

Поставить NodeJs и запустить \*.js-файлы из консоли

```
C:\vasya>node index.js
C:\vasya>node index.js
ураааа
C:\vasya>
```

Написать скрипт, который считывает текстовый файл с диска и записывает его в обратном порядке ('абв' -> 'вба') в тот же файл.

```
JS index.js > ...
1 let fs = require('fs');
2 let fileContent = fs.readFileSync('file.txt', 'utf8');
3 let mas = fileContent.split('').reverse()
4 fileContent = ""
5 for (let i = 0; i < mas.length; i++)
6   fileContent += mas[i]
7 fs.writeFile('file.txt', fileContent, () => console.log('сработало'))
```

file.txt – Блокнот

Файл Правка Формат Вид Справка

Лляф еж тот в )'абв' >- 'вба'( екдяроп монтарбо в оге теавысипаз и аксид с лляф йывотскет теавытичс йыроток ,тпиркс ьтасипаН

Написать сервер, который при запросе GET / отправляет содержимое некоего файла на диске (см. предыдущую задачу). POST / добавляет в файл тело запроса (body).

```
JS index.js > server > http.createServer() callback
1 const http = require("http");
2 const parse = require('co-body');
3
4
5 const server = http.createServer(async function(req, res) {
6   if (req.method === 'GET') {
7     let fs = require('fs');
8     let fileContent = fs.readFileSync('file.txt', 'utf8');
9     res.write(fileContent)
10    res.end()
11  } else if (req.method === 'POST') {
12    const body = await parse.json(req)
13
14    res.end(JSON.stringify(body))
15  }
16
17 });
18 server.listen(8080);
```

## Задание по NPM

Инициализировать проект

```
PS C:\Users\admin\Desktop\proekt> npm init -y
Wrote to C:\Users\admin\Desktop\proekt\package.json:

{
  "name": "proekt",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
```

Поставить библиотеку lodash

```
PS C:\Users\admin\Desktop\proekt> npm install lodash
added 1 package, and audited 2 packages in 1s
found 0 vulnerabilities
```

Подключить её и отсортировать массив объектов по полю age (можно использовать метод sortBy):

```
let lod = require('lodash')

const users = [
  { user: 'fred', age: 48 },
  { user: 'barney', age: 36 },
  { user: 'fred', age: 40 },
  { user: 'barney', age: 34 }
]

let sortUsers = lod.sortBy(users, 'age')

console.log(sortUsers)
```

```
PS C:\Users\admin\Desktop\proekt> node index.js
[
  { user: 'barney', age: 34 },
  { user: 'barney', age: 36 },
  { user: 'fred', age: 40 },
  { user: 'fred', age: 48 }
]
```

Вызвать написанный код через npm run (используется поле script).

```
1  {
2    "name": "proekt",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "test": "echo \"Error: no test specified\" && exit 1",
8      "sort": "node index.js"
9    },
10   "keywords": [],
11   "author": "",
12   "license": "ISC",
13   "dependencies": {
14     "lodash": "^4.17.21"
15   }
16 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\Users\admin\Desktop\proekt> npm run sort
Debugger attached.
```

```
> proekt@1.0.0 sort
> node index.js
```

```
Debugger attached.
```

```
[
  { user: 'barney', age: 34 },
  { user: 'barney', age: 36 },
  { user: 'fred', age: 40 },
  { user: 'fred', age: 48 }
]
```

```
Waiting for the debugger to disconnect...
```

```
Waiting for the debugger to disconnect...
```

```
PS C:\Users\admin\Desktop\proekt> 
```