Работа с файлами. Чтение. Синхронное

Для **работы с файлами** в Node.js используется модуль **fs**

Для примера создадим текстовый файл **hello.txt** (кодировка **UTF–8**):

```
Hello World!
```

Для **чтения** данных из файла в **синхронном** режиме используется метод **readFileSync()**, параметры:

1 – путь к файлу, относительно файла приложения
2 – кодировка

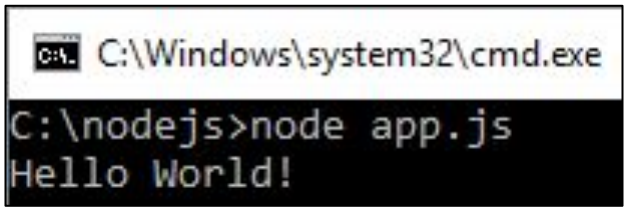
Результат: считанные данные

В каталоге где расположен файл hello.txt, создадим файл приложения **app.js**:

```
const fs = require("fs");

let data = fs.readFileSync("./hello.txt", "utf8");
console.log(data);
```

Запуск **app.js**:



Работа с файлами. Чтение. Асинхронное

Для **чтения** данных из файла в **асинхронном** режиме используется метод **readFile()**, параметры:

1 – путь к файлу

2 – кодировка

3 – функция обратного вызова, которая выполняется после завершения чтения, параметры:

1 – информацию об ошибке, при ее наличии

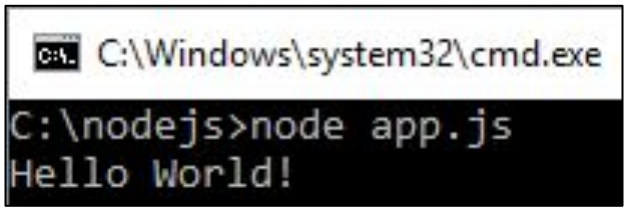
2 – считанные данные

Файл **app.js**:

```
const fs = require("fs");

fs.readFile("./hello.txt", "utf8", (error, data) => {
    if (error) {
        throw error;
    }
    console.log(data);
});
```

Запуск **app.js**:



Работа с файлами. Запись. Синхронная

Для **записи** данных в файл в **синхронном** режиме используется метод **writeFileSync()**, параметры:

1 – путь к файлу
2 – записываемые данные
3 – кодировка (необязательный, по умолчанию **utf8**)

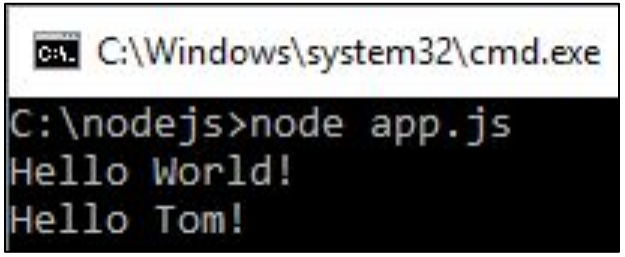
Обратите внимание, если файл для записи не существует, он будет создан

Файл **app.js**:

```
const fs = require("fs");

console.log(fs.readFileSync("./hello.txt", "utf8"));
fs.writeFileSync("./hello.txt", "Hello Tom!", "utf8");
console.log(fs.readFileSync("./hello.txt", "utf8"));
```

Запуск **app.js**:



Работа с файлами. Запись. Асинхронная

Для **записи** данных в файл в **асинхронном** режиме используется метод **writeFile()**, параметры:

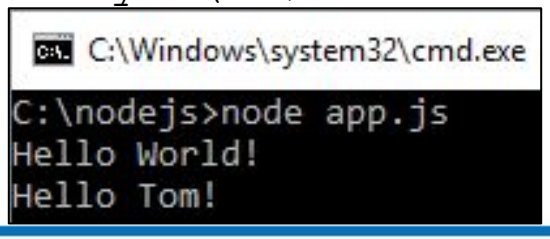
1 – путь к файлу
2 – записываемые данные
3 – кодировка (необязательный, по умолчанию **utf8**)
4 – функция обратного вызова, которая выполняется после завершения записи, параметр: информацию об ошибке, при ее наличии

Файл **app.js**:

```
const fs = require("fs");

console.log(fs.readFileSync("./hello.txt", "utf8"));
fs.writeFile("./hello.txt", "Hello Tom!", "utf8", (error) => {
    if (error) {
        throw error;
    }
    console.log(fs.readFileSync("./hello.txt", "utf8"));
});
```

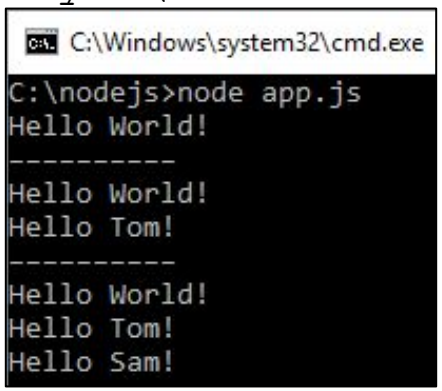
Запуск **app.js**:



Работа с файлами. Дозапись

Обратите внимание, методы записи **полностью перезаписывают** файл. Для **дозаписи** данных используются методы **appendFileSync()/appendFile()**, использование методов аналогично методам writeFileSync()/writeFile()

Файл **app.js**:

```
const fs = require("fs");
console.log(fs.readFileSync("./hello.txt", "utf8"));
console.log("----------");
fs.appendFileSync("./hello.txt", "\nHello Tom!", "utf8");
console.log(fs.readFileSync("./hello.txt", "utf8"));
console.log("----------");
fs.appendFile("./hello.txt", "\nHello Sam!", "utf8", (error) => {
    if (error) { throw error; }
    console.log(fs.readFileSync("./hello.txt", "utf8"));
});
```

Запуск **app.js**: