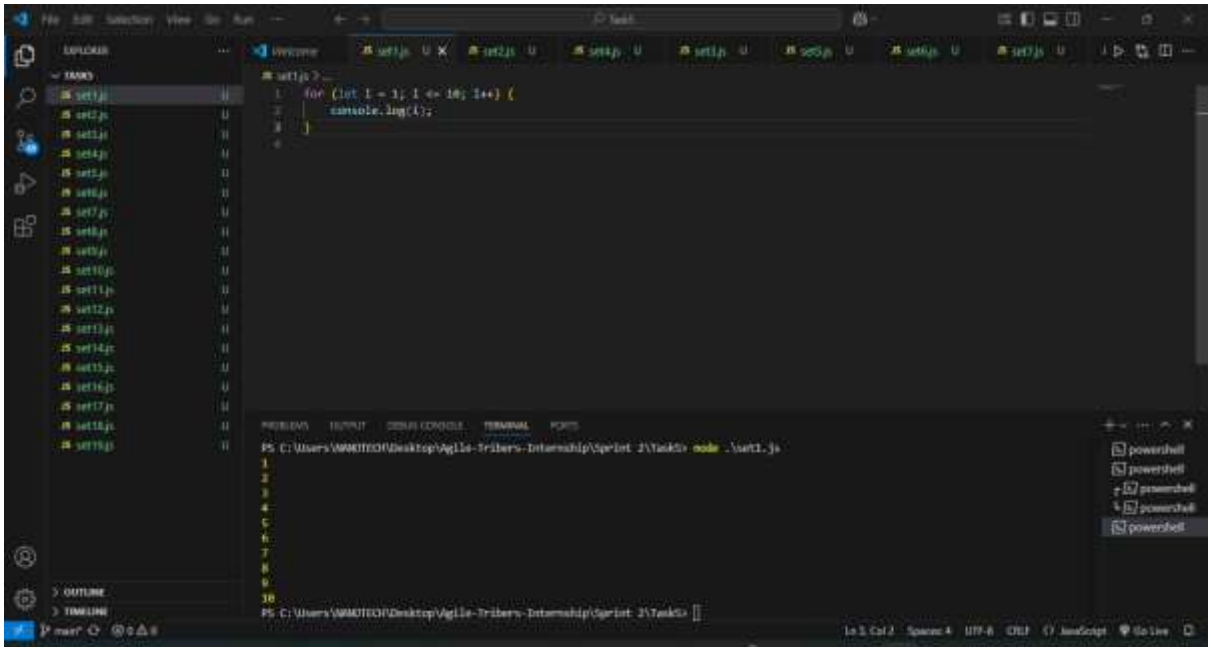
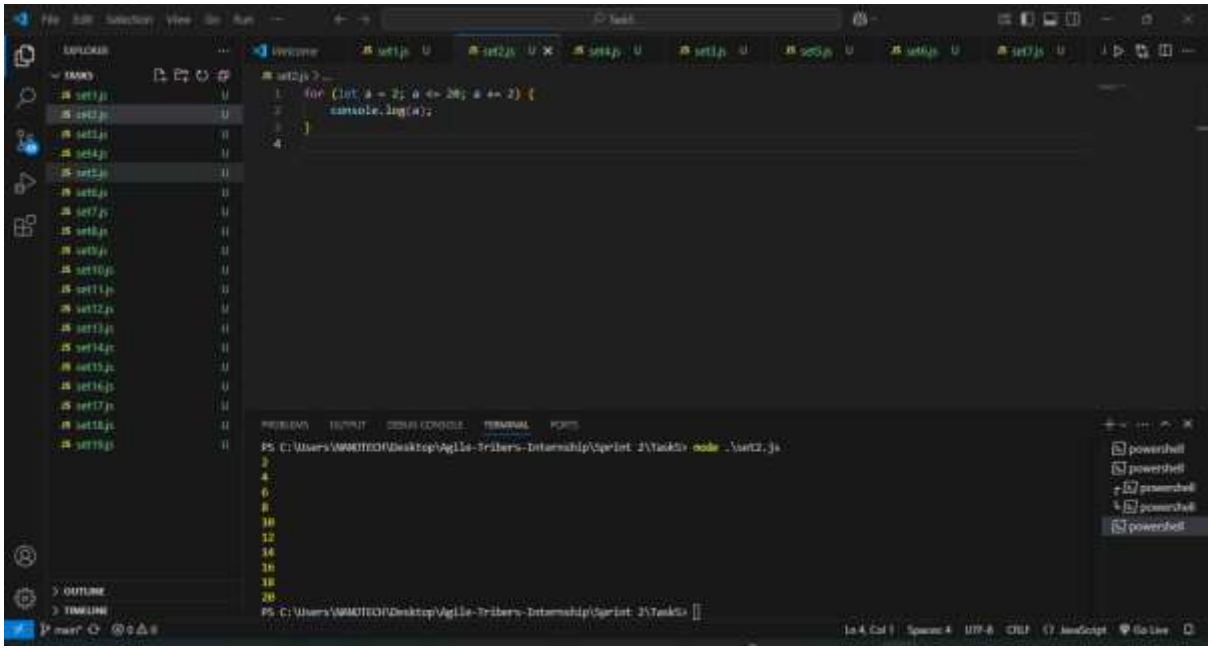
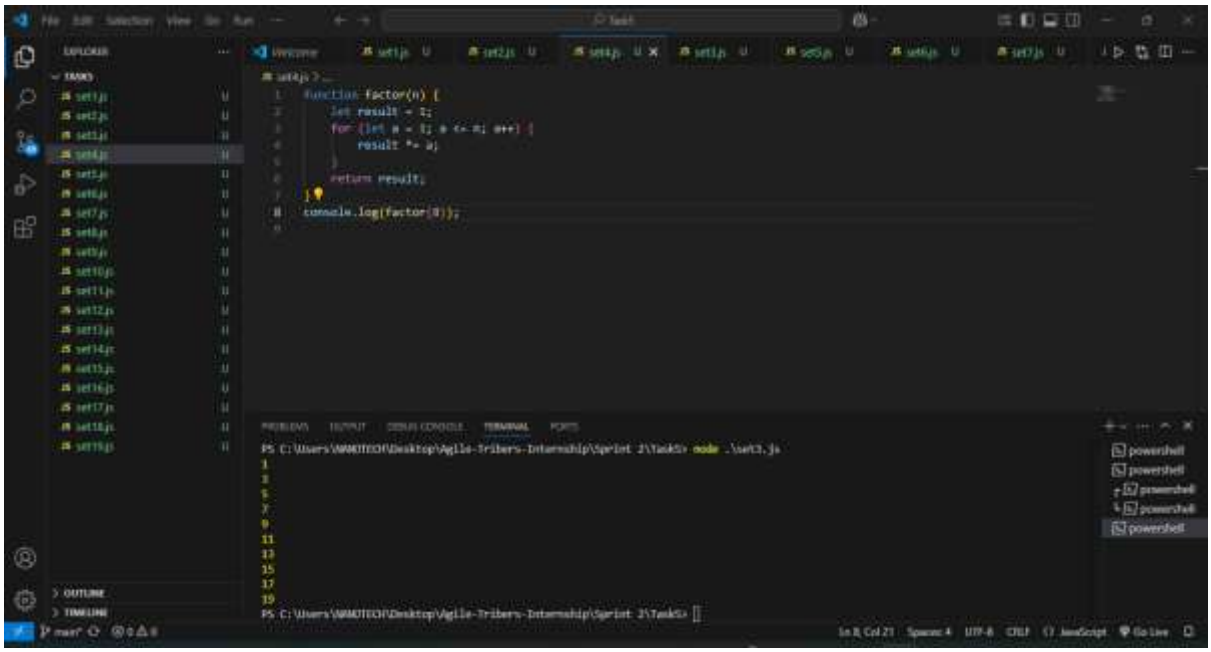
Set1


for (let i = 1; i <= 10; i++) {
console.log(i);
}
PS C:\Users\VAMSITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set1.js
1
2
3
4
5
6
7
8
9
10
PS C:\Users\VAMSITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5>

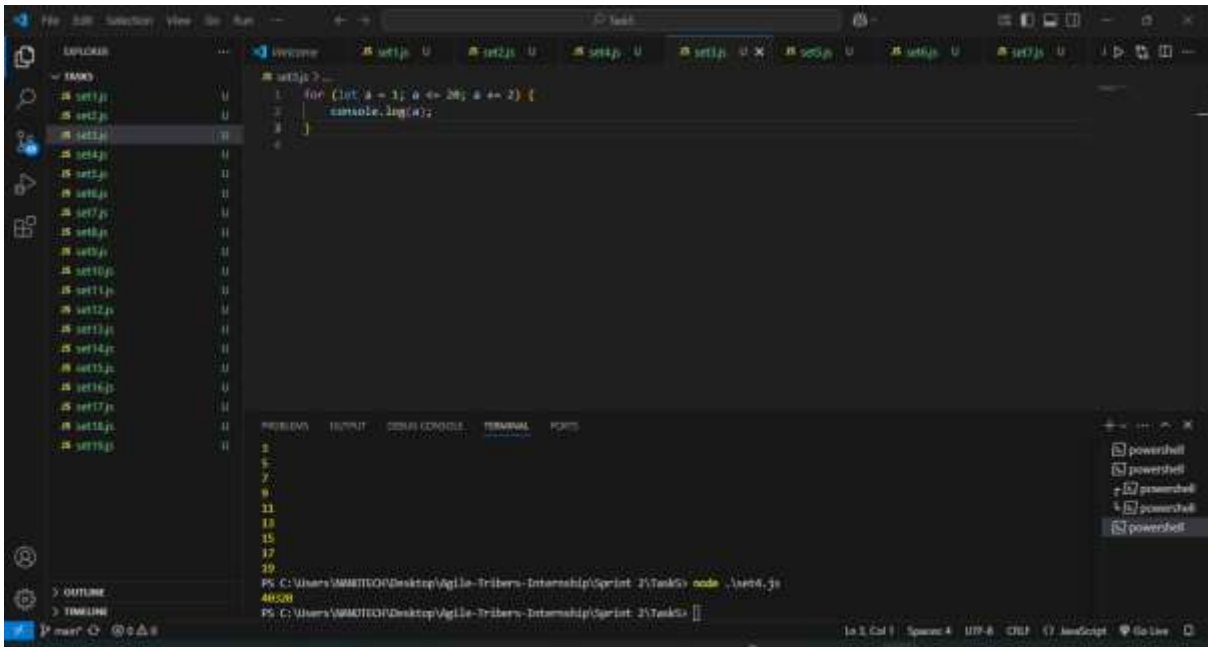
Set2


for (let a = 2; a <= 20; a += 2) {
console.log(a);
}
PS C:\Users\VAMSITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set2.js
2
4
6
8
10
12
14
16
18
20
PS C:\Users\VAMSITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5>

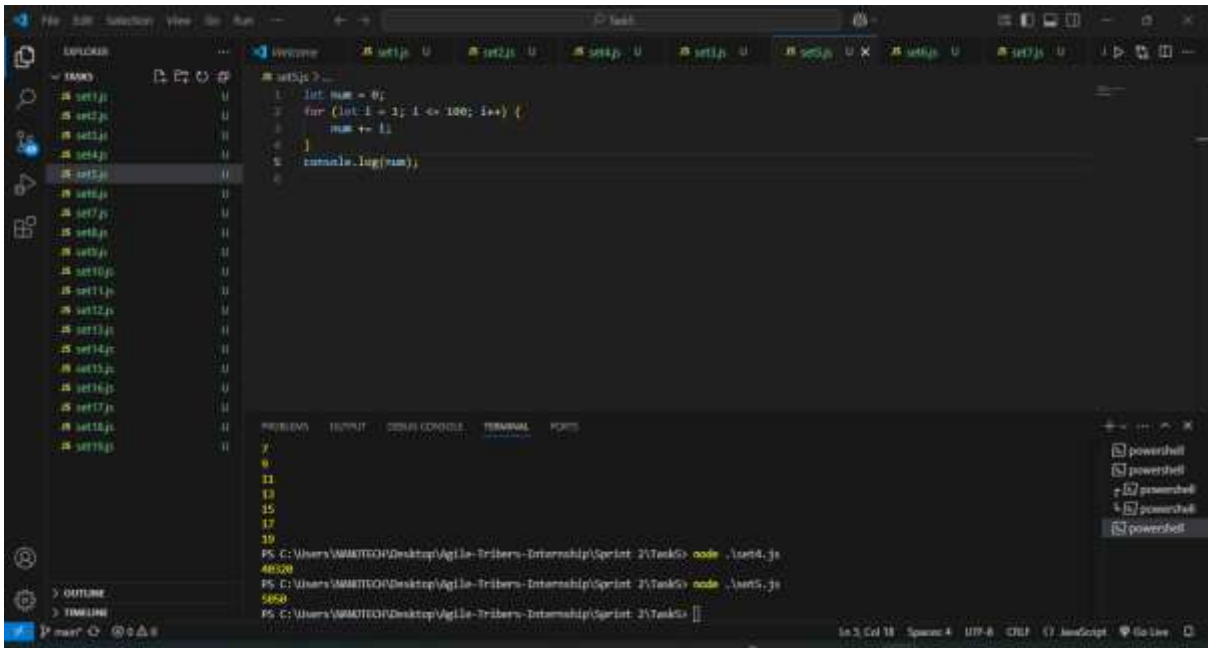
Set 3


function factor(n) {
let result = 1;
for (let a = 1; a <= n; a++) {
result *= a;
}
return result;
}
console.log(factor(8));
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set3.js
1
3
5
7
9
11
13
15
17
19
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5>

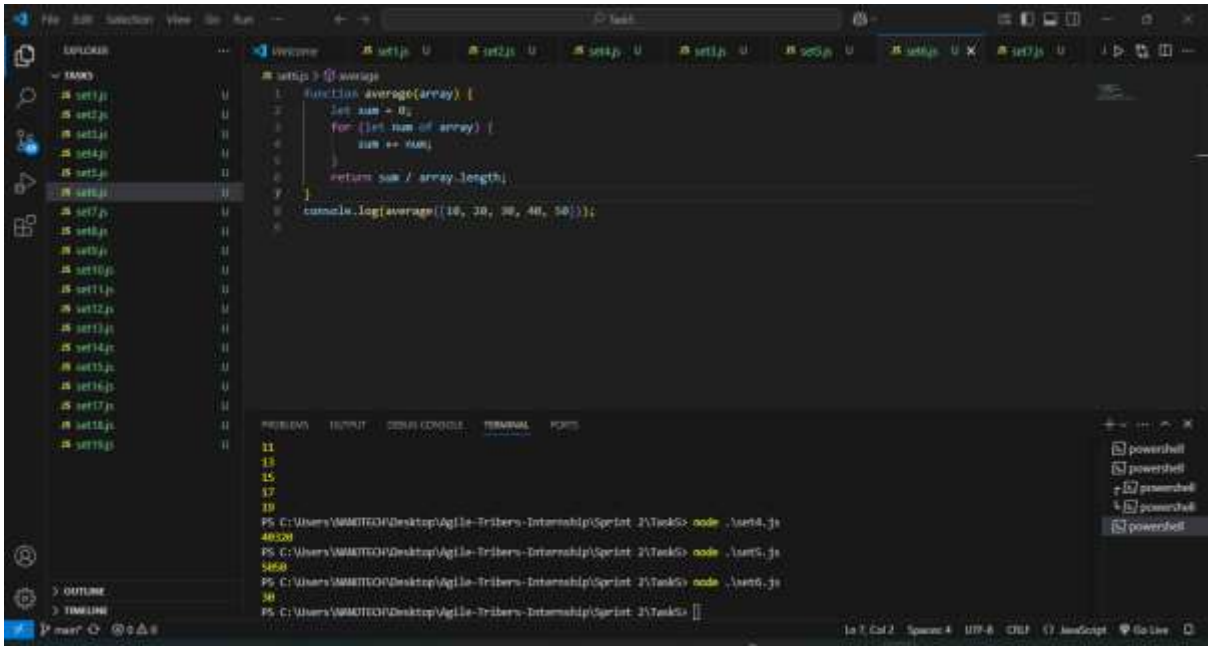
Set4


for (let a = 1; a <= 20; a += 2) {
console.log(a);
}
1
3
5
7
9
11
13
15
17
19
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set4.js
40320
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5>

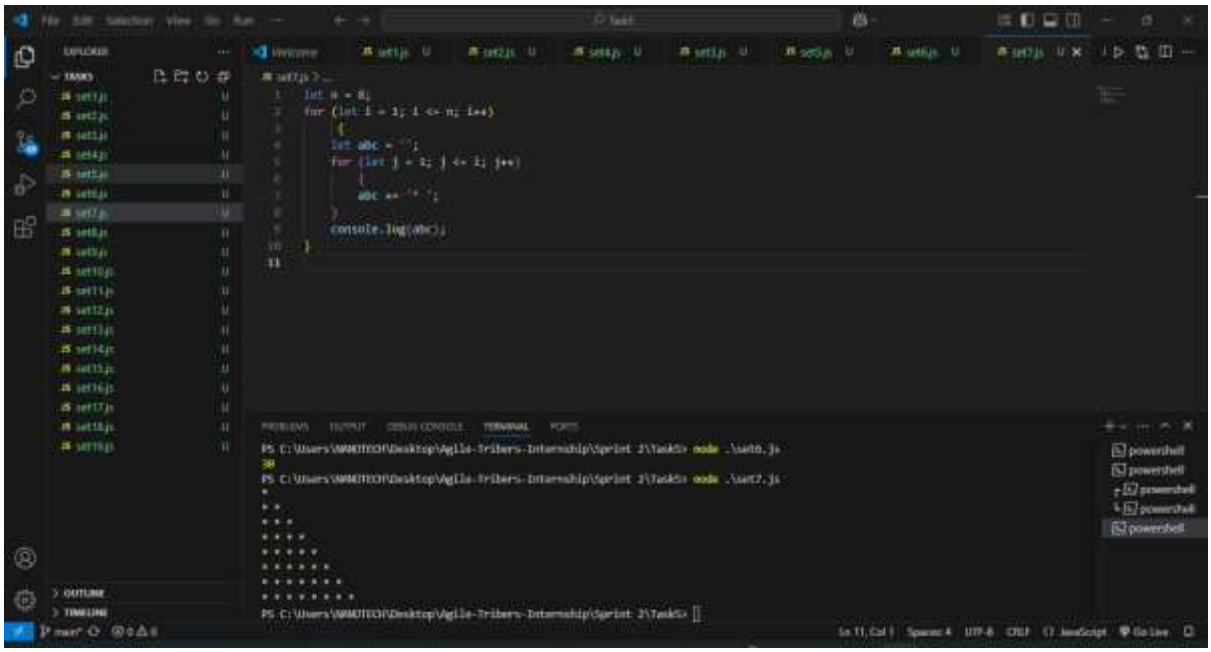
Set5


let num = 0;
for (let i = 1; i <= 100; i++) {
num += i;
}
console.log(num);
7
9
11
13
15
17
19
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set4.js
40320
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set5.js
5050
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5>

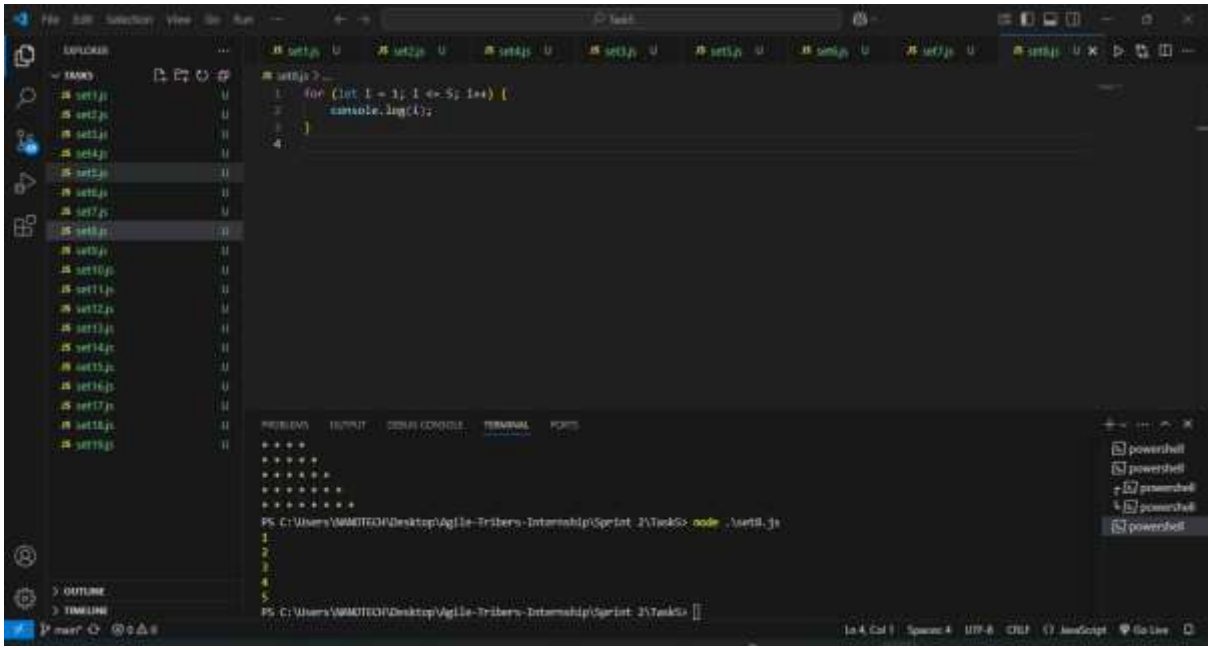
Set6


function average(array) {
let sum = 0;
for (let num of array) {
sum += num;
}
return sum / array.length;
}
console.log(average([10, 20, 30, 40, 50]));
11
13
15
17
19
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set4.js
40320
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set5.js
5050
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set6.js
30
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5>

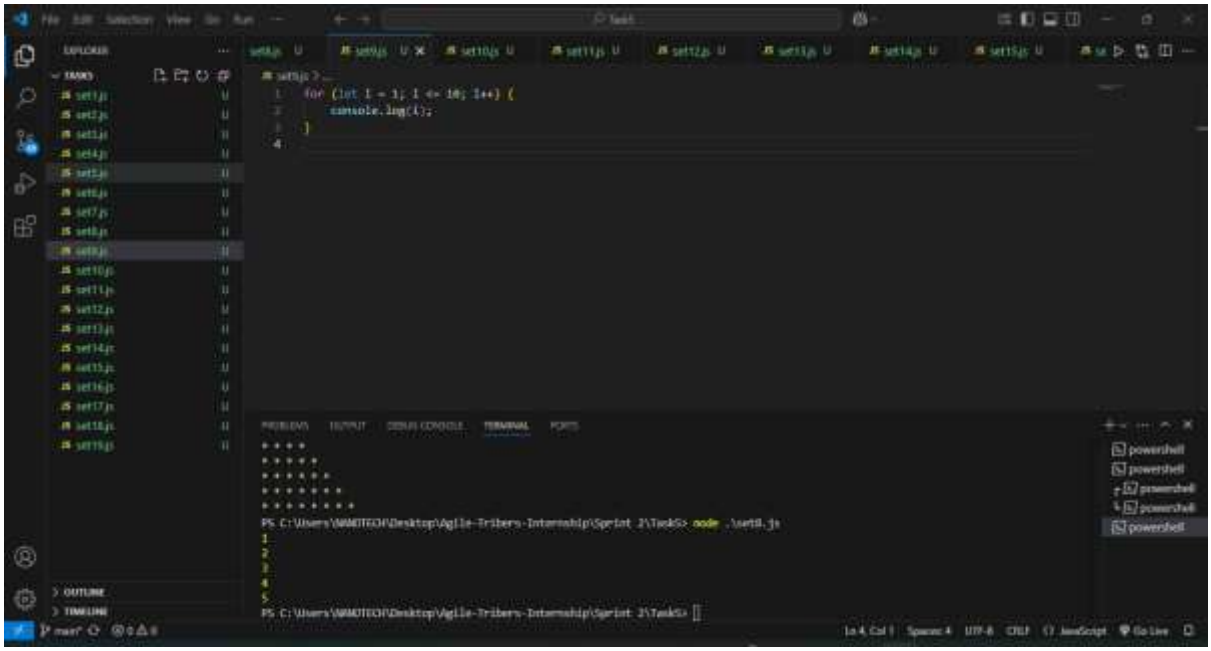
Set7


let n = 8;
for (let i = 1; i <= n; i++)
{
let abc = '';
for (let j = 1; j <= i; j++)
{
abc += '* ';
}
console.log(abc);
}
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set8.js
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set7.js
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5>

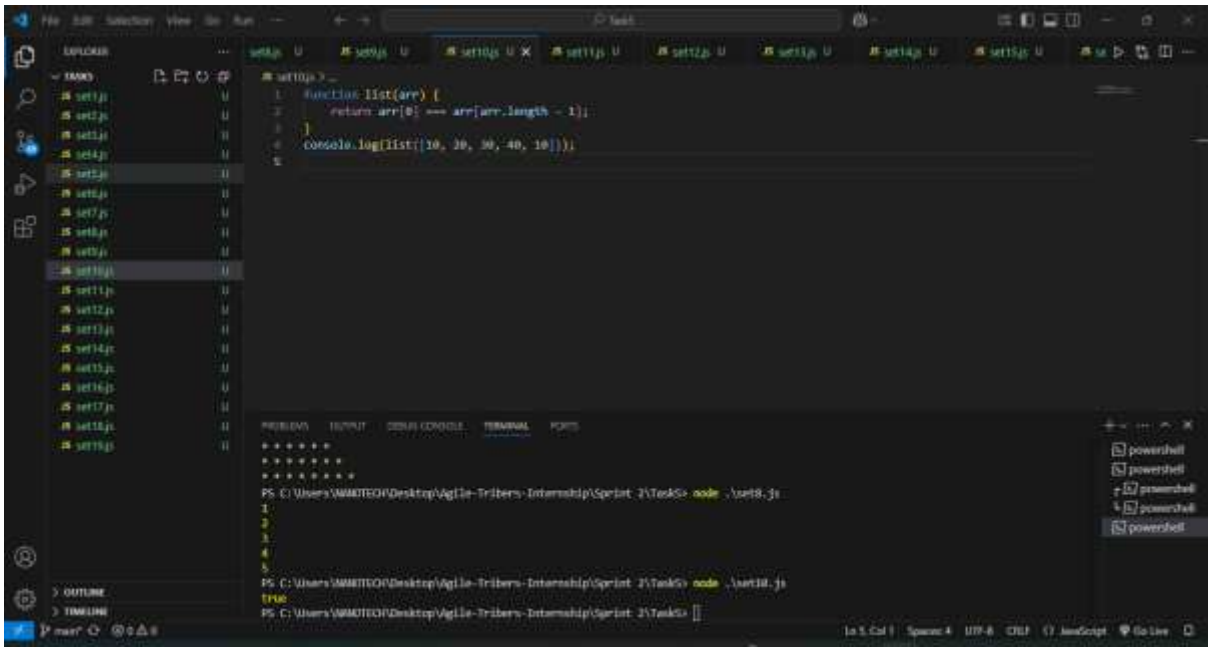
Set8


for (let i = 1; i <= 5; i++) {
console.log(i);
}
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set8.js
1
2
3
4
5
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5>

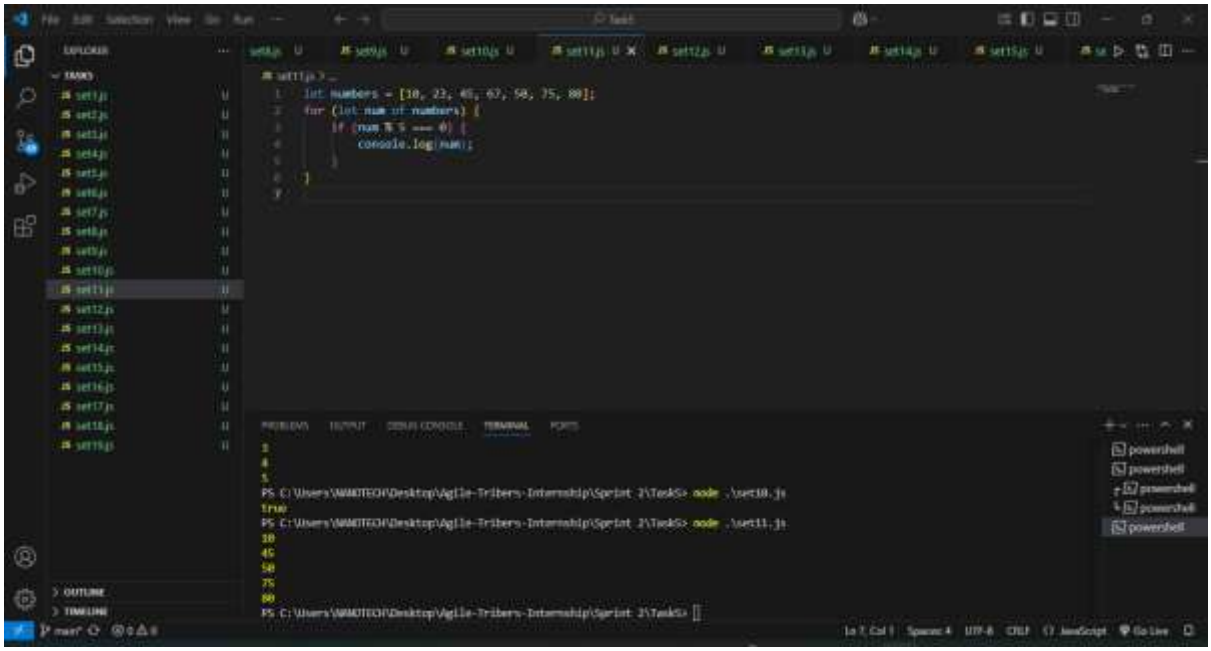
Set9


for (let i = 1; i <= 10; i++) {
console.log(i);
}
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\TaskS> node .\set8.js
1
2
3
4
5

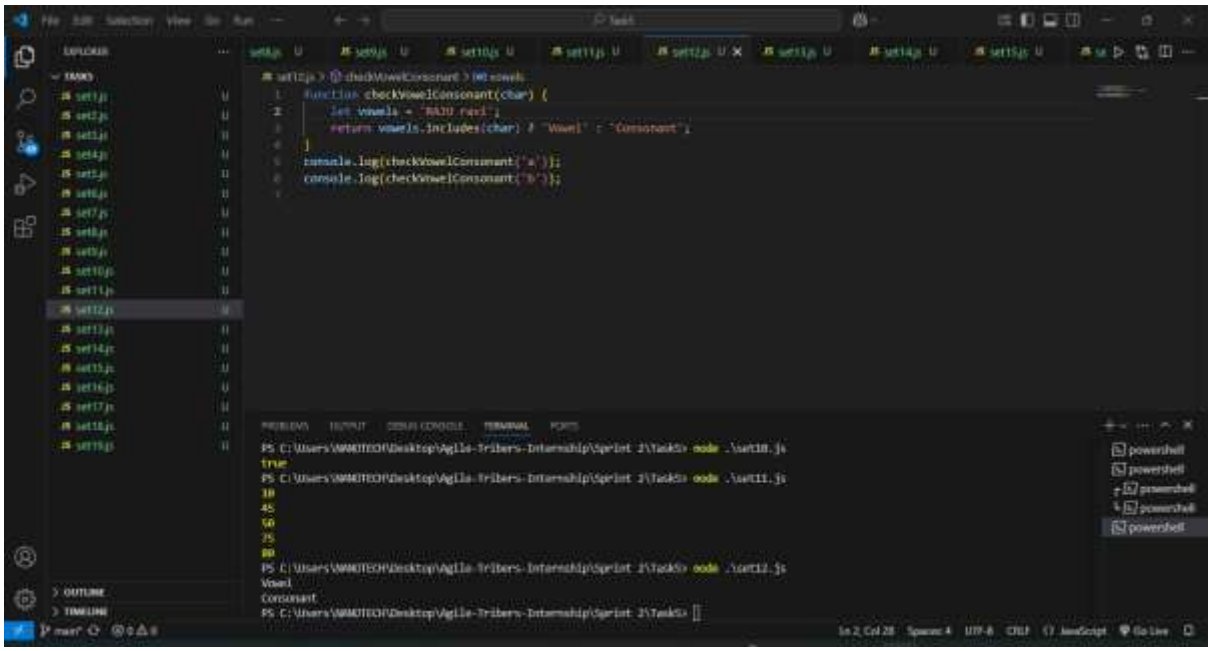
Set10


function list(arr) {
return arr[0] === arr[arr.length - 1];
}
console.log(list([10, 20, 30, 40, 10]));
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\TaskS> node .\set8.js
1
2
3
4
5
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\TaskS> node .\set10.js
true

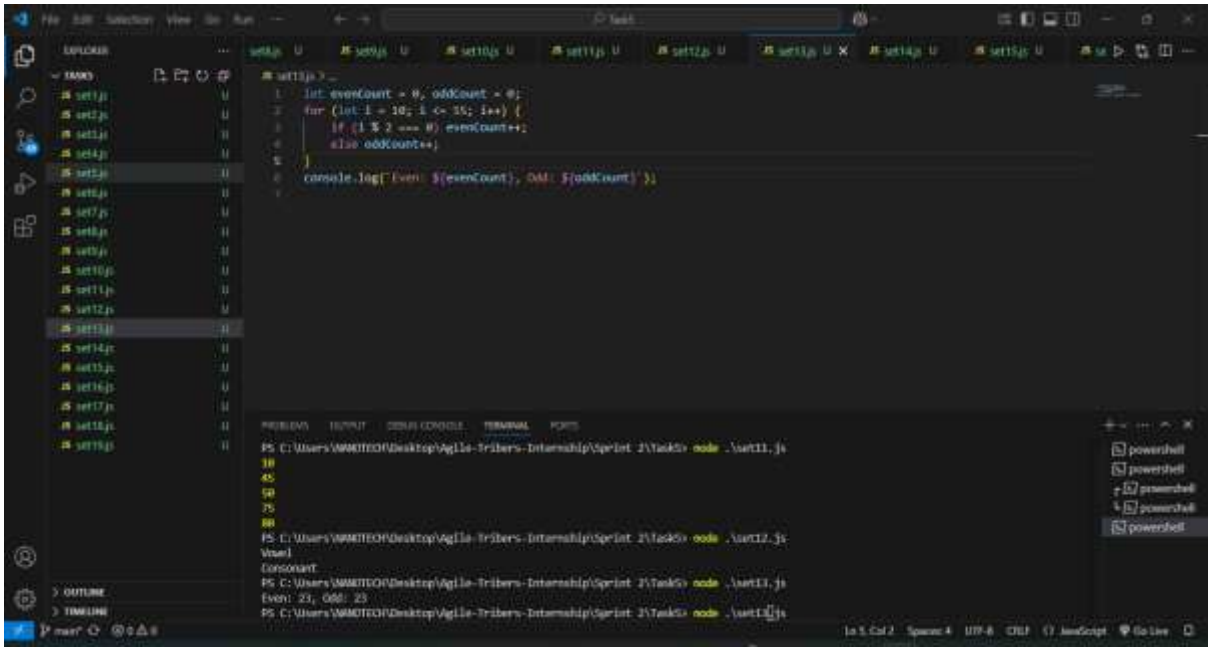
Set11


let numbers = [10, 23, 45, 67, 50, 75, 80];
for (let num of numbers) {
if (num % 5 === 0) {
console.log(num);
}
}
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set10.js
true
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set11.js
10
45
50
75
80
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5>

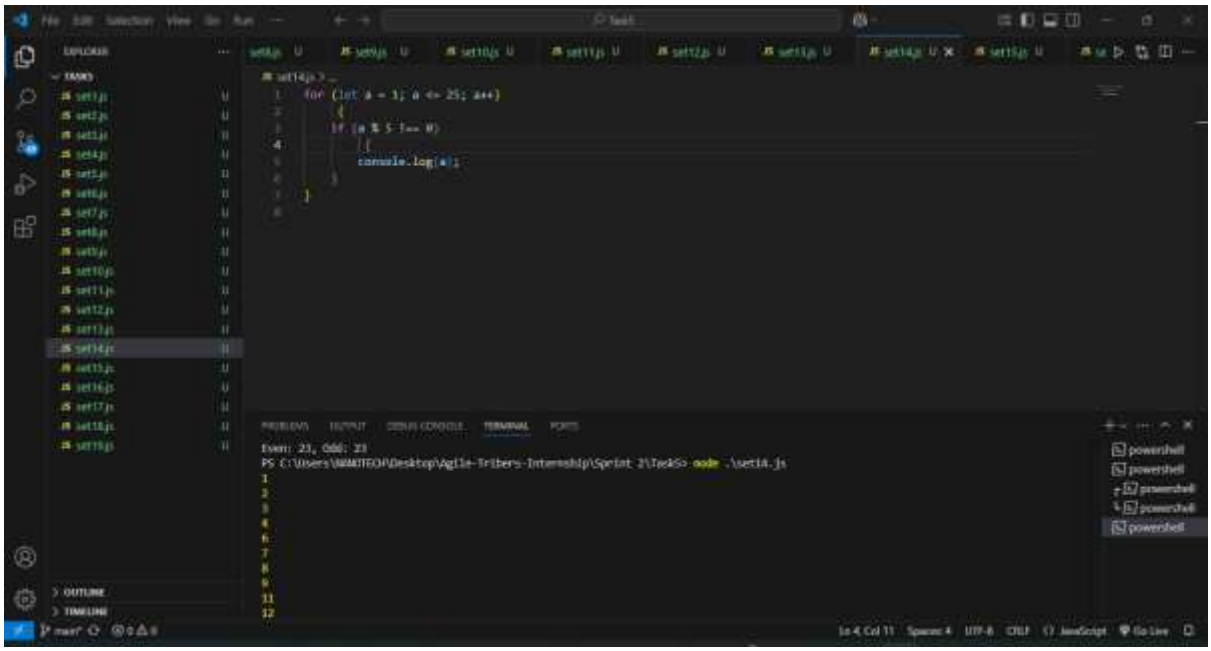
Set12


function checkVowelConsonant(char) {
let vowels = 'AEIOU';
return vowels.includes(char) ? 'Vowel' : 'Consonant';
}
console.log(checkVowelConsonant('a'));
console.log(checkVowelConsonant('b'));
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set10.js
true
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set11.js
10
45
50
75
80
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set12.js
Vowel
Consonant
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5>

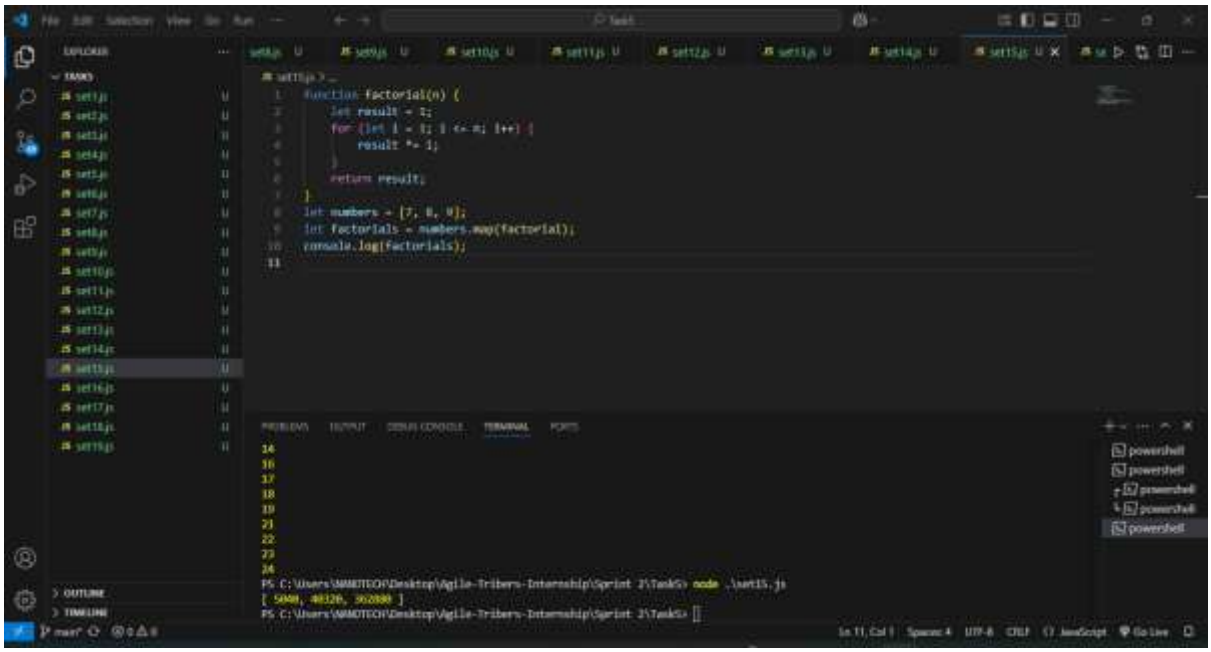
Set13


let evenCount = 0, oddCount = 0;
for (let i = 10; i <= 15; i++) {
if (i % 2 === 0) evenCount++;
else oddCount++;
}
console.log(`Even: ${evenCount}, Odd: ${oddCount}`);
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set11.js
10
45
50
75
80
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set12.js
Vowel
Consonant
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set13.js
Even: 23, Odd: 23
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set13.js

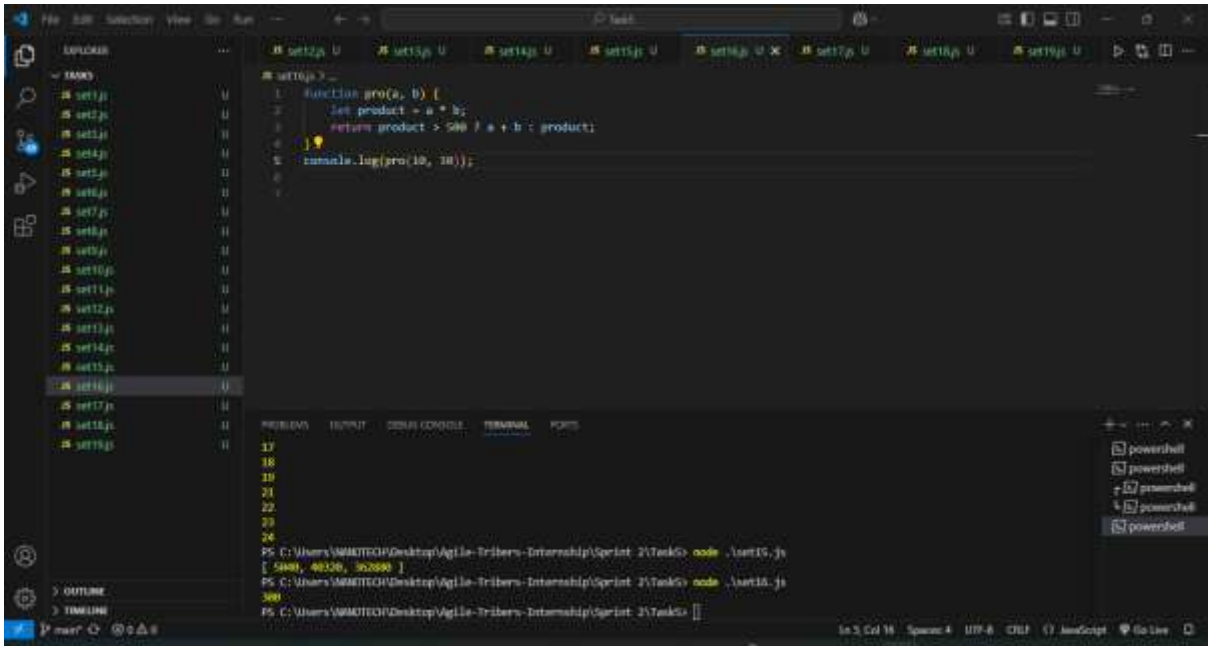
Set14


for (let a = 1; a <= 25; a++)
{
if (a % 5 !== 0)
{
console.log(a);
}
}
Even: 23, Odd: 23
PS C:\Users\WAMITECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set14.js
1
2
3
4
6
7
8
9
11
12

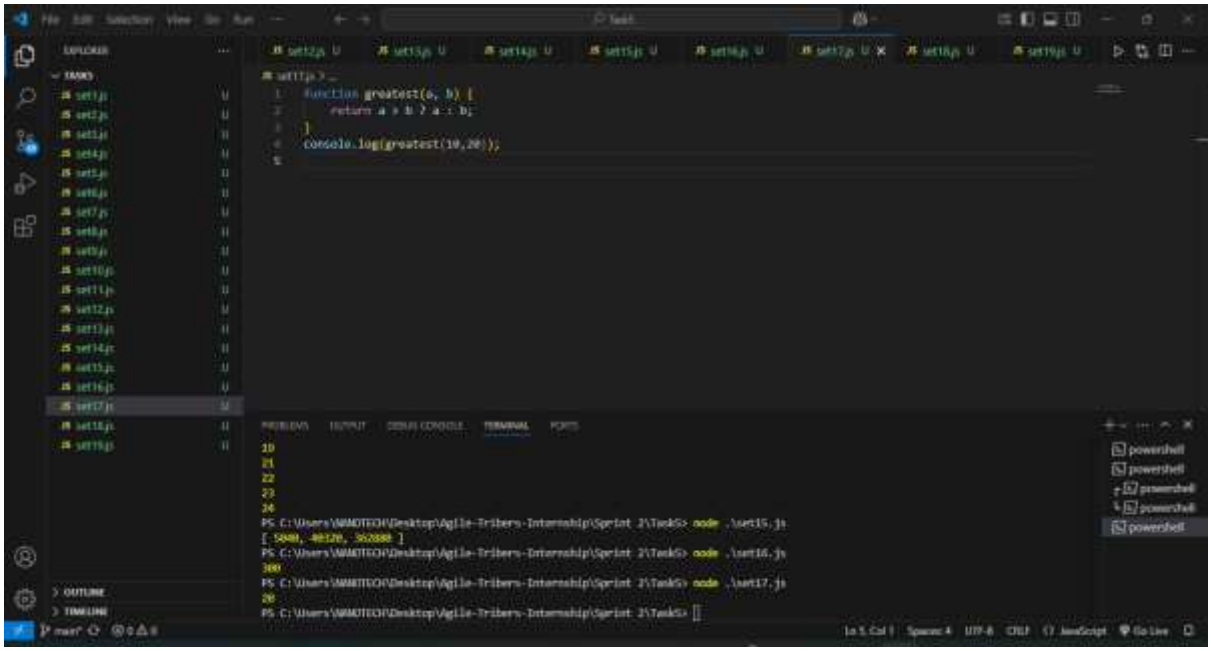
Set15


function factorial(n) {
let result = 1;
for (let i = 1; i <= n; i++) {
result *= i;
}
return result;
}
let numbers = [7, 8, 9];
let factorials = numbers.map(factorial);
console.log(factorials);
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set15.js
[5040, 40320, 362880]
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5>

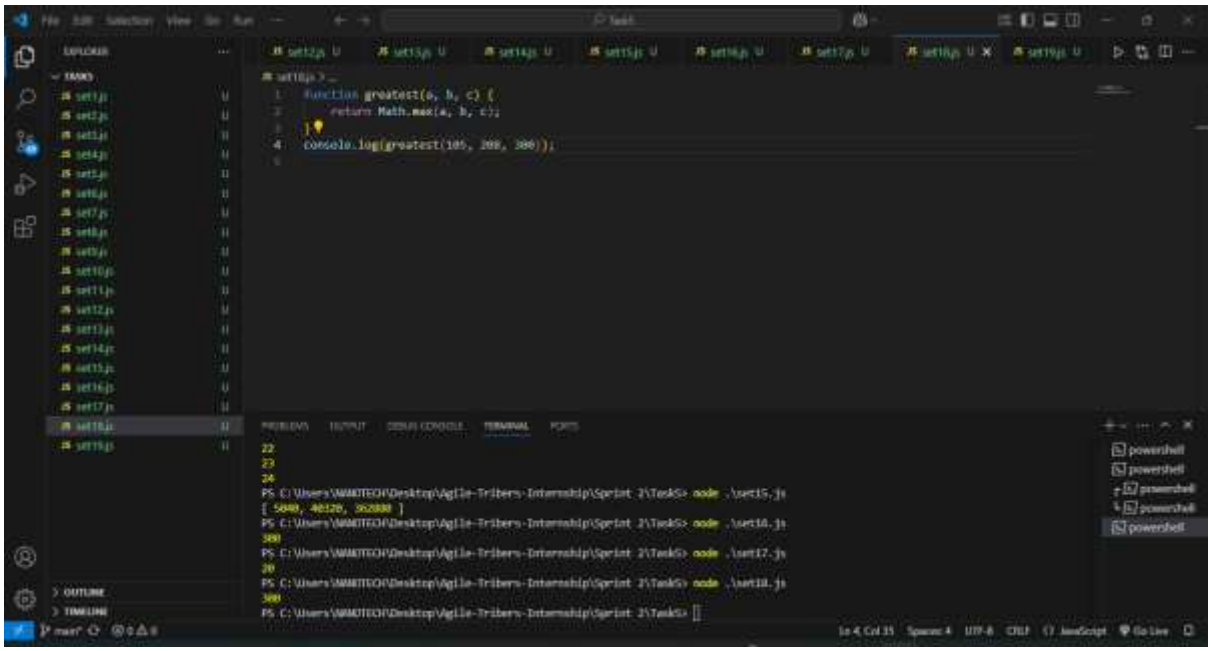
Set16


function pro(a, b) {
let product = a * b;
return product > 500 ? a + b : product;
}
console.log(pro(10, 30));
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set15.js
[5040, 40320, 362880]
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set16.js
500
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5>

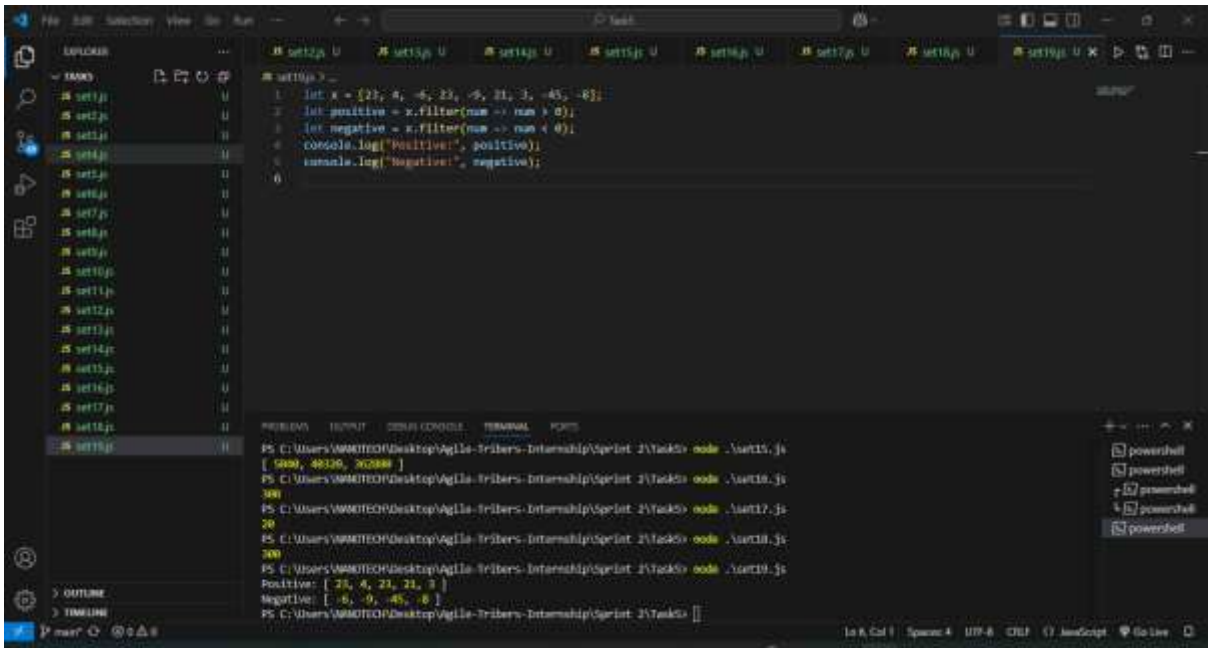
Set17


function greatest(a, b) {
return a > b ? a : b;
}
console.log(greatest(10,20));
10
21
22
23
24
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\TaskS> node .\set15.js
[5040, 40320, 362880]
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\TaskS> node .\set16.js
300
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\TaskS> node .\set17.js
20
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\TaskS>

Set18


function greatest(a, b, c) {
return Math.max(a, b, c);
}
console.log(greatest(105, 208, 300));
22
23
24
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\TaskS> node .\set15.js
[5040, 40320, 362880]
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\TaskS> node .\set16.js
300
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\TaskS> node .\set17.js
20
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\TaskS> node .\set18.js
300
PS C:\Users\WAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\TaskS>

Set19


let x = [21, 4, -6, 23, -9, 21, 3, -45, -8];
let positive = x.filter(num => num > 0);
let negative = x.filter(num => num < 0);
console.log("Positive:", positive);
console.log("Negative:", negative);
PS C:\Users\VAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set15.js
[5000, 40320, 362880]
PS C:\Users\VAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set16.js
500
PS C:\Users\VAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set17.js
20
PS C:\Users\VAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set18.js
200
PS C:\Users\VAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5> node .\set19.js
Positive: [21, 4, 23, 21, 3]
Negative: [-6, -9, -45, -8]
PS C:\Users\VAMOTECH\Desktop\Agile-Tribers-Internship\Sprint 2\Task5>