JavaScript Notes and Exercises

let and var

- Let and var are keywords. Both are used to define variables in js.

- Major difference is:

- *let* gives you the privilege to declare variables that are limited in scope to the block

- *var* is rather a keyword which defines a variable globally regardless of block scope.

- *let* variables cannot be accessed in the window object because they cannot be globally accessed.

- *let* variables are usually used when there is a limited use of those variables. Ex: for, while,if

- var variable can be accessed by for loop

- *let* variables cannot be re-declared while var variable can be re-declared in the same scope.(using strict)

Ex:

'use strict';
var temp = "this is a temp variable";
var temp = "this is a second temp variable"; //replaced easily

- *let* and *var* variables work the same way when used in a function block.

Sum of array elements

```html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Find the Sum of Values in an Array in
JavaScript</title>
</head>
<body>

  <script>

    var array = [1, 2, 3, 4, 5];

    // Getting sum of array using for loop
    var i,sum1=0;
    for(i=0;i<array.length;i++)
    {
        sum1=sum1+array[i];
    }
    document.write(sum1);

    //Getting sum of array using reduce function
    var sum = array.reduce(function(a, b)
        {
            return a + b;
        }, 0);

        document.write(sum);

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

String and operators: program that removes spaces and appends into new string

```html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>String and operators in JS</title>
</head>
  <body>
    <script type="text/javascript">

      var str="h e l l o";
      alert(str)  //This pops an alert box on the
browser
      var a="";
      var b="";
      for(let i=0;i<str.length;i++)
      {
        if(str[i]!=" ")
        {
          a+=str[i];
        }
```

```html
        else if(str[i]== " ")
        {
          for(j=i;j<str.length-i;j++)
          {
            b+=str[j];
            break;
          }
        }
      }
      var c=b+" "+a;
      document.write(c);

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

Functions and Variables

Output??	Output??	Output??	Output??	

`<html>`
`<body>`
`<script>`
`function a()`
`{`
`var x=50;`
`console.log("func a"+x);`
`}`
 	
`console.log(x);`
	`for(let i=0;i<2;i++)`
	`        console.log(i);`
	
	`console.log(i);`
	
	`for(var i=0;i<2;i++)`
	`        console.log(i);`
	
	`console.log(i);`
		`const x=51;`
		`x=52;`
		`x=x+1;`
		`console.log(x);`
			`for(const i=0;i<2;i++)`
			`        console.log(i);`
				`function a()`
				`{`
				`var x=50;`
				`console.log("func a"+x);`
				`}`
				
				`a()`
				
				`let z=100;`
				`function b1()`
				`{`
				`z=500;`
				`console.log("in func"+z);`
				`}`
				`console.log("out func"+z);`

Built in constructors

- <script>
- var a = new Object();
- var a = new String();
- var a = new Number();
- var a = new Boolean();
- var a = new Array();
- var a = new RegExp();
- </script>

User defined Objects

- <html>
- <body>
- <script>
-   function Business(name, property, age, designation) {
-     this.Name = name;
-     this.prop = property;
-     this.age = age;
-     this.designation = designation;
-     this.name = function() {
-       return this.Name
-     };
-   }
-   var person1 = new Business("Trump", "$28.05billion", "73", "President");
-   document.write(person1.name());
- </script>
- </body>
- </html>

Forms and event handling in js

- Event: JavaScript's interaction with HTML is handled through events that occur when the user or the browser manipulates a page.

- Like button click, page load, mouse moved, mouse clicked, button click....

- Events are a part of the Document Object Model (DOM) Level 3 and every HTML element contains a set of events which can trigger JavaScript Code.

Event handling- Button

- <html>
-   <head>
-     <script type = "text/javascript">
-             function validation()
-   {
-         }
- </script>
- </head>
- <body>
-     <form method = "POST" onsubmit = "return validate()">
-      <input type = "submit" value = "Submit" />
-      </form>
- </body>
- </html>

```html
<button onclick="alert('Hello, this is my old-fashioned event handler!');">Press me</button>
```

```html
<form>
  <div>
    <label for="fname">First name:
</label>
    <input id="fname" type="text">
    <p id="para">
    <input id="submit"
type="submit">
  </div>
</form>
```

```javascript
form.onsubmit = function(abc)
{
fname =
document.getElementById('fnam
e');
if (fname.value === ' ')
  alert("Please enter first name")
p.innerHTML="This is para"
}
```