Aula JavaScript

***Getters e Setters***

No JavaScript moderno, temos duas formas de usar os ***getters*** e ***setters*** utilizando as ferramentas modernas oferecidas, ou seja, nas classes. O primeiro é usando a keyword ***set*** e ***get***, que já são bem intuitivas no que elas fazem, e a segunda forma é utilizando métodos.

***Por que usar ?***

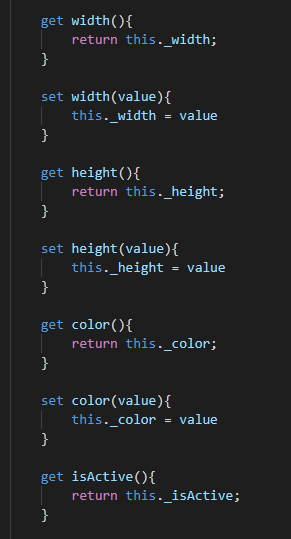
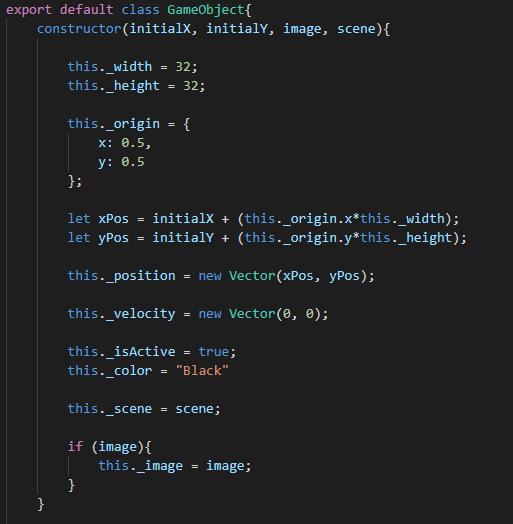
Diferente de outras linguagens, o JavaScript não possui variáveis privadas, devido a sua natureza extremamente dinâmica e flexível.

Por isso, muitos desenvolvedores convencionaram de que as variáveis que possuem o “**\_**” (underline) na frente não devem ser acessadas diretamente (mas ainda continuam acessíveis), isso foi feito para prevenir que outros colaboradores cometam algum erro.

Logo, é muito comum encontrarmos códigos em que as variáveis das classes possuem o “\_” (underline) na frente, e depois são atribuídos formas de acessar a variável.

***Usando as novas funcionalidades do ECMAScript 6***

Como citado anteriormente, podemos usar as Keywords ***get*** e ***set***, elas são escritas na frente do método, e o nome do método será a forma de acessa-las. Elas também nos provém uma certa segurança pois a keyword ***get*** apenas aceita um retorno, e a ***set***, apenas uma definição.



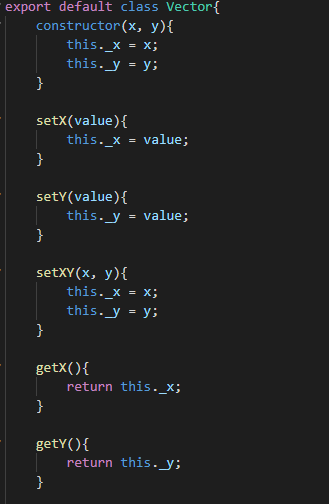
Como podemos observar, nesse exemplo temos uma classe chamada GameObject, que dentro dela são criadas variáveis com o “**\_**”(underline) na frente, indicando que essas variáveis são privadas, ou seja, não foram desenvolvidas/pensadas para serem acessadas por um objeto externo **diretamente**. **Entretanto**, como dito anteriormente, não existe variáveis privadas em JavaScript, então nada impede de acessá-las assim:



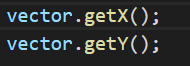
Mas, usando os ***get*** e ***setters***, podemos criar formas acessá-las usando um método como intermediário, que por convenção, colocamos igual ao nome da variável sem o underline. Logo, também podemos acessá-la assim:



Podemos também usar métodos para isso, não há nada de errado, alguns até preferem, seria algo do tipo:



Logo, criamos métodos que retornam ou definem os valores das variáveis, então para acessá-las/defini-las devemos usar como:



**Observação:** Já existe uma proposta de atualização para o JavaScript em adicionar o símbolo ***#*** na frente de variáveis para defini-las como privadas, navegadores mais modernos também aceitam, entretanto, muitos desenvolvedores não gostaram da proposta pelo fato do ***#*** poluir visualmente o código.