

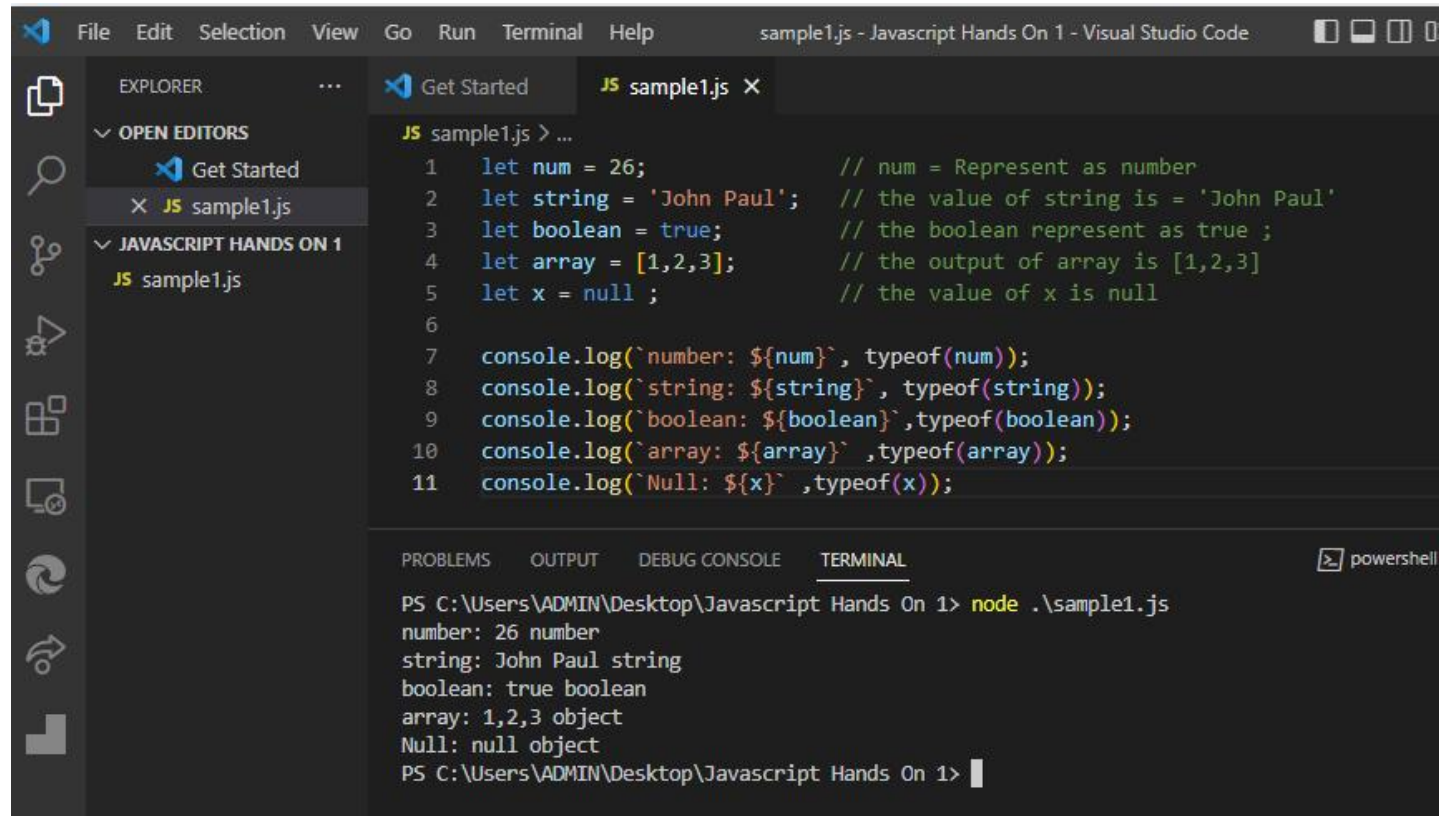
Name :John Paul S, Baluyot

Batch “WD37”

Activities

Javascript Hands On 1

1, Create a Js file that will show different Data Types(Number , String , Objects). Used typeof to show the data types.



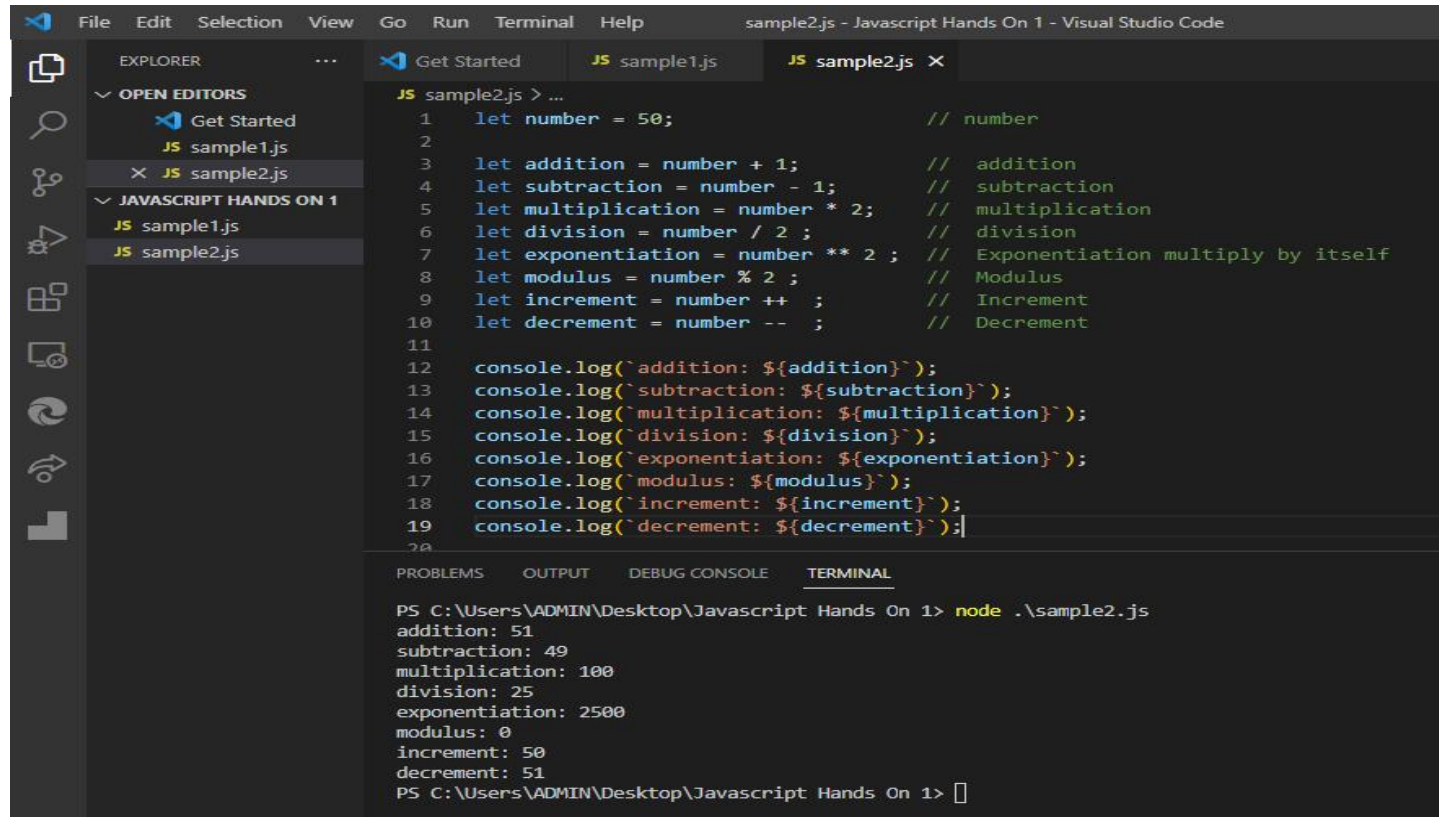
```
File Edit Selection View Go Run Terminal Help sample1.js - Javascript Hands On 1 - Visual Studio Code

EXPLORER
  OPEN EDITORS
    Get Started
    JS sample1.js
  JAVASCRIPT HANDS ON 1
    JS sample1.js

JS sample1.js > ...
1 let num = 26;           // num = Represent as number
2 let string = 'John Paul'; // the value of string is = 'John Paul'
3 let boolean = true;     // the boolean represent as true ;
4 let array = [1,2,3];    // the output of array is [1,2,3]
5 let x = null ;         // the value of x is null
6
7 console.log(`number: ${num}`, typeof(num));
8 console.log(`string: ${string}`, typeof(string));
9 console.log(`boolean: ${boolean}`,typeof(boolean));
10 console.log(`array: ${array}`,typeof(array));
11 console.log(`Null: ${x}`,typeof(x));

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Users\ADMIN\Desktop\Javascript Hands On 1> node .\sample1.js
number: 26 number
string: John Paul string
boolean: true boolean
array: 1,2,3 object
Null: null object
PS C:\Users\ADMIN\Desktop\Javascript Hands On 1>
```

2, Create Js file to show different Mathematical Operations (+ , - , * , / , ** , % , ++ , --).



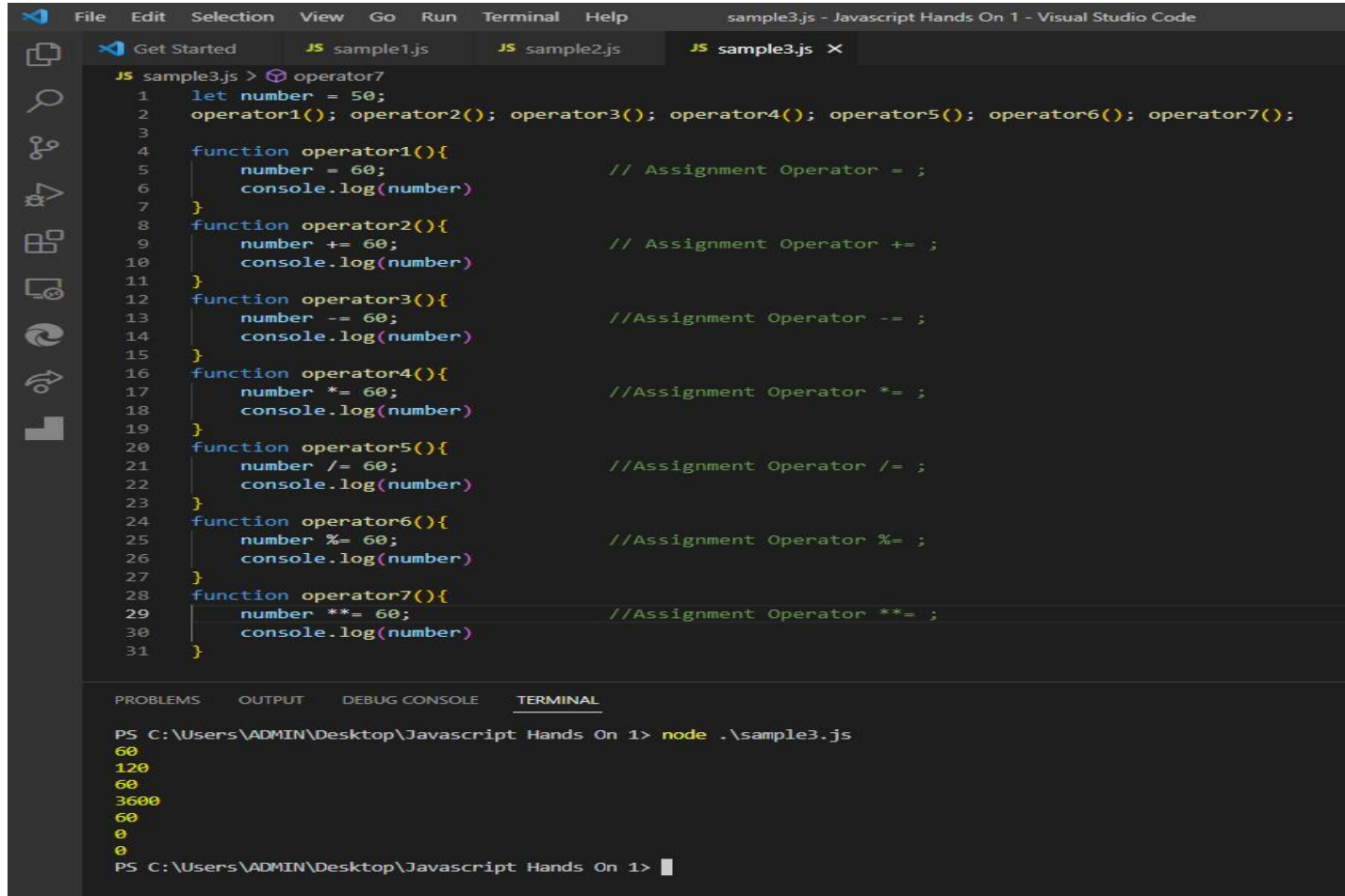
The screenshot displays the Visual Studio Code interface. The Explorer panel on the left shows the file structure with 'sample2.js' selected. The main editor area shows the code for 'sample2.js', which defines variables for various mathematical operations using the 'let' keyword. The code includes comments for each operation. The bottom panel shows the Terminal output, which displays the results of these operations as logged by 'console.log' statements.

```
JS sample2.js > ...
1  let number = 50;           // number
2
3  let addition = number + 1;  // addition
4  let subtraction = number - 1; // subtraction
5  let multiplication = number * 2; // multiplication
6  let division = number / 2 ; // division
7  let exponentiation = number ** 2 ; // Exponentiation multiply by itself
8  let modulus = number % 2 ; // Modulus
9  let increment = number ++ ; // Increment
10 let decrement = number -- ; // Decrement
11
12 console.log(`addition: ${addition}`);
13 console.log(`subtraction: ${subtraction}`);
14 console.log(`multiplication: ${multiplication}`);
15 console.log(`division: ${division}`);
16 console.log(`exponentiation: ${exponentiation}`);
17 console.log(`modulus: ${modulus}`);
18 console.log(`increment: ${increment}`);
19 console.log(`decrement: ${decrement}`);
20
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\Users\ADMIN\Desktop\Javascript Hands On 1> node .\sample2.js
addition: 51
subtraction: 49
multiplication: 100
division: 25
exponentiation: 2500
modulus: 0
increment: 50
decrement: 51
PS C:\Users\ADMIN\Desktop\Javascript Hands On 1> 
```

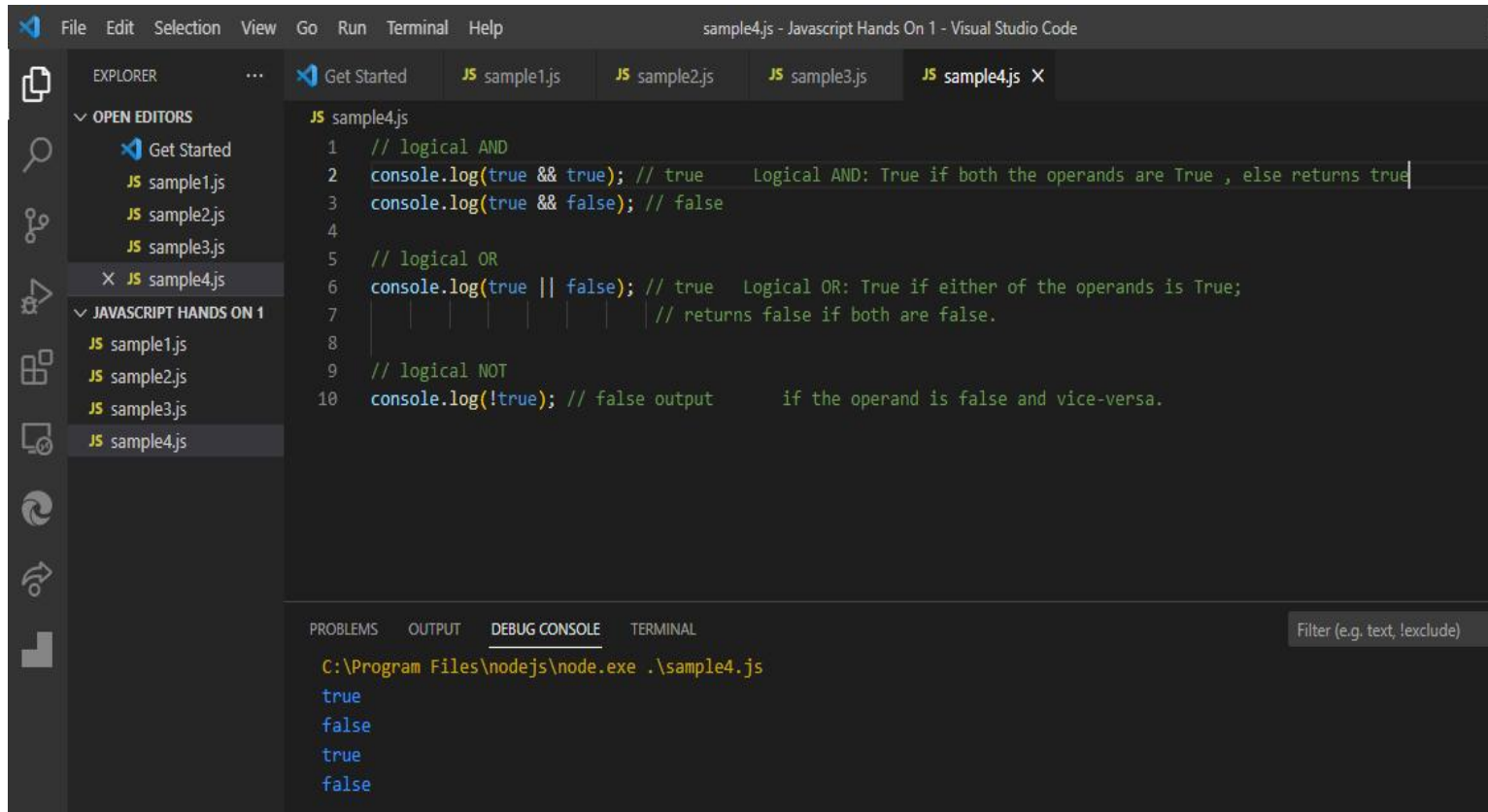
3, Create Js file to show different Assignment Operators (= , += , -= , *= , /= , %= , **=).



```
File Edit Selection View Go Run Terminal Help sample3.js - Javascript Hands On 1 - Visual Studio Code
JS sample3.js > operator7
1 let number = 50;
2 operator1(); operator2(); operator3(); operator4(); operator5(); operator6(); operator7();
3
4 function operator1(){
5     number = 60; // Assignment Operator = ;
6     console.log(number)
7 }
8 function operator2(){
9     number += 60; // Assignment Operator += ;
10    console.log(number)
11 }
12 function operator3(){
13     number -= 60; //Assignment Operator -= ;
14    console.log(number)
15 }
16 function operator4(){
17     number *= 60; //Assignment Operator *= ;
18    console.log(number)
19 }
20 function operator5(){
21     number /= 60; //Assignment Operator /= ;
22    console.log(number)
23 }
24 function operator6(){
25     number %= 60; //Assignment Operator %= ;
26    console.log(number)
27 }
28 function operator7(){
29     number **= 60; //Assignment Operator **= ;
30    console.log(number)
31 }
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Users\ADMIN\Desktop\Javascript Hands On 1> node .\sample3.js
60
120
60
3600
60
0
0
PS C:\Users\ADMIN\Desktop\Javascript Hands On 1>
```

4, Create Js file to show different Logical Operators (`&&` , `||` , `!`).



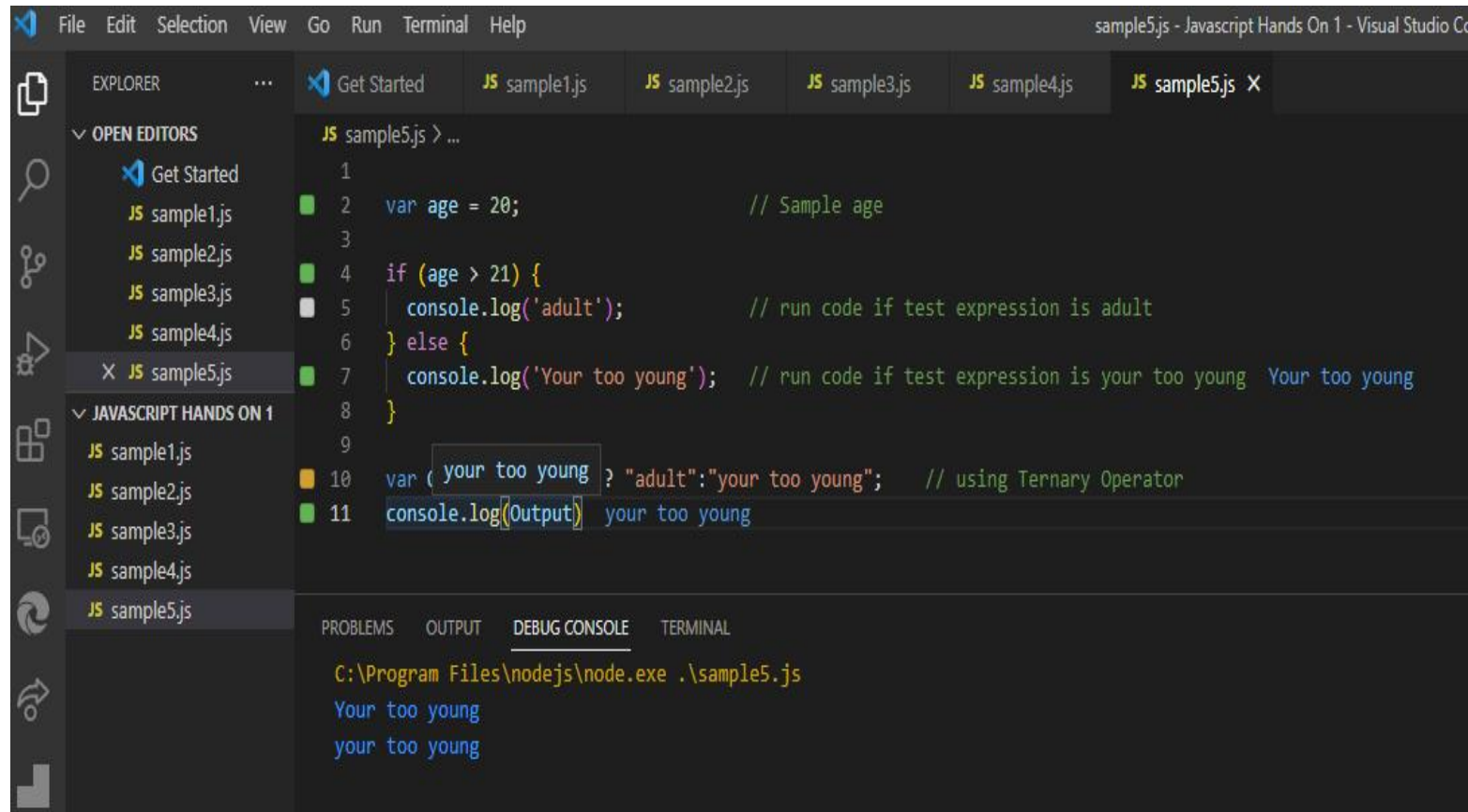
The screenshot shows the Visual Studio Code interface with a file named `sample4.js` open. The file contains JavaScript code demonstrating logical operators: `&&` (AND), `||` (OR), and `!` (NOT). The code is as follows:

```
1 // logical AND
2 console.log(true && true); // true   Logical AND: True if both the operands are True , else returns true
3 console.log(true && false); // false
4
5 // logical OR
6 console.log(true || false); // true   Logical OR: True if either of the operands is True;
7                                     // returns false if both are false.
8
9 // logical NOT
10 console.log(!true); // false output   if the operand is false and vice-versa.
```

The bottom panel shows the terminal output of the code execution:

```
C:\Program Files\nodejs\node.exe .\sample4.js
true
false
true
false
```

5, Create Js file to program an if else statement and convert this to ternary operator.



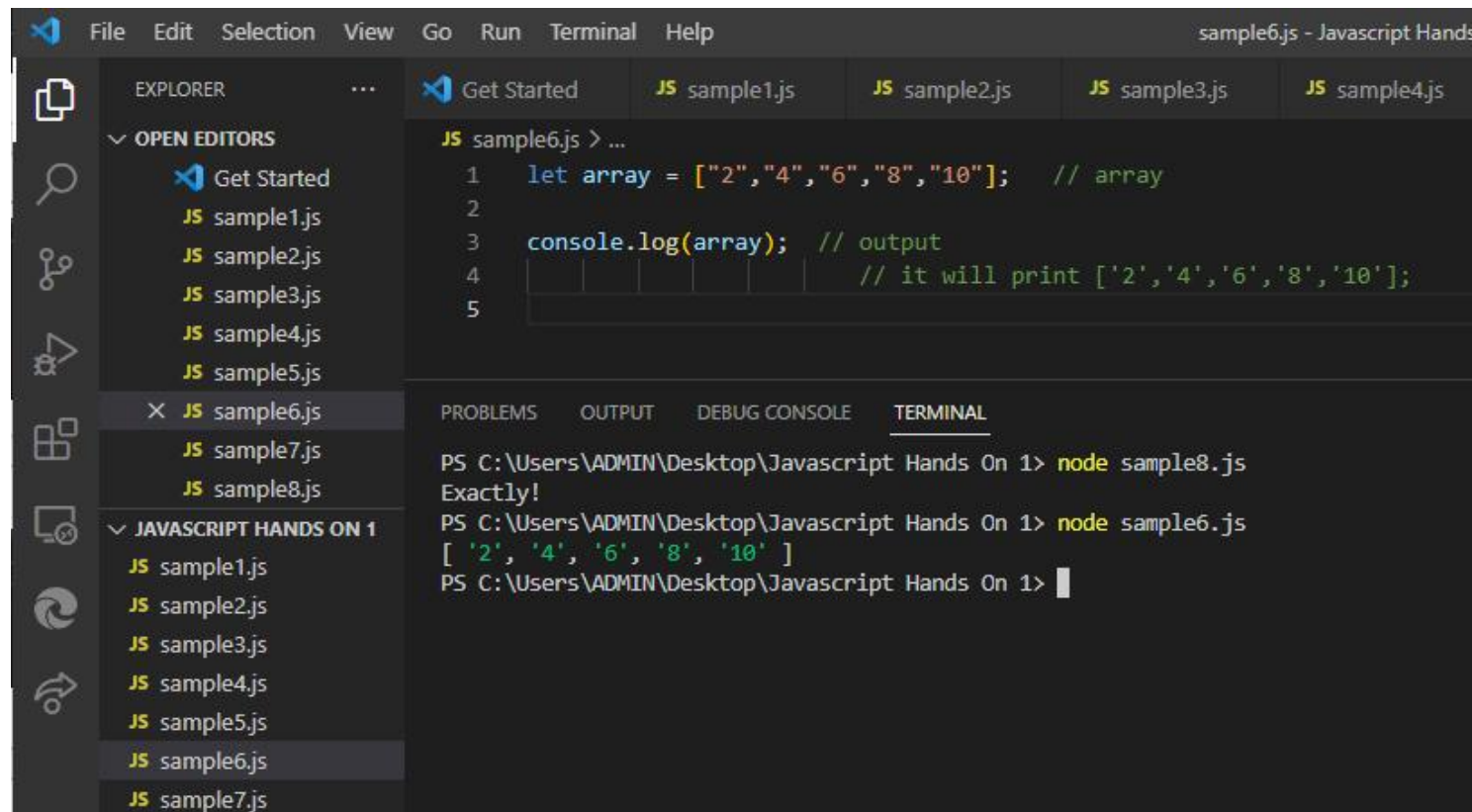
```
File Edit Selection View Go Run Terminal Help
sample5.js - Javascript Hands On 1 - Visual Studio Code

EXPLORER
  OPEN EDITORS
    Get Started
    JS sample1.js
    JS sample2.js
    JS sample3.js
    JS sample4.js
    X JS sample5.js
  JAVASCRIPT HANDS ON 1
    JS sample1.js
    JS sample2.js
    JS sample3.js
    JS sample4.js
    JS sample5.js

JS sample5.js > ...
1
2 var age = 20; // Sample age
3
4 if (age > 21) {
5   console.log('adult'); // run code if test expression is adult
6 } else {
7   console.log('Your too young'); // run code if test expression is your too young Your too young
8 }
9
10 var ( your too young ? "adult":"your too young"; // using Ternary Operator
11 console.log((Output) your too young

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
C:\Program Files\nodejs\node.exe .\sample5.js
Your too young
your too young
```


6, Create Js file the has blank array, then push or assigned 5 elements inside the array, after you add items in the array loop and show all the elements of the array.



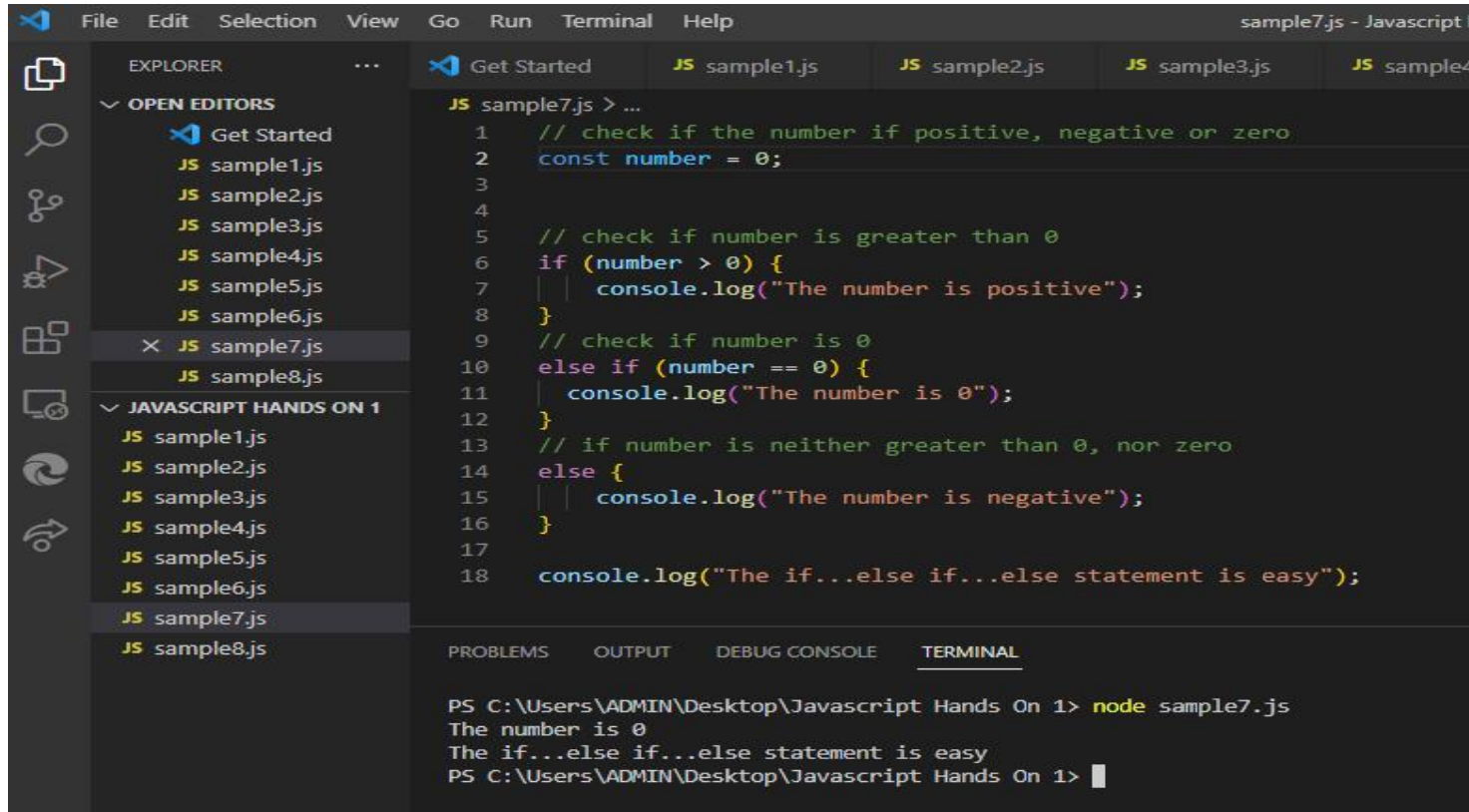
The screenshot shows the Visual Studio Code editor interface. The Explorer panel on the left displays a project named "JAVASCRIPT HANDS ON 1" containing several JavaScript files. The file "sample6.js" is selected and open in the editor. The code in "sample6.js" is as follows:

```
JS sample6.js > ...
1  let array = ["2","4","6","8","10"]; // array
2
3  console.log(array); // output
4  // it will print ['2','4','6','8','10'];
5
```

The TERMINAL panel at the bottom shows the command prompt output for running the file:

```
PS C:\Users\ADMIN\Desktop\Javascript Hands On 1> node sample8.js
Exactly!
PS C:\Users\ADMIN\Desktop\Javascript Hands On 1> node sample6.js
[ '2', '4', '6', '8', '10' ]
PS C:\Users\ADMIN\Desktop\Javascript Hands On 1>
```

7. Create Js file to program if, else if , else statement.



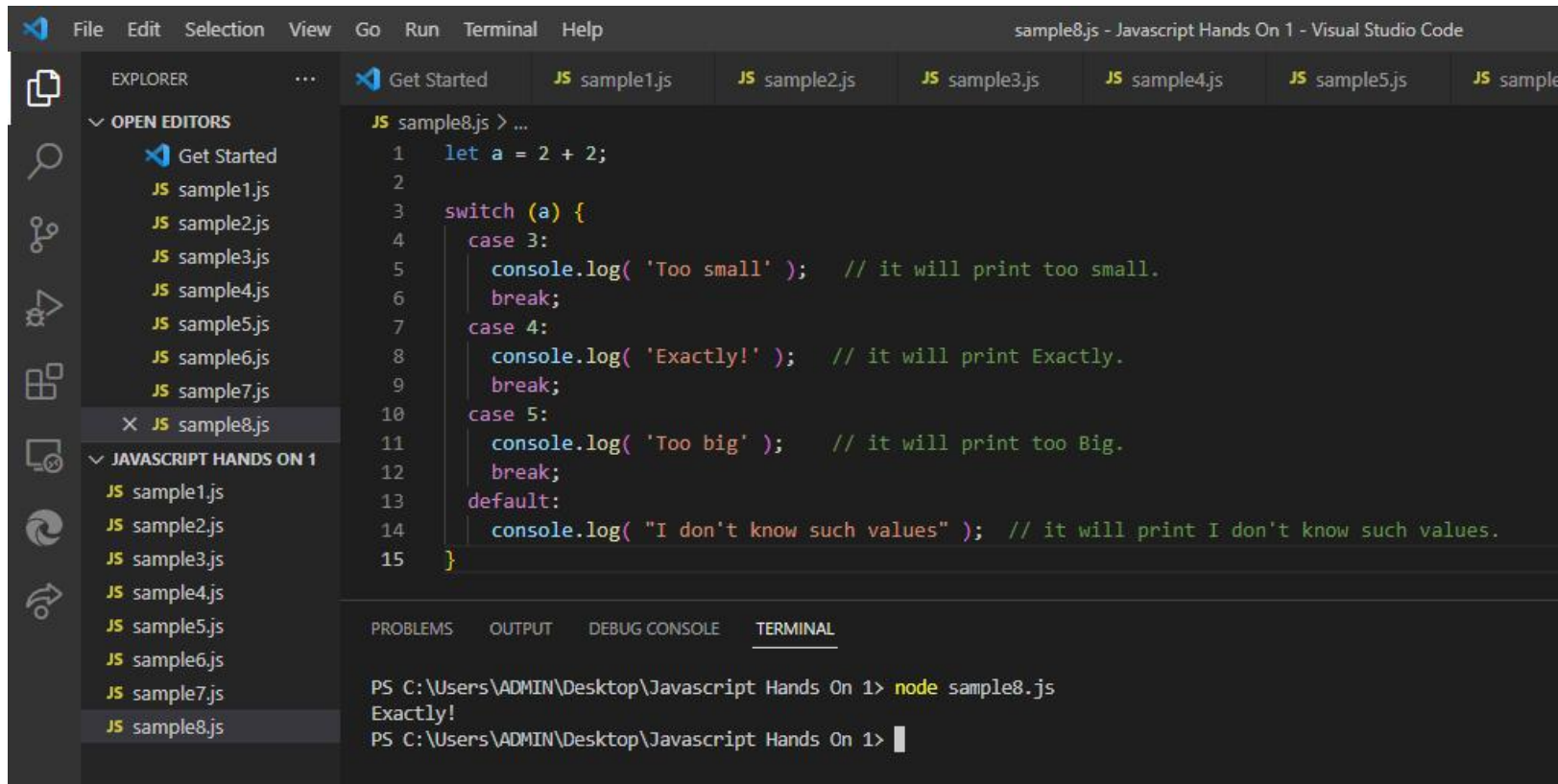
```
File Edit Selection View Go Run Terminal Help sample7.js - Javascript H

EXPLORER
OPEN EDITORS
  Get Started
  JS sample1.js
  JS sample2.js
  JS sample3.js
  JS sample4.js
  JS sample5.js
  JS sample6.js
  X JS sample7.js
  JS sample8.js
JAVASCRIPT HANDS ON 1
  JS sample1.js
  JS sample2.js
  JS sample3.js
  JS sample4.js
  JS sample5.js
  JS sample6.js
  JS sample7.js
  JS sample8.js

JS sample7.js > ...
1  // check if the number if positive, negative or zero
2  const number = 0;
3
4
5  // check if number is greater than 0
6  if (number > 0) {
7    | console.log("The number is positive");
8  }
9  // check if number is 0
10 else if (number == 0) {
11   | console.log("The number is 0");
12 }
13 // if number is neither greater than 0, nor zero
14 else {
15   | console.log("The number is negative");
16 }
17
18 console.log("The if...else if...else statement is easy");

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Users\ADMIN\Desktop\Javascript Hands On 1> node sample7.js
The number is 0
The if...else if...else statement is easy
PS C:\Users\ADMIN\Desktop\Javascript Hands On 1> 
```


8. Create Js file to program switch case statement.



```
File Edit Selection View Go Run Terminal Help
sample8js - Javascript Hands On 1 - Visual Studio Code

EXPLORER
  OPEN EDITORS
    Get Started
    JS sample1.js
    JS sample2.js
    JS sample3.js
    JS sample4.js
    JS sample5.js
    JS sample6.js
    JS sample7.js
    JS sample8.js
  JAVASCRIPT HANDS ON 1
    JS sample1.js
    JS sample2.js
    JS sample3.js
    JS sample4.js
    JS sample5.js
    JS sample6.js
    JS sample7.js
    JS sample8.js

JS sample8.js > ...
1  let a = 2 + 2;
2
3  switch (a) {
4    case 3:
5      console.log( 'Too small' ); // it will print too small.
6      break;
7    case 4:
8      console.log( 'Exactly!' ); // it will print Exactly.
9      break;
10   case 5:
11     console.log( 'Too big' ); // it will print too Big.
12     break;
13   default:
14     console.log( "I don't know such values" ); // it will print I don't know such values.
15  }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Users\ADMIN\Desktop\Javascript Hands On 1> node sample8.js
Exactly!
PS C:\Users\ADMIN\Desktop\Javascript Hands On 1>
```