МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХЕРСОНСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
КАФЕДРА ПРОГРАМНИХ ЗАСОБІВ І ТЕХНОЛОГІЙ

Лабораторна робота №1
з дисципліни Розробка бек-енд

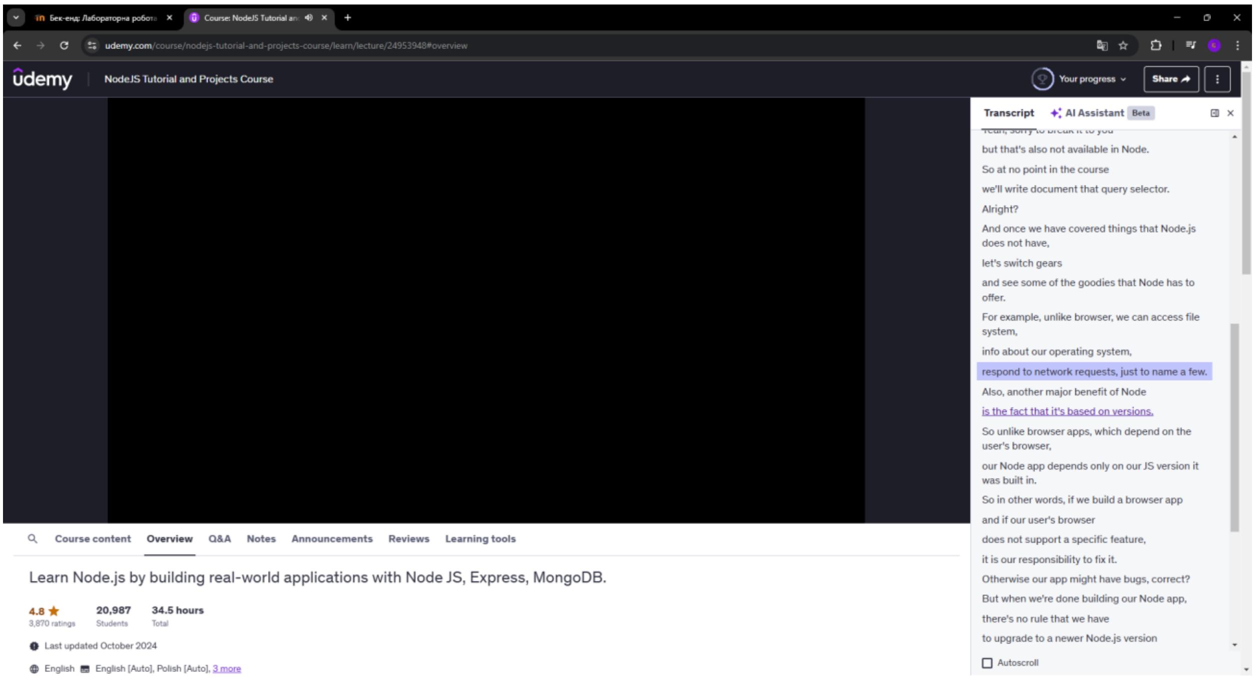
Виконав	
студент групи ЗПР1	Бойко М.Є
Перевірив	
викладач	Щербанюк С.А

Херсон 2025

Тема: Знайомство з NodeJS

Завдання

Опрацювати матеріали відео курсу по модулях: 1 – 23



Бек-енд: Лабораторна робота
Course: NodeJS Tutorial an
udemy.com/course/nodejs-tutorial-and-projects-course/learn/lecture/24953948#overview
ůdemy
NodeJS Tutorial and Projects Course
Your progress
Share
Transcript
AI Assistant
Beta
Course content
Overview
Q&A
Notes
Announcements
Reviews
Learning tools

Learn Node.js by building real-world applications with Node JS, Express, MongoDB.

4.8 ★ 3,870 ratings
20,987 Students
34.5 hours Total

Last updated October 2024

English English [Auto], Polish [Auto], 3 more

Yeah, sorry to break it to you

but that's also not available in Node.

So at no point in the course

we'll write document that query selector.

Alright?

And once we have covered things that Node.js does not have,

let's switch gears

and see some of the goodies that Node has to offer.

For example, unlike browser, we can access file system,

info about our operating system,

respond to network requests, just to name a few.

Also, another major benefit of Node

is the fact that it's based on versions.

So unlike browser apps, which depend on the user's browser,

our Node app depends only on our JS version it was built in.

So in other words, if we build a browser app

and if our user's browser

does not support a specific feature,

it is our responsibility to fix it.

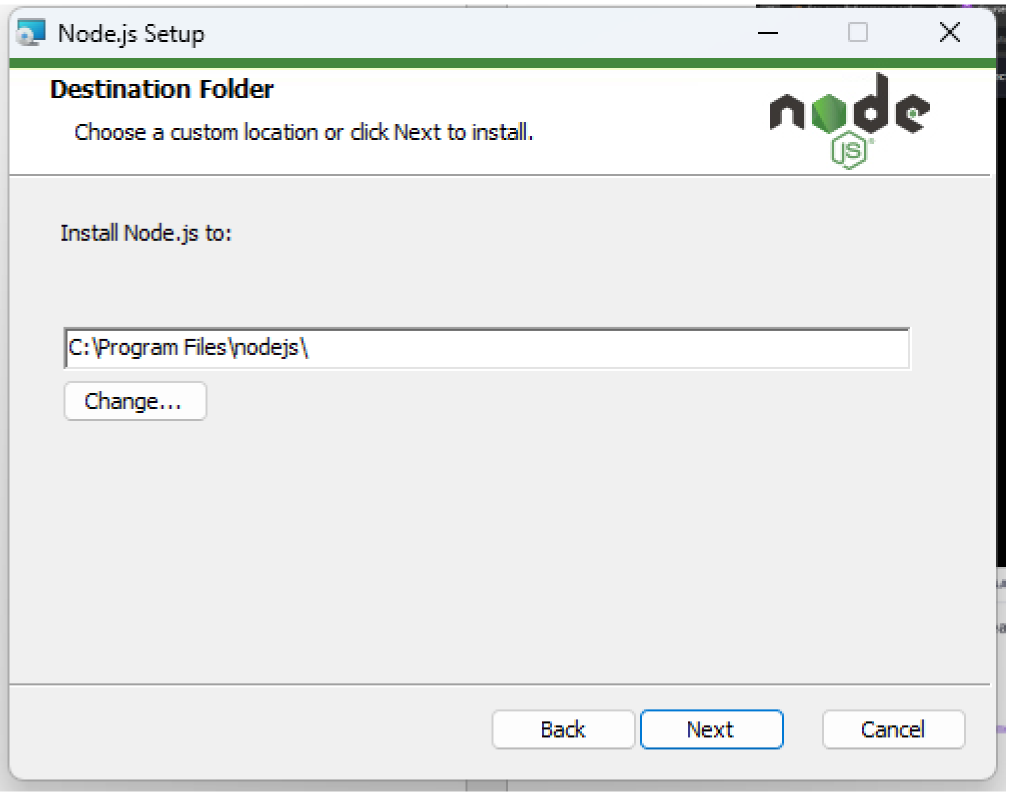
Otherwise our app might have bugs, correct?

But when we're done building our Node app,

there's no rule that we have

to upgrade to a newer Node.js version

□ Autoscroll



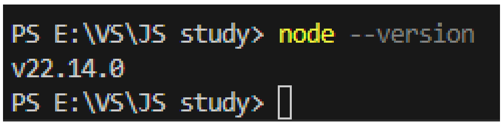
Destination Folder

Choose a custom location or click Next to install.

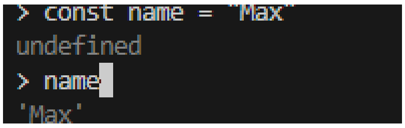
Install Node.js to:

C:\Program Files\nodejs\

Change...

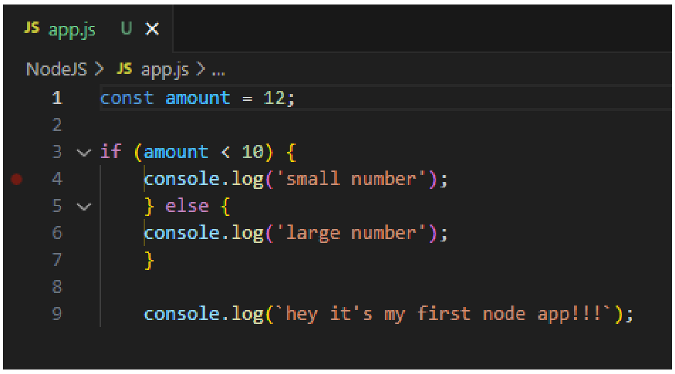


```
PS E:\VS\JS study> node --version
v22.14.0
PS E:\VS\JS study> 
```

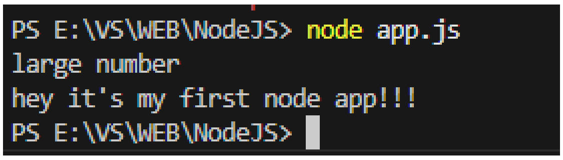


```
> const name = "Max"
undefined
> name
'Max'
```

Globals

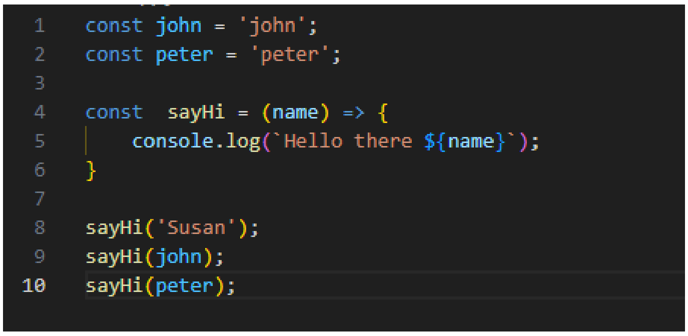


```javascript
const amount = 12;

if (amount < 10) {
    console.log('small number');
    } else {
    console.log('large number');
    }

    console.log(`hey it's my first node app!!!`);
```

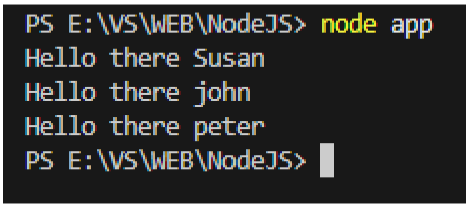


```
PS E:\VS\WEB\NodeJS> node app.js
large number
hey it's my first node app!!!
PS E:\VS\WEB\NodeJS> 
```

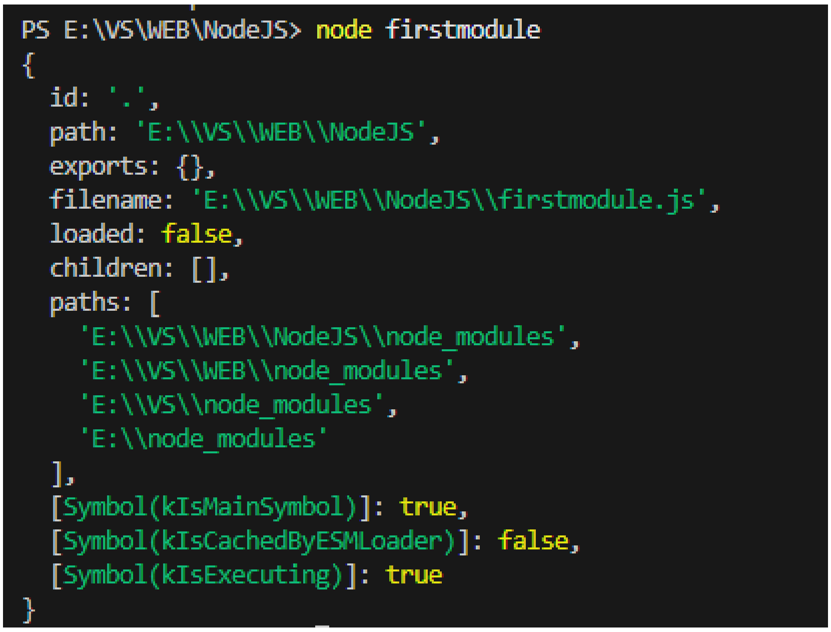
Modules



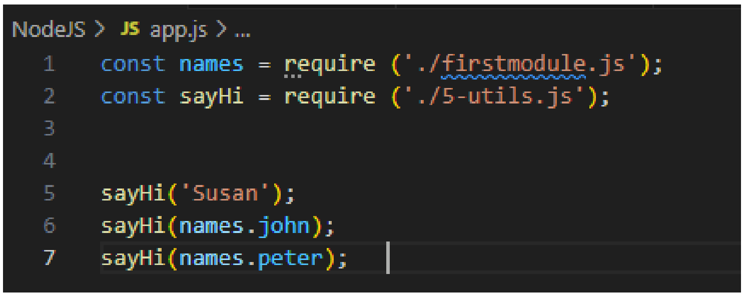
```javascript
const john = 'john';
const peter = 'peter';

const  sayHi = (name) => {
    console.log(`Hello there ${name}`);
}

sayHi('Susan');
sayHi(john);
sayHi(peter);
```



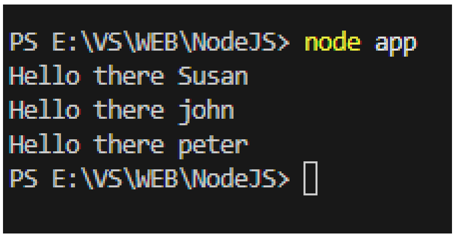
```
PS E:\VS\WEB\NodeJS> node app
Hello there Susan
Hello there john
Hello there peter
PS E:\VS\WEB\NodeJS> 
```



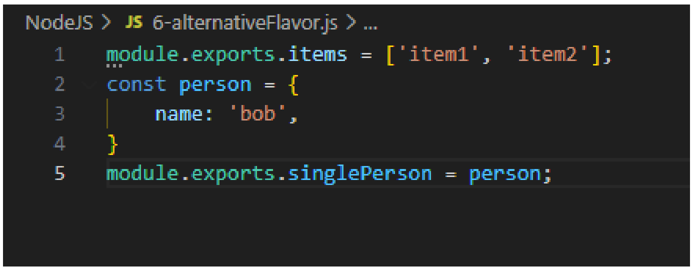
```
PS E:\VS\WEB\NodeJS> node firstmodule
{
  id: '.',
  path: 'E:\\VS\\WEB\\NodeJS',
  exports: {},
  filename: 'E:\\VS\\WEB\\NodeJS\\firstmodule.js',
  loaded: false,
  children: [],
  paths: [
    'E:\\VS\\WEB\\NodeJS\\node_modules',
    'E:\\VS\\WEB\\node_modules',
    'E:\\VS\\node_modules',
    'E:\\node_modules'
  ],
  [Symbol(kIsMainSymbol)]: true,
  [Symbol(kIsCachedByESMLoader)]: false,
  [Symbol(kIsExecuting)]: true
}
```



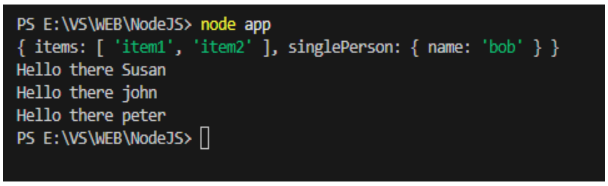
```
const names = require ('./firstmodule.js');
const sayHi = require ('./5-utils.js');


sayHi('Susan');
sayHi(names.john);
sayHi(names.peter);
```



```
PS E:\VS\WEB\NodeJS> node app
Hello there Susan
Hello there john
Hello there peter
PS E:\VS\WEB\NodeJS> 
```



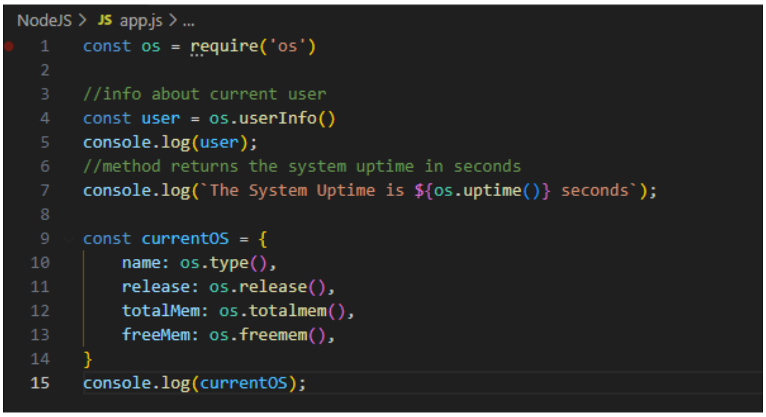
```
module.exports.items = ['item1', 'item2'];
const person = {
    name: 'bob',
}
module.exports.singlePerson = person;
```



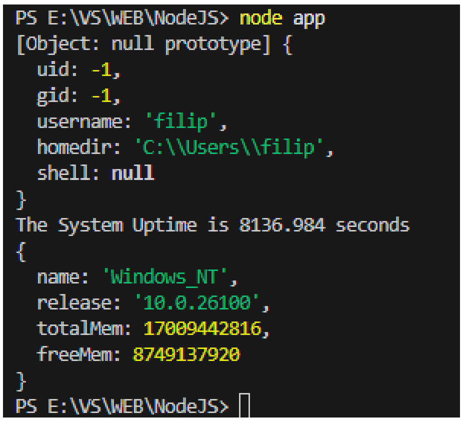
```
PS E:\VS\WEB\NodeJS> node app
{ items: [ 'item1', 'item2' ], singlePerson: { name: 'bob' } }
Hello there Susan
Hello there john
Hello there peter
PS E:\VS\WEB\NodeJS> 
```

Built-in Modules

OS:

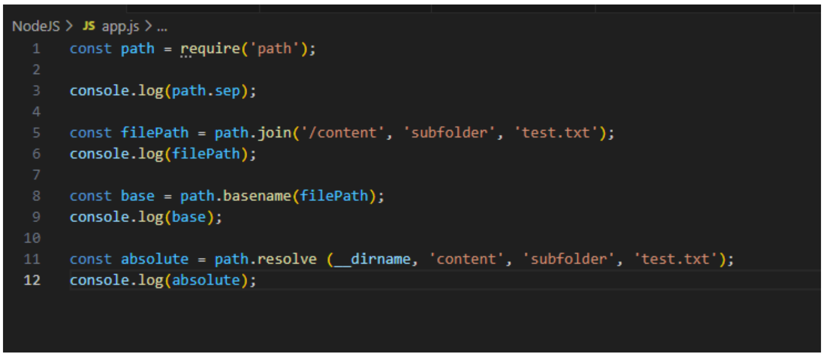


```javascript
const os = require('os')

//info about current user
const user = os.userInfo()
console.log(user);
//method returns the system uptime in seconds
console.log(`The System Uptime is ${os.uptime()} seconds`);

const currentOS = {
    name: os.type(),
    release: os.release(),
    totalMem: os.totalmem(),
    freeMem: os.freemem(),
}
console.log(currentOS);
```

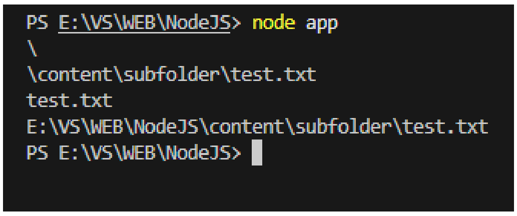


```
PS E:\VS\WEB\NodeJS> node app
[Object: null prototype] {
  uid: -1,
  gid: -1,
  username: 'filip',
  homedir: 'C:\\Users\\filip',
  shell: null
}
The System Uptime is 8136.984 seconds
{
  name: 'Windows_NT',
  release: '10.0.26100',
  totalMem: 17009442816,
  freeMem: 8749137920
}
PS E:\VS\WEB\NodeJS>
```

Path:

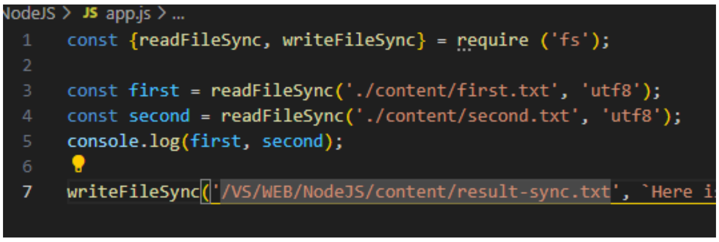


```javascript
const path = require('path');

console.log(path.sep);

const filePath = path.join('/content', 'subfolder', 'test.txt');
console.log(filePath);

const base = path.basename(filePath);
console.log(base);

const absolute = path.resolve (__dirname, 'content', 'subfolder', 'test.txt');
console.log(absolute);
```

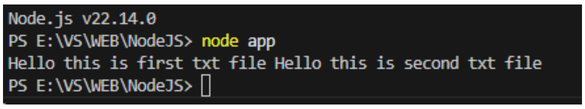


```
PS E:\VS\WEB\NodeJS> node app
\
\content\subfolder\test.txt
test.txt
E:\VS\WEB\NodeJS\content\subfolder\test.txt
PS E:\VS\WEB\NodeJS>
```

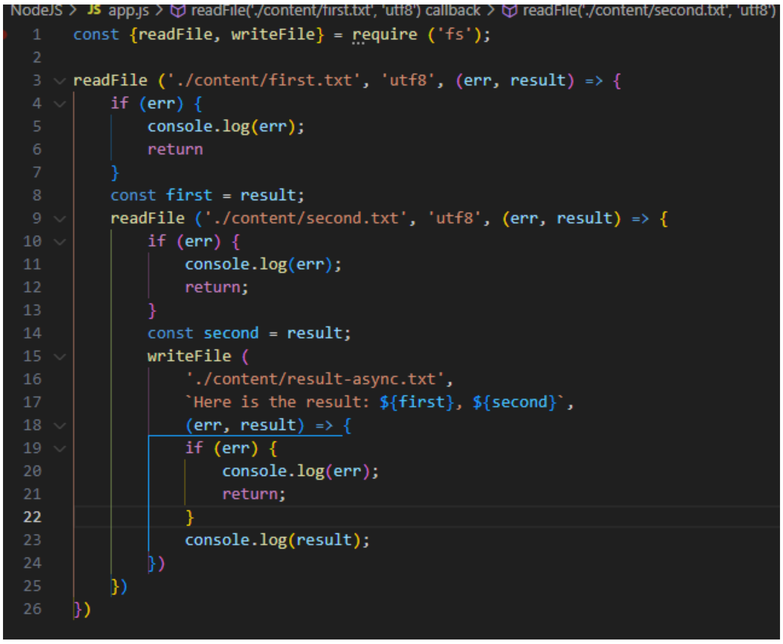
FS:



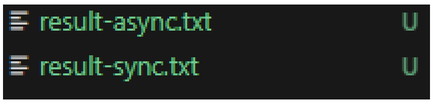
```javascript
const {readFileSync, writeFileSync} = require ('fs');

const first = readFileSync('./content/first.txt', 'utf8');
const second = readFileSync('./content/second.txt', 'utf8');
console.log(first, second);

writeFileSync('/VS/WEB/NodeJS/content/result-sync.txt', `Here i
```



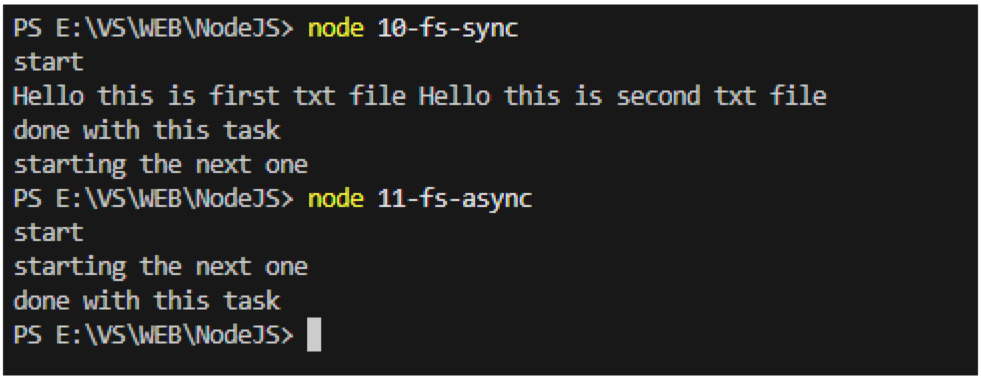
```
Node.js v22.14.0
PS E:\VS\WEB\NodeJS> node app
Hello this is first txt file Hello this is second txt file
PS E:\VS\WEB\NodeJS>
```


NodeJS > JS app.js > readFile('./content/first.txt', 'utf8') callback > readFile('./content/second.txt', 'utf8')

```
const {readFile, writeFile} = require ('fs');

readFile ('./content/first.txt', 'utf8', (err, result) => {
    if (err) {
        console.log(err);
        return
    }
    const first = result;
    readFile ('./content/second.txt', 'utf8', (err, result) => {
        if (err) {
            console.log(err);
            return;
        }
        const second = result;
        writeFile (
            './content/result-async.txt',
            `Here is the result: ${first}, ${second}`,
            (err, result) => {
            if (err) {
                console.log(err);
                return;
            }
            console.log(result);
        })
    })
})
```



Sync vs Async



```
PS E:\VS\WEB\NodeJS> node 10-fs-sync
start
Hello this is first txt file Hello this is second txt file
done with this task
starting the next one
PS E:\VS\WEB\NodeJS> node 11-fs-async
start
starting the next one
done with this task
PS E:\VS\WEB\NodeJS>
```