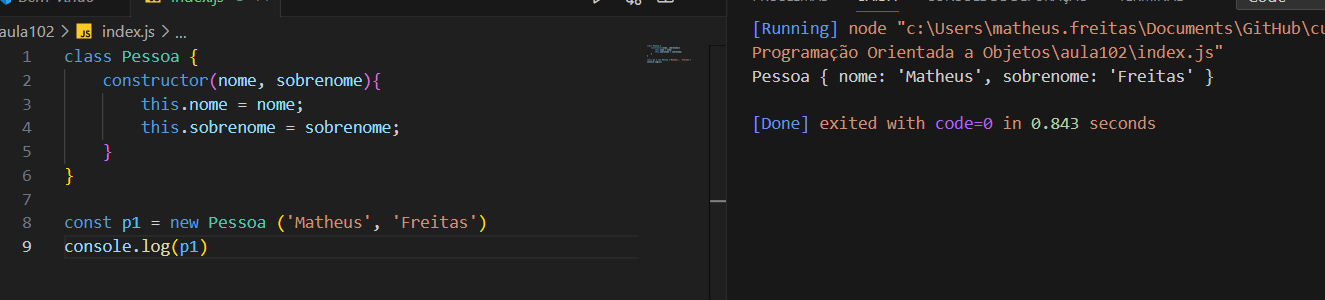
**AULA 102**

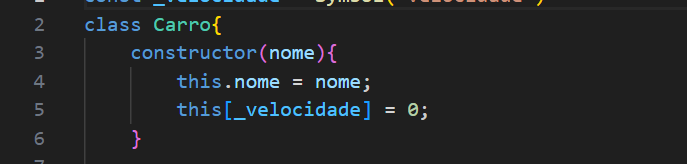
* Aula sobre classes
* É apenas uma outra forma de criar uma função construtora, mas na prática, funcionam de forma semelhante (Como um molde para criar objetos)
* Para criarmos uma classe, utilizamos o comando “class”
* Sempre que criamos uma classe, automaticamente já devemos criar o método “constructor”
* Com o método “constructor”, conseguimos passar parâmetros para a classe
* Exemplo de “Classe”



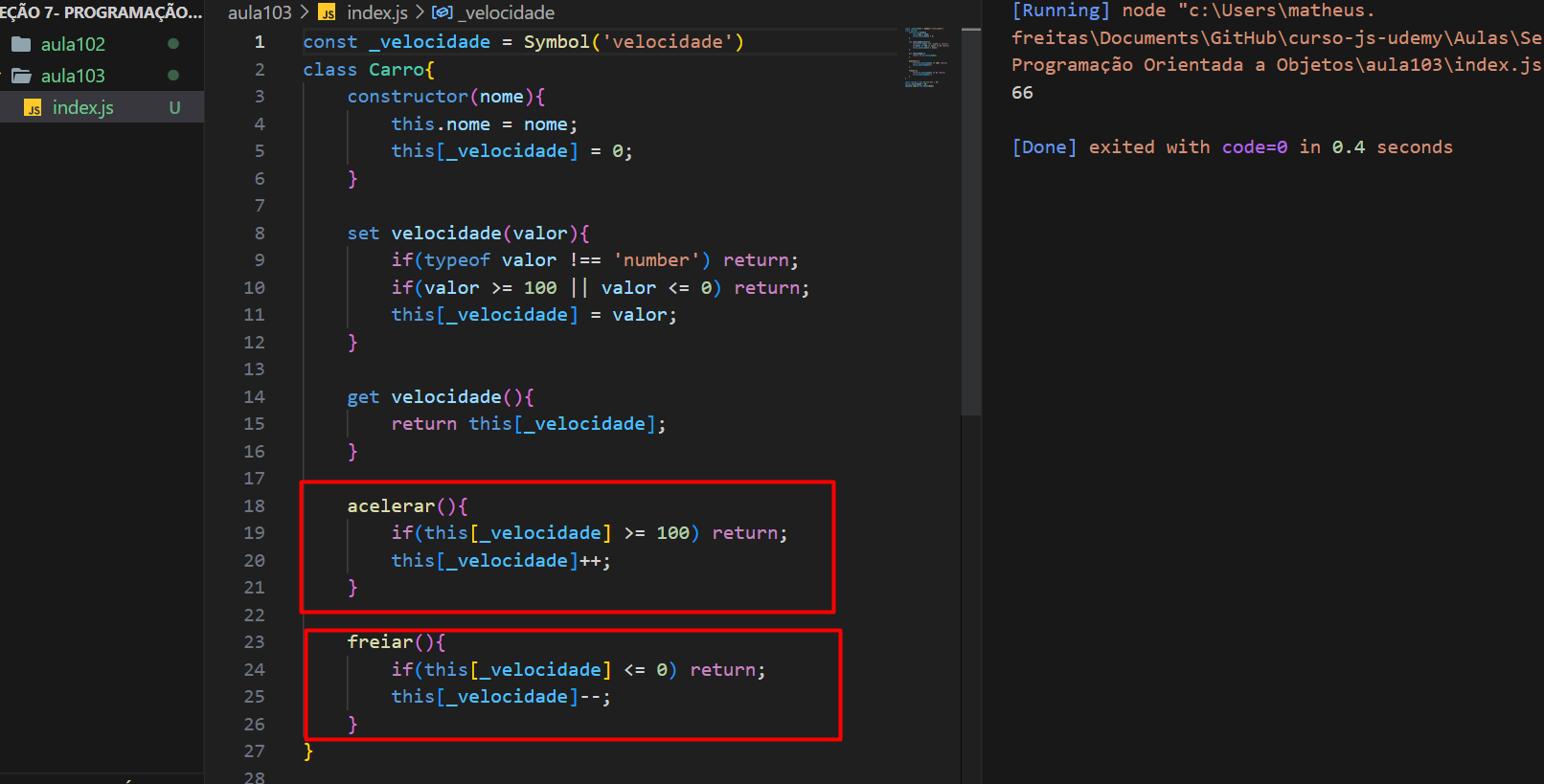
* Quando criamos algum método dentro de uma classe, ele automaticamente é inserido dentro do seu prototype, que ajuda consumindo menos recursos dos computadores

**AULA 103**

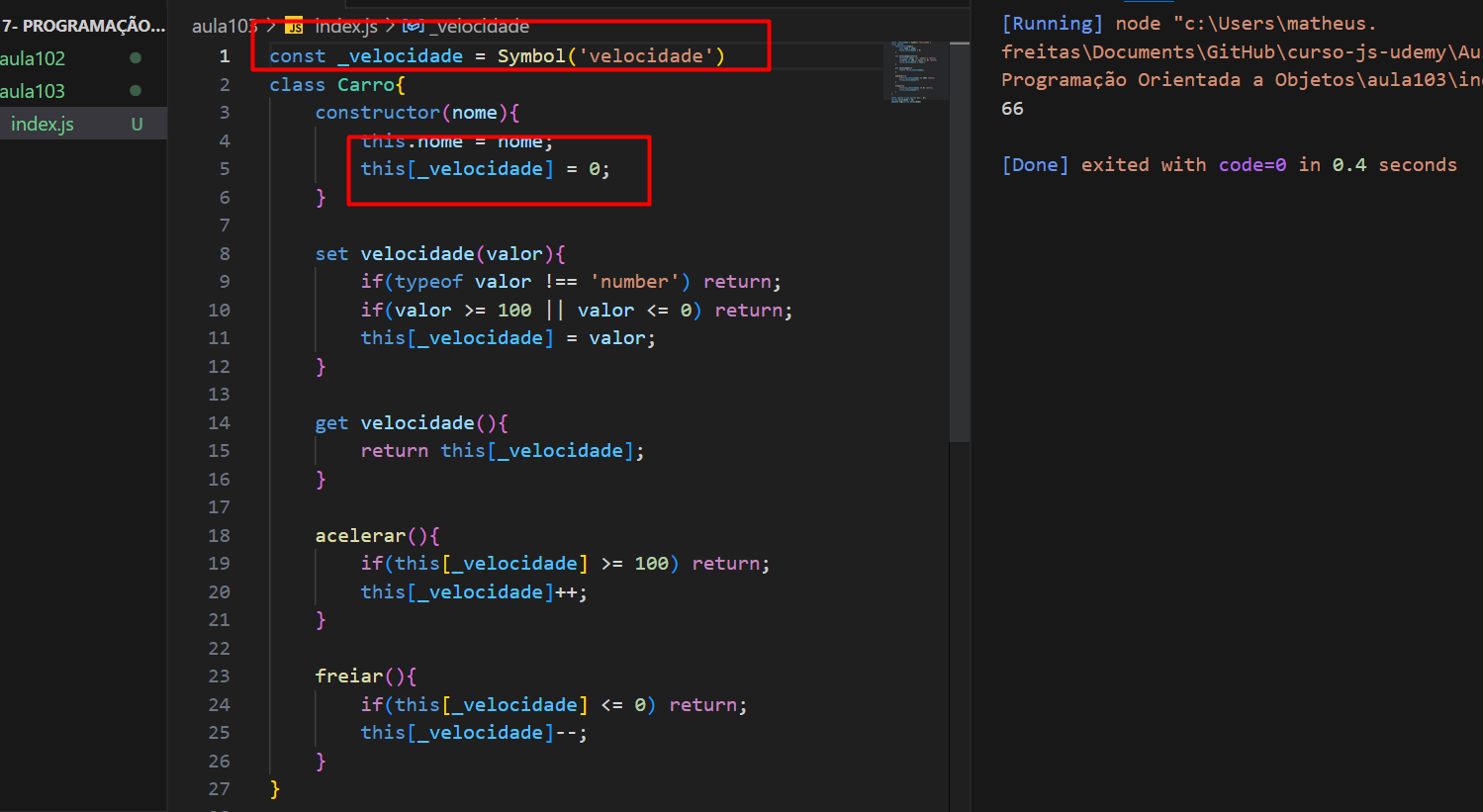
* Aula sobre getter e setter
* Criamos uma classe que representa um carro, com os atributos nome e velocidade



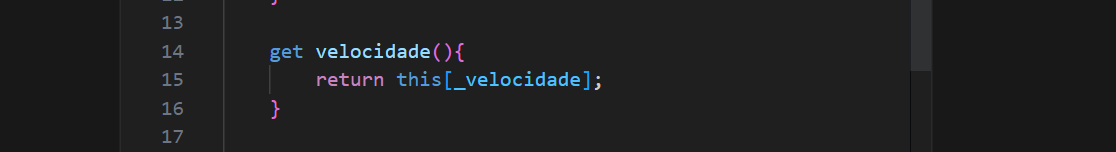
* Criamos dois métodos que modificam a velocidade do carro, “acelerar()” e “freiar()”



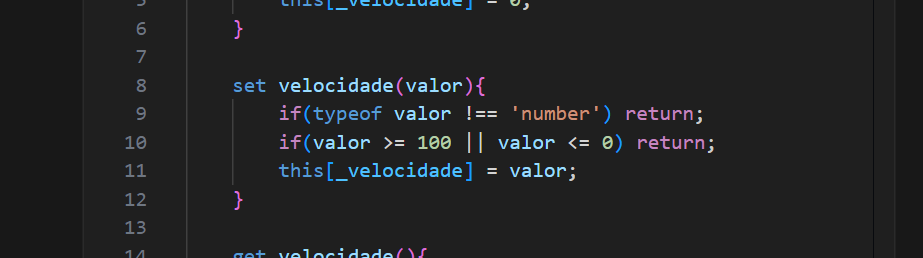
* Utilizamos o tipo de dado Symbol para que não seja possível alterar o valor do atributo velocidade fora da classe



* Criamos um getter “velocidade” que retorna o valor do Symbol velocidade, ligando assim os valores dos elementos, desta forma, conseguimos manipular o valor do Symbol

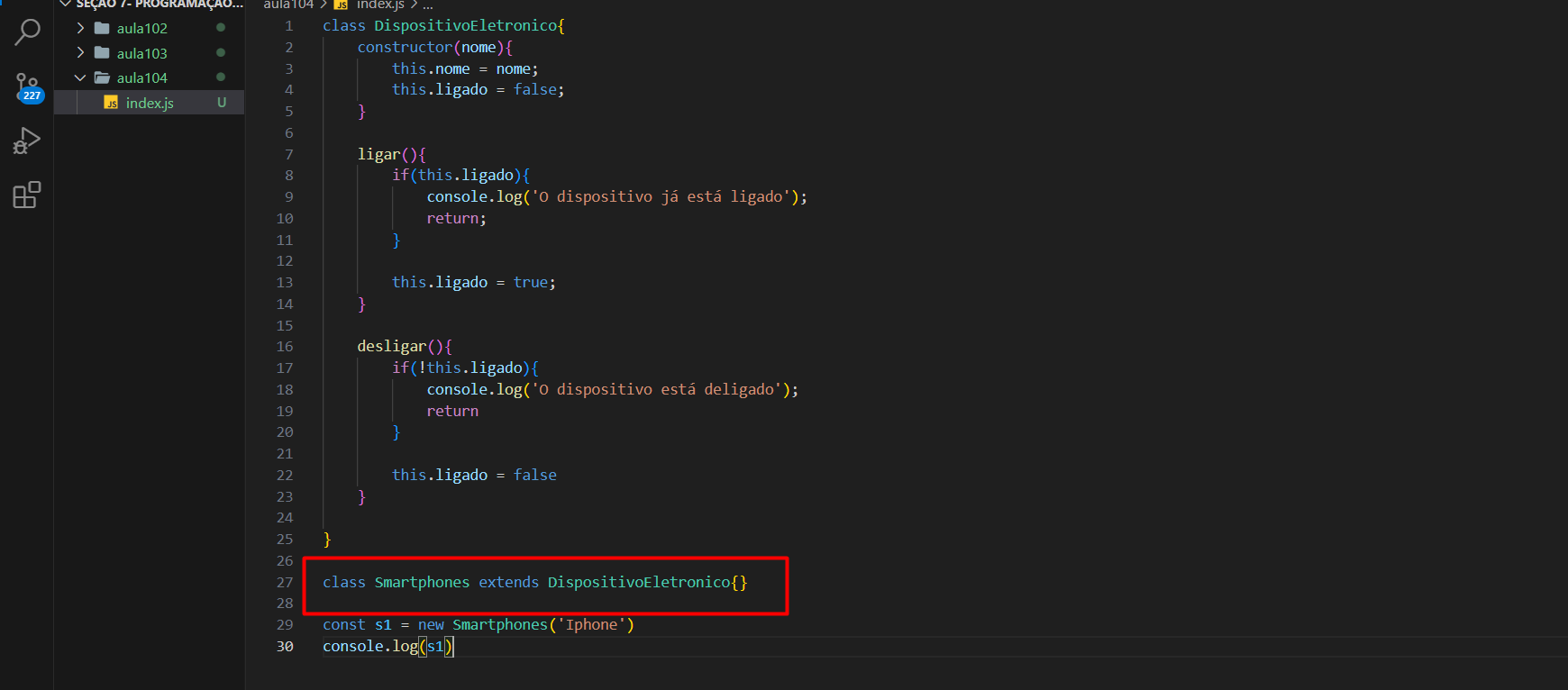


* Criamos um setter que verifica se o valor que será inserido é maior ou igual a 100 ou menor ou igual a 0, também verifica se o dado inserido é um number, se passar nessas validações, o valor pode ser atributo ao valor Symbol

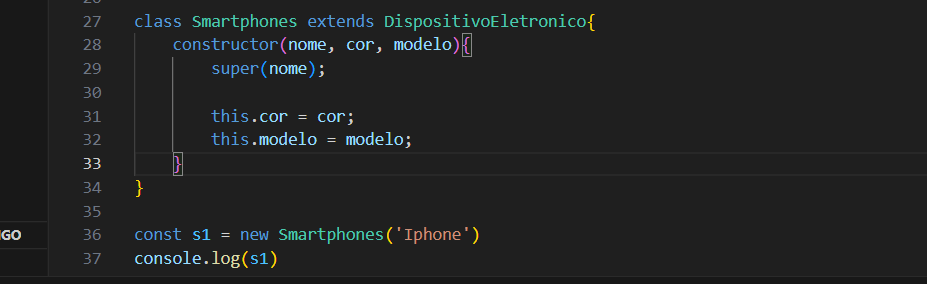


**AULA 104**

* Herança com classes
* Para que uma classe herde elementos de outra classe, utilizamos o comando extends



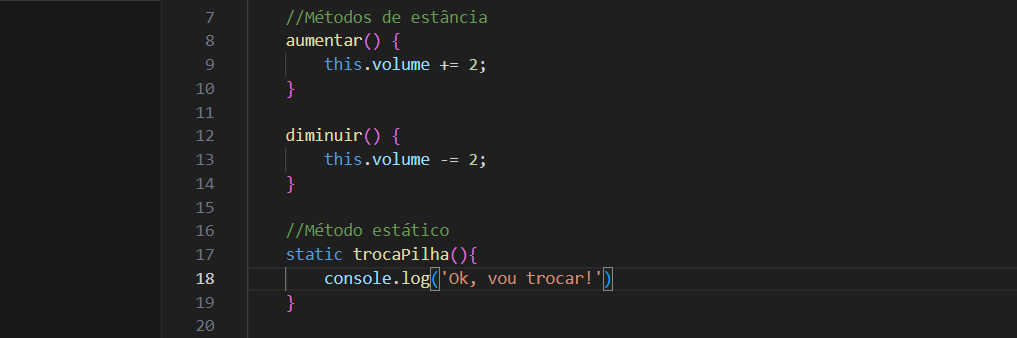
* Quando utilizamos apenas o comando extends, ele vai atribuir todos os atributos do construtor da classe pai
* Para atribuir alguns atributos específicos do elemento pai, utilizamos o comando Super()
* Para utilizar o comando super, primeiro utilize o comando extends para puxar todos os atribuitos, crie um constructor na chave filha e dentro do construtor, utilize o comando Super() e dentro dos parênteses, insira o parâmetro que você deseja utilizar da classe pai



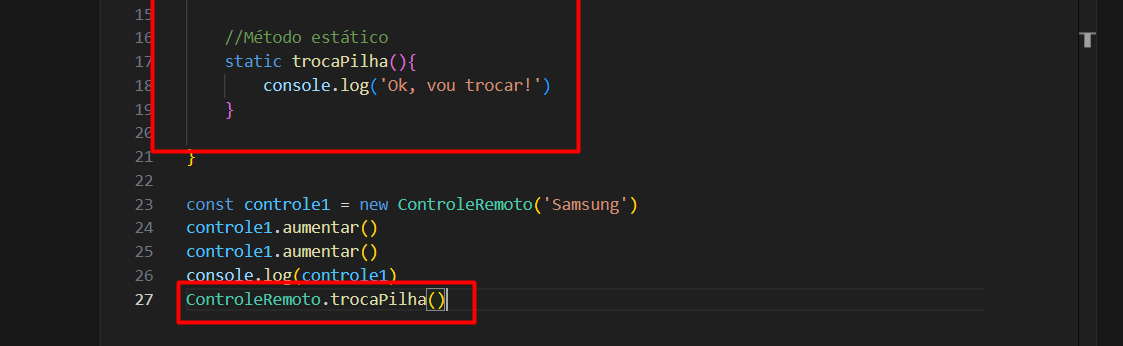
* Podemos atribuir os elementos de uma classe filho para outra classe
* Não é recomendável utilizar uma hierarquia muito grande, recomendado uma ou duas hierarquias, no máximo 3
* As classes “pais” não recebem heranças de suas classes filhas

**AULA 105**

* Métodos de instancia e estáticos
* Métodos estáticos são métodos que podemos acessar na classe sem instanciar ela
* Métodos estáticos não ficam disponíveis dentro da instância
* São como funções dentro de uma classe para uma organização
* Para criar um método estático, utilizamos o comando static



* Para acessar um método estático, não precisamos acionar a sua instância, podemos acionarmos direto pela classe



* Podem receber parâmetros para efetuar operações

**AULA 106**

* Utilizando as classes
* Exercício validador de CPF utilizando classes

**AULA 107**

* Exercício validação de CPF utilizando classes
* Criamos um formulário em HTML e CSS
* Criamos o link para o arquivo JavaScript
* Criamos uma classe chamada ValidaFormulario();
* Na classe, criamos as chaves formulário (que foi lincado ao formulário no HTML) e o método eventos, que irá capturar os eventos da página HTML
* Criamos um evento que será acionado ao clicar no botão “Enviar” do formulário
* Dentro do método eventos, criamos uma função de call-back que aciona o método que impende do formulário ser enviado ao clicarmos no botão “submit” (utilizando o comando preventDefault()
* Dentro do nosso método que para o envio do formulário, acionamos um outro método que irá fazer as validações dos campos
* Adicionamos uma classe para os inputs dos dados dos formulários, para capturar os valores
* Dentro do método de validação, acionamos um outro método que irá criar a mensagem de erro após o campo que não passar na validação (utilizamos um novo método para isso, insertAdjacentElemente()
* Criamos uma mensagem de erro após o campo que não passar na validação, criamos uma classe para personalizar a mensagem de erro
* Criamos um for que para de enviar a mensagem de erro quando ela já está apresentada e não adiciona quando o campo for preenchido