

```
//MODULE.EXPORTS:
//Os módulos no Node são objetos onde temos um conjunto de chave e valor, em que os atributos de um módulo podem ser acessados por
qualquer aplicação que importe esses módulos. Para isso temos algumas formas de exportar esses atributos:

//EXPORTANDO COM THIS:
console.log('1', module.exports === this) //Perceba que o this referencia diretamente para o module.exports...
this.ola = 'Fala pessoal' //Veja que criamos um atributo dentro do modulo exports chamado "ola", embora possamos usar o "this" esse
método é pouco indicado...


//EXPORTANDO COM EXPORTS:
console.log('\n2', module.exports === exports) //Perceba que assim como o "this" o "exports" também faz referencia direta ao
module.exports...
exports.bemVindo = 'Bem vindo ao node' //E assim como o this podemos usar o exports para criar atributos diretamente no module
exports dando um nome ao atributo"...


//EXPORTANDO PELO MODULE.EXPORTS:
module.exports.ateLogo = 'Até logo' //E claro, podemos criar um objeto direto usando module.exports...


//VARIÁVEIS NÃO FICARÃO VISÍVEIS NO ARQUIVO EXTERNO:
let naoExports = 'Não fui exportado' //Veja que valores que não foram exportados não ficaram visíveis em outro arquivo...
console.log('\n3', naoExports)


//MODO MAIS USADO DE EXPORTAR ATRIBUTOS:
//****DESCOMENTE AQUI!!!
```

```
// module.exports = { //O "module.exports" com a atribuição direta é o módulo mais comum a ser criado, onde geramos chaves
// diretamente dentro do objeto. Nesse método de sobrescrita de todo o módulo temos que usar o nome completo "module.exports" se
// tentarmos usar "this" ou "exports" o nosso arquivo vai gerar erro ou undefined...
//     bomDia: 'Bom dia',
//     boaNoite() {
//         return 'Boa noite'
//     }
// }

//FORMAS ERRADAS DE GERAR ATRIBUTOS:
//****DESCOMENTE AQUI!!!
// this = { //Perceba que embora o this faça referência ao module.exports não podemos gerar um atributo dentro dele diretamente,
// temos que usar a notação de objeto...
//     oi: 'Olá tudo bem'
// } //Perceba que this também dá erro...

exports = { //O mesmo serve para o exports, não podemos gerar um objeto nele diretamente, assim como o this ele é apenas um método de
// acesso para o module.
//     comoVai: 'Como vai?'
// }
console.log('\n4)', module.exports) //Veja que nenhum valor foi incluso...

//PARA VER COMO OS MÓDULOS SÃO IMPORTADOS CONSULTE A AULA 134-module.exports_Exportando_Modulos-Cliente.js...
```

ARQUIVO CLIENTE QUE ESTÁ IMPORTANDO...

```
//IMPORTAÇÃO DE MÓDULOS:
//Os módulos são importados por através do método "require()" do Node. Podemos atribuir todos os módulos exportados para uma variável
// a nossa escolha.
//OBS: Quaisquer variáveis e funções criadas dentro dos módulos que não estiverem usando "module.exports" não poderão ser
// importadas...
```

```
const modulo = require('./134-module.exports__Exportando_Modulos') //0 "./" indica o caminho relativo para que a nossa aplicação
encontre o módulo, sempre que o nosso módulo estiver dentro do mesmo servidor, temos que percorrer todo o caminho á partir do "./"...

console.log(modulo.ola) //Perceba que para chamar os valores basta usarmos a notação de objeto que o valor virá naturalmente...
console.log(modulo.bemVindo)
console.log(modulo.atéLogo)
console.log(modulo) //Veja como muda quando o module.exports direto é descomentado, o module do arquivo é sobrescrito, e os atributos
passados anteriormente são descontinuados, gerando valores undefined. Além disso, veja que a variável criada no módulo não pôde ser
exportada por não usar "module.exports"...
console.log(modulo.boaNoite())
```

RESULTADO NO CONSOLE COMENTADO...

```
[Running] node "c:\Users\Almoxarifado\Documents\javascript\arquivos_das_aulas\134-module.exports__Exportando_Modulos-Cliente.js"
1) true

2 true

3) Não fui exportado

4) {
  ola: 'Fala pessoal',
  bemVindo: 'Bem vindo ao node',
  atéLogo: 'Até logo'
}
Fala pessoal
Bem vindo ao node
Até logo
{
  ola: 'Fala pessoal',
  bemVindo: 'Bem vindo ao node',
  atéLogo: 'Até logo'
}
c:\Users\Almoxarifado\Documents\javascript\arquivos_das_aulas\134-module.exports__Exportando_Modulos-Cliente.js:11
console.log(modulo.boaNoite())
      ^
```

```
TypeError: modulo.boaNoite is not a function
    at Object.<anonymous> (c:\Users\Almoxarifado\Documents\javascript\arquivos_das_aulas\134-module.exports__Exportando_Modulos-Cliente.js:11:20)
    at Module._compile (internal/modules/cjs/loader.js:1072:14)
    at Object.Module._extensions..js (internal/modules/cjs/loader.js:1101:10)
    at Module.load (internal/modules/cjs/loader.js:937:32)
    at Function.Module._load (internal/modules/cjs/loader.js:778:12)
    at Function.executeUserEntryPoint [as runMain] (internal/modules/run_main.js:76:12)
    at internal/main/run_main_module.js:17:47

[Done] exited with code=1 in 0.119 seconds
```

RESULTADO NO CONSOLE COM MODULE.EXPORTS DESCOMENTADO...

```
undefined
undefined
undefined
{ bomDia: 'Bom dia', boaNoite: [Function: boaNoite] }
Boa noite
```

RESULTADO NO CONSOLE COM THIS DESCOMENTADO...

```
[Running] node "c:\Users\Almoxarifado\Documents\javascript\arquivos_das_aulas\134-module.exports__Exportando_Modulos.js"
c:\Users\Almoxarifado\Documents\javascript\arquivos_das_aulas\134-module.exports__Exportando_Modulos.js:45
this = { //Perceba que embora o this faça referência ao module.exports não podemos gerar um atributo dentro dele diretamente, temos
que usar a notação de objeto...
^^^^

SyntaxError: Invalid left-hand side in assignment
    at wrapSafe (internal/modules/cjs/loader.js:988:16)
    at Module._compile (internal/modules/cjs/loader.js:1036:27)
    at Object.Module._extensions..js (internal/modules/cjs/loader.js:1101:10)
    at Module.load (internal/modules/cjs/loader.js:937:32)
    at Function.Module._load (internal/modules/cjs/loader.js:778:12)
    at Function.executeUserEntryPoint [as runMain] (internal/modules/run_main.js:76:12)
```

```
at internal/main/run_main_module.js:17:47
```

```
[Done] exited with code=1 in 0.115 seconds
```