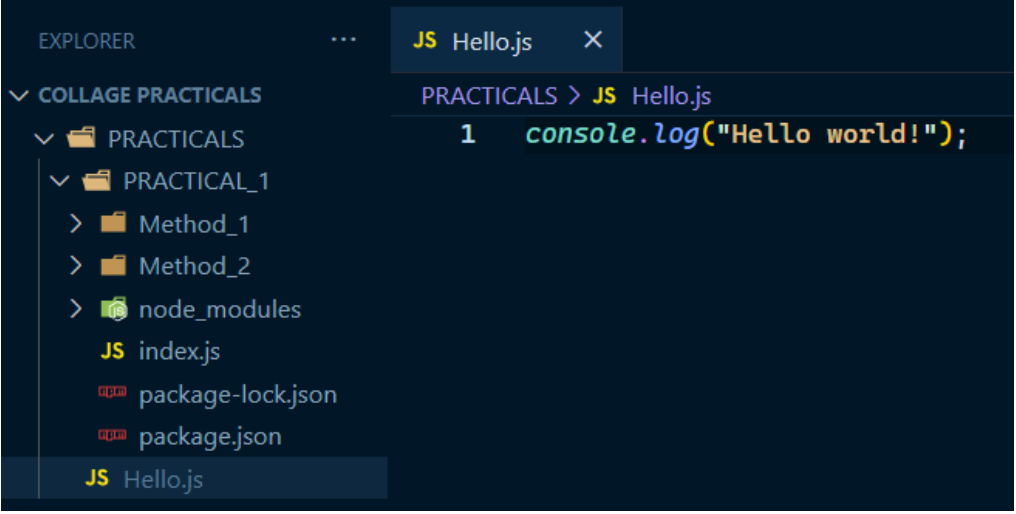


PRACTICAL 1

1. Create Hello World – First NODE Program & Run it.

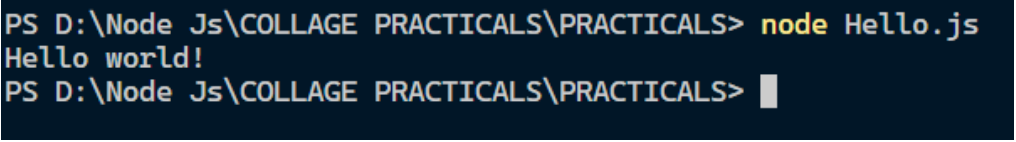
CODE:



```
EXPLORER  ...  JS Hello.js  X
COLLAGE PRACTICALS
  PRACTICALS
    PRACTICAL_1
      Method_1
      Method_2
      node_modules
      JS index.js
      package-lock.json
      package.json
      JS Hello.js

PRACTICALS > JS Hello.js
1  console.log("Hello world!");
```

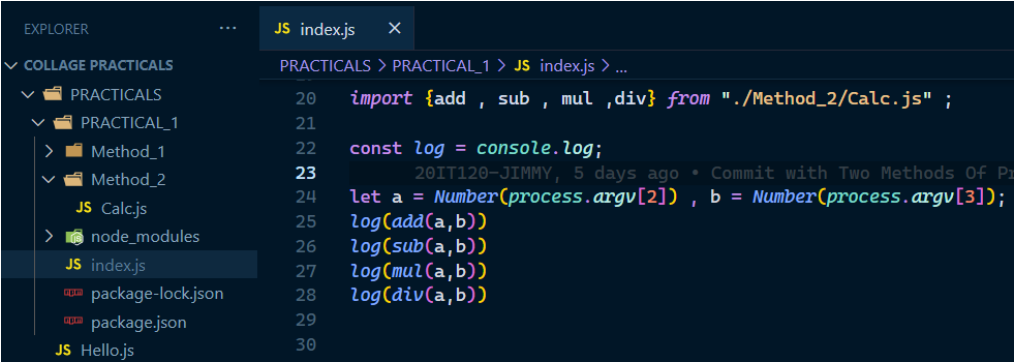
OUTPUT :



```
PS D:\Node Js\COLLAGE PRACTICALS\PRACTICALS> node Hello.js
Hello world!
PS D:\Node Js\COLLAGE PRACTICALS\PRACTICALS> █
```

2. Create local NodeJS Custom Module “Calculator” and import it and use the module in current project.

CODE :



```
EXPLORER  ...  JS index.js  X
COLLAGE PRACTICALS
  PRACTICALS
    PRACTICAL_1
      Method_1
      Method_2
      JS Calc.js
      node_modules
      JS index.js
      package-lock.json
      package.json
      JS Hello.js

PRACTICALS > PRACTICAL_1 > JS index.js > ...
20  import {add , sub , mul ,div} from './Method_2/Calc.js' ;
21
22  const log = console.log;
23
24  let a = Number(process.argv[2]) , b = Number(process.argv[3]);
25  log(add(a,b))
26  log(sub(a,b))
27  log(mul(a,b))
28  log(div(a,b))
29
30
```

```

1 //Addition
2 export const add = (a, b) => {
3   let sum = a + b;
4   return "Addition Is : " + sum;
5 };
6
7 // Subtraction
8 export const sub = (a, b) => {
9   let diff = a - b;
10  return "Subtraction Is : " + diff;
11 };
12
13 //Multiplication
14 export const mul = (a, b) => {
15   let prod = a * b;
16   return "Multiplication Is : " + prod;
17 };
18
19 //Division
20 export const div = (a, b) => {
21   let divide = a / b;
22   return "Division Is : " + divide;
23 };

```

OUTPUT :

```

PS D:\Node Js\COLLAGE PRACTICALS\PRACTICALS\PRACTICAL_1> node index.js 10 5
Addition Is : 15
Subtraction Is : 5
Multiplication Is : 50
Division Is : 2

```

3. Asynchronously Read the file “input.txt” and apply specific operation given in file using local custom calculator.js module and write the result in file called “output.txt” (new line for each result) .

CODE :

```

1 export const add =(a,b)=>{
2   return a + b;
3 }
4 export const sub = (a,b)=>{
5   return a - b;
6 }
7 export const mul= (a,b)=>{
8   return a * b;
9 }
10 export const div= (a,b)=>{
11   return a / b;
12 }
13

```

```

1  import fs from "fs";
2  import { add, sub, mul, div } from "./Calc.js";
3  const cal = fs.readFileSync("./input.txt", "utf-8");
4  const arr = [];
5
6  cal.split(/\r?\n/).forEach((line) => {
7    arr.push(line);
8  });
9
10 for (let i = 0; i < arr.length / 3; i++) {
11   let a = arr[i * 3 + 0];
12   let b = arr[i * 3 + 1];
13   let c = arr[i * 3 + 2];
14
15   a = parseInt(a);
16   b = parseInt(b);
17
18   if (c === "ADD") {
19     let d = add(a, b);
20     fs.appendFileSync("output.txt", d.toString() + "\n");
21   } else if (c === "SUB") {
22     let d = sub(a, b);
23     fs.appendFileSync("output.txt", d.toString() + "\n");
24   } else if (c === "MUL") {
25     let d = mul(a, b);
26     fs.appendFileSync("output.txt", d.toString() + "\n");
27   } else if (c === "DIV") {
28     let d = div(a, b);
29     fs.appendFileSync("output.txt", d.toString() + "\n");
30   }
31 }
32

```

OUTPUT :

input.txt	output.txt
1 5	1 11
2 6	2 5
3 ADD	3 28
4 8	4 4
5 3	5
6 SUB	
7 4	
8 7	
9 MUL	
10 8	
11 2	
12 DIV	