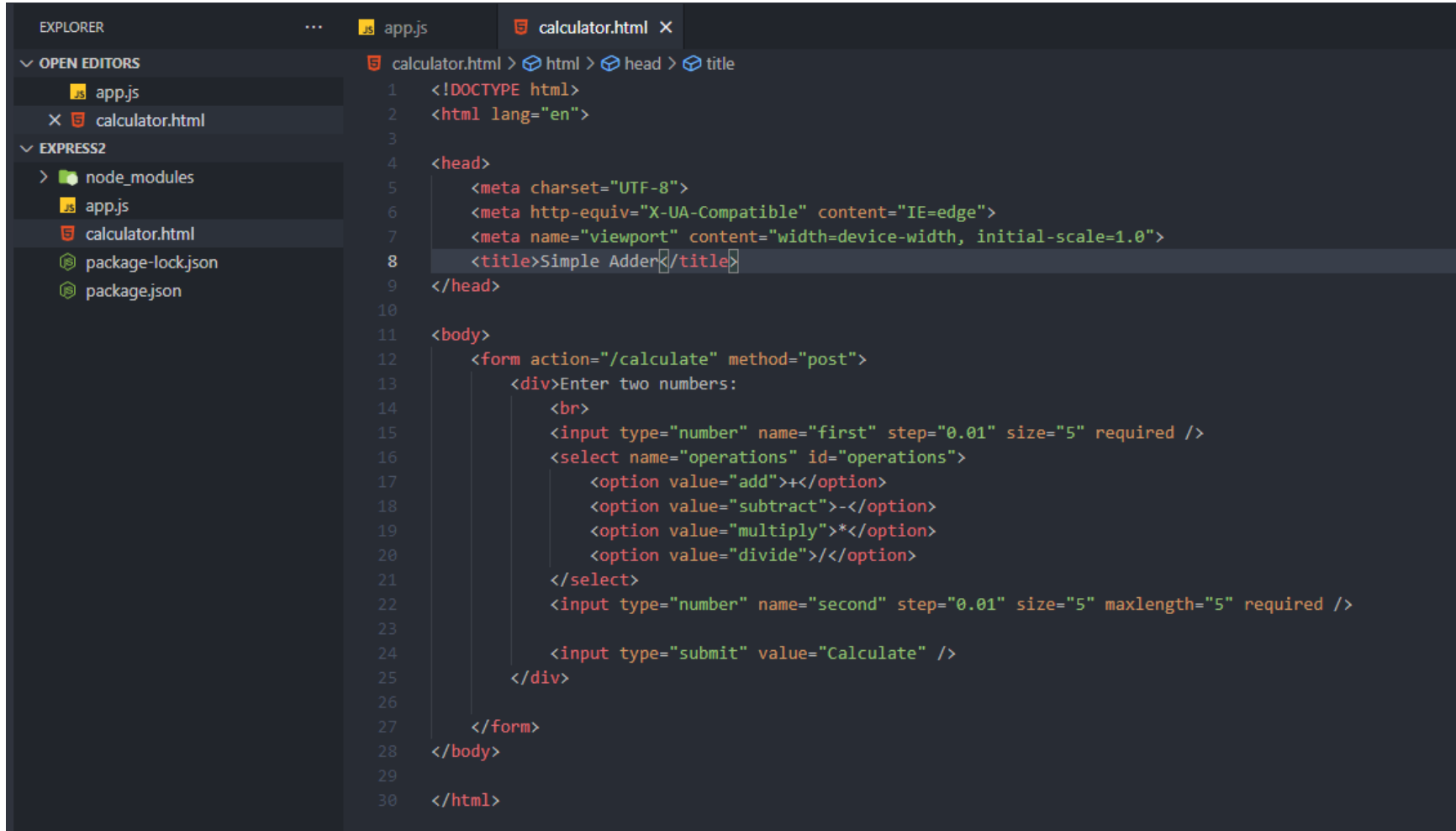


Simple Calculator Lab solution

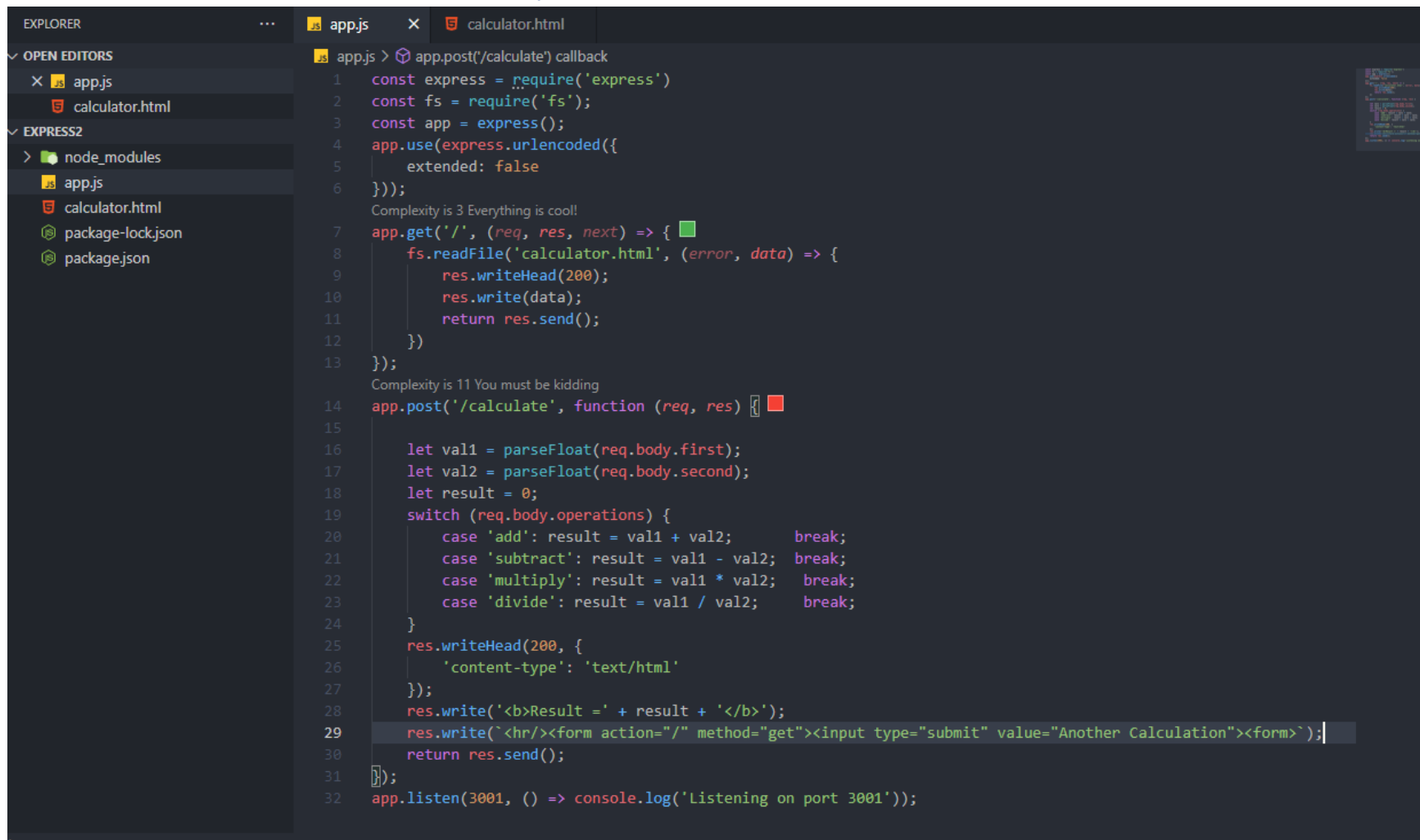
Code



```
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>Simple Adder</title>
9  </head>
10
11 <body>
12     <form action="/calculate" method="post">
13         <div>Enter two numbers:
14             <br>
15             <input type="number" name="first" step="0.01" size="5" required />
16             <select name="operations" id="operations">
17                 <option value="add">+</option>
18                 <option value="subtract">-</option>
19                 <option value="multiply">*</option>
20                 <option value="divide">/</option>
21             </select>
22             <input type="number" name="second" step="0.01" size="5" maxlength="5" required />
23
24             <input type="submit" value="Calculate" />
25         </div>
26     </form>
27 </body>
28 </html>
```

Figure 1 : Calculator.html

Simple Calculator Lab solution

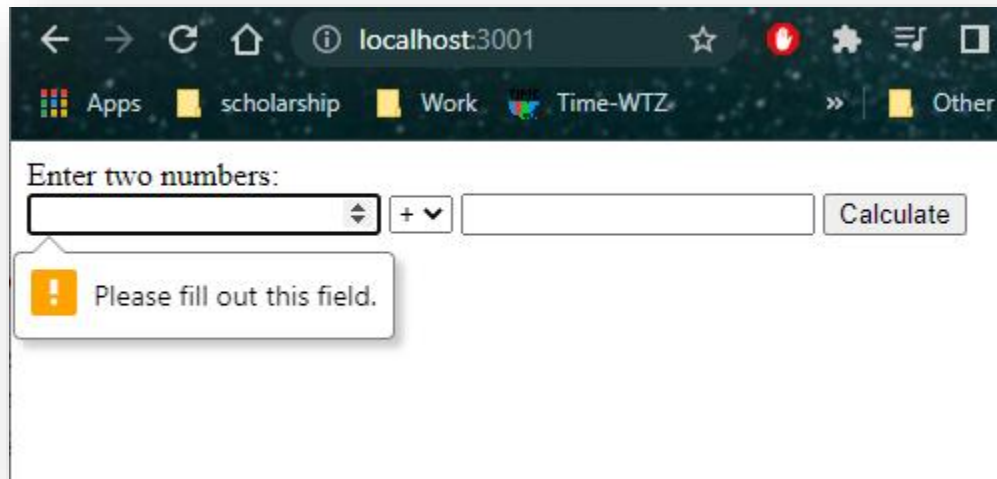


```
1  const express = require('express')
2  const fs = require('fs');
3  const app = express();
4  app.use(express.urlencoded({
5    extended: false
6  }));
7  Complexity is 3 Everything is cool!
8  app.get('/', (req, res, next) => {
9    fs.readFile('calculator.html', (error, data) => {
10      res.writeHead(200);
11      res.write(data);
12      return res.send();
13    });
14  });
15  Complexity is 11 You must be kidding
16  app.post('/calculate', function (req, res) {
17
18    let val1 = parseFloat(req.body.first);
19    let val2 = parseFloat(req.body.second);
20    let result = 0;
21    switch (req.body.operations) {
22      case 'add': result = val1 + val2; break;
23      case 'subtract': result = val1 - val2; break;
24      case 'multiply': result = val1 * val2; break;
25      case 'divide': result = val1 / val2; break;
26    }
27    res.writeHead(200, {
28      'content-type': 'text/html'
29    });
30    res.write('<b>Result = ' + result + '</b>');
31    res.write('<hr><form action="/" method="get"><input type="submit" value="Another Calculation"></form>');
32    return res.send();
33  });
34  app.listen(3001, () => console.log('Listening on port 3001'));
```

Figure 2 : app.js

Simple Calculator Lab solution

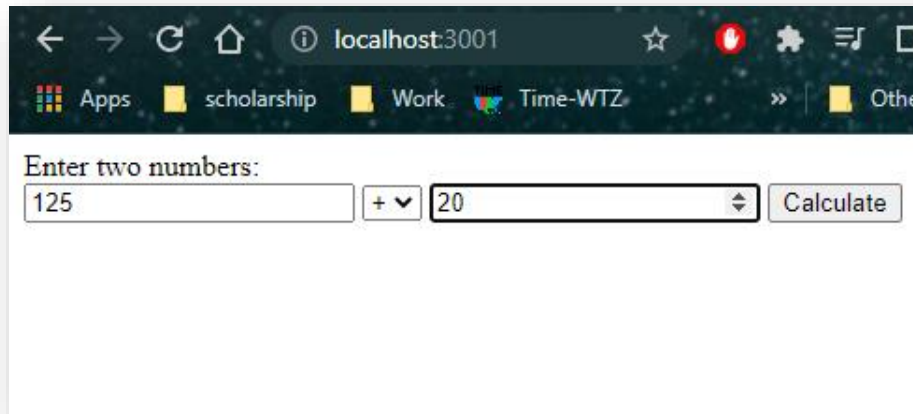
Screenshots



The screenshot shows a web browser window with the address bar displaying 'localhost:3001'. The browser's tab bar includes 'Apps', 'scholarship', 'Work', 'Time-WTZ', and 'Other'. The main content area of the browser displays a web form titled 'Enter two numbers:'. The form consists of two input fields, a dropdown menu with a '+' sign, and a 'Calculate' button. The first input field is empty and has a red border, indicating a validation error. A tooltip with an exclamation mark icon and the text 'Please fill out this field.' is pointing to the first input field.

Figure 3 : Calculator Home

Simple Calculator Lab solution



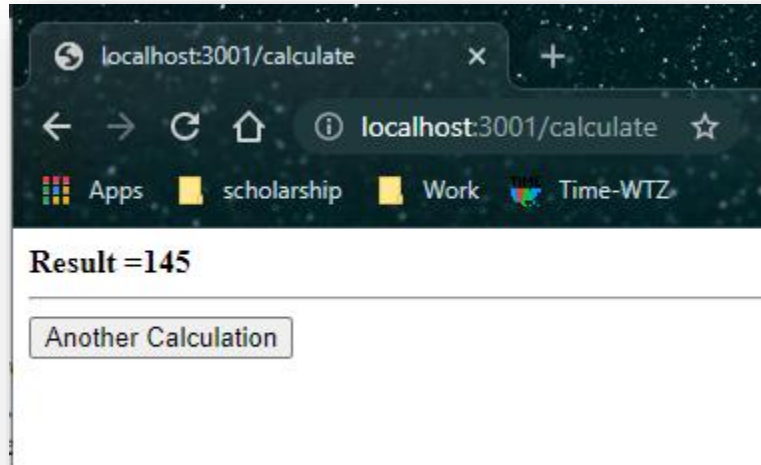
localhost:3001

Apps scholarship Work Time-WTZ Other

Enter two numbers:

125 + 20 Calculate

Figure 4: Sample 01: Addition



localhost:3001/calculate

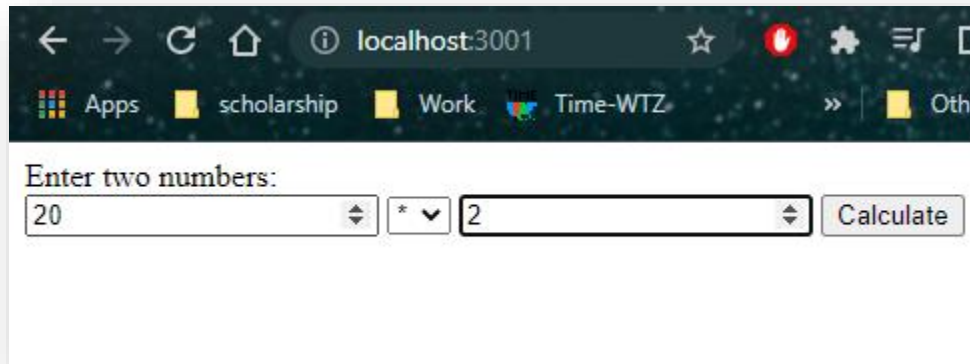
Apps scholarship Work Time-WTZ

Result =145

Another Calculation

Figure 5 : Addition Result

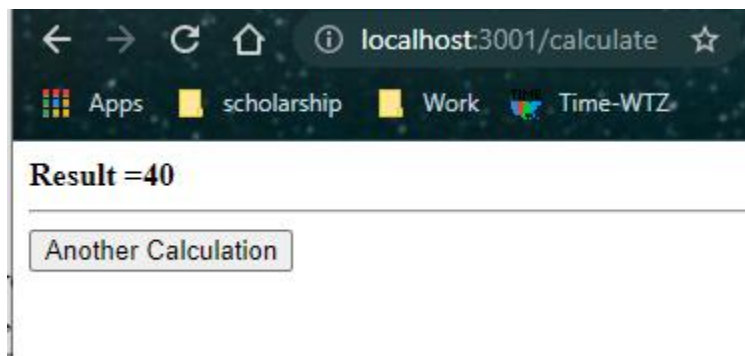
Simple Calculator Lab solution



Enter two numbers:

20 * 2 Calculate

Figure 6: Sample input: Multiplication



Result =40

Another Calculation

Figure 7 : Sample output :Multiplication