Application Development Lab Assignment

> Submitted by **Aniruddha Mukherjee [2205533] CSE-16**

Q1:

Design A JavaScript to display whether given number is prime or not.

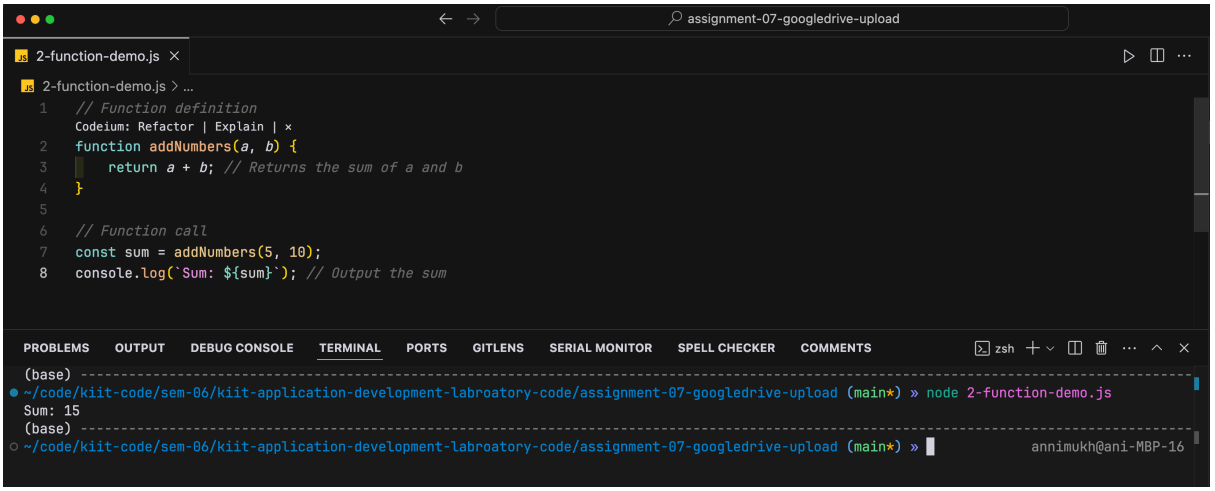
prime-check.js

```javascript
// Function to check if a number is prime
function isPrime(num) {
    if (num < 2) return false; // Numbers less than 2 are not prime

    for (let i = 2; i <= Math.sqrt(num); i++) { // Loop from 2 to square root of num
        if (num % i === 0) {
            return false; // If divisible, it's not a prime number
        }
    }
    return true; // If no divisor found, it's prime
}

// Example usage
const number = 17; // Change this number to test other values
console.log(`${number} is ${isPrime(number) ? 'a prime' : 'not a prime'} number.`);
```

```
(base) ~/code/kiit-code/sem-06/kiit-application-development-labroatory-code/assignment-07-googledrive-upload (main*) » node prime-check.js
17 is a prime number.
(base) ~/code/kiit-code/sem-06/kiit-application-development-labroatory-code/assignment-07-googledrive-upload (main*) »
```

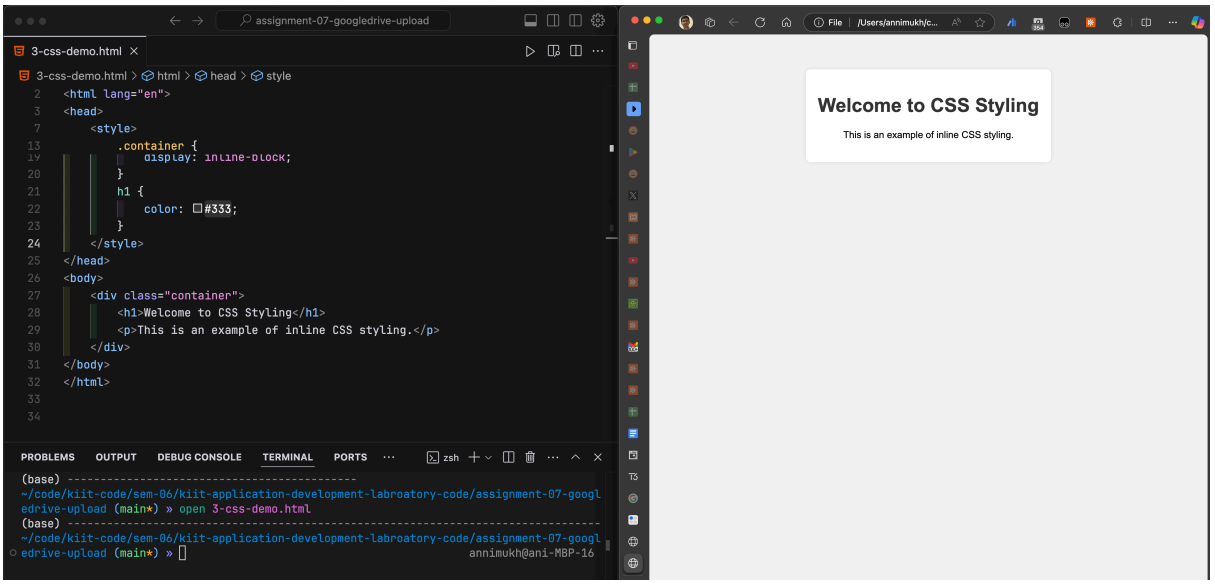
Q2:

Explain about Function definition, Function calling, Function parameter, return type with a suitable program in JavaScript.



Q3:

Explain about Cascading Style Sheets with a program



Q4:

Explain various operators and data types available in java script with examples

4-operators-data-types.js

```javascript
// Arithmetic Operators
let a = 10, b = 5;
console.log(`Addition: ${a + b}`);
console.log(`Subtraction: ${a - b}`);
console.log(`Multiplication: ${a * b}`);
console.log(`Division: ${a / b}`);

// Comparison Operators
console.log(`Equal: ${a == b}`);
console.log(`Not Equal: ${a != b}`);
console.log(`Greater than: ${a > b}`);

// Logical Operators
console.log(`AND: ${true && false}`);
console.log(`OR: ${true || false}`);
console.log(`NOT: ${!true}`);

// Data Types
let str = "Hello"; // String
var boolean = true; // Boolean
let num = 42; // Number
let obj = { name: "John" }; // Object
console.log(`String: ${str}, Boolean: ${boolean}, Number: ${num}, Object: ${JSON.stringify(obj)}`);
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS SERIAL MONITOR SPELL CHECKER COMMENTS

```
(base) ----------
~/code/kiit-code/sem-06/kiit-application-development-labroatory-code/assignment-07-googledrive-upload (main*) » node 4-operators-data-types.js
Addition: 15
Subtraction: 5
Multiplication: 50
Division: 2
Equal: false
Not Equal: true
Greater than: true
AND: false
OR: true
NOT: false
String: Hello, Boolean: true, Number: 42, Object: {"name":"John"}
(base) ----------
~/code/kiit-code/sem-06/kiit-application-development-labroatory-code/assignment-07-googledrive-upload (main*) »
```

annimukh@ani-MBP-16

Q5:

Build a JavaScript program to convert temperature from Celsius to Fahrenheit and vice versa

5-temperature-conversion.js

```javascript
// Function to convert Celsius to Fahrenheit
Codeium: Refactor | Explain | ×
function celsiusToFahrenheit(celsius) {
    return (celsius * 9/5) + 32;
}

// Function to convert Fahrenheit to Celsius
Codeium: Refactor | Explain | ×
function fahrenheitToCelsius(fahrenheit) {
    return (fahrenheit - 32) * 5/9;
}

// Example usage
let celsius = 25;
console.log(`${celsius}°C = ${celsiusToFahrenheit(celsius)}°F`);

let fahrenheit = 77;
console.log(`${fahrenheit}°F = ${fahrenheitToCelsius(fahrenheit)}°C`);
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS SERIAL MONITOR

```
(base) ----------
~/code/kiit-code/sem-06/kiit-application-development-labroatory-code/assignment-07-googledrive-upload (main*) » node 5-temperature-conversion.js
25°C = 77°F
77°F = 25°C
(base) ----------
~/code/kiit-code/sem-06/kiit-application-development-labroatory-code/assignment-07-googledrive-upload (main*) »
```

Q6:

Define Form tag. Design a Registration page by using all Form controls

6-registration-form.html

```html
<html>
<body>
    <form action="#" method="post">
        <label for="name">Name:</label>
        <input type="text" id="name" name="name" required><br><br>

        <label for="email">Email:</label>
        <input type="email" id="email" name="email" required><br><br>

        <label for="password">Password:</label>
        <input type="password" id="password" name="password" required><br><br>

        <label for="gender">Gender:</label>
        <select id="gender" name="gender">
            <option value="male">Male</option>
            <option value="female">Female</option>
        </select><br><br>

        <input type="submit" value="Register">
    </form>
</body>
</html>
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
(base) ----------
~/code/kiit-code/sem-06/kiit-application-development-labroatory-code/assignment-07-googledrive-upload (main*) » open 6-registration-form.html  annimukh@ani-MBP-16
(base) ----------
~/code/kiit-code/sem-06/kiit-application-development-labroatory-code/assignment-07-googledrive-upload (main*) »  annimukh@ani-MBP-16
```

Registration Form

Name:

Email:

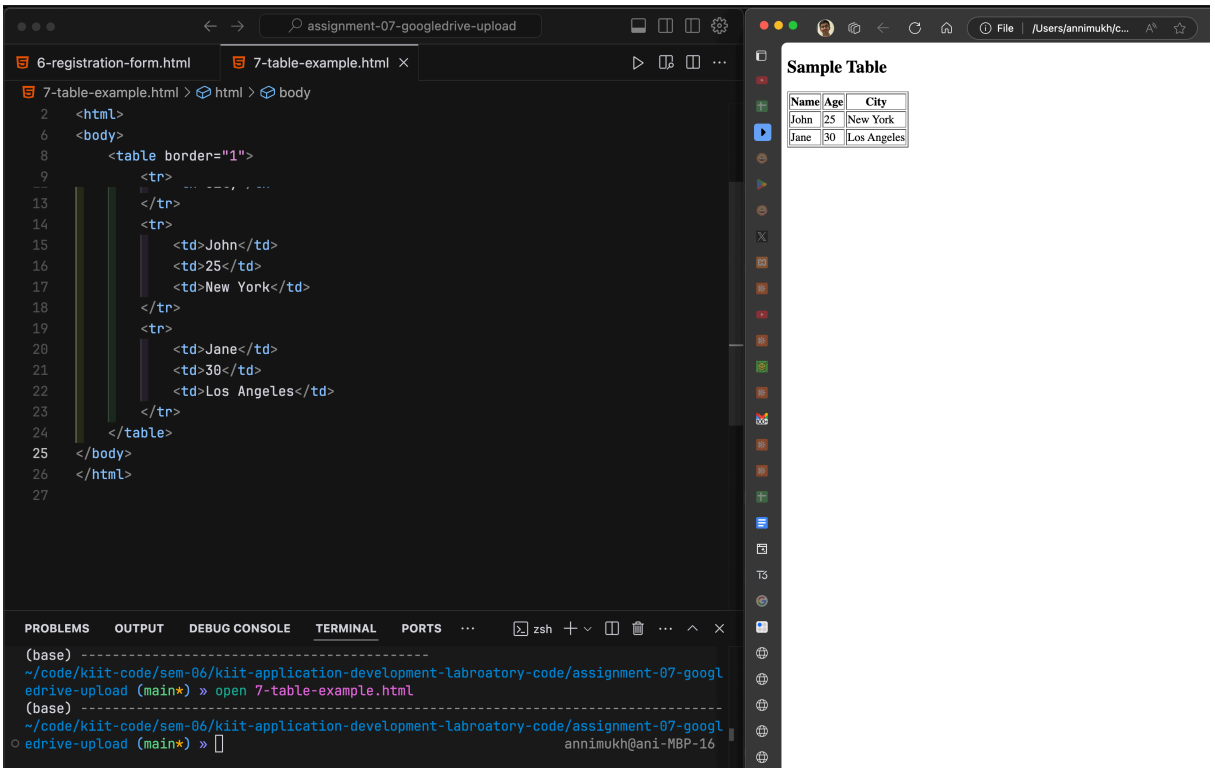
Password:

Gender: Male

Register

Q7:

Define Table tag and their attributes with an example.



Q8:

Write a JavaScript program to print all indices of an array where perfect numbers are present.

```javascript
function isPerfect(num) {
    let sum = 0;
    for (let i = 1; i < num; i++) {
        if (num % i === 0) {
            sum += i;
        }
    }
    return sum === num;
}

// Function to find indices of perfect numbers in an array
Codeium: Refactor | Explain | ×
function findPerfectIndices(arr) {
    return arr.map((num, index) => isPerfect(num) ? index : -1).filter(index => index !== -1);
}

// Example usage
let numbers = [6, 28, 12, 496, 8128, 7];
console.log(`Indices of perfect numbers: ${findPerfectIndices(numbers)}`);
```

```
(base) ----------------------------------------------------------------------------------------------------
~/code/kiit-code/sem-06/kiit-application-development-labroatory-code/assignment-07-googledrive-upload (main*) » node 8-perfect-numbers.js
Indices of perfect numbers: 0,1,3,4
(base) ----------------------------------------------------------------------------------------------------
~/code/kiit-code/sem-06/kiit-application-development-labroatory-code/assignment-07-googledrive-upload (main*) »
```

Q9:

Write a JavaScript program to validate e-mail id, phone number and password

9-validation.js

```javascript
// Function to validate email
function validateEmail(email) {
    const regex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
    return regex.test(email);
}

// Function to validate phone number (10-digit format)
function validatePhone(phone) {
    const regex = /^\d{10}$/;
    return regex.test(phone);
}

// Function to validate password (at least 8 characters, one number, one special character)
function validatePassword(password) {
    const regex = /^(?=.*[A-Za-z])(?=.*\d)(?=.*[@$!%*?&])[A-Za-z\d@$!%*?&]{8,}$/;
    return regex.test(password);
}

// Example usage
const email = "test@example.com";
const phone = "1234567890";
const password = "Test@1234";

console.log(`Email valid: ${validateEmail(email)}`);
console.log(`Phone valid: ${validatePhone(phone)}`);
console.log(`Password valid: ${validatePassword(password)}`);
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS SERIAL MONITOR

```
(base) ----------------------------------------------------------------------------------------------------
~/code/kiit-code/sem-06/kiit-application-development-labroatory-code/assignment-07-googledrive-upload (main*) » node 9-validation.js
Email valid: true
Phone valid: true
Password valid: true
(base) ----------------------------------------------------------------------------------------------------
~/code/kiit-code/sem-06/kiit-application-development-labroatory-code/assignment-07-googledrive-upload (main*) »
```