

TRABALHANDO COM ARQUIVOS.

No Node.js podemos importar módulos criados pessoalmente, módulos originais do Node.js ou módulos de uma terceira pessoa.

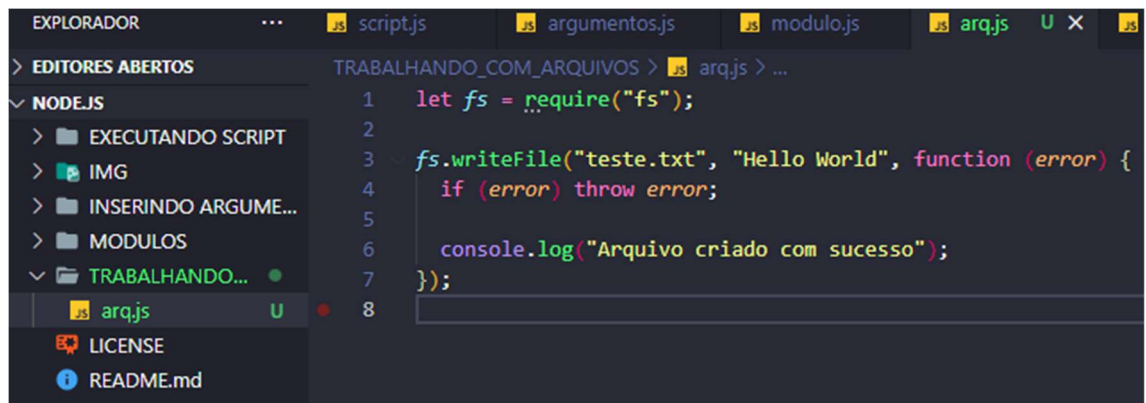
FS (file system) é um modulo padrão do Node.

CRIANDO UM ARQUIVO

Para criar um arquivo utiliza o módulo FS junto com a função **writeFile()**;

A função writeFile têm três argumentos:

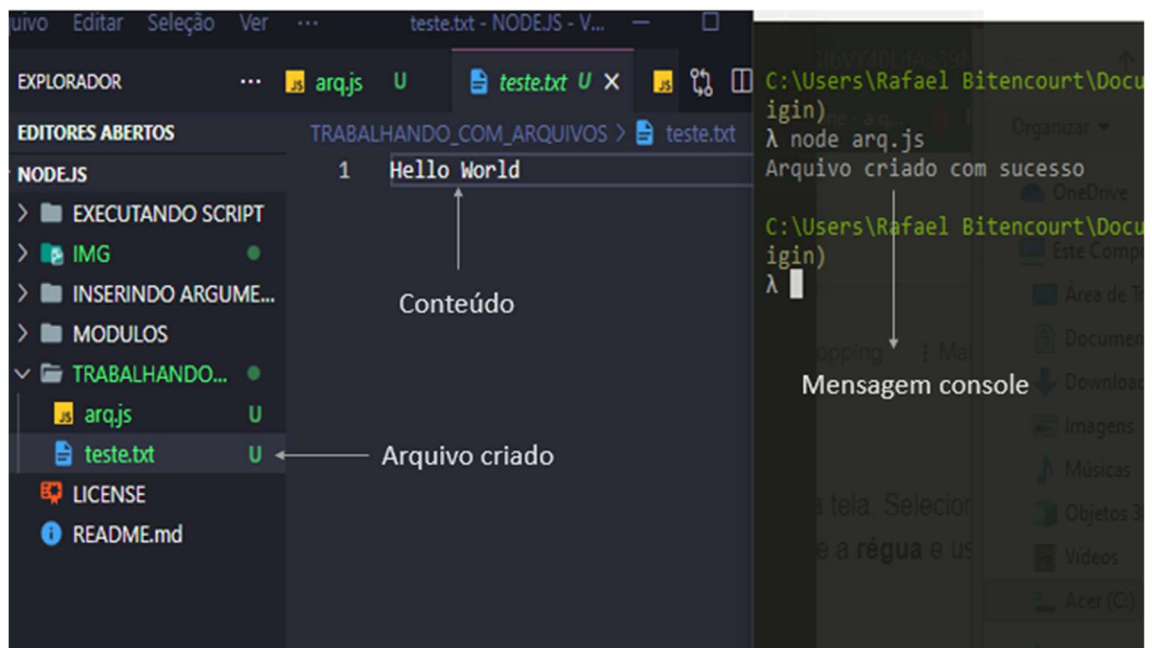
`fs.writeFile ('nome do arquivo', 'conteudo do arquivo', 'callback')`.



The screenshot shows the Visual Studio Code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project named 'TRABALHANDO_COM_ARQUIVOS' with a file 'arq.js' selected. The code editor shows the following JavaScript code:

```
1 let fs = require("fs");
2
3 fs.writeFile("teste.txt", "Hello World", function (error) {
4   if (error) throw error;
5
6   console.log("Arquivo criado com sucesso");
7 });
8
```

Resultado:



ALTERANDO UM ARQUIVO

`fs.appendFile` - O método `fs.appendFile()` é usado para anexar de forma assíncrona os dados fornecidos a um arquivo. Um novo arquivo é criado se ele não existir.

`fs.appendFile('nome do arquivo', 'conteúdo a acrescentar', 'callback')`.

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

/* fs.writeFile("teste.txt", "Hello World", function (error) {
  if (error) throw error;

  console.log("Arquivo criado com sucesso");
}); */

fs.appendFile("teste.txt", " - Olá Mundo", function (error) {
  if (error) throw error;

  console.log("Arquivo modificado com sucesso");
});
```

Resultado:



APAGANDO UM ARQUIVO

Utiliza-se o método `unlink()` com dois argumentos:

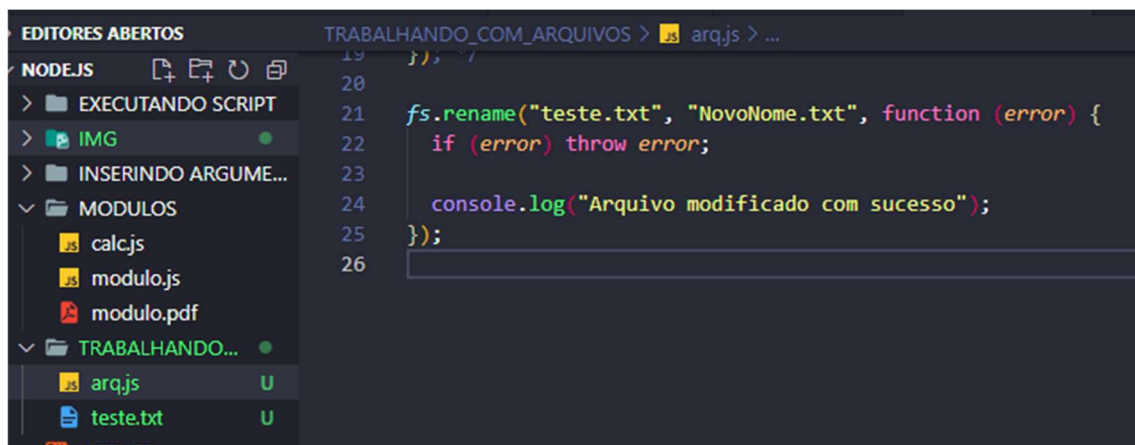
`fs.unlink('nome do arquivo', callback).`

```
fs.unlink("teste.txt", function (error) {  
  if (error) throw error;  
  
  console.log("Arquivo apagado com sucesso");  
});
```

RENOMEANDO UM ARQUIVO

Utiliza-se o método `rename()` com três argumentos:

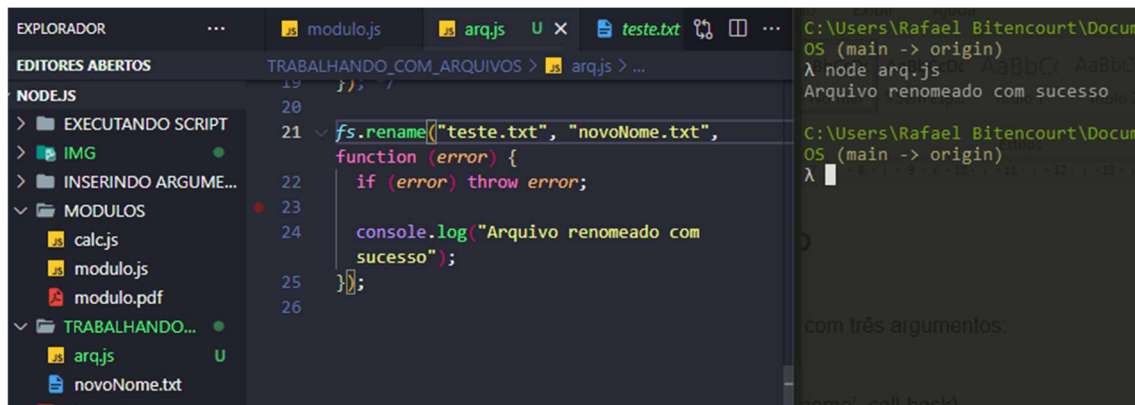
`fs.rename('nome do arquivo', 'novo nome', call-back).`



The screenshot shows the VS Code editor interface. On the left, the 'EXPLORADOR' sidebar shows a project named 'TRABALHANDO...' with files 'arq.js' and 'teste.txt'. The main editor area shows the 'arq.js' file with the following code:

```
19  },  
20  
21  fs.rename("teste.txt", "NovoNome.txt", function (error) {  
22    if (error) throw error;  
23  
24    console.log("Arquivo modificado com sucesso");  
25  });  
26
```

Resultado:



The screenshot shows the VS Code editor interface after the file rename operation. The 'EXPLORADOR' sidebar now shows the file 'novoNome.txt' instead of 'teste.txt'. The main editor area shows the 'arq.js' file with the following code:

```
21  fs.rename("teste.txt", "novoNome.txt",  
22    function (error) {  
23      if (error) throw error;  
24      console.log("Arquivo renomeado com  
25      sucesso");  
26    });
```

On the right side of the screenshot, there is a terminal window showing the command prompt output:

```
C:\Users\Rafael Bitencourt\Docum  
OS (main -> origin)  
λ node arq.js  
Arquivo renomeado com sucesso
```

LENDO ARQUIVO

Utiliza-se o método `readFile()`, é necessário especificar o tipo de texto e colocar nos argumentos os dados através do ***data***.

```
fs.readFile("novoNome.txt", "utf8", function (error, data) {  
  if (error) throw error;  
  
  console.log(data);  
});
```