

## TASK: 1

### BMI CALCULATOR

#### HTML CODE:

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>BMI Calculator</title>

  <link rel="stylesheet" href="styles.css">

</head>

<body>

  <div class="container">

    <h1>BMI Calculator</h1>

    <form id="bmi-form">

      <label for="weight">Weight (kg):</label>

      <input type="number" id="weight" required>

      <label for="height">Height (cm):</label>

      <input type="number" id="height" required>

      <button type="button" onclick="calculateBMI()">Calculate BMI</button>

    </form>

    <div id="result"></div>

  </div>

</body>

</html>
```

```
</div>

<script src="script.js"></script>

</body>

</html>
```

## CSS CODE:

```
body{

  font-family: Arial, sans-serif;

  background-color: #f4f4f4;

  display: flex;

  justify-content: center;

  align-items: center;

  height: 100vh;

  margin: 0;

}

.container{

  background-color: #fff;

  padding: 20px;

  border-radius: 10px;

  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

  text-align: center;

  max-width: 400px;
```

```
    width: 100%;  
}
```

```
h1 {  
    margin-bottom: 20px;  
}
```

```
form {  
    display: flex;  
    flex-direction: column;  
}
```

```
label {  
    margin-bottom: 5px;  
}
```

```
input {  
    margin-bottom: 15px;  
    padding: 10px;  
    border: 1px solid #ddd;  
    border-radius: 5px;  
}
```

```
button {  
    padding: 10px;  
    border: none;  
    border-radius: 5px;  
    background-color: #007BFF;  
    color: white;  
    font-size: 16px;  
    cursor: pointer;  
}
```

```
button:hover {  
    background-color: #0056b3;  
}
```

```
#result {  
    margin-top: 20px;  
    font-size: 18px;  
}
```

## **JAVA SCRIPT CODE:**

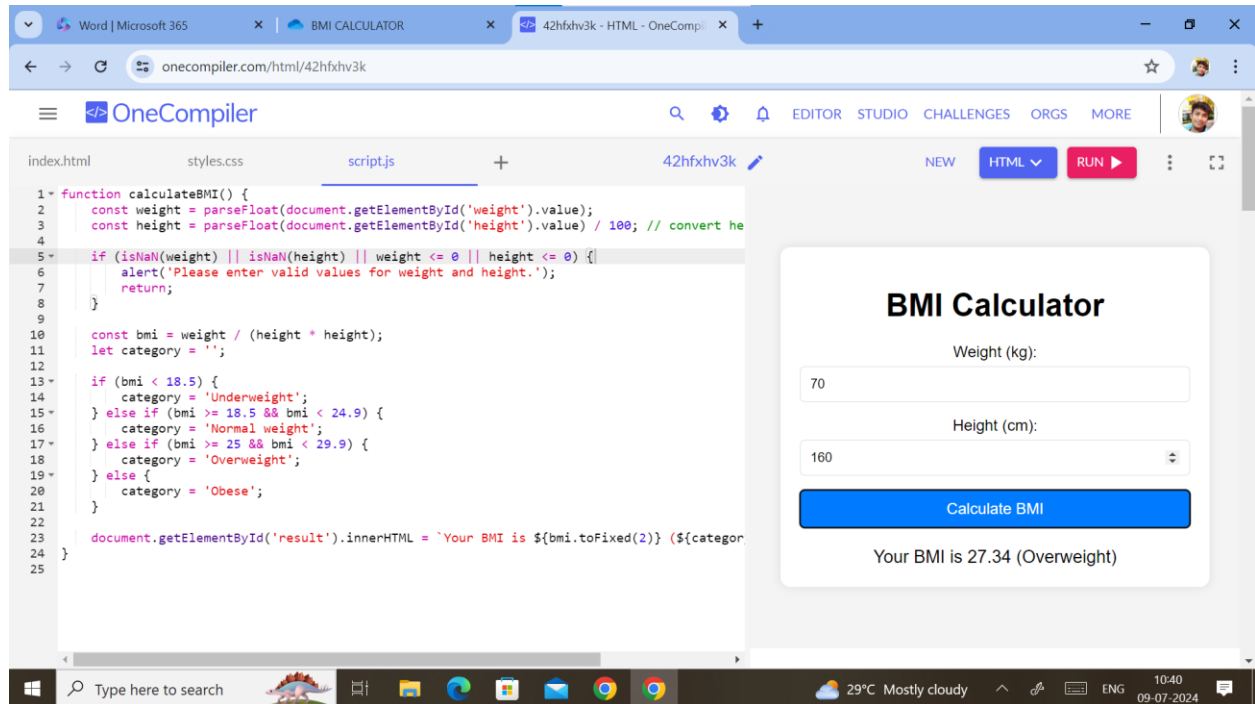
```
function calculateBMI() {  
    const weight = parseFloat(document.getElementById('weight').value);  
    const height = parseFloat(document.getElementById('height').value) / 100;  
    // convert height from cm to meters
```

```
if (isNaN(weight) || isNaN(height) || weight <= 0 || height <= 0) {  
    alert('Please enter valid values for weight and height.');
```

```
    return;  
}  
  
const bmi = weight / (height * height);  
let category = '';  
  
if (bmi < 18.5) {  
    category = 'Underweight';  
} else if (bmi >= 18.5 && bmi < 24.9) {  
    category = 'Normal weight';  
} else if (bmi >= 25 && bmi < 29.9) {  
    category = 'Overweight';  
} else {  
    category = 'Obese';  
}  
  
document.getElementById('result').innerHTML = `Your BMI is  
${bmi.toFixed(2)} (${category})`;  
}
```

## OUTPUT:



## TASK 2:

## COUNTDOWN TIMER

## HTML CODE:

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Countdown Timer</title>

  <link rel="stylesheet" href="styles.css">

</head>

<body>

  <div class="countdown-container">

    <h1>Countdown Timer</h1>

    <div id="countdown">

      <div class="time-box">

        <span id="days">00</span>

        <p>Days</p>

      </div>

      <div class="time-box">

        <span id="hours">00</span>

        <p>Hours</p>

      </div>

      <div class="time-box">

        <span id="minutes">00</span>

        <p>Minutes</p>

      </div>

    </div>

  </div>

</body>

</html>
```

```
</div>

<div class="time-box">

  <span id="seconds">00</span>

  <p>Seconds</p>

</div>

</div>

<div class="input-container">

  <label for="target-date">Set Target Date and Time:</label>

  <input type="datetime-local" id="target-date">

  <button id="start-button">Start Countdown</button>

</div>

</div>

<script src="script.js"></script>

</body>

</html>
```

## CSS:

```
/* styles.css */
```

```
body{

  font-family: Arial, sans-serif;

  display: flex;

  justify-content: center;
```



```
    align-items: center;
    height: 100vh;
    margin: 0;
    background-color: #f0f0f0;
}
```

```
.countdown-container {
    text-align: center;
    background-color: white;
    padding: 20px;
    border-radius: 10px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}
```

```
#countdown {
    display: flex;
    justify-content: center;
    gap: 20px;
    margin-bottom: 20px;
}
```

```
.time-box {
    background-color: #333;
```

```
    color: white;

    padding: 20px;

    border-radius: 5px;
}
```

```
.time-box span {

    font-size: 2rem;

    display: block;
}
```

```
.input-container {

    margin-top: 20px;
}
```

```
input[type="datetime-local"] {

    padding: 5px;

    font-size: 1rem;
}
```

```
button {

    padding: 5px 10px;

    font-size: 1rem;

    cursor: pointer;
}
```

```
}
```

## JAVA SCRIPT CODE:

```
// script.js
```

```
document.getElementById('start-button').addEventListener('click', function() {  
    const targetDateInput = document.getElementById('target-date').value;  
    if (!targetDateInput) {  
        alert('Please select a target date and time.');        return;  
    }  
  
    const targetDate = new Date(targetDateInput);  
  
    function updateCountdown() {  
        const now = new Date().getTime();  
        const timeDifference = targetDate - now;  
  
        if (timeDifference < 0) {  
            clearInterval(interval);  
            document.getElementById('countdown').innerHTML = "<p>Countdown ended</p>";  
            return;  
        }  
  
        const days = Math.floor(timeDifference / (1000 * 60 * 60 * 24));  
        const hours = Math.floor((timeDifference % (1000 * 60 * 60 * 24)) / (1000 * 60 * 60));  
        const minutes = Math.floor((timeDifference % (1000 * 60 * 60)) / (1000 * 60));  
        const seconds = Math.floor((timeDifference % (1000 * 60)) / 1000);
```

```

document.getElementById('days').innerText = days.toString().padStart(2, '0');

document.getElementById('hours').innerText = hours.toString().padStart(2, '0');

document.getElementById('minutes').innerText = minutes.toString().padStart(2, '0');

document.getElementById('seconds').innerText = seconds.toString().padStart(2, '0');

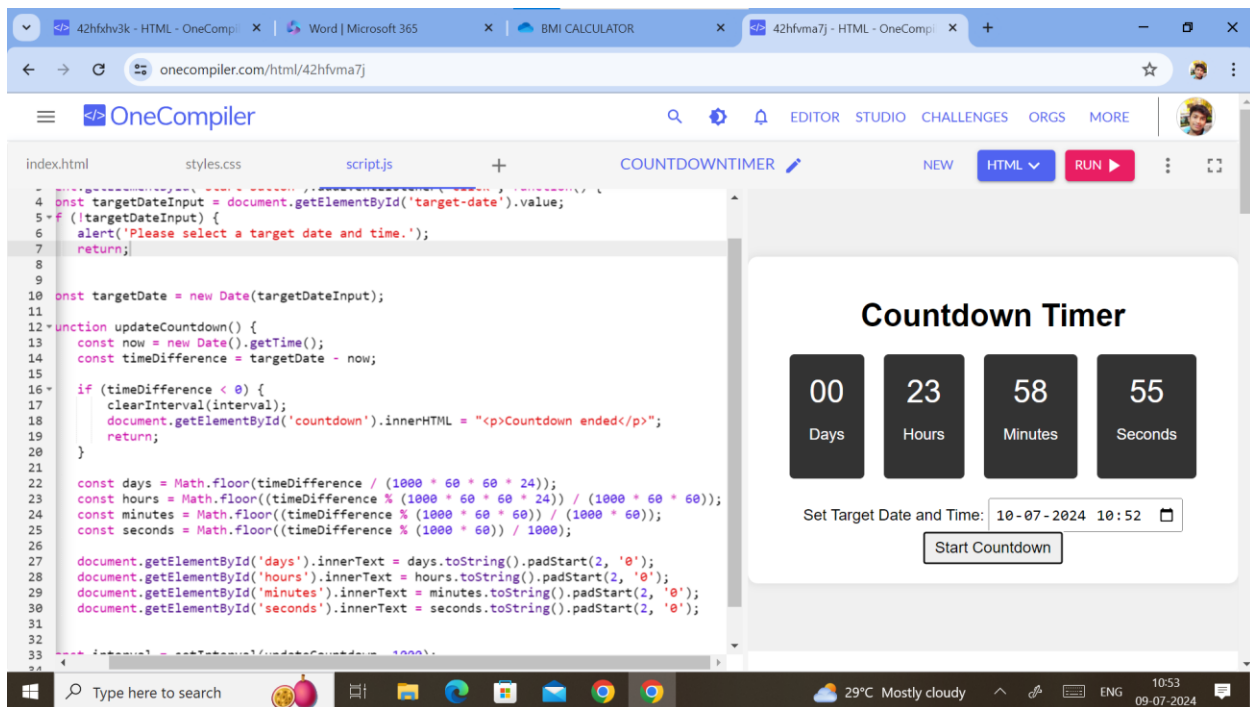
}

const interval = setInterval(updateCountdown, 1000);

});

```

## OUTPUT:



## TASK 3:

### CURRENCY CONVERTER

#### HTML CODE:

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Currency Converter</title>

  <link rel="stylesheet" href="styles.css">

</head>

<body>

  <div class="container">

    <h1>Currency Converter</h1>

    <div class="converter">

      <div>

        <label for="amount">Amount:</label>

        <input type="number" id="amount" placeholder="Enter amount">

      </div>

      <div>

        <label for="from-currency">From:</label>
```

```
        <select id="from-currency"></select>

    </div>

    <div>

        <label for="to-currency">To:</label>

        <select id="to-currency"></select>

    </div>

    <button id="convert-btn">Convert</button>

</div>

<div id="result"></div>

</div>

<script src="script.js"></script>

</body>

</html>
```

## CSS CODE:

```
body{

    font-family: Arial, sans-serif;

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

    margin: 0;

    background-color: #f0f0f0;

}
```

```
.container {  
    background: #fff;  
    padding: 20px;  
    border-radius: 8px;  
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
    text-align: center;  
}
```

```
.converter div {  
    margin-bottom: 10px;  
}
```

```
input, select {  
    padding: 8px;  
    width: 200px;  
}
```

```
button {  
    padding: 10px 20px;  
    background-color: #007bff;  
    color: white;  
    border: none;
```

```
border-radius: 5px;

cursor: pointer;
}

button:hover {
  background-color: #0056b3;
}

#result {
  margin-top: 20px;
  font-size: 1.2em;
}
```

## **JAVA SCRIPT CODE:**

```
const apiKey = 'YOUR_API_KEY';

const apiUrl = `https://open.exchangerate-api.com/v6/latest`;

// Elements

const amountEl = document.getElementById('amount');

const fromCurrencyEl = document.getElementById('from-currency');

const toCurrencyEl = document.getElementById('to-currency');
```



```
const convertBtn = document.getElementById('convert-btn');  
const resultEl = document.getElementById('result');
```

```
// Fetch exchange rates and populate currency options
```

```
async function populateCurrencyOptions() {  
  try {  
    const response = await fetch(apiUrl);  
    const data = await response.json();  
  
    const currencies = Object.keys(data.rates);  
  
    currencies.forEach(currency => {  
      const option1 = document.createElement('option');  
      const option2 = document.createElement('option');  
      option1.value = option2.value = currency;  
      option1.textContent = option2.textContent = currency;  
      fromCurrencyEl.appendChild(option1);  
      toCurrencyEl.appendChild(option2);  
    });  
  } catch (error) {  
    console.error('Error fetching exchange rates:', error);  
  }  
}
```

```
// Perform conversion
```

```
async function convertCurrency() {
```

```
const amount = parseFloat(amountEl.value);
const fromCurrency = fromCurrencyEl.value;
const toCurrency = toCurrencyEl.value;

if (isNaN(amount) || fromCurrency === '' || toCurrency === '') {
  resultEl.textContent = 'Please enter a valid amount and select currencies.';
  return;
}

try {
  const response = await fetch(apiUrl);
  const data = await response.json();

  const fromRate = data.rates[fromCurrency];
  const toRate = data.rates[toCurrency];
  const convertedAmount = (amount / fromRate) * toRate;

  resultEl.textContent = `${amount} ${fromCurrency} = ${convertedAmount.toFixed(2)}
  ${toCurrency}`;
} catch (error) {
  console.error('Error converting currency:', error);
}

convertBtn.addEventListener('click', convertCurrency);
```

```
// Initialize currency options on page load  
populateCurrencyOptions();
```

## OUTPUT:

The screenshot displays the OneCompiler web interface. On the left, the `script.js` file contains the following JavaScript code:

```
30  
31  
32 Perform conversion  
33 *  
34 *  
35 *  
36 *  
37 *  
38 *  
39 *  
40 *  
41 *  
42 *  
43 *  
44 *  
45 *  
46 *  
47 *  
48 *  
49 *  
50 *  
51 *  
52 *  
53 *  
54 *  
55 *  
56 *  
57 *  
58 *  
59 *  
60 *  
61 *  
62 *  
63 *  
64 *  
65 *  
66 *  
67 *  
68 *  
69 *  
70 *  
71 *  
72 *  
73 *  
74 *  
75 *  
76 *  
77 *  
78 *  
79 *  
80 *  
81 *  
82 *  
83 *  
84 *  
85 *  
86 *  
87 *  
88 *  
89 *  
90 *  
91 *  
92 *  
93 *  
94 *  
95 *  
96 *  
97 *  
98 *  
99 *  
100 *
```

The output on the right shows a "Currency Converter" form with the following fields and values:

- Amount: 200
- From: USD
- To: INR
- Convert button
- Result: 200 USD = 16700.48 INR