
UNIVERSITI MALAYSIA TERENGGANU

LAB MODULE 3

NAME : ILHAM HANINA MADIHA BINTI OTHMAN

MATRIC NO : S63762

PROGRAMME: BACHELOR OF COMPUTER SCIENCE (MOBILE COMPUTING)

LECTURER : DR RABIEI B MAMAT

GITHUB LINK : https://github.com/ilhamhanina/Code-Lab-CSM3103-S63762.git

Task 1 – JavaScript Function

1. Write a function to find the square of a given number

```javascript
// Define the square calculation function
function calculateSquare(number) {
  return number * number;
}

// Prompt the user for input
var userInput = prompt("Enter a number to calculate its square:");
var parsedInput = parseFloat(userInput);

// Validate user input
if (isNaN(parsedInput)) {
  console.log("Invalid input. Please enter a valid number.");
} else {
  var square = calculateSquare(parsedInput);
  console.log("The square of " + parsedInput + " is: " + square);
}
```

2. Write a function to find sum of cubes of two numbers

```javascript
// Define the function to calculate the sum of cubes
function calculateSumOfCubes(number1, number2) {
  var cube1 = number1 * number1 * number1;
  var cube2 = number2 * number2 * number2;
  var sum = cube1 + cube2;
  return sum;
}

var number1 = 3;
var number2 = 4;
var sumOfCubes = calculateSumOfCubes(number1, number2);
console.log("The sum of cubes of " + number1 + " and " + number2 + " is: " + sumOfCubes);

// Prompt the user for input
var userInput1 = prompt("Enter the first number:");
var userInput2 = prompt("Enter the second number:");
var parsedInput1 = parseFloat(userInput1);
var parsedInput2 = parseFloat(userInput2);

// Validate user input
if (isNaN(parsedInput1) || isNaN(parsedInput2)) {
  console.log("Invalid input. Please enter valid numbers.");
} else {
  var sum = calculateSumOfCubes(parsedInput1, parsedInput2);
  console.log("The sum of cubes of " + parsedInput1 + " and " + parsedInput2 + " is: " + sum);
}
```

3. Write a function to reverse a number

[Hint n = 12345 output : 54321 ]

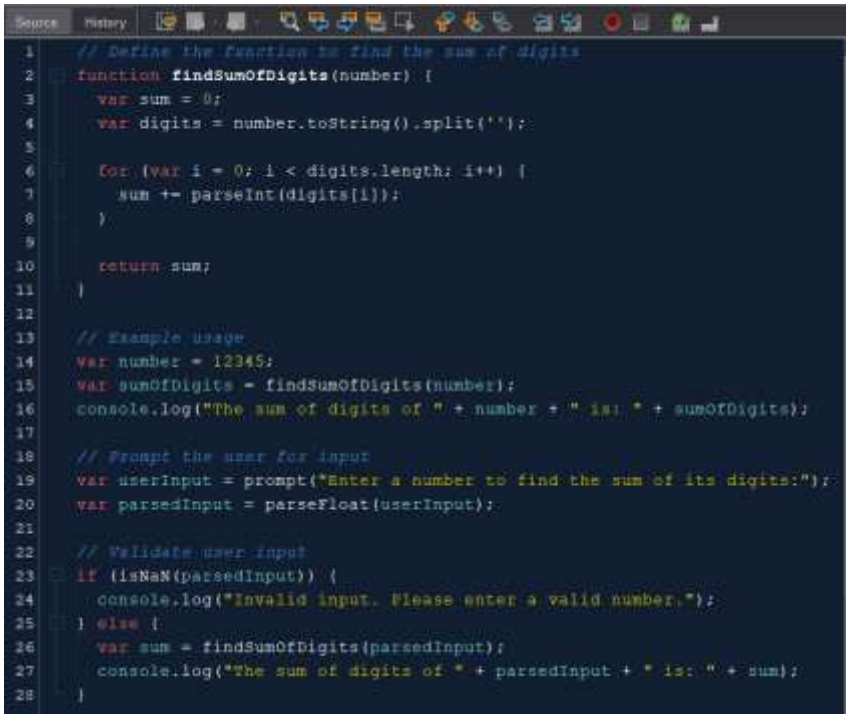
```javascript
// Define the function to reverse a number
function reverseNumber(number) {
  var reversedNumber = 0;
  while (number !== 0) {
    reversedNumber = (reversedNumber * 10) + (number % 10);
    number = Math.floor(number / 10);
  }
  return reversedNumber;
}

var number = 12345;
var reversed = reverseNumber(number);
console.log("The reversed number of " + number + " is: " + reversed);

// Prompt the user for input
var userInput = prompt("Enter a number to reverse:");
var parsedInput = parseFloat(userInput);

// Validate user input
if (isNaN(parsedInput)) {
  console.log("Invalid input. Please enter a valid number.");
} else {
  var reversedInput = reverseNumber(parsedInput);
  console.log("The reversed number of " + parsedInput + " is: " + reversedInput);
}
```

4. Write a function to print all numbers between 1 and 100 which is divisible by given number z

```javascript
// Define the function to print numbers divisible by z
function printDivisibleNumbers(z) {
  for (var i = 1; i <= 100; i++) {
    if (i % z === 0) {
      console.log(i);
    }
  }
}

// Example usage
var givenNumber = 5;
console.log("Numbers between 1 and 100 divisible by " + givenNumber + ":");
printDivisibleNumbers(givenNumber);

// Prompt the user for input
var userInput = prompt("Enter a number:");
var parsedInput = parseInt(userInput);

// Validate user input
if (isNaN(parsedInput)) {
  console.log("Invalid input. Please enter a valid number.");
} else {
  console.log("Numbers between 1 and 100 divisible by " + parsedInput + ":");
  printDivisibleNumbers(parsedInput);
}
```

Task 2 – JavaScript Recursion Function

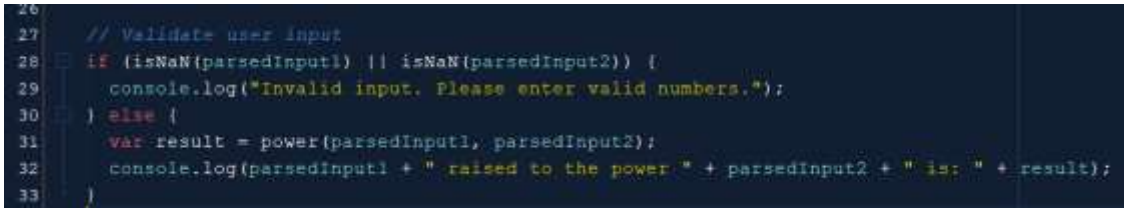
1. Write a JavaScript function to find sum of digits of a number



```
// Define the function to find the sum of digits
function findSumOfDigits(number) {
  var sum = 0;
  var digits = number.toString().split('');

  for (var i = 0; i < digits.length; i++) {
    sum += parseInt(digits[i]);
  }

  return sum;
}

// Example usage
var number = 12345;
var sumOfDigits = findSumOfDigits(number);
console.log("The sum of digits of " + number + " is: " + sumOfDigits);

// Prompt the user for input
var userInput = prompt("Enter a number to find the sum of its digits:");
var parsedInput = parseFloat(userInput);

// Validate user input
if (isNaN(parsedInput)) {
  console.log("Invalid input. Please enter a valid number.");
} else {
  var sum = findSumOfDigits(parsedInput);
  console.log("The sum of digits of " + parsedInput + " is: " + sum);
}
```

2. Write a JavaScript program to compute x raise to the power y using recursion

```
// Define the function to compute x raised to the power y
function power(x, y) {
  if (y === 0) {
    return 1;
  } else if (y < 0) {
    return 1 / power(x, -y);
  } else if (y % 2 === 0) {
    var temp = power(x, y / 2);
    return temp * temp;
  } else {
    return x * power(x, y - 1);
  }
}

// Example usage
var base = 2;
var exponent = 5;
var result = power(base, exponent);
console.log(base + " raised to the power " + exponent + " is: " + result);

// Prompt the user for input
var userInput1 = prompt("Enter the base number:");
var userInput2 = prompt("Enter the exponent:");
var parsedInput1 = parseFloat(userInput1);
var parsedInput2 = parseFloat(userInput2);
```



```
// Validate user input
if (isNaN(parsedInput1) || isNaN(parsedInput2)) {
  console.log("Invalid input. Please enter valid numbers.");
} else {
  var result = power(parsedInput1, parsedInput2);
  console.log(parsedInput1 + " raised to the power " + parsedInput2 + " is: " + result);
}
```

Task 3 – JavaScript Object and Prototype

1. Write a JavaScript program to create object product,

    a. Add the property Product Name, Quantity and price.

    b. Access all the properties and display them.

```javascript
// Create the Product constructor
function Product(productName, quantity, price) {
  this.productName = productName;
  this.quantity = quantity;
  this.price = price;
}

// Add a method to calculate the total value of the product
Product.prototype.calculateTotalValue = function() {
  return this.quantity * this.price;
};

// Create a new product object
var product = new Product("Skincare Product", 10, 49.99);

// Access and display the properties
console.log("Product Name: " + product.productName);
console.log("Quantity: " + product.quantity);
console.log("Price: $" + product.price.toFixed(2));

// Calculate and display the total value
console.log("Total Value: $" + product.calculateTotalValue().toFixed(2));
```

2. Write a JavaScript program to create object book

    a. Add the property book name, author name

    b. Add the prototype property price

    c. Display all the properties.

```javascript
// Create the Book constructor
function Book(bookName, authorName) {
  this.bookName = bookName;
  this.authorName = authorName;
}

// Add the prototype property Price
Book.prototype.price = 0;

// Add a method to set the price of the book
Book.prototype.setPrice = function(price) {
  this.price = price;
};

// Add a method to get the formatted price
Book.prototype.getFormattedPrice = function() {
  return "$" + this.price.toFixed(2);
};
```

```
19
20    // Create a new book object
21    var book = new Book("Front End for Beginner", "Ilham Hanina");
22
23    // Access and display the properties
24    console.log("Book Name: " + book.bookName);
25    console.log("Author Name: " + book.authorName);
26    console.log("Price: " + book.getFormattedPrice());
27
28    // Set the price of the book
29    book.setPrice(49.99);
30
31    // Display the updated price
32    console.log("Updated Price: " + book.getFormattedPrice());
33
```

3. Write a JavaScript program to create Parent object employee (Property : Employee Name , Employee Id , Salary) and Child object Manager (Property : Manager Name , Branch). Inherit all the properties of employee and display all the properties.

```
1   // Create the Employee parent object
2   function Employee(employeeName, employeeId, salary) {
3     this.employeeName = employeeName;
4     this.employeeId = employeeId;
5     this.salary = salary;
6   }
7
8   // Create the Manager child object that inherits from Employee
9   function Manager(employeeName, employeeId, salary, managerName, branch) {
10    Employee.call(this, employeeName, employeeId, salary);
11    this.managerName = managerName;
12    this.branch = branch;
13  }
14
15  // Set up the inheritance
16  Manager.prototype = Object.create(Employee.prototype);
17  Manager.prototype.constructor = Manager;
18
19  // Create a new Manager object
20  var manager = new Manager("Ilham Hanina", "E123", 5700, "Imtiyaz Wirni", "Branch Promotion");
21
22  // Access and display the properties
23  console.log("Employee Name: " + manager.employeeName);
24  console.log("Employee ID: " + manager.employeeId);
25  console.log("Salary: $" + manager.salary.toFixed(2));
26  console.log("Manager Name: " + manager.managerName);
27  console.log("Branch: " + manager.branch);
```

Task 4 – Event Manager

1. Create a HTML page with <p> paragraph. Change the paragraph color according to the following mouse events.

a. Onclick, yellow background

b. ondblclick, blue background

c. onmouseover , red background

d. onmouseout, green background


```html
<!DOCTYPE html>
<html>
<head>
  <title>Event Manager Task 4.1</title>
  <style>
    #myParagraph {
      width: 300px;
      padding: 10px;
      border: 1px solid black;
    }
  </style>
</head>
<body>
  <p id="myParagraph">Click me!</p>

  <script>
    var paragraph = document.getElementById("myParagraph");

    // Onclick - yellow background
    paragraph.onclick = function() {
      paragraph.style.backgroundColor = "yellow";
    };

    // Ondblclick - blue background
    paragraph.ondblclick = function() {
      paragraph.style.backgroundColor = "blue";
    };

```

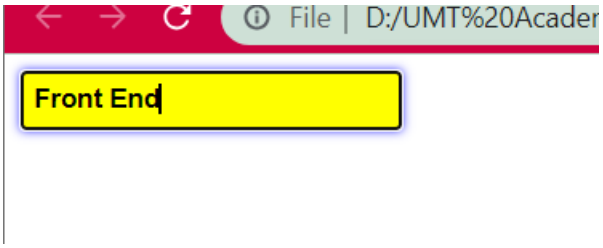
```
28
29        // Onmouseover - red background
30        paragraph.onmouseover = function() {
31          paragraph.style.backgroundColor = "red";
32        };
33
34        // Onmouseout - green background
35        paragraph.onmouseout = function() {
36          paragraph.style.backgroundColor = "green";
37        };
38      </script>
39    </body>
40    </html>
```

2. Create a HTML page with textfield. Show some effects on the textfield when the following events occurred:

a. Onchange

b. Onfocus

c. onblur




```
<!DOCTYPE html>
<html>
<head>
  <title>Event Manager Task 4.2</title>
  <style>
    .effect {
      border: 2px solid gray;
      padding: 5px;
      transition: border-color 0.3s ease-in-out;
    }

    .effect:focus {
      border-color: blue;
      box-shadow: 0 0 5px blue;
    }
  </style>
</head>
<body>
  <input type="text" id="myInput" class="effect" placeholder="Enter text" />

```

```
  <script>
    var input = document.getElementById("myInput");

    // Onchange
    input.onchange = function() {
      input.style.backgroundColor = "yellow";
    };

    // Onfocus
    input.onfocus = function() {
      input.style.fontWeight = "bold";
    };

    // Onblur
    input.onblur = function() {
      input.style.fontWeight = "normal";
    };
  </script>
</body>
</html>
```

Task 5

Given the following HTML table

| | | | |
|---|---|---|---|
| 1 | Ahmad Faisal | ahmadfaisal@gmail.com | 0199088888 |
| 2. | Ismail Sabri | isabri@mail.com | 0199076760 |
| 3 | Fateh Yakin | ffateh@hotmail.com | 0176067762 |

1. Using javascript add the following record into table
    a. Name: Mukhriz Jamil Asoka
    b. Email: mukriz@corp.jo
    c. Phone: 651181187223
2. Using javascript add the table header as follow:
    a. #, Name, Email, Phone #
3. Using javascript, delete any row from table when clicked on that row

| # | Name | Email | Phone # |
|---|---|---|---|
| 1 | Ahmad Faisal | ahmadfaisal@gmail.com | 0199088888 |
| 2 | Ismail Sabri | isabri@mail.com | 0199076760 |
| 3 | Fateh Yakin | ffateh@hotmail.com | 0176067762 |
| 4 | Mukhriz Jamil Asoka | mukriz@corp.jo | 651181187223 |

| # | Name | Email | Phone # |
|---|---|---|---|
| 1 | Ahmad Faisal | ahmadfaisal@gmail.com | 0199088888 |
| 2 | Ismail Sabri | isabri@mail.com | 0199076760 |
| 3 | Fateh Yakin | ffateh@hotmail.com | 0176067762 |

```
<!DOCTYPE html>
<html>
<head>
  <title>Table Manipulation Example</title>
  <style>
    table {
      border-collapse: collapse;
    }

    th, td {
      border: 1px solid black;
      padding: 5px;
    }

    tr:hover {
      background-color: lightgray;
      cursor: pointer;
    }
  </style>
</head>
```

```html
<body>
  <table id="myTable">
    <!-- Existing table rows -->
    <tr>
      <td>1</td>
      <td>Ahmad Faisal</td>
      <td>ahmadfaisal@gmail.com</td>
      <td>0199088888</td>
    </tr>
    <tr>
      <td>2</td>
      <td>Ismail Sabri</td>
      <td>isabri@mail.com</td>
      <td>01990767760</td>
    </tr>
    <tr>
      <td>3</td>
      <td>Fateh Yakin</td>
      <td>ffateh@hotmail.com</td>
      <td>0176067762</td>
    </tr>
  </table>
```

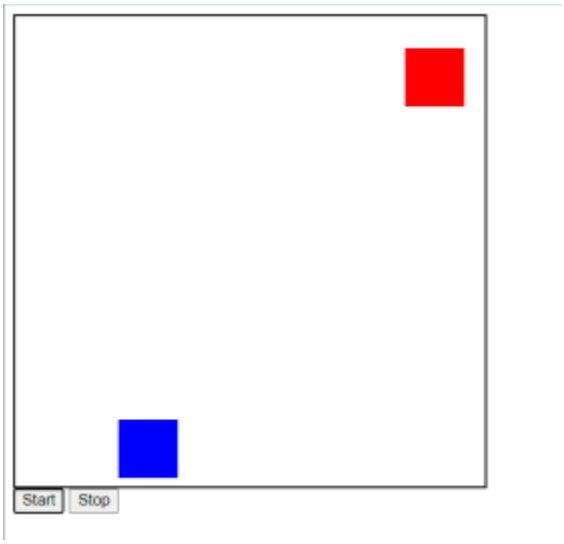
```javascript
  <script>
    // Add a new record to the table
    var table = document.getElementById("myTable");
    var newRow = table.insertRow();

    var cell1 = newRow.insertCell();
    cell1.textContent = "4";

    var cell2 = newRow.insertCell();
    cell2.textContent = "Mukhriz Jamil Asoka";

    var cell3 = newRow.insertCell();
    cell3.textContent = "mukriz@corp.jo";

    var cell4 = newRow.insertCell();
    cell4.textContent = "651181187223";

    // Add the table header
    var headerRow = table.insertRow(0);

    var header1 = headerRow.insertCell();
    header1.textContent = "#";
```

```
66
67          var header2 = headerRow.insertCell();
68          header2.textContent = "Name";
69
70          var header3 = headerRow.insertCell();
71          header3.textContent = "Email";
72
73          var header4 = headerRow.insertCell();
74          header4.textContent = "Phone #";
75
76          // Delete row when clicked
77          var rows = table.getElementsByTagName("tr");
78          for (var i = 0; i < rows.length; i++) {
79            rows[i].onclick = function() {
80              this.parentNode.removeChild(this);
81            };
82          }
83        </script>
84      </body>
85      </html>
```

Task 6

Write a JavaScript program to move two small squares inside one big square in a random manner. User should be able to start and stop this animationusing button based events

**Math.floor(Math.random() * Math.floor(max)) will give you a random number that is less than max value**




```
<!DOCTYPE html>
<html>
<head>
  <title>Animation Example</title>
  <style>
    #container {
      position: relative;
      width: 400px;
      height: 400px;
      border: 2px solid black;
    }

    .square {
      position: absolute;
      width: 50px;
      height: 50px;
      background-color: blue;
    }

    #square1 {
      top: 0;
      left: 0;
    }

    #square2 {
      bottom: 0;
      right: 0;
      background-color: red;
    }
  </style>
```

```
</head>
<body>
  <div id="container">
    <div id="square1" class="square"></div>
    <div id="square2" class="square"></div>
  </div>

  <button id="startBtn">Start</button>
  <button id="stopBtn">Stop</button>

  <script>
    var square1 = document.getElementById("square1");
    var square2 = document.getElementById("square2");
    var container = document.getElementById("container");
    var animationInterval;

    function moveSquares() {
      var containerWidth = container.clientWidth;
      var containerHeight = container.clientHeight;
      var squareSize = square1.clientWidth;

      var maxX = containerWidth - squareSize;
      var maxY = containerHeight - squareSize;

      var x1 = Math.floor(Math.random() * Math.floor(maxX));
      var y1 = Math.floor(Math.random() * Math.floor(maxY));

      var x2 = Math.floor(Math.random() * Math.floor(maxX));
      var y2 = Math.floor(Math.random() * Math.floor(maxY));
```

```
        square1.style.left = x1 + "px";
        square1.style.top = y1 + "px";

        square2.style.right = x2 + "px";
        square2.style.bottom = y2 + "px";
      }

      function startAnimation() {
        animationInterval = setInterval(moveSquares, 1000);
      }

      function stopAnimation() {
        clearInterval(animationInterval);
      }

      var startButton = document.getElementById("startBtn");
      startButton.addEventListener("click", startAnimation);

      var stopButton = document.getElementById("stopBtn");
      stopButton.addEventListener("click", stopAnimation);
    </script>
  </body>
  </html>
```

Link GitHub for the lab : https://github.com/ilhamhanina/Code-Lab-CSM3103-S63762.git