Javascript

Functions

A function is a block of statements that can be used repeatedly in a program.
A function will not execute automatically when a page loads.
A function will be executed by a call to the function.
A function can called multiple times

Syntax

```
function_keyword name_of_function()
{
  code
}
```

There are three types of function:

- normal function
- anonymous function
- arrow function(es6)

```
1) Simple function : it can be called anywhere in code

    example:
   // calling function
    add() // it will run 2times
   function add(){
    var a=16;
    var b=29;
    var add= a + b;
    console.log(`addition of ${a} & ${b} is ${add}`);
    };

    // calling function
    add()

    output:
    addition of 16 & 29 is 45

2) Anonymous function : can be called only after function is created

     // calling function
    sum() // it will throw error

    var sum = function() {
    var a=-16;
    var b=19;
    var add= a + b;
    console.log("addition of a & b is "+ add);
    };

    // calling function
    sum()

    output:
    addition of a & b is 3
```

```
3) Arrow function: Introduced in es6

     //calling function
    sum(); // it will throw error
    var sum =()=> {
    var a=6;
    var b=1;
    var add= a + b;
    console.log("addition of a & b is "+ add);
    };

    //calling function
    sum();

    output:
    addition of a & b is 7
```

Events in js

There are various types of events in js

- **click events**
  - onclick
  - dblclick
- **mouse events**
  - onmouseover
  - onmouseenter
  - onmouseleave
  - onmousemove
  - onmouseup
  - onmousedown
  - onmousewheel
- **keyboard events**
  - onkeypress
  - onkeyup
  - onkeydown

example:Function without parameter

```
    in html
    <button onclick="add()">click me to add</button>

    in script
    function add(){
    var a=16;
    var b=29;
    var add= a + b;
    console.log(`addition of ${a} & ${b} is ${add}`);
    };

    output:
    addition of 16 & 29 is 45
```

example:Function with parameter

```
    in html
    <button onclick="sum(10,2,3)">sum of 3nos</button>

    in script
    function sum(a,b,c) {
    var add =  a + b + c;
    console.log(`function with 3params ${a} + ${b} + ${c} is`,add);
    document.writeln(add)
  };

  output:
  function with 3params 10 + 2 + 3 add is 15
```

Functions with parameter with some validations

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

 In func.js file

 // step 1
 // Get inputs for name and email
 const userName = prompt("Enter your name:");
 const userEmail = prompt("Enter your email:");

 // creating variable to throw error and calling function with inputs
 const nameError = validateName(userName); // username means passing from prompt
 const emailError = validateEmail(userEmail); // userEmail means passing from prompt

 // ---------------------------------------------------------------------------
```

```
// step 2
// Validation name
// single parameter with name
function validateName(name) {
  if (!name) {
    return "Name is required";
  }

  else if (name.length < 5) {
    return "Name must be at least 2 characters";
  }

  else{
    return "";
  }
}

// Validation email
// single parameter with email
function validateEmail(email) {
  if (!email || !email.includes("@") || !email.includes(".")) {
    return "Email is invalid";
  }

  else{
    return "";
  }
}

// ----------------------------------------------------------------------------------------

// step 3
// show result
if (nameError || emailError) {
  alert(`${nameError}\n${emailError}`); // template literal
} else {
  alert("Validation passed");
}
```