GET","searchCat",true);
  }
```

4. Call send () method for send request
There are two version of send method
send(): this method normally when we use get method in open() method
send(string): this method normally use when we use post method In open() method

```
function callAjax(){
   let xhttp=new XMLHttpRequest();
   xhttp.onreadystatechange=function(){
         if(this.readyState==4 && this.status==200)
             {  request send send successfully and response get property from a server
             }
  }
  xhttp.open("GET","searchCat",true);
xhttp.send();
  }
```

Example:
testajax.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
<script type='text/javascript'>
 function ajaxCall(){
         var xhttp=new XMLHttpRequest();
        xhttp.onreadystatechange=function(){
            if(this.readyState==4 && this.status==200){
                document.getElementById("h").innerHTML=this.responseText;
            }
        }
        xhttp.open("GET","test123",true);
        xhttp.send();
 }
</script>
</head>
<body>
  <input type='button' name='s' value="Send Request" onclick="ajaxCall()"></input>
 <br><br>
 <h1 id="h"></h1>
</body>
</html>
```

Servlet

```
package org.techhub.controller;

import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

@WebServlet("/test123")
public class TestAjaxCall extends HttpServlet {
	protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
			response.setContentType("text/html");
			PrintWriter out=response.getWriter();
			out.println("Hey i am sending response");
	}

	/**
	 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
	 */
	protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
		// TODO Auto-generated method stub
		doGet(request, response);
	}
}
```

Regular Expression in JavaScript

Regular expression is object in java script which is used for pattern matching purpose. The pattern is used for searching and replacing characters in string.
JavaScript provide regular expression object to us to perform searching in string as well as Perform replacement in string as well as regular expression provide the some inbuilt properties and method to us for perform pattern matching.

How to work with regular expression objects

if we want to create regular expression object we have following syntax

let variable name= /expression/;
or
let variable name=new RegExp(/expression/);

Example: let n=/good/; //regular expression object if we want to work with regular expression object we have some modifiers

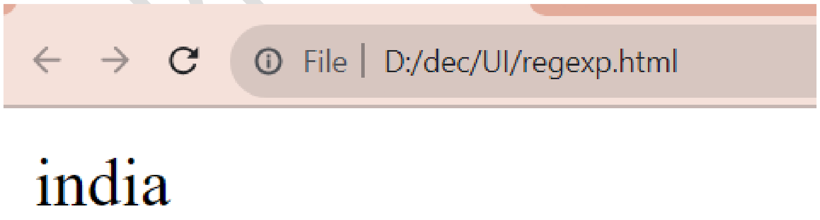
Modifier	Description
g	Perform a global match(find all matches rather than stopping after first match)
i	Perform case insensitive matching
m	Perform multiple matching

How we can use modifier with regular expression

____

let variablename=/expression/modifier

```
 <html>
<head>
 <title>regular expression example</title>
<script type='text/javascript'>
    let s="good morning india good evening india good night india";
    let reg=/india/;
    let result = s.match(reg);
  document.write(result);
</script>
</head>
<body>

</body>
</html>
```

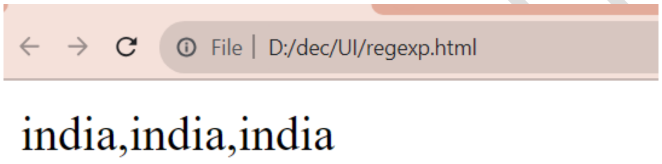


```
india
```

if we think about above output then we found india only once but india present three times in string means match method search data in string and if first match found the stop the matching if we want to search data globally in striing then we can use /g modifier with regular expression shown in following code

Source code

```
<html>
<head>
 <title>regular expression example</title>
<script type='text/javascript'>
    let s="good morning india good evening india good night india";
    let reg=/india/g;
    let result = s.match(reg);
  document.write(result);
</script>
</head>
<body>

</body>
</html>
```



```
india,india,india
```

Example of case insensitive match

```
<html>
<head>
 <title>regular expression example</title>
<script type='text/javascript'>
    let s="good morning india good evening india good night india";
    let reg=/INDIA/;
    let result = s.match(reg);
  document.write(result);
</script>
</head>
```

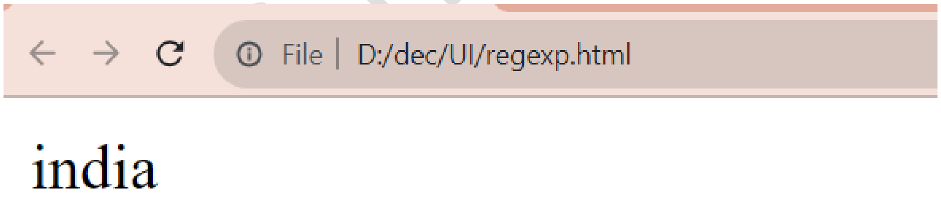
```
<body>
</body>
</html>
```

if we think about above code we get null result because we want to search capital INDIA in String but we have small india present in string but regular expression compare the case also but if we want to perform search without case checking then we have one more modifier i.e /i

Example:

```
<html>
<head>
 <title>regular expression example</title>
<script type='text/javascript'>
   let s="good morning india good evening india good night india";
   let reg=/INDIA/i;
   let result = s.match(reg);
  document.write(result);
</script>
</head>
<body>

</body>
</html>
```

Output:

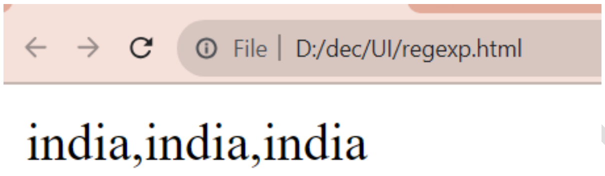


```
india
```

You can combine more than one modifier with regular expression means here we are going to combine ig modifier at time means we want to perform without case checking as well as perform global search at time shown in following code.

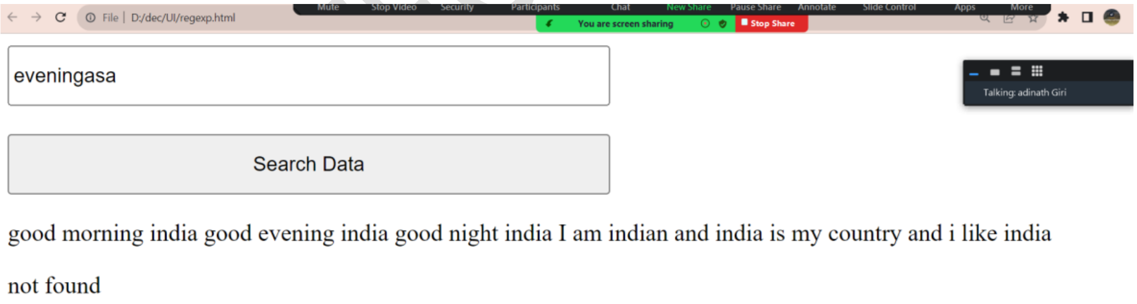
Source code

```
<html>
<head>
 <title>regular expression example</title>
<script type='text/javascript'>
    let s="good morning india good evening india good night india";
    let reg=/INDIA/ig;
    let result = s.match(reg);
  document.write(result);
</script>
</head>
<body>

</body>
</html>
```



```
india,india,india
```

Example search paragraph data



eveningasa

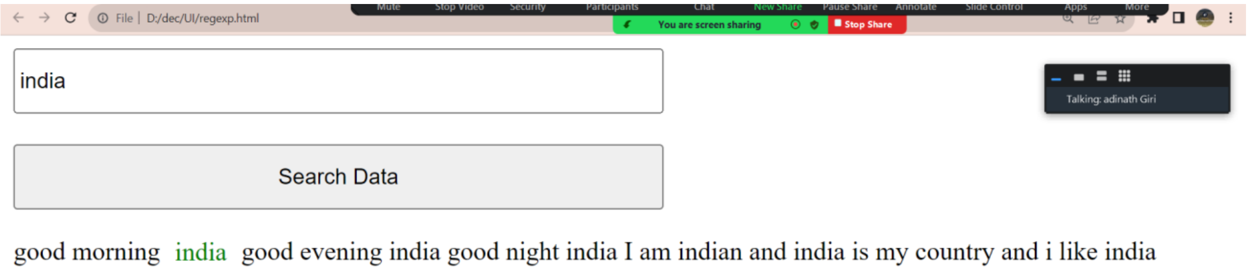
good morning india good evening india good night india I am indian and india is my country and i like india

not found

Source code

```
<html>
<head>
 <title>regular expression example</title>
<script type='text/javascript'>
    function search(){
      let s=document.getElementById("t").value;
      let s1=document.getElementById("p1").innerHTML;

      let reg=new RegExp(s,"i");

   let result = s1.match(reg);
       if(result!=null){
        document.getElementById("msg").innerHTML="found";
      }
      else{
           document.getElementById("msg").innerHTML="not found";

      }
    }

</script>
</head>
<body>
 <input type='text' name='n' id='t' value='' style='width:400px;height:40px;'
 /><br><br>
<input type='submit' name='s' value='Search Data' style='width:400px;height:40px;'
onclick="search()"  />
 <p id="p1">
     good morning india
     good evening india
     good night india
     I am indian and india is my country and i like india
 </p>
<span id='msg'></span>
</body>
</html>
```

Example



india

good morning india good evening india good night india I am indian and india is my country and i like india

Source code

```
<html>
<head>
 <title>regular expression example</title>
<script type='text/javascript'>
    function search(){
      let s=document.getElementById("t").value;
     let para=document.getElementById("p1");
      let s1=para.innerHTML;

      let reg=new RegExp(s,"i");

   let result = s1.search(reg);
        if(result!=-1)
        {
                let prevString=s1.substring(0,result);
                let nextString=s1.substring((prevString.length+s.length),s1.length);
                let newSpan=document.createElement("span");
           newSpan.style.color="green";
           newSpan.style.padding="5px";
           newSpan.innerHTML=s;

               let newSpan1=document.createElement("span");
               newSpan1.innerHTML=prevString;

          let newSpan2=document.createElement("span");
               newSpan2.innerHTML=nextString;
```

```
         para.innerHTML="";
        para.appendChild(newSpan1);
        para.appendChild(newSpan);
        para.appendChild(newSpan2);

       }
            else{
 document.getElementById("msg").innerHTML="Data Not Found";
      }
   }

</script>
</head>
<body>
 <input type='text' name='n' id='t' value='' style='width:400px;height:40px;'
 /><br><br>
<input type='submit' name='s' value='Search Data' style='width:400px;height:40px;'
onclick="search()"  />
 <p id="p1">good morning india
     good evening india
     good night india
     I am indian and india is my country and i like india
 </p>
<span id='msg'></span>
</body>
</html>
```

Brackets of Regular Expression

____

Brackets are helping us to find the range of characters

Expression	Description
[abc]	Find the any character between bracket
[^abc]	Find any character not between the brackets
[0-7]	Find any digit between bracket
[^0-9]	Find any character not between bracket

(x|y)	Find any alternative specified

Example:

```
<html>
 <head>
 <title>regular expression application</title>
<script type='text/javascript'>
  let s="good morning";
  let regExp=/[0-9]/;
   let result= s.match(regExp);
   if(result!=null)
    { document.write("digit found");
    }
  else{
      document.write("Digit not found");
   }
</script>
</head>
<body>

</body>
</html>
```

Meta Characters: Meta Characters are the characters with a special meaning

Meta Characters	Description
.	Find single character or new line termination

\w	Find word of character
	
	let s="good morn8ing";
	  let regExp=/\w[0-9]/;
	   let result= s.match(regExp);
	   document.write(result);
	Output: n8

\W	Find non-word character
\d	Find a digit

\D	Find non digit character
\t	Find tab character
\udddd	Find Unicode characters
\xdd	Find hexa decimal characters

Quantifier

Quantifier	**Description**
n+	Match any string that contain at least one character
n*	Matches any string that contains zero or one occurrence of n
n{X}	Matches a string that contain sequence of string
n$	Matches any string with n at the ending of it
^n	Matches any string with n at the beginning of it
?=n	Matches any string that is followed by specific string of n
?!n	Matches any string that is not followed by specific string n

Method of Regular Expression

Method	Description
exec()	Test for a match in string and return first match
test()	Test for match in a string return true or false.
toString()	Return string value regular expression

Example: perform number validation using a regular expression

WAP to create textbox and perform number validation using a regular expression

```
<html>
 <head>
 <title>regular expression application</title>
```

```
<script type='text/javascript'>
  function checkNum(str){
     let regExp=/^[0-9]+$/;
     let result=regExp.test(str);
     if(result && (str.length==10))
     { document.getElementById("s").innerHTML="";
     }
     else{
             document.getElementById("s").innerHTML="Invalid Number";
    }
  }
</script>
<style>
   input{
    width:400px;
    height:40px;
   }
</style>
</head>
<body>
  <input type='text' name='name' value='' onkeyup='checkNum(this.value)'/>
&nbsp;&nbsp;<span id='s'></span>
</body>
</html>
```

OR

```
<html>
 <head>
 <title>regular expression application</title>
<script type='text/javascript'>
  function checkNum(str){
    (/^[0-9]+$/.test(str) &&
(str.length==10))?document.getElementById("s").innerHTML="":document.getElementById("s")
.innerHTML="Invalid Number";
```

```
  }
</script>
<style>
   input{
    width:400px;
    height:40px;
   }
</style>
</head>
<body>
  <input type='text' name='name' value='' onkeyup='checkNum(this.value)'/>
&nbsp;&nbsp;<span id='s'></span>
</body>
</html>
```

Example: WAP to create a textbox and perform name validation using a java script using a regular expression

```
<html>
 <head>
 <title>regular expression application</title>
<script type='text/javascript'>
  function checkNum(str){
    (/^[a-zA-z '
']+$/.test(str))?document.getElementById("s").innerHTML="":document.getElementById("s").in
nerHTML="Invalid Name";
  }
</script>
<style>
   input{
    width:400px;
    height:40px;
   }
</style>
</head>
```

```
<body>
  <input type='text' name='name' value='' onkeyup='checkNum(this.value)'/>
&nbsp;&nbsp;<span id='s'></span>
</body>
</html>
```

Example: WAP to perform email validation using a Regular Expression
Example: WAP to perform credit card and debit card validation using a Regular Expression
Example: WAP to perform password validation using a regular expression
Example: WAP to perform date of birth validation using a regular expression?

Advance java script

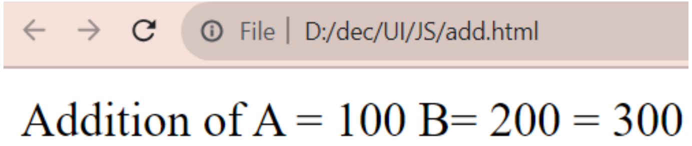
1. Template string
2. Array rest operator
3. Arrow function
4. Array destructing
5. Object Literals
6. OOP concept
7. Promises
8. JSON
9. Modules

Template string

Template string help us to avoid the concatenation using “”+ when we want to pass run time data in string

```
<html>
 <head>
  <title>template string example</title>
 <script type='text/javascript'>
     let a=100;
     let b=200;
     let c=a+b;
   document.write("Addition of  A = "+a+" B= "+b+" = "+c);
 </script>
 </head>
 <body>
 </body>
</html>
```

Output:



```
Addition of A = 100 B= 200 = 300
```

If we think about above code we have three variables A B and C
So we want to display its value so we concat it in document. write () for add run time values in string using “”+ but it is very complicated task
so better way you can use template string
if we want to template string we have to use backtick operator `

Example:

```
document. write(`good morning`);
```

when we use template string we can pass run time parameter in template string using ${expression}

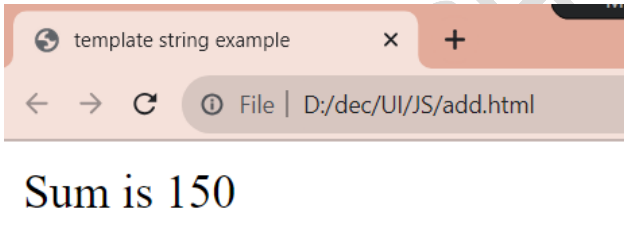
Example:

```
<html>
 <head>
  <title>template string example</title>
 <script type='text/javascript'>
     let a=100;
     let b=200;
     let c=a+b;
   document.write(`Addition is  ${a} + ${b} = ${c} `);
 </script>
 </head>
 <body>
 </body>
</html>
```

Rest operator

Rest operator helps us to pass infinite parameter to function from function calling to definition.

```
<html>
 <head>
  <title>template string example</title>
 <script type='text/javascript'>
   function sum(...x){
     let s=0;
    for(var i=0; i<x.length; i++)
    {  s= s+x[i];
    }
      document.write(`Sum is ${s}`);
   }
  sum(10,20,30,40,50);
 </script>
 </head>
 <body>
 </body>
</html>
```

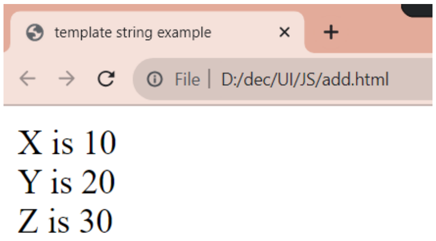


```
Sum is 150
```

Array Destruction: means we initialize array values in different variables by putting it at left hand side.

Example:

```
<html>
 <head>
  <title>template string example</title>
 <script type='text/javascript'>
    let a=[10,20,30];
     let [x,y,z]=a;
 document.write(`X is ${x} <br>`);
 document.write(`Y is ${y} <br>`);
 document.write(`Z is ${z} <br>`);



 </script>
 </head>
 <body>
 </body>
</html>
```



```
X is 10
Y is 20
Z is 30
```

Java Script Object literals

JavaScript provide the facility to us we can store different data in java script in the form of name and value pair.

How to declare java script object

---

if we want to declare java script object we have following syntax

```
let var={
             name:value,
            name:value
            };
```

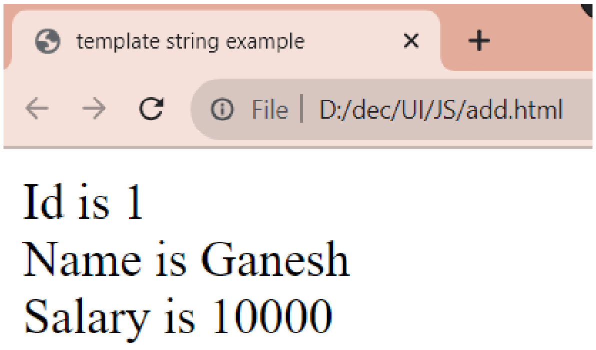
Example:

```
 let emp ={
                    id:1,
                    name:”ABC”,
                    sal:10000
              }
```

if we think about above code we have javascript object which contain three field id,name and sal it work as key means if we want to access data from java script object we have to use the name or key of object.

Source code and output given below

```
<html>
 <head>
  <title>template string example</title>
 <script type='text/javascript'>
   let emp={
                    id:1,
                    name:"Ganesh",
                    sal:10000
             };
document.write(`Id is ${emp.id}<br>`);
document.write(`Name is ${emp.name}<br>`);
document.write(`Salary is ${emp.sal}<br>`);


 </script>
 </head>
 <body>
 </body>
</html>
```



```
Id is 1
Name is Ganesh
Salary is 10000
```

Arrow function

arrow function is used for create lamda expression and the goal lamda expression is if we have smaller logics then we need to write that logic in single line

Syntax:

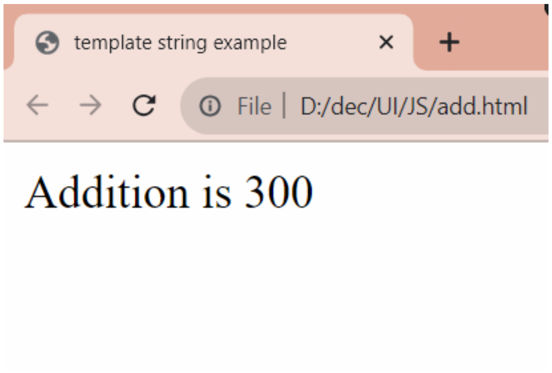
```
let  functionname=()=>{
        write here logics
   }
```

Example using lamda expression

```
<html>
 <head>
  <title>template string example</title>
 <script type='text/javascript'>
    let show=(a,b)=>{
           return a+b;
     }
   let result= show(100,200);
   document.write("Addition is  "+result);
 </script>
 </head>
 <body>
 </body>
```

```
</html>
```

Output



Addition is 300

We can create lamda expression using arrow function

```
<html>
 <head>
  <title>template string example</title>
 <script type='text/javascript'>
    let show=(a,b)=>a+b;
   let result= show(100,200);
   document.write("Addition is  "+result);
 </script>
 </head>
 <body>
 </body>
</html>
```

or

```
<html>
 <head>
  <title>template string example</title>
 <script type='text/javascript'>
    let show=(a,b)=>a+b;
   document.write("Addition is  "+show(100,200));
 </script>
 </head>
 <body>
 </body>
</html>
```

You can implement OOP concept using JavaScript

How to use class in java script

Syntax:

```
class classname{
}
```

you can define function within like as

```
class classname{
   functionname(localvariable){
            this.variable=localvariablename;
    }
}
```

if we use variable using this keyword it mark as instance variable

Example:

```
class Add{
  setValue(a,b){
     this.x=a;
     this.y=b;
   }
  getAdd(){
            return this.x+this.y;
   }
}
```

If we want to use class you have to create object of class.

Syntax:

```
  let variablename = new classname();
```

Example let ad = new Add();

Example:

```
<html>
 <head>
  <title>template string example</title>
 <script type='text/javascript'>
    class Add{
            setValue(a,b){
              this.x=a;
              this.y=b;
            }
            getAdd(){
             return this.x+this.y;
            }
    }
 let ad = new Add();
 ad.setValue(10,20);
 let result=ad.getAdd();
  document.write("Addition is  "+result);
 </script>
 </head>
 <body>
 </body>
</html>
```

How to use constructor in JavaScript

---

if we want to use constructor in JavaScript we have to constructor() function and which is executed automatically when we create object of class.

```
<html>
 <head>
  <title>template string example</title>
 <script type='text/javascript'>
    class Add{
            constructor(){
              document.write("I am constructor");
            }

    }
 let ad = new Add();

 </script>
 </head>
 <body>
 </body>
</html>
```

How to perform inheritance in JavaScript
if we want to perform inheritance in java we have to use extends keyword.

```
<html>
 <head>
  <title>template string example</title>
 <script type='text/javascript'>
   class Value{
     constructor(a,b){
      this.x=a;
      this.y=b;
    }
   }
    class Add extends Value{
            constructor(a,b){
               super(a,b);
            }
       getAdd(){
          return this.x+this.y;
       }

    }
 let ad = new Add(100,200);
 let result=ad.getAdd();
 document.write("Addition is "+result);
 </script>
 </head>
 <body>
 </body>
</html>
```

Example of overriding

```
<html>
 <head>
  <title>template string example</title>
 <script type='text/javascript'>
     class A{
            show(){
              document.write("A method");
            }
     }
     class B extends A{
             show(){
              document.write("B method");
            }
    }
   let b = new B();
    b.show();
 </script>
 </head>
 <body>
 </body>
</html>
```

JSON

____

Q. what is JSON?

____

JSON stands for java script object notations

JSON is a text format for storing and transporting data.

Q. Why use JSON?

____

JSON normally use in Restful web services.

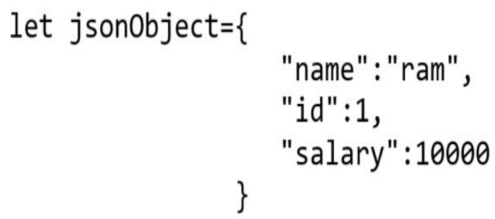
Q. what is web services and why use it?

____

Web services is used for provide the language interoperability

How to use JSON object

JSON object is similar like as java script object just difference json key always present in "" and js object state not in " "



```
let jsonObject={
                    "name":"ram",
                    "id":1,
                    "salary":10000
                 }
```

If we want to access JSON data we have to use following syntax

jsonObject.key

```
<html>
 <head>
  <title>JSON Object App</title>
<script type='text/javascript'>
   let jsonObj={
                    "name":"ABC",
                    "id":1,
                    "salary":10000
                }
document.write("Name is  "+jsonObj.name+"<br>");
document.write("Id is  "+jsonObj.id+"<br>");
document.write("Salary is  "+jsonObj.salary+"<br>");

</script>
 </head>
 <body>
 </body>
</html>
```

Important points related with JSON
1. it help us to store data
2. Transfer data in different technologies
3. Universal format in different languages.

JSON support to following types of data type

---

1. string
2. number
3. object
4. array
5. boolean
6. null

Example:

```
<html>
 <head>
  <title>JSON Object App</title>
<script type='text/javascript'>
  let jsonObj={
                 "name":"ABC",
                 "id":1,
                 "salary":10000,
          "marks":[60,60,60,60,60,60],
                 "skillSet":{language:"JAVA",duration:8},
          "exp":false,

              }
document.write("Name is  "+jsonObj.name+"<br>");
document.write("Id is  "+jsonObj.id+"<br>");
document.write("Salary is  "+jsonObj.salary+"<br>");
let marks=jsonObj.marks;
for(var i=0; i<marks.length; i++)
{ document.write(marks[i]+"&nbsp;&nbsp;");
}
let skills=jsonObj.skillSet;
```

```
document.write("Language "+skills.language+"<br>");
document.write("Duration "+skills.duration);
</script>
 </head>
 <body>
 </body>
</html>
```

if we want to store of json you can store data of json in separate file with extension .json

Methods of JSON

JSON.stringify(object): this method is used for convert json object in to stringify
JSON.parse(): convert JSON object in to JS object

```
<html>
 <head>
  <title>JSON Object App</title>
<script type='text/javascript'>
   let jsonObj={
                    "name":"ABC",
                    "id":1,
                    "salary":10000,
            "marks":[60,60,60,60,60,60],
                    "skillSet":{language:"JAVA",duration:8},
            "exp":false,

              }
 document.write(typeof(JSON.stringify(jsonObj)));

</script>
 </head>
 <body>
 </body>
</html>
```