```
//INSTANCIALIZAÇÃO DO NODE:

//No Node, toda vez que criamos um novo objeto por através do "module.exports", nos deparamos com um determinado comportamento que é a criação e uma única instância para aquele objeto. Isso significa que, se nós tentarmos referenciar um determinado objeto existente dentro de um módulo, mesmo que criemos mais de uma variável chamadora para ele, vamos estar referenciando ao mesmo objeto, que será contido no mesmo espaço de memória.

//Isso acontece por causa do cache do Node, que armazena os valores em cache para cada instância.

//Esse é um assunto que devemos tomar cuidado, pois quando o assunto é instancialização, qualquer alteração nos valores de memória pode causar um resultado inesperado. Vejamos um exemplo logo abaixo...

//EXEMPLOS DE VALORES ARMAZENADOS EM CACHE:
module.exports = { //Perceba que criamos 2 objetos, onde um possuí o valor 1, e o outro é uma função que incrementa esse valor, caso a função de incremento seja chamada, o atributo "valor" será incrementado e vai mudar, passando de 1 para 2 e assim por diante quantas vezes ele for chamado durante a execução do programa, pois seus valores serão armazenados em cache...
    valor: 1,
    inc() {
        this.valor++
    }
}


//Para ver o incremento da instancialização na prática acesse a aula 137-Instanciacao_Client.js
```

```
//INSTANCIAÇÃO FACTORY:

//Podemos resolver o problema da instancia única do Node por gerar objetos dentro de uma função factoty, lembrando que funções factory criam novos objetos para cada execução, fazendo com que um atributo seja chamado á partir de um único valor de instancia, invés de referenciar ao mesmo valor...

//CRIAÇÃO DE FUNÇÃO FACTORY DENTRO DE UM MODULE.EXPORTS:
module.exports = () => { //Perceba que dentro do module.exports temos uma factory que irá criar um objeto novo para cada variável que chamar o atributo valor, invés de pegar os valores de uma instância que já existe...
    return {
        valor: 1,
```

```
        inc() {
            this.valor++
        }
    }
}

//Para ver como que a instanciação única por através da factory funciona acesse a aula "137-Instanciacao_Client.js"
```

```
//CHAMANDO OS ATRIBUTOS NODE EM CACHE:
//Perceba que estamos instanciando o mesmo atributo em variáveis diferentes...
const contadorNodeComum1 = require('./137-Instanciacao_Server_Comum')
const contadorNodeComum2 = require('./137-Instanciacao_Server_Comum')
contadorNodeComum1.inc()
contadorNodeComum1.inc()
//Veja que mesmo incrementando os valores da variável 1, o valor da variável 2 também foram incrementados...

console.log('Veja que os 2 valores foram incrementados:', contadorNodeComum1.valor, contadorNodeComum2.valor)



//CHAMANDO OS VALORES DO MÓDULO INSTANCIADO POR ATRAVÉS DE UMA FACTORY...
const contadorFactory1 = require('./137-Instanciacao_Server_Factory')() //Detalhe importante, quando usamos o factory é que temos que chamar a função no momento da chamada do require adicionando os () de chamada da função factory...
const contadorFactory2 = require('./137-Instanciacao_Server_Factory')()
contadorFactory1.inc()
contadorFactory1.inc()

console.log('Veja que só o valor da variável 1 foi incrementado:', contadorFactory1.valor, contadorFactory2.valor)
```

RESULTADO NO CONSOLE...

```
[Running] node "c:\Users\Almoxarifado\Documents\javascript\arquivos_das_aulas\137-Instanciacao_do_Node\137-Instanciacao_Client.js"
Veja que os 2 valores foram incrementados: 3 3
Veja que só o valor da variável 1 foi incrementado: 3 1

[Done] exited with code=0 in 0.12 seconds
```