

Practical 3-

The screenshot shows the VS Code interface with a file explorer on the left containing files for 'PRACTICAL 3 NODE'. The main editor displays 'app.js' with the following code:

```
1 const { add, subtract } = require('./math');
2
3 console.log('=== Math Operations Demo ===');
4
5 const num1 = 10;
6 const num2 = 5;
7
8 const sum = add(num1, num2);
9 console.log(`${num1} + ${num2} = ${sum}`);
10
11
12 const difference = subtract(num1, num2);
13 console.log(`${num1} - ${num2} = ${difference}`);
14
15 console.log('Additional examples:');
16 console.log(`20 + 15 = ${add(20, 15)}`);
17 console.log(`50 - 25 = ${subtract(50, 25)}`);
18
```

The terminal at the bottom shows the command `node app.js` being executed, resulting in the following output:

```
PS D:\Practical 3 node> node app.js
=== Math Operations Demo ===
10 + 5 = 15
10 - 5 = 5
Additional examples:
20 + 15 = 35
50 - 25 = 25
PS D:\Practical 3 node>
```

The screenshot shows the VS Code interface with a file explorer on the left containing files for 'PRACTICAL 3 NODE'. The main editor displays 'Blocking.js' with the following code:

```
1
2 const fs = require('fs');
3
4 console.log('1. FILE READING EXAMPLE ');
5 console.log('1. Program started...');
6
7 try {
8   console.log('2. Reading file now...');
9
10  const fileContent = fs.readFileSync('sample.txt', 'utf8');
11
12  console.log('3. File loaded successfully!');
13  console.log('4. Here is the content:');
14  console.log('-----');
15  console.log(fileContent);
16  console.log('-----');
17
18 } catch (err) {
19   console.error('File error:', err.message);
20 }
```

The terminal at the bottom shows the command `node Blocking.js` being executed. The output includes student and course information:

```
PS D:\Practical 3 node> node Blocking.js
name: RITU
Student ID: GF202346594
Course: Computer Science Engineering
Semester: 5th Semester
Email: ritu@university.edu

Practical for Node js
```

PRACTICAL 3 NODE

JS app.js

JS Blocking.js

JS math.js

JS nonblocking.js

package.json

sample.txt

setTimeout.js

```
1  const fs = require('fs');
2
3  console.log('ASYNC FILE READ EXAMPLE');
4  console.log('1. Program initialization...');
5
6  console.log('2. Starting file read operation (ASYNC)...');
7
8  fs.readFile('sample.txt', 'utf8', (err, fileContent) => {
9    if (err) {
10      console.error('File reading failed:', err.message);
11      return;
12    }
13
14    console.log('4. File loaded successfully! (callback executed)');
15    console.log('5. Content below:');
16    console.log('=====');
17    console.log(fileContent);
18    console.log('=====');
19    console.log('6. Async operation completed!');
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POLYGLOT NOTEBOOK

PS D:\Practical 3 node> node nonblocking.js

2. Starting file read operation (ASYNC)...

3. This executes RIGHT AWAY after file read starts

7. Main thread execution done

8. Note: File content will appear after this line

4. File loaded successfully! (callback executed)

5. Content below:

=====

Student Information System

Name: Ritu

Student ID: GF202346594

Course: Computer Science Engineering

Semester: 5th Semester

Email: ritu@university.edu

Practical for Node is

powerShell

powerShell

powerShell

powerShell

PRACTICAL 3 NODE

JS app.js

JS Blocking.js

JS math.js

JS nonblocking.js

JS p3.js

package.json

sample.txt

```
1  p3.js > loadPersonInfo > setTimeout() callback > personInfo > emailAddress
2
3  function loadPersonInfo() {
4    console.log('Loading person information...');
5
6    setTimeout(() => {
7      const personInfo = {
8        fullName: 'Ritu Singh',
9        emailAddress: 'Ritu99@gmail.com'
10      };
11
12      console.log('Information loaded successfully!');
13      console.log('Person Details:', personInfo);
14    }, 2000);
15  }
16
17  loadPersonInfo();
18
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POLYGLOT NOTEBOOK

PS D:\Practical 3 node> node nonblocking.js

Practical for Node js

=====

6. Async operation completed!

9. Background tasks can run while file loads!

PS D:\Practical 3 node> node p3.js

Loading person information...

Information loaded successfully!

Person Details: { fullName: 'Ritu Singh', emailAddress: 'Ritu99@gmail.com' }

powerShell

powerShell

powerShell

powerShell