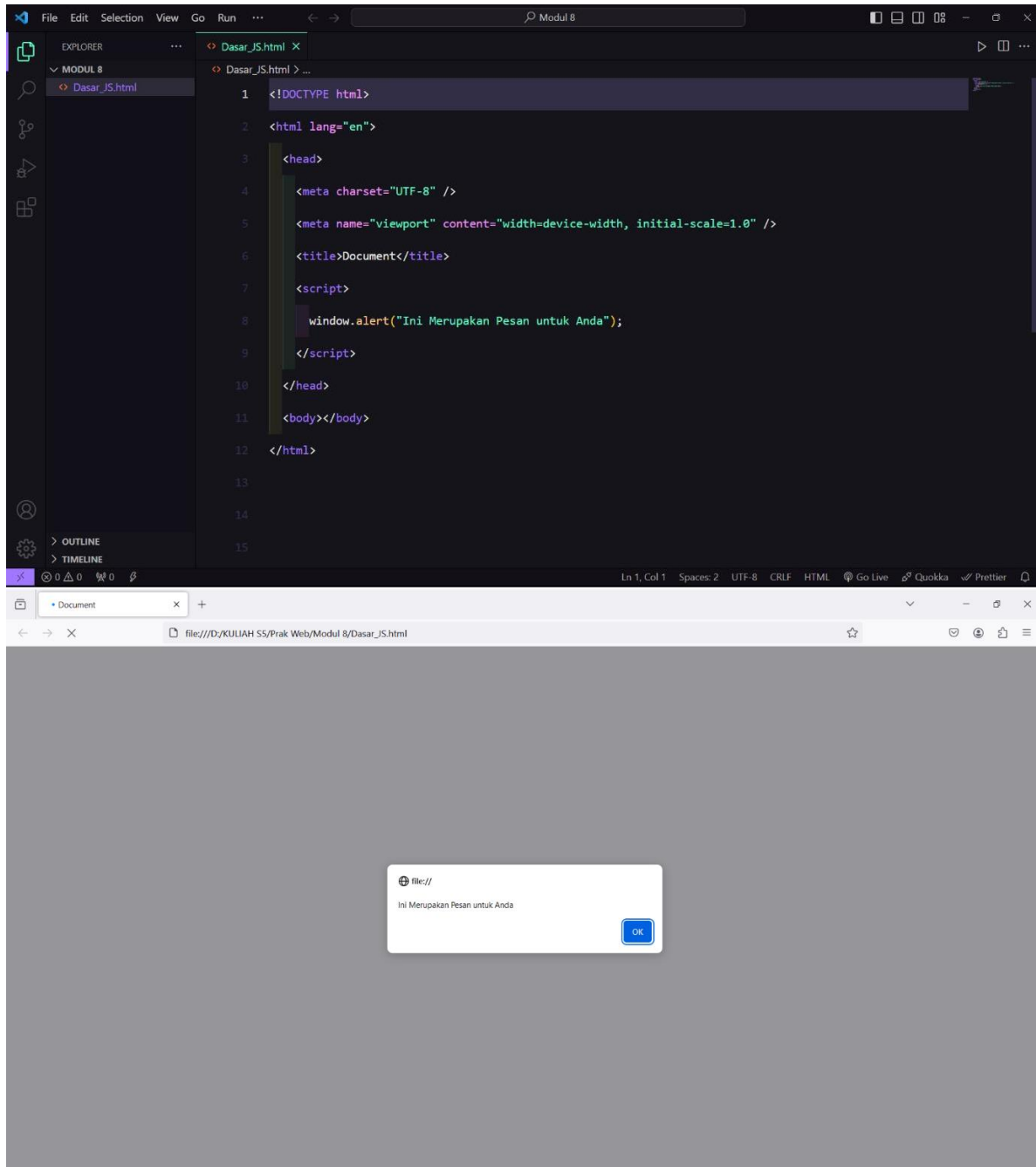


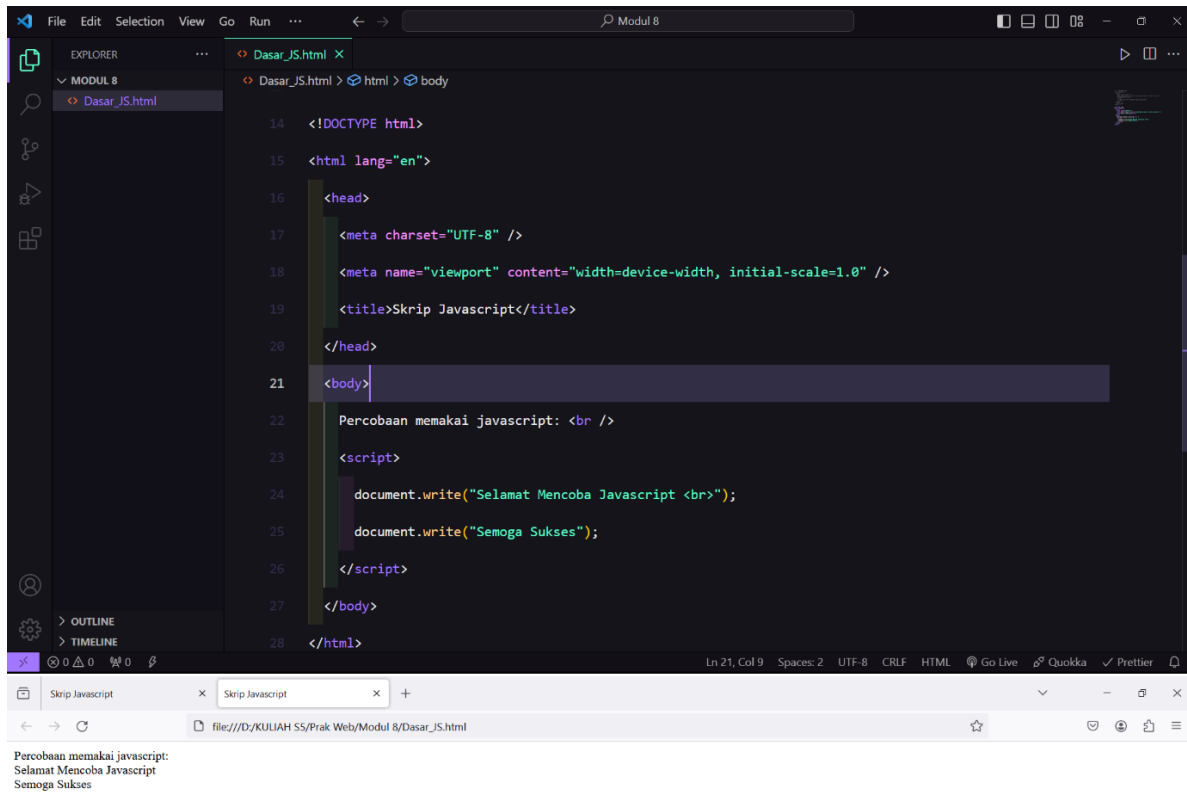
Modul 8 Javascript

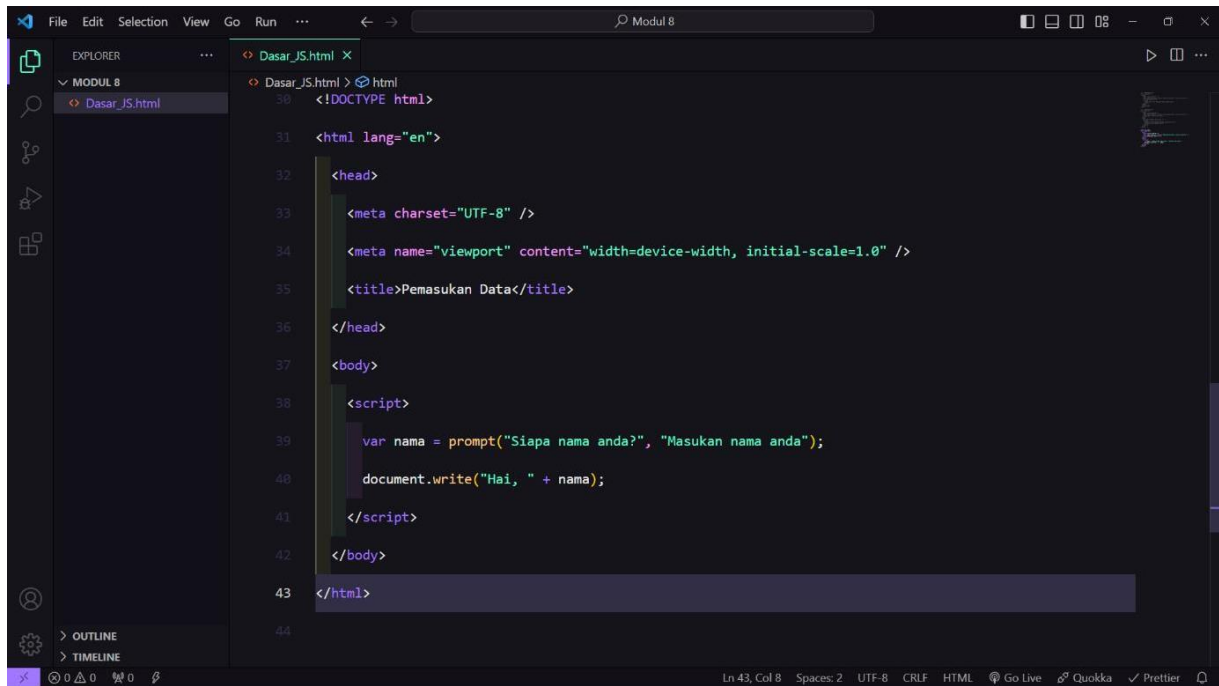
Nama : Valliant Dheka F

NIM : 2105101019

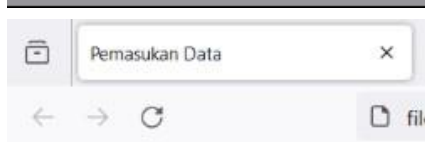
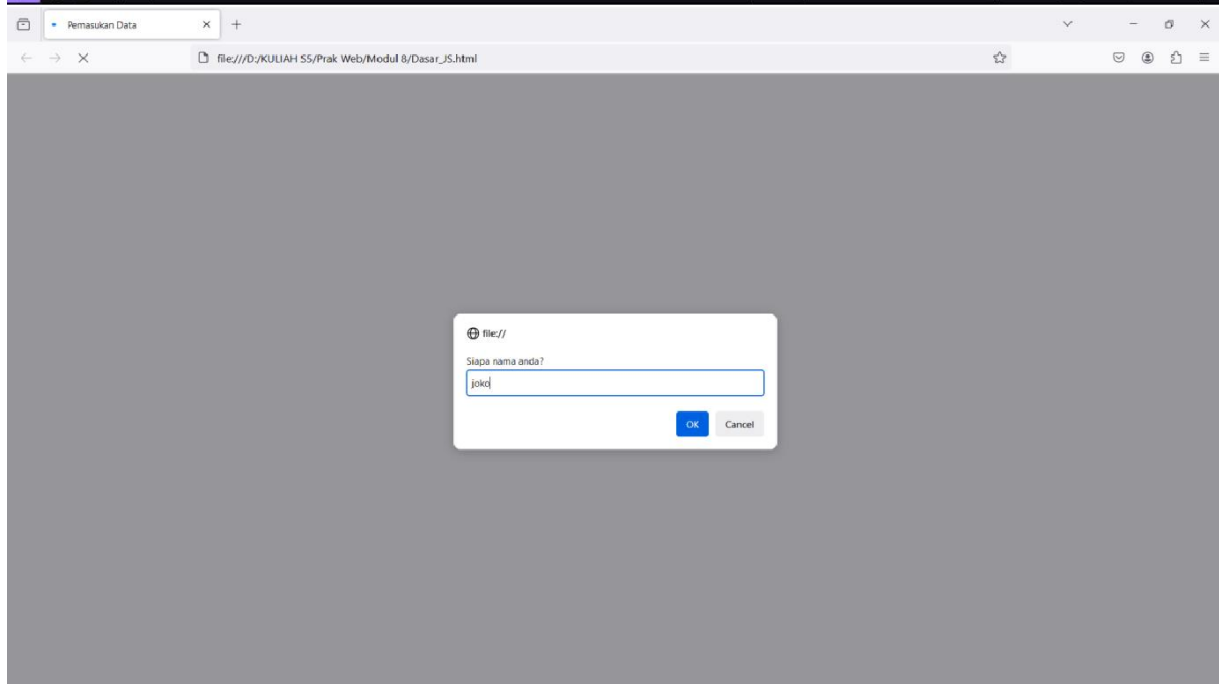
Dasar – Dasar Javascript





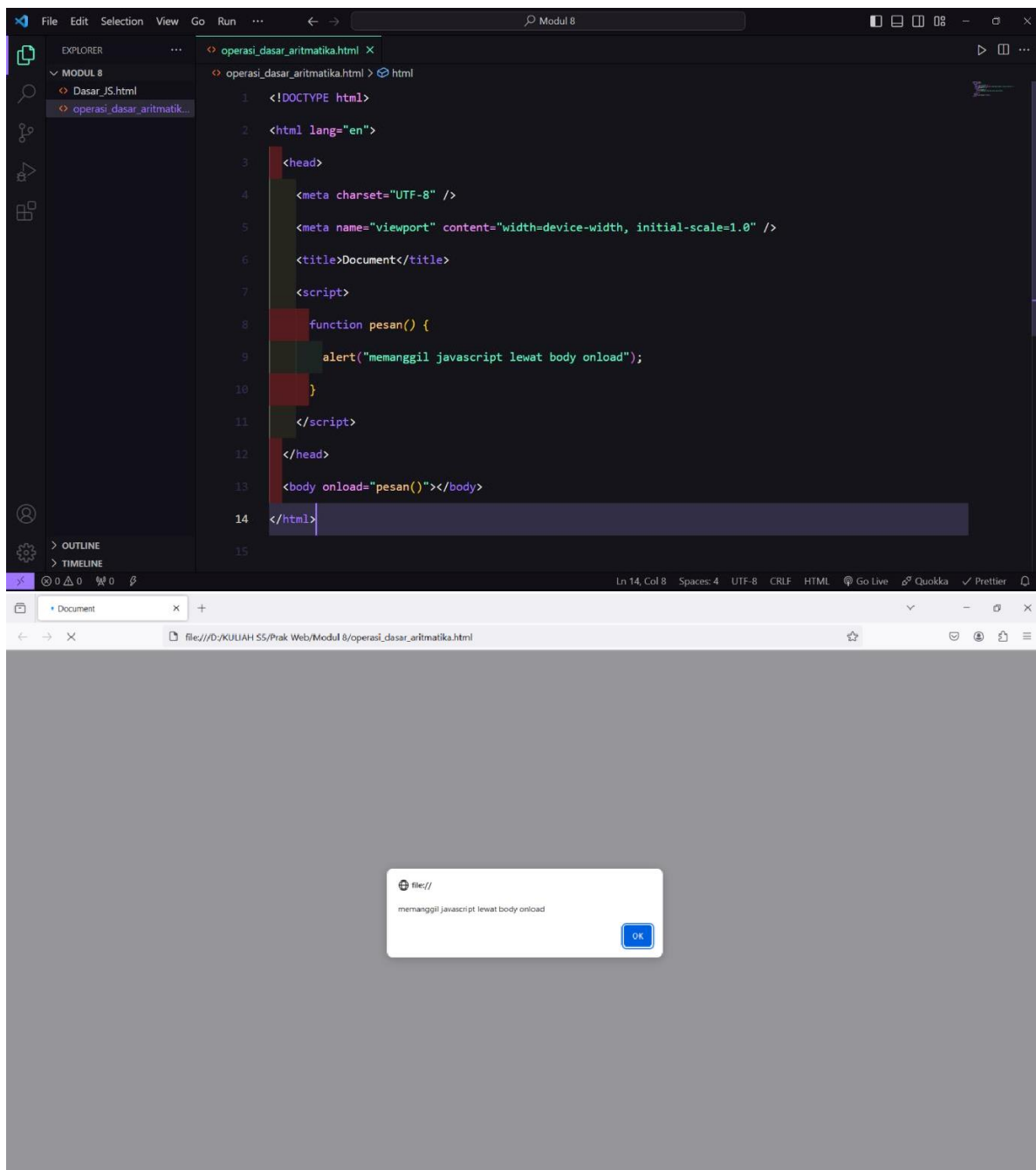


```
30 <!DOCTYPE html>
31 <html lang="en">
32 <head>
33   <meta charset="UTF-8" />
34   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
35   <title>Pemasukan Data</title>
36 </head>
37 <body>
38   <script>
39     var nama = prompt("Siapa nama anda?", "Masukan nama anda");
40     document.write("Hai, " + nama);
41   </script>
42 </body>
43 </html>
```



Hai, joko

Operasi Dasar Aritmatika



The screenshot shows a VS Code editor with a dark theme. The Explorer sidebar on the left displays the file structure: 'MODUL 8' contains 'Dasar_JS.html' and 'operasi_dasar_aritmatika.html'. The main editor area shows the content of 'operasi_dasar_aritmatika.html', which is an HTML file with a script section. The script defines a function 'test(val1, val2)' that performs multiplication, division, addition, subtraction, and modulus operations, displaying the results using 'document.write'.

```

<script>
  function test(val1, val2){
    document.write("<br> " + "Perkalian : val1*val2 "+"<br>")
    document.write(val1*val2)
    document.write("<br> " + "Pembagian : val1/val2 "+"<br>")
    document.write(val1/val2)
    document.write("<br> " + "Penjumlahan : val1+val2 "+"<br>")
    document.write(val1+val2)
    document.write("<br> " + "Pengurangan : val1-val2 "+"<br>")
    document.write(val1-val2)
    document.write("<br> " + "Modulus : val1%val2 "+"<br>")
    document.write(val1%val2)
  }
</script>
</head>

```

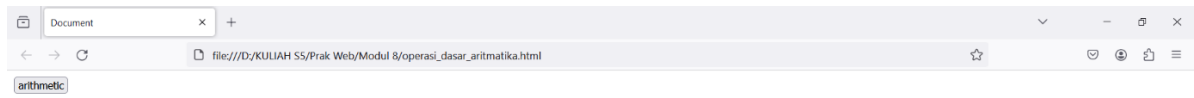
The screenshot shows the VS Code editor interface. The Explorer sidebar on the left lists the files: 'MODUL 8', 'Dasar_JS.html', and 'operasi_dasar_aritmatika.html'. The main editor area displays the content of 'operasi_dasar_aritmatika.html', which includes a head section with a title and a body section containing a button element. The button has the attributes 'type="button"', 'name="button1"', 'value="arithmetic"', and an 'onclick="test(9,4)" event.

```

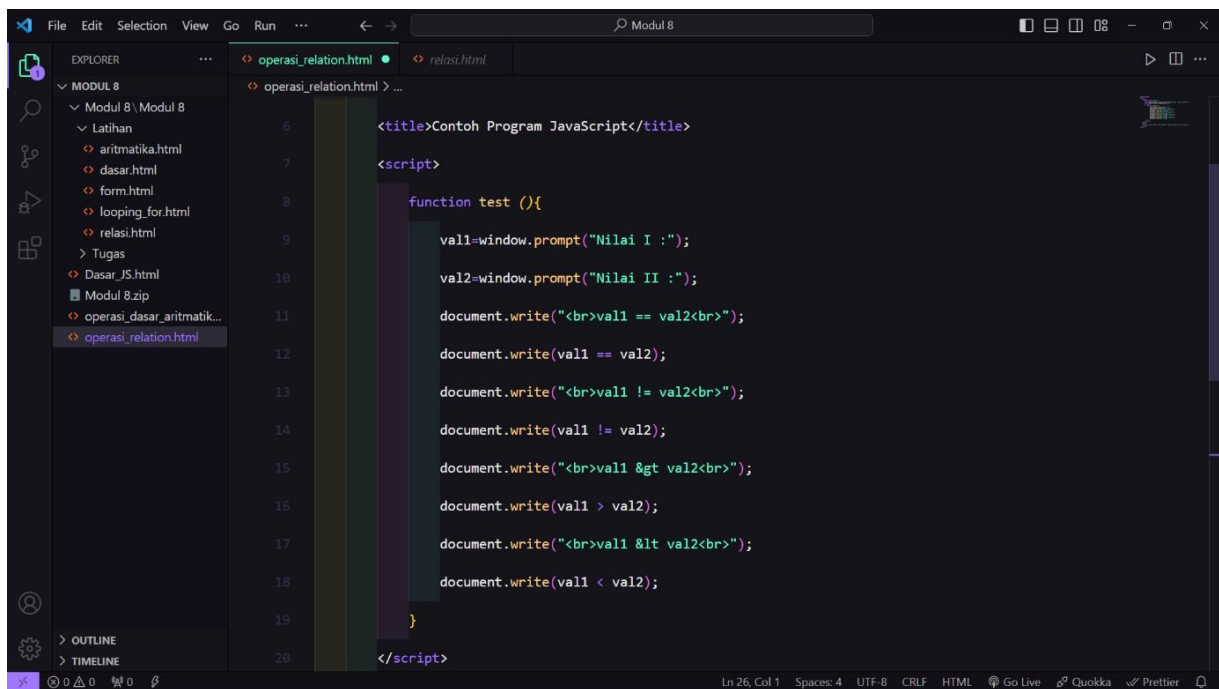
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>Dasar Aritmatika</title>
5  </head>
6  <body>
7    <input type="button" name="button1" value="arithmetic" onclick=test(9,4)>
8  </body>
9  </html>

```

The status bar at the bottom indicates the current position is Line 31, Column 30, with 4 spaces, UTF-8 encoding, CRLF line endings, and the HTML language mode.

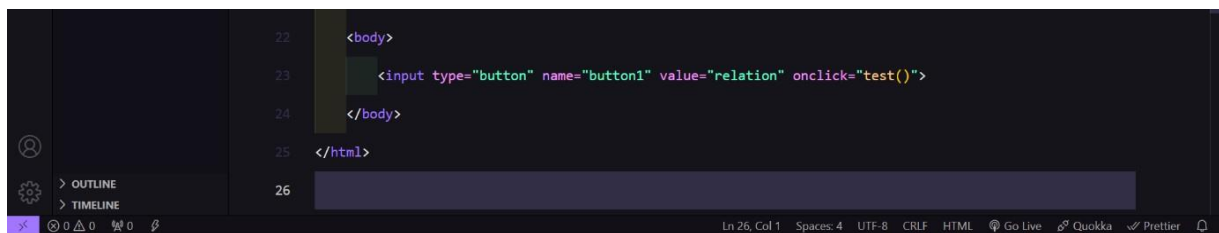


Operasi relational



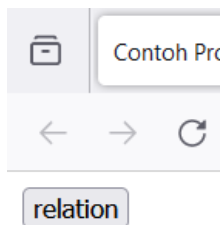
The screenshot shows the VS Code editor with the file `operasi_relation.html` open. The code defines a `test()` function that prompts the user for two values, `val1` and `val2`, and then displays the results of various relational operations using `document.write()`.

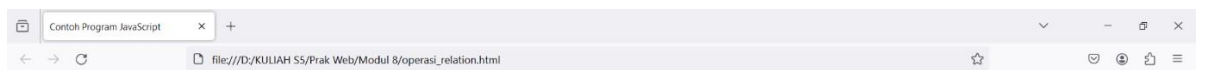
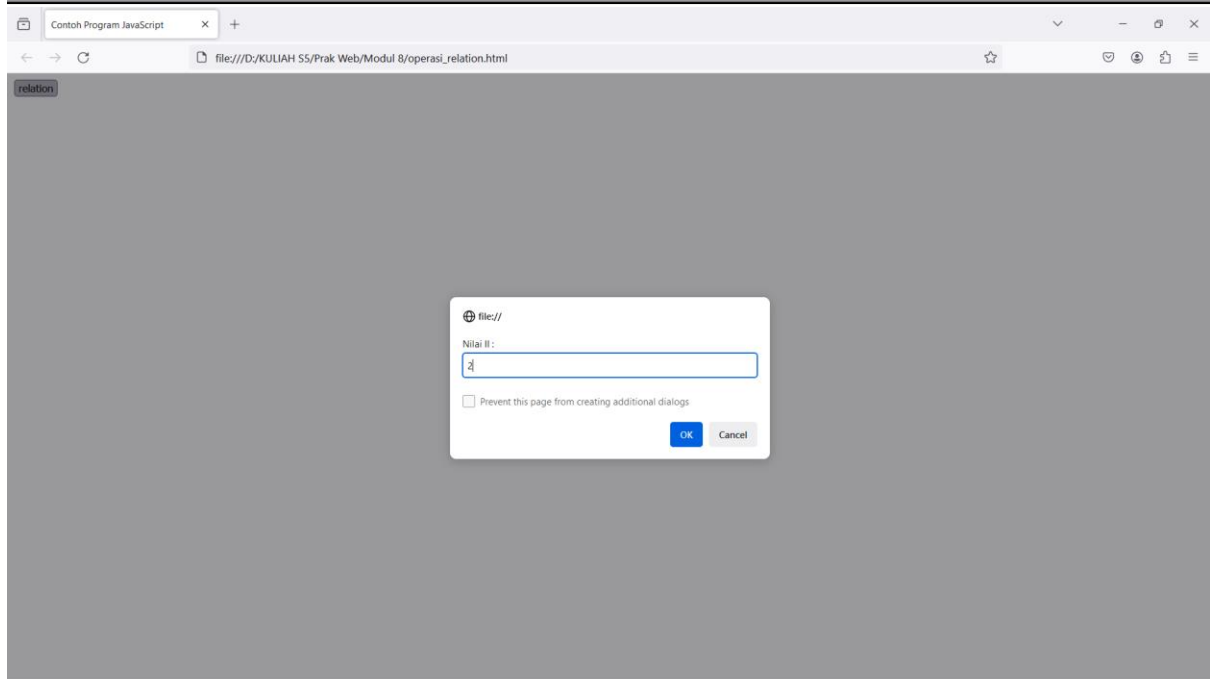
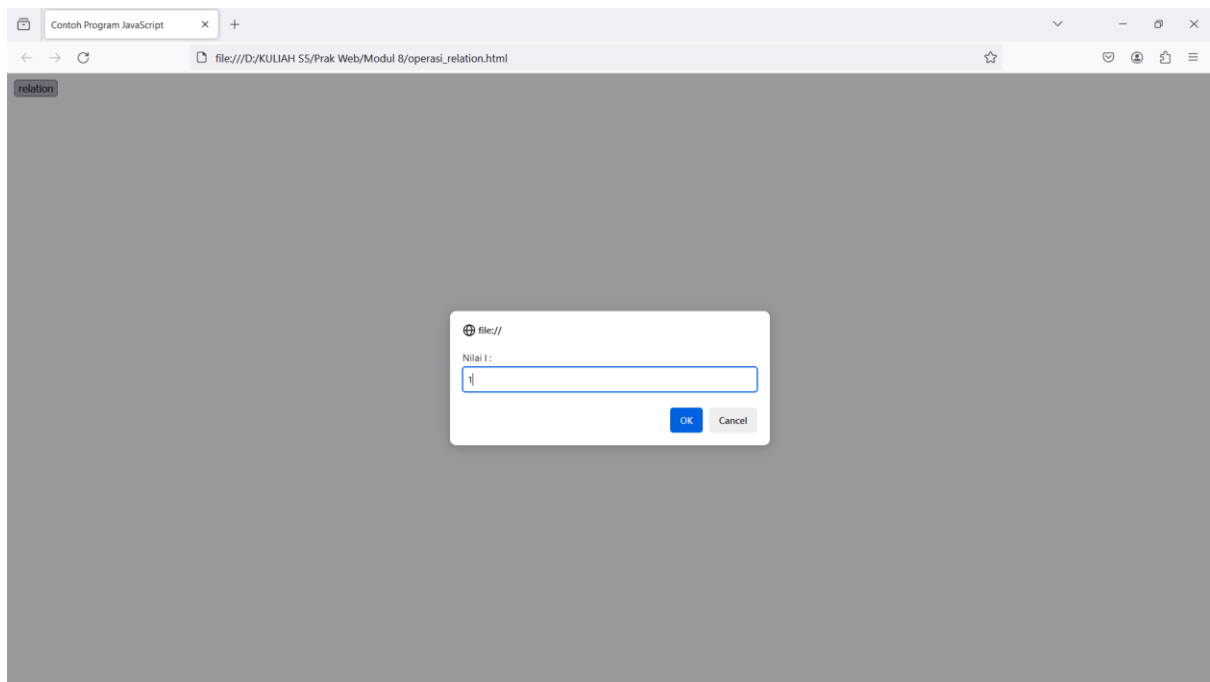
```
6 <title>Contoh Program JavaScript</title>
7 <script>
8     function test (){
9         val1=window.prompt("Nilai I :");
10        val2=window.prompt("Nilai II :");
11        document.write("<br>val1 == val2<br>");
12        document.write(val1 == val2);
13        document.write("<br>val1 != val2<br>");
14        document.write(val1 != val2);
15        document.write("<br>val1 &gt; val2<br>");
16        document.write(val1 > val2);
17        document.write("<br>val1 &lt; val2<br>");
18        document.write(val1 < val2);
19    }
20 </script>
```



The screenshot shows the VS Code editor with the file `operasi_relation.html` open, displaying the HTML structure. It includes a `<body>` tag with an `<input>` button that calls the `test()` function on click, followed by `</body>` and `</html>`.

```
22 <body>
23     <input type="button" name="button1" value="relation" onclick="test()">
24 </body>
25 </html>
26
```

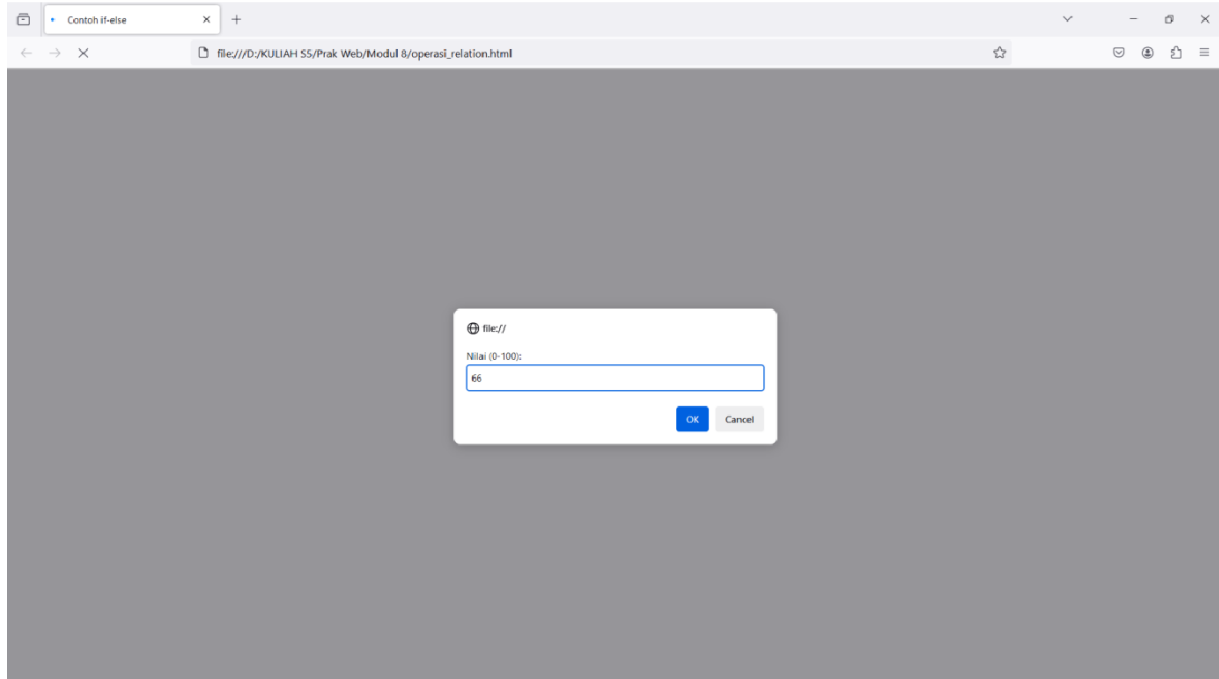




```
val1 == val2  
false  
val1 != val2  
true  
val1 > val2  
false  
val1 < val2  
true
```

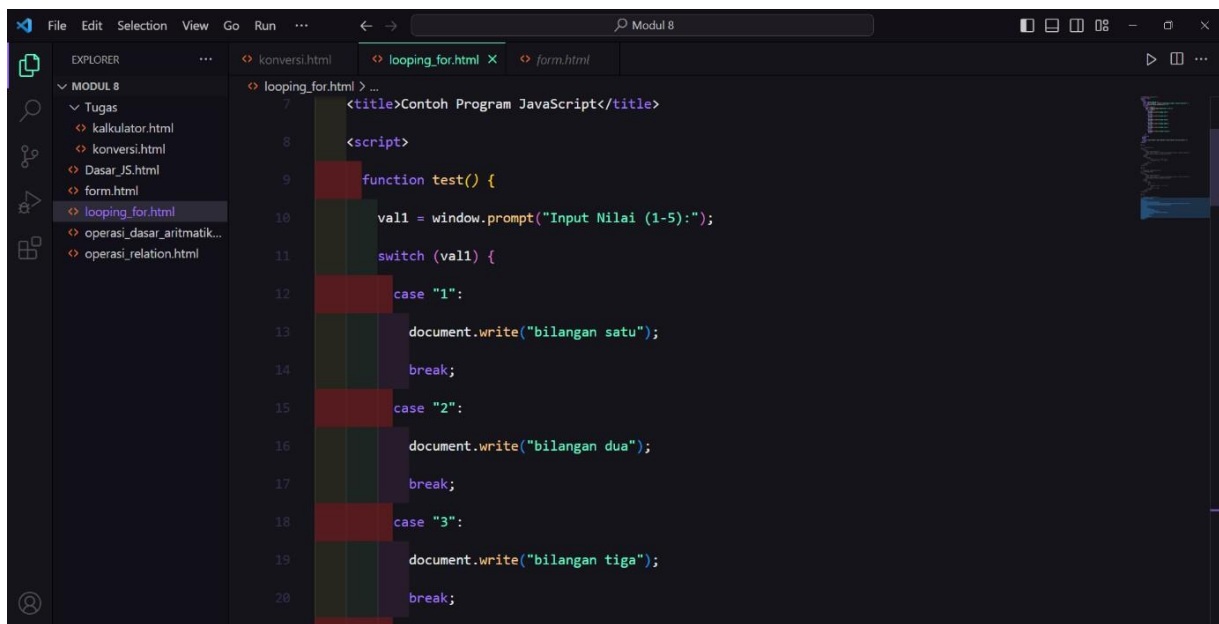


```
33 </head>
34 <body>
35 <script>
36     var nilai = prompt("Nilai (0-100): ", 0);
37     var hasil = "";
38     if (nilai >= 60) {
39         hasil = "Lulus";
40     } else {
41         hasil = "Tidak Lulus";
42     }
43     document.write("Hasil: " + hasil);
44 </script>
45 </body>
46 </html>
47 TIF FT UNIPMA
```

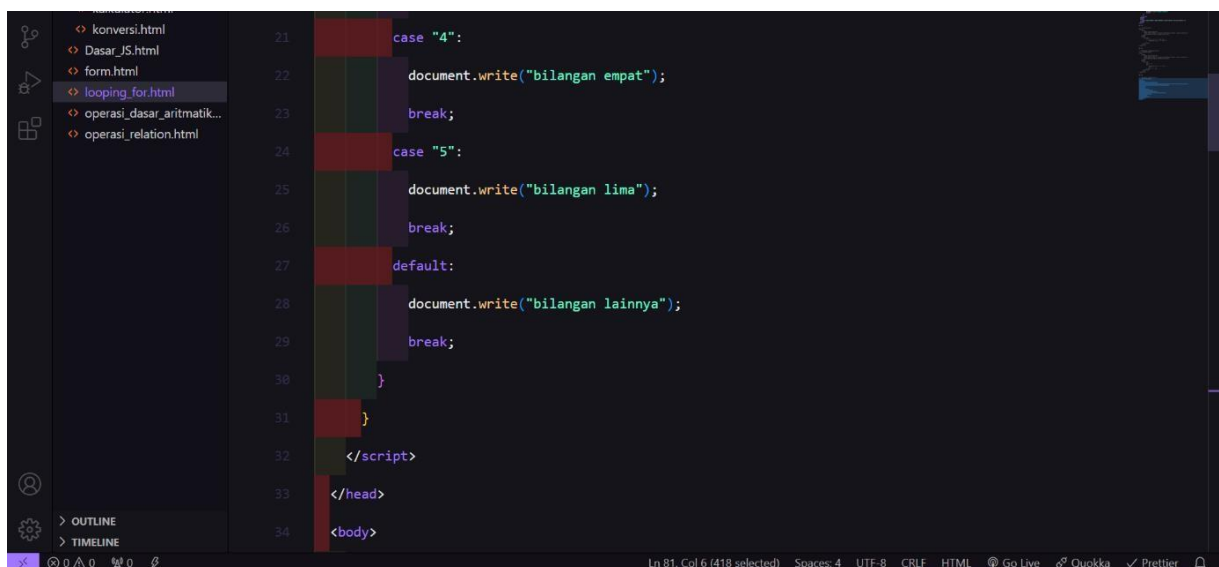


Hasil: Lulus TIF FT UNIPMA

Looping for



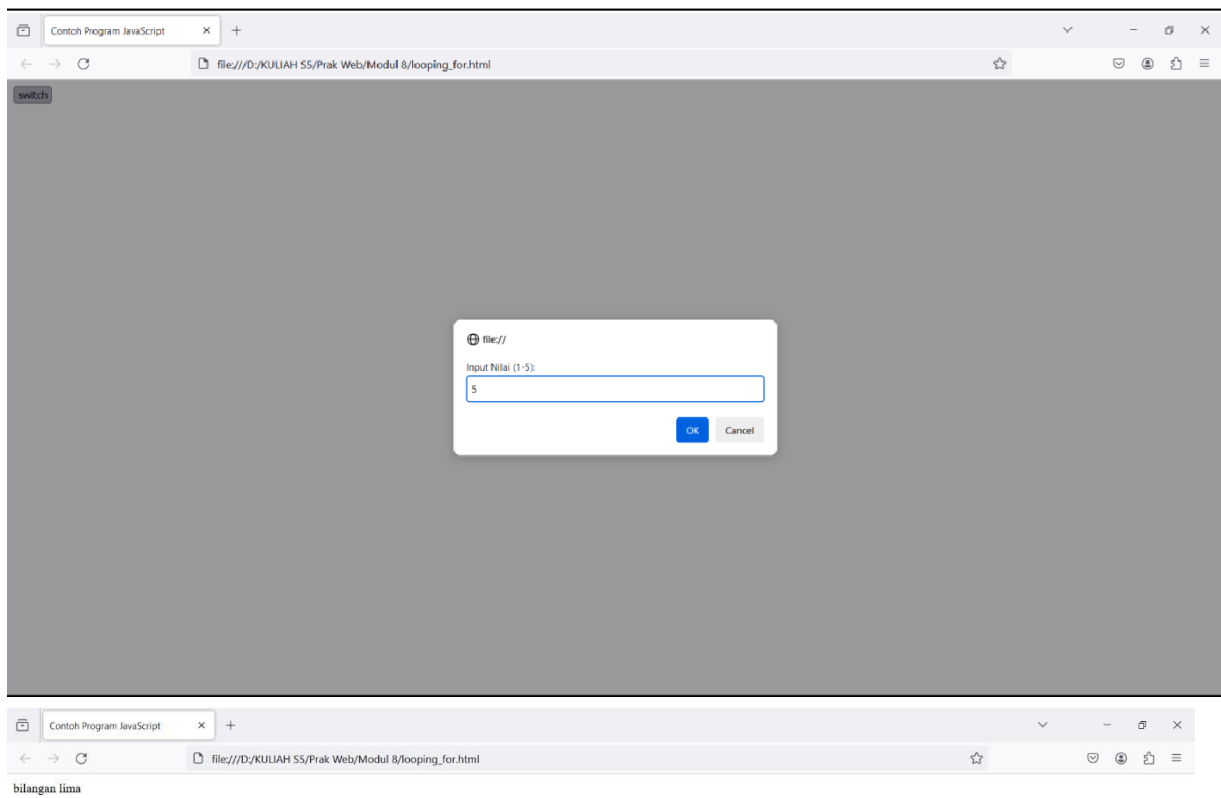
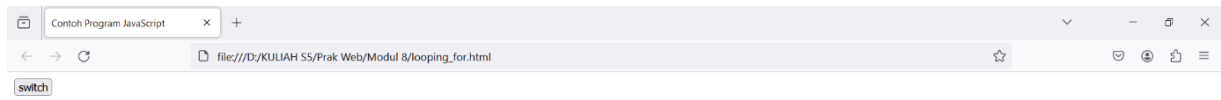
```
7 <title>Contoh Program JavaScript</title>
8
9 <script>
10 function test() {
11     val1 = window.prompt("Input Nilai (1-5):");
12     switch (val1) {
13         case "1":
14             document.write("bilangan satu");
15             break;
16         case "2":
17             document.write("bilangan dua");
18             break;
19         case "3":
20             document.write("bilangan tiga");
```



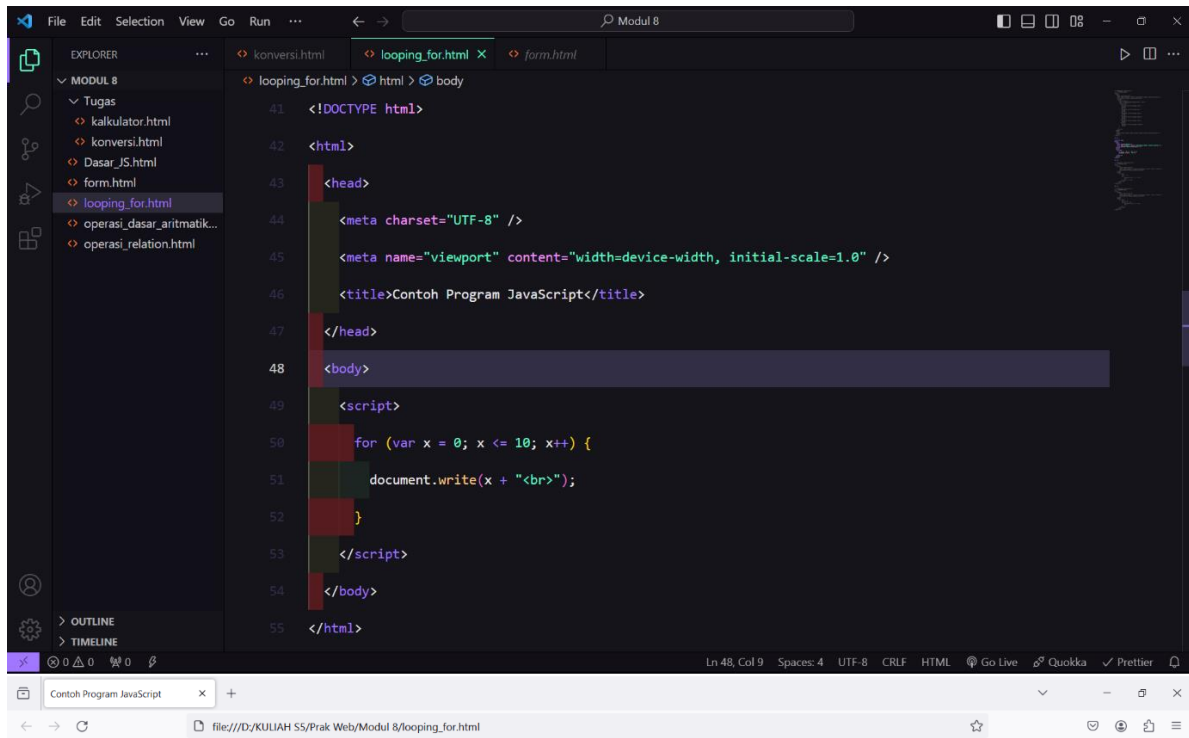
```
21         case "4":
22             document.write("bilangan empat");
23             break;
24         case "5":
25             document.write("bilangan lima");
26             break;
27         default:
28             document.write("bilangan lainnya");
29             break;
30     }
31 }
32 </script>
33 </head>
34 <body>
```



```
35 <input type="button" name="button1" value="switch" onclick="test()" />
36 </body>
37 </html>
38
39 <br />
```

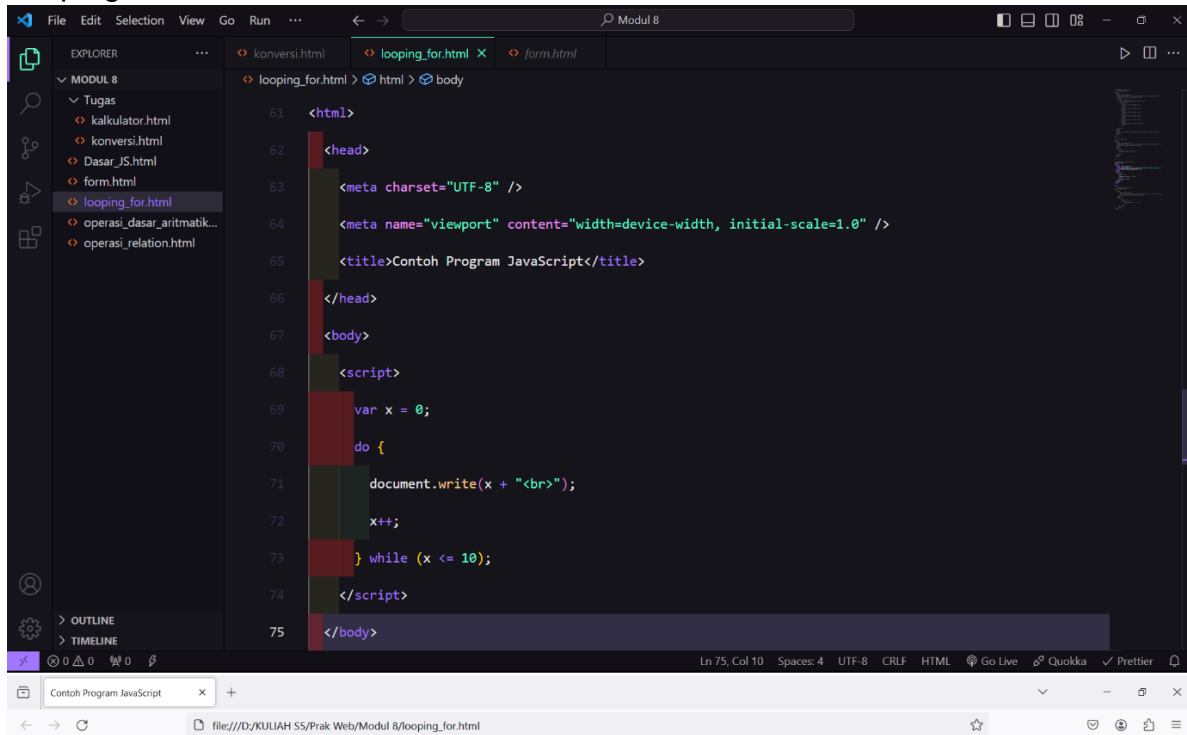


bilangan lima



0
1
2
3
4
5
6
7
8
9
10

Looping do while



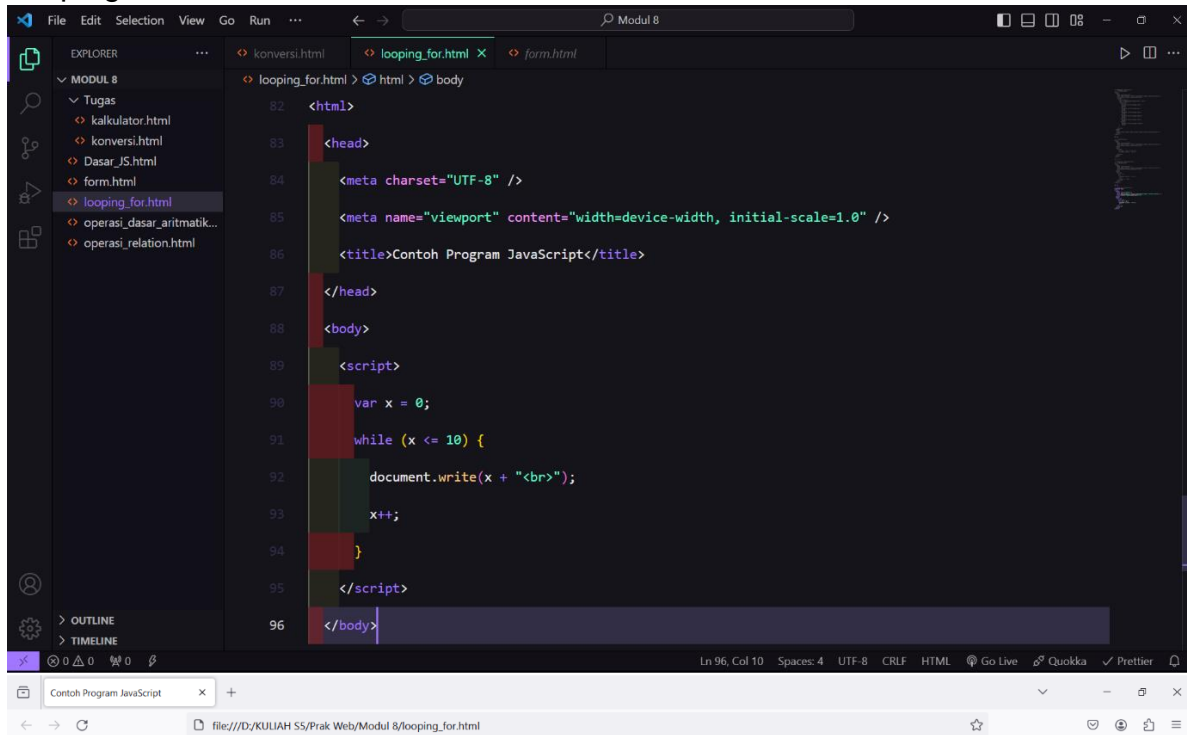
The screenshot shows a Visual Studio Code editor window with a dark theme. The Explorer sidebar on the left shows a project structure for 'MODUL 8' with files like 'kalkulator.html', 'konversi.html', 'Dasar_JS.html', 'form.html', 'looping_for.html', 'operasi_dasar_aritmatik...', and 'operasi_relation.html'. The 'looping_for.html' file is open in the main editor. The code in the editor is as follows:

```
61 <html>
62 <head>
63   <meta charset="UTF-8" />
64   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
65   <title>Contoh Program JavaScript</title>
66 </head>
67 <body>
68   <script>
69     var x = 0;
70     do {
71       document.write(x + "<br>");
72       x++;
73     } while (x <= 10);
74   </script>
75 </body>
```

The status bar at the bottom indicates 'Ln 75, Col 10', 'Spaces: 4', 'UTF-8', 'CRLF', 'HTML', 'Go Live', 'Quokka', and 'Prettier'. Below the editor, a web browser window is visible, showing the title 'Contoh Program JavaScript' and the file path 'file:///D:/KULIAH SS/Prak Web/Modul 8/looping_for.html'.

0
1
2
3
4
5
6
7
8
9
10

Looping while



The screenshot shows a Visual Studio Code editor window with a file explorer on the left and a code editor in the center. The file explorer shows a project named 'MODUL 8' with a subfolder 'Tugas' containing several HTML files. The active file is 'looping_for.html'. The code editor shows the following HTML structure:

```
82 <html>
83   <head>
84     <meta charset="UTF-8" />
85     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
86     <title>Contoh Program JavaScript</title>
87   </head>
88   <body>
89     <script>
90       var x = 0;
91       while (x <= 10) {
92         document.write(x + "<br>");
93         x++;
94       }
95     </script>
96   </body>
```

The status bar at the bottom indicates the current line and column (Ln 96, Col 10), encoding (UTF-8), line endings (CRLF), and the file type (HTML). The bottom of the image shows a browser window with the address bar displaying the file path: file:///D:/KULIAH SS/Prak Web/Modul 8/looping_for.html.

0
1
2
3
4
5
6
7
8
9
10

Form input

The image shows a Visual Studio Code editor window with a dark theme. The Explorer sidebar on the left shows a project structure for 'MODUL 8' with files like 'kalkulator.html', 'konversi.html', 'Dasar_JS.html', 'form.html', 'looping_for.html', 'operasi_dasar_aritmatik...', and 'operasi_relation.html'. The 'form.html' file is selected and open in the main editor. The code in the editor is as follows:

```
9 <body>
10 <script>
11 function test() {
12     var val1 = document.kirim.T1.value;
13     if (val1 % 2 == 0) document.kirim.T2.value = "bilangan genap";
14     else document.kirim.T2.value = "bilangan ganjil";
15 }
16 </script>
17 </body>
18 <form method="post" name="kirim">
19     <p>BIL <input type="text" name="T1" size="20" />Merupakan Bil <input type="text" name="T2" size="20"
20     <p><input type="button" value="TEBAK" name="B1" onclick="test()" /></p>
21 </form>
22 </html>
23
```

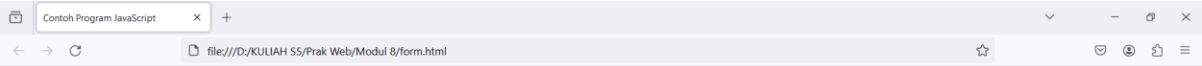
Below the editor, a browser window is open showing the rendered form. The address bar shows the file path: 'file:///D:/KULIAH SS/Prak Web/Modul 8/form.html'. The form contains a text input field labeled 'BIL' with the value '1', followed by the text 'Merupakan Bil', another text input field with the value 'bilangan ganjil', and a button labeled 'TEBAK'.

Form Button

The image shows a development environment with VS Code and a web browser. In VS Code, the file `form.html` is open, displaying the following code:

```
35 <script>
36     function ubahWarnaLB(warna) {
37         document.body.style.backgroundColor = warna;
38     }
39     function ubahWarnaLD(warna) {
40         document.body.style.color = warna;
41     }
42 </script>
43 <h1>TES</h1>
44 <form>
45     <input type="button" value="Latar Belakang Hijau" onclick="ubahWarnaLB('green')">
46     <input type="button" value="Latar Belakang Putih" onclick="ubahWarnaLB('white')">
47     <input type="button" value="Text Kuning" onclick="ubahWarnaLD('yellow')">
48     <input type="button" value="Text Biru" onclick="ubahWarnaLD('blue')">
49 </form>
```

The web browser below shows the rendered page. The background is green, and the text "TES" is displayed in a large font. Below the text, there are four buttons: "Latar Belakang Hijau", "Latar Belakang Putih", "Text Kuning", and "Text Biru".



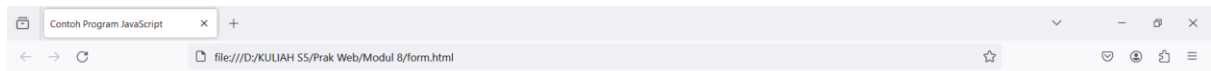
TES

Latar Belakang Hijau Latar Belakang Putih Text Kuning Text Biru



TES

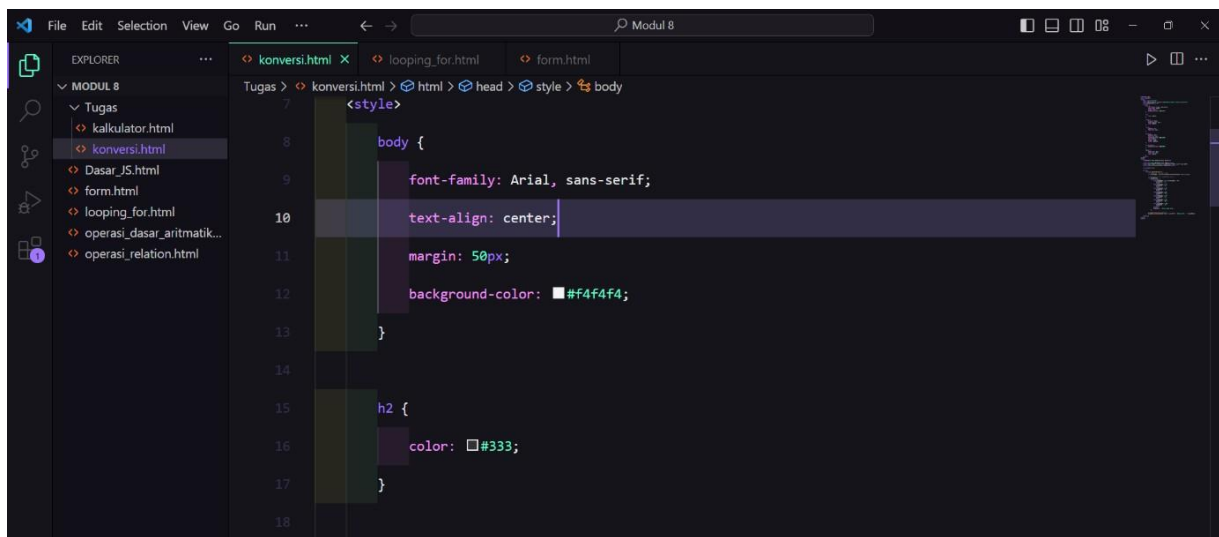
Latar Belakang Hijau Latar Belakang Putih Text Kuning Text Biru



TES

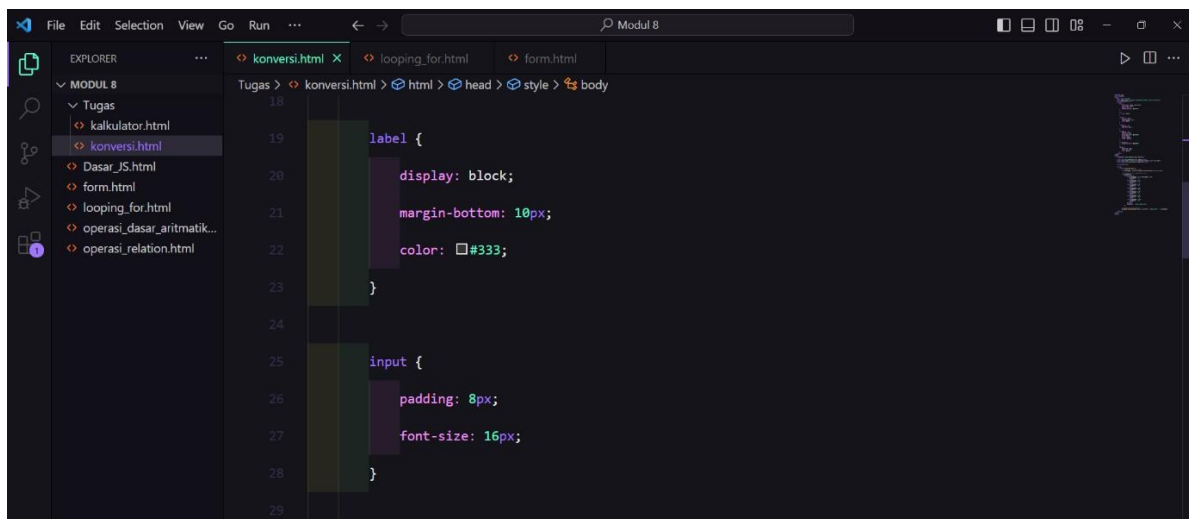
Latar Belakang Hijau Latar Belakang Putih Text Kuning Text Biru

Tugas No 1



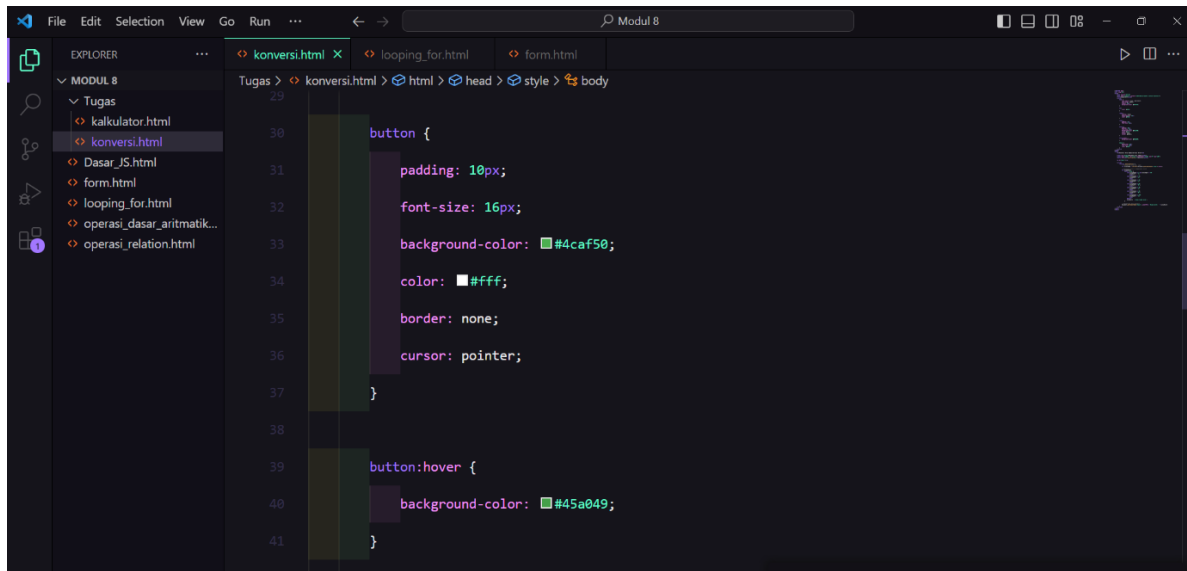
This screenshot shows the Visual Studio Code editor with a file named `konversi.html` open. The breadcrumb navigation at the top indicates the path: `Tugas > konversi.html > html > head > style > body`. The Explorer sidebar on the left shows a project structure for 'MODUL 8' containing a 'Tugas' folder with several HTML files, including `konversi.html`. The main editor area displays CSS code for the `body` and `h2` elements. The `body` selector is defined with a font family, text alignment, margin, and background color. The `h2` selector is defined with a color.

```
7 <style>
8
9     body {
10         font-family: Arial, sans-serif;
11         text-align: center;
12         margin: 50px;
13         background-color: #f4f4f4;
14     }
15
16     h2 {
17         color: #333;
18     }
```



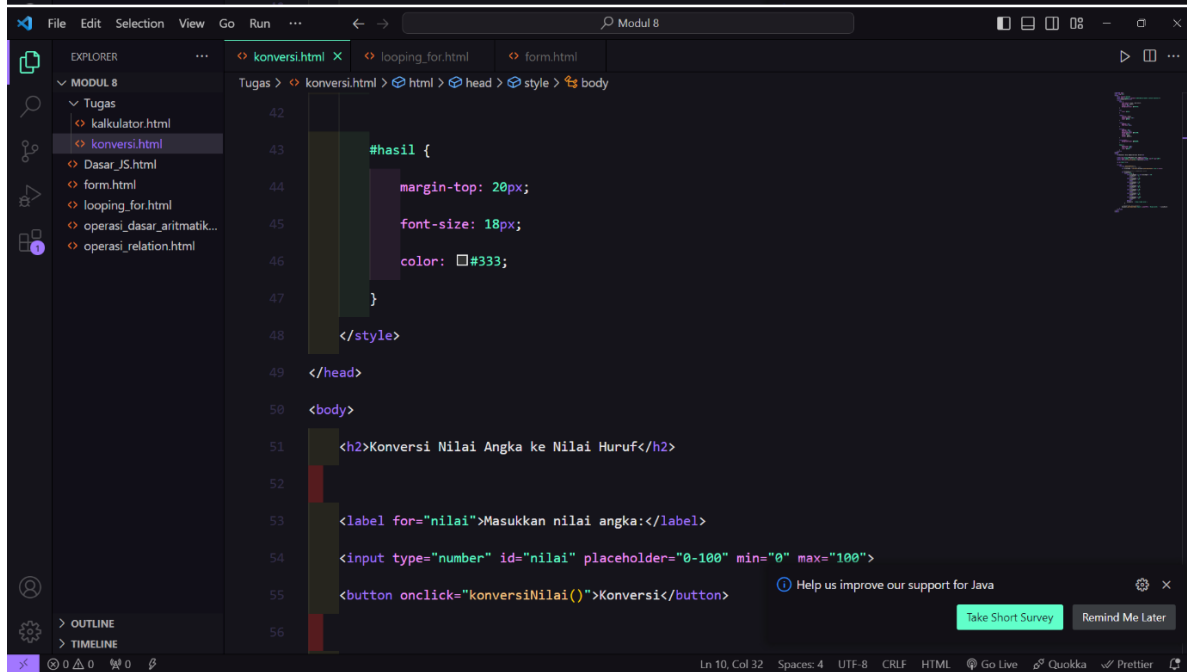
This screenshot shows the Visual Studio Code editor with the same `konversi.html` file open. The breadcrumb navigation at the top indicates the path: `Tugas > konversi.html > html > head > style > body`. The Explorer sidebar on the left shows the same project structure. The main editor area displays CSS code for the `label` and `input` elements. The `label` selector is defined with display, margin-bottom, and color. The `input` selector is defined with padding and font-size.

```
18
19     label {
20         display: block;
21         margin-bottom: 10px;
22         color: #333;
23     }
24
25     input {
26         padding: 8px;
27         font-size: 16px;
28     }
29
```



This screenshot shows the VS Code editor with the file explorer on the left displaying a project structure under 'MODUL 8' including files like 'kalkulator.html', 'konversi.html', 'Dasar_JS.html', 'form.html', 'looping_for.html', 'operasi_dasar_aritmatik...', and 'operasi_relation.html'. The main editor window shows the 'konversi.html' file with the following CSS code:

```
29
30     button {
31         padding: 10px;
32         font-size: 16px;
33         background-color: #4caf50;
34         color: #fff;
35         border: none;
36         cursor: pointer;
37     }
38
39     button:hover {
40         background-color: #45a049;
41     }
```



This screenshot shows the same VS Code editor with the 'konversi.html' file. The main editor window displays the following HTML code:

```
42
43     #hasil {
44         margin-top: 20px;
45         font-size: 18px;
46         color: #333;
47     }
48 </style>
49 </head>
50 <body>
51     <h2>Konversi Nilai Angka ke Nilai Huruf</h2>
52
53     <label for="nilai">Masukkan nilai angka:</label>
54     <input type="number" id="nilai" placeholder="0-100" min="0" max="100">
55     <button onclick="konversiNilai()">Konversi</button>
56
```

At the bottom of the editor, there is a status bar showing 'Ln 10, Col 32', 'Spaces: 4', 'UTF-8', 'CRLF', 'HTML', and icons for 'Go Live', 'Quokka', and 'Prettier'. A notification banner at the bottom right says 'Help us improve our support for Java' with buttons for 'Take Short Survey' and 'Remind Me Later'.

This screenshot shows the first part of the JavaScript code in the file 'konversi.html'. The Explorer sidebar on the left shows a project structure with 'MODUL 8' containing 'Tugas' and several HTML files. The breadcrumb at the top of the editor indicates the current path: 'Tugas > konversi.html > html > head > style > body'. The code in the editor includes a closing paragraph tag, a script tag, and the start of a function named 'konversiNilai()' which contains a comment about getting input and a variable declaration for 'nilaiAngka'.

```
56
57 <p id="hasil"></p>
58
59 <script>
60     function konversiNilai() {
61         // Mendapatkan nilai dari input
62         var nilaiAngka = parseInt(document.getElementById('nilai').value);
63
64         // Melakukan konversi menggunakan switch
65         var nilaiHuruf;
66         switch (true) {
67             case nilaiAngka >= 0 && nilaiAngka <= 40:
68                 nilaiHuruf = 'E';
69                 break;
```

This screenshot shows the continuation of the 'konversiNilai()' function. It contains several 'case' statements in the switch block, each assigning a letter grade to 'nilaiHuruf' based on the value of 'nilaiAngka'. The cases are for ranges 40-55 (E), 55-60 (D), 60-65 (C), 65-70 (BC), 70-80 (B), and 80-85 (AB). Each case is followed by a 'break;' statement. The status bar at the bottom indicates the current cursor position is at line 10, column 32.

```
69         break;
70         case nilaiAngka <= 55:
71             nilaiHuruf = 'D';
72             break;
73         case nilaiAngka <= 60:
74             nilaiHuruf = 'C';
75             break;
76         case nilaiAngka <= 65:
77             nilaiHuruf = 'BC';
78             break;
79         case nilaiAngka <= 70:
80             nilaiHuruf = 'B';
81             break;
82         case nilaiAngka <= 80:
83             nilaiHuruf = 'AB';
```

kalkulator.html

konversi.html

Dasar_JS.html

form.html

looping_for.html

operasi_dasar_aritmatik...

operasi_relation.html

82

83

84

85

86

87

88

89

90

91

92

93

94

95

OUTLINE

TIMELINE

```
case nilaiAngka <= 80:
    nilaiHuruf = 'AB';
    break;
case nilaiAngka <= 100:
    nilaiHuruf = 'A';
    break;
default:
    nilaiHuruf = 'Nilai tidak valid';
}

// Menampilkan hasil konversi
document.getElementById('hasil').innerHTML = 'Nilai huruf: ' + nilaiHuruf;
}
```

Help us improve our support for Java

Take Short Survey

Remind Me Later

Ln 10, Col 32 Spaces: 4 UTF-8 CRLF HTML Go Live Quokka Prettier

Converter

file:///D:/KULIAH SS/Prak Web/Modul 8/Tugas/konversi.html

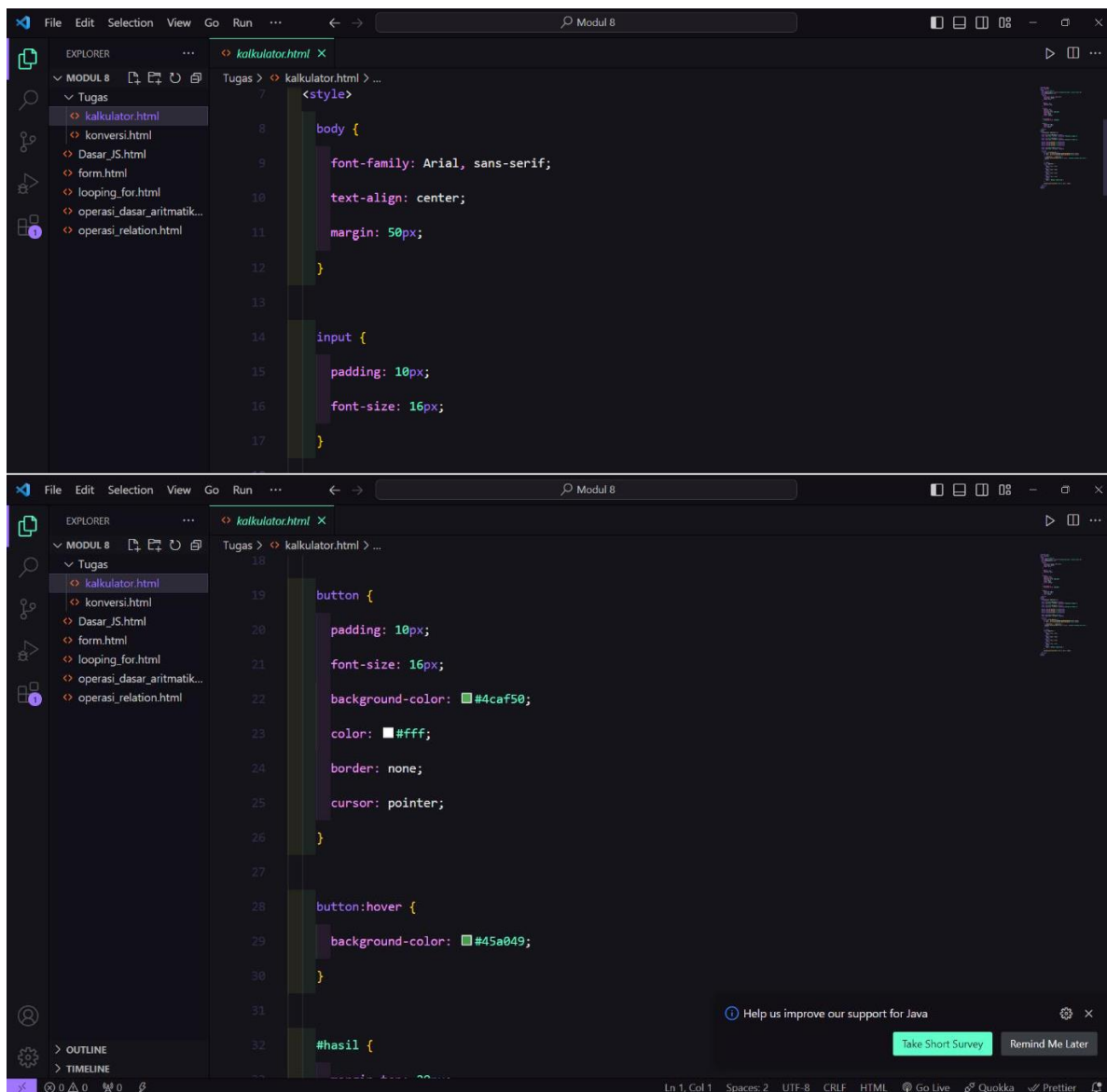
Konversi Nilai Angka ke Nilai Huruf

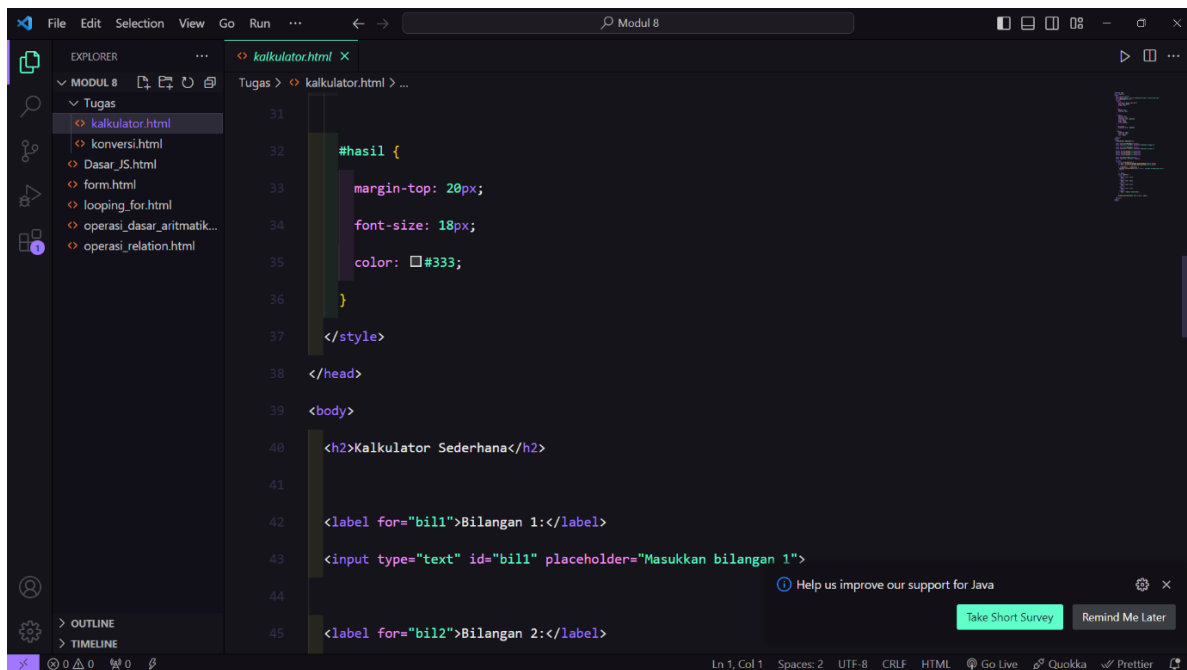
Masukkan nilai angka:

Konversi

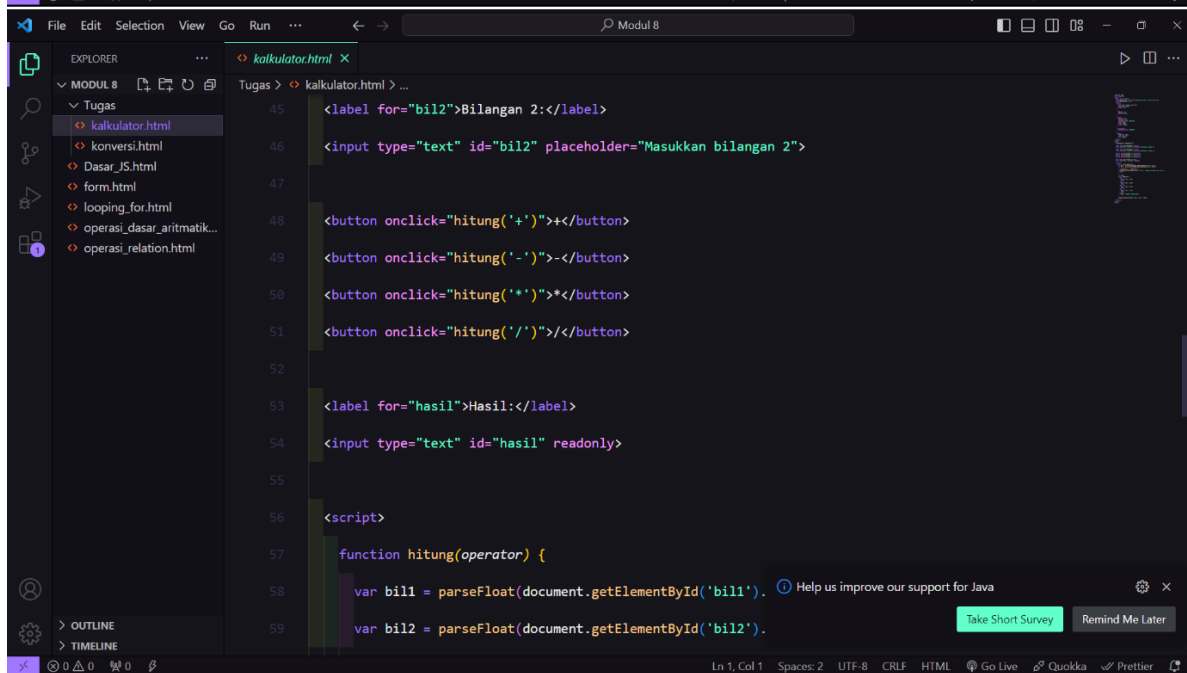
Nilai huruf: AB

Tugas No 2





```
31
32 #hasil {
33     margin-top: 20px;
34     font-size: 18px;
35     color: #333;
36 }
37 </style>
38 </head>
39 <body>
40 <h2>Kalkulator Sederhana</h2>
41
42 <label for="bil1">Bilangan 1:</label>
43 <input type="text" id="bil1" placeholder="Masukkan bilangan 1">
44
45 <label for="bil2">Bilangan 2:</label>
```



```
45 <input type="text" id="bil2" placeholder="Masukkan bilangan 2">
46
47 <button onclick="hitung('+")>+</button>
48 <button onclick="hitung('-")>-</button>
49 <button onclick="hitung('*")>*</button>
50 <button onclick="hitung('/')>/</button>
51
52
53 <label for="hasil">Hasil:</label>
54 <input type="text" id="hasil" readonly>
55
56 <script>
57     function hitung(operator) {
58         var bil1 = parseFloat(document.getElementById('bil1'));
59         var bil2 = parseFloat(document.getElementById('bil2'));
```

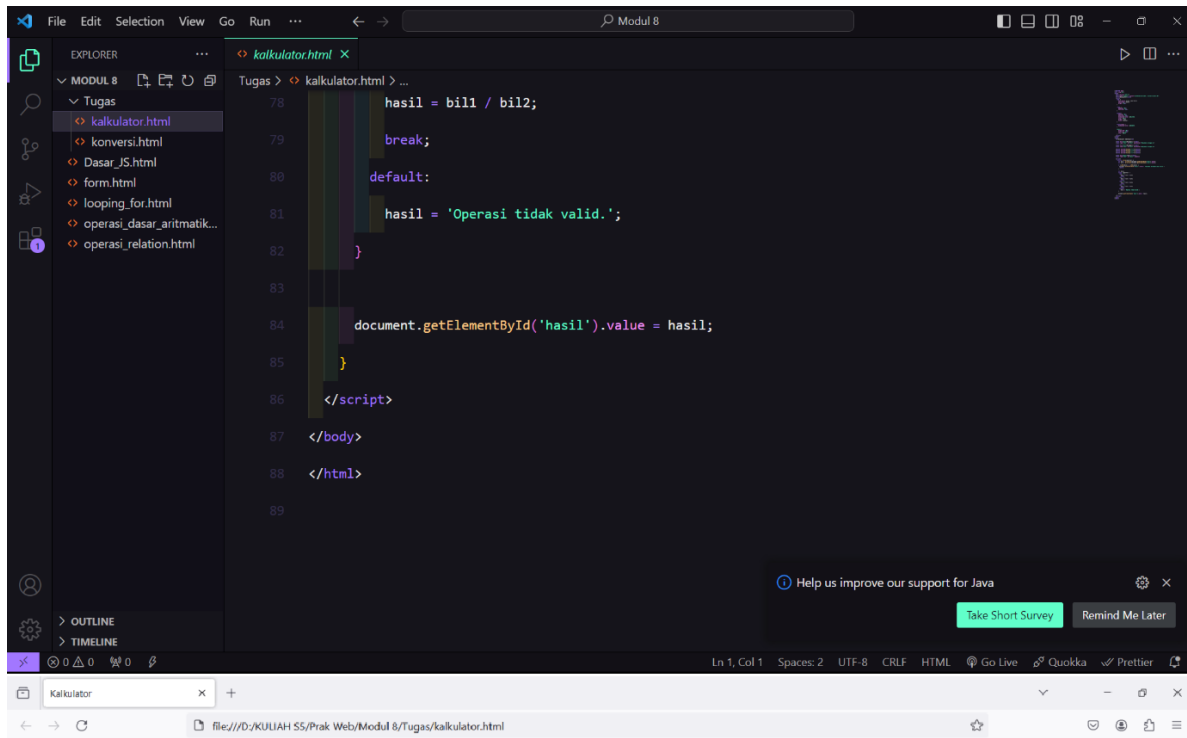

This screenshot shows the first part of a JavaScript script in a file named `kalkulator.html`. The script defines a `hitung(operator)` function that takes an operator as an argument. It first retrieves the values of `bil1` and `bil2` from the document. It then checks if either value is NaN. If so, it sets the `hasil` element's value to "Masukkan bilangan yang valid." and returns. Otherwise, it enters a switch statement to perform the calculation based on the operator.

```
<script>
56
57   function hitung(operator) {
58       var bil1 = parseFloat(document.getElementById('bil1').value);
59       var bil2 = parseFloat(document.getElementById('bil2').value);
60
61       if (isNaN(bil1) || isNaN(bil2)) {
62           document.getElementById('hasil').value = 'Masukkan bilangan yang valid.';
63           return;
64       }
65
66       var hasil;
67       switch (operator) {
68           case '+':
69               hasil = bil1 + bil2;
70               break;
```

This screenshot shows the continuation of the JavaScript script from the previous block. It completes the switch statement with cases for subtraction, multiplication, and division, each followed by a `break` statement. A `default` case is also present at the end of the switch block.

```

71       case '-':
72           hasil = bil1 - bil2;
73           break;
74       case '*':
75           hasil = bil1 * bil2;
76           break;
77       case '/':
78           hasil = bil1 / bil2;
79           break;
80       default:
```



Kalkulator Sederhana

Bilangan 1: Bilangan 2: Hasil: