

## Assignment - 5

Read file " myfile.txt" in nodejs using synchronous and asynchronous file function and find out How much time saved by non -blocking asynchronous I/O operation.

Note: Create separate nodejs program for blocking and non-blocking I/O

### Txt file:

This is NodeJs  
This is NodeJs  
This is NodeJs  
This is NodeJs  
This is NodeJs  
This is NodeJs  
This is NodeJs  
This is NodeJs  
This is NodeJs  
This is NodeJs  
This is NodeJs

### Blocking I/O:

```
var fs = require('fs');  
console.log("Serving User 1");  
console.time();  
var data = fs.readFileSync('myfile.txt');  
console.log(data.toString());  
console.timeEnd();  
console.log("Serving user 2")  
console.log("Serving user 3");  
console.log("Program ended");
```

## Non Blocking I/O:

```
console.time();
var fs = require('fs');
console.log("Serving User 1");
console.time();
var data = fs.readFile('myfile.txt', function (err, data) {
    if (err) return console.error(err);
    console.log(data.toString());
});
console.timeEnd();
console.log("Serving User 2");
console.log("Serving User 3");
console.log("Serving User 4")
```

### Output of Both Files :

The screenshot shows the Visual Studio Code interface with the following components:

- Explorer Panel:** Displays a file tree with folders `NEEL` and `node`. The `node` folder contains files `block.js`, `hello.js`, `myfile.txt`, and `non.js`. The `node` folder is selected.
- Terminal Panel:** Shows the output of Node.js commands. The first command is `node block.js`, which outputs "This is Node.js" 10 times. The second command is `node non.js`, which outputs "This is Node.js" 10 times.
- Output Panel:** Shows the output of the `node` command, which is "This is Node.js" repeated 10 times.
- Terminal Window:** A PowerShell terminal window is open, showing the command `node block.js` and its output.