

# Atividade de Fixação

métodos

*Fabício Curvello*

# Projeto *ExFixacaoCarro*

Carro
- fabricante: String - modelo: String - ano: int - motorLigado: boolean - velocidade: int
+ detalhes(): void + situacaoMotor(): String + ligarMotor(): String + desligarMotor(): String + acelerar(valor: int): String + frear(valor: int): String

1 - Crie o projeto com o referido nome

2 - Crie a classe **Carro** no pacote **model**, inicialmente com:

- atributos
- construtor simples
- construtor completo
- métodos get e set

# Terceiro método construtor:

Carro
- fabricante: String - modelo: String - ano: int - motorLigado: boolean - velocidade: int
+ detalhes(): void + situacaoMotor(): String + ligarMotor(): String + desligarMotor(): String + acelerar(valor: int): String + frear(valor: int): String

Na Classe Carro, elabore um método construtor personalizado com as seguintes funcionalidades:

- Deve conseguir criar um objeto da Classe Carro recebendo apenas valores para os atributos *fabricante*, *modelo* e *ano*.
- Ele deve atribuir *false* ao atributo *motorLigado*;
- Ele deve atribuir 0 (zero) ao atributo *velocidade*.

# Classe Carro – Método *detalhes()*

Carro
- fabricante: String - modelo: String - ano: int - motorLigado: boolean - velocidade: int
+ detalhes(): void + situacaoMotor(): String + ligarMotor(): String + desligarMotor(): String + acelerar(valor: int): String + frear(valor: int): String

Implemente o método ***detalhes()***, que ao ser acionado deve exibir os valores inerentes ao referido objeto para os seguintes atributos:

- fabricante
- modelo
- ano

# Classe Carro – Método *situacaoMotor()*

Carro
<ul style="list-style-type: none"><li>- fabricante: String</li><li>- modelo: String</li><li>- ano: int</li><li>- motorLigado: boolean</li><li>- velocidade: int</li></ul>
<ul style="list-style-type: none"><li>+ detalhes(): void</li><li>+ situacaoMotor(): String</li><li>+ ligarMotor(): String</li><li>+ desligarMotor(): String</li><li>+ acelerar(valor: int): String</li><li>+ frear(valor: int): String</li></ul>

Implemente o método *situacaoMotor()*, que ao ser acionado deve verificar se o motor do carro está ligado, **retornando** uma das seguintes *Strings* conforme a situação do motor:

**<modelo do carro> --> Motor LIGADO.**

ou

**<modelo do carro> --> Motor DESLIGADO.**

# Classe Carro – Método *ligarMotor()*:

Carro
<ul style="list-style-type: none"><li>- fabricante: String</li><li>- modelo: String</li><li>- ano: int</li><li>- motorLigado: boolean</li><li>- velocidade: int</li></ul>
<ul style="list-style-type: none"><li>+ detalhes(): void</li><li>+ situacaoMotor(): String</li><li>+ ligarMotor(): String</li><li>+ desligarMotor(): String</li><li>+ acelerar(valor: int): String</li><li>+ frear(valor: int): String</li></ul>

Implemente o método ***ligarMotor()***, que ao ser acionado deve verificar se o motor do carro está ligado, **retornando** uma das seguintes *Strings* conforme a situação do motor:

***O motor do <modelo do carro> já estava ligado.***

ou

***O motor do <modelo do carro> foi ligado agora.***

**OBS:** Caso o motor esteja desligado, modificar o valor do atributo *motorLigado* para ***true***.

# Classe Carro – Método *desligarMotor()*:

Carro
<ul style="list-style-type: none"><li>- fabricante: String</li><li>- modelo: String</li><li>- ano: int</li><li>- motorLigado: boolean</li><li>- velocidade: int</li></ul>
<ul style="list-style-type: none"><li>+ detalhes(): void</li><li>+ situacaoMotor(): String</li><li>+ ligarMotor(): String</li><li>+ desligarMotor(): String</li><li>+ acelerar(valor: int): String</li><li>+ frear(valor: int): String</li></ul>

Implemente o método ***desligarMotor()***, que ao ser acionado deve verificar se o motor do carro está desligado, **retornando** uma das seguintes *Strings* conforme a situação do motor:

***O motor do <modelo do carro> foi desligado agora.***

ou

***ATENÇÃO: O motor do <modelo do carro> não foi desligado pois o carro está em movimento.***

ou

***O motor do <modelo do carro> já estava desligado.***

**OBS:** Caso o motor esteja ligado, modificar o valor do atributo *motorLigado* para ***false***.

# Classe Carro – Método *acelerar()*:

Carro
<ul style="list-style-type: none"><li>- fabricante: String</li><li>- modelo: String</li><li>- ano: int</li><li>- motorLigado: boolean</li><li>- velocidade: int</li></ul>
<ul style="list-style-type: none"><li>+ detalhes(): void</li><li>+ situacaoMotor(): String</li><li>+ ligarMotor(): String</li><li>+ desligarMotor(): String</li><li>+ acelerar(valor: int): String</li><li>+ frear(valor: int): String</li></ul>

Implemente o método ***acelerar()***, que deve ser acionado **recebendo** o valor a ser acrescentado na velocidade do carro:

1 – Se o motor do carro estiver **ligado**:

1.1 – Aumentar a velocidade do carro de acordo com o valor recebido

1.2 – **Retornar** a *String*:

**<modelo do carro> --> ACELEROU!!! Velocidade atual: <velocidade> Km/h.**

2 – Se o motor do carro **não** estiver ligado:

2.1 – Retornar a *String*:

**<modelo do carro> --> ATENÇÃO: Não é possível acelerar com o motor desligado!**



# Classe Carro – Método *frear( )*:

## Carro

- fabricante: String  
- modelo: String  
- ano: int  
- motorLigado: boolean  
- velocidade: int

+ detalhes(): void  
+ situacaoMotor(): String  
+ ligarMotor(): String  
+ desligarMotor(): String  
+ acelerar(valor: int): String  
+ frear(valor: int): String

Implemente o método *frear( )*, que deve ser acionado recebendo o valor a ser reduzido na velocidade do carro:

1 – Se o motor do carro estiver **desligado**, retornar a *String*:

<modelo do carro> --> **ATENÇÃO: O motor está desligado! O carro já está parado!**

2 – Se o motor do carro estiver **ligado** e a velocidade for **0 (zero)**, retonar a *String*:

<modelo do carro> --> **ATENÇÃO: O carro já está parado!**

3 – Se o motor do carro estiver ligado e a velocidade for diferente de zero:

3.1 – Reduzir da velocidade o valor recebido no método, sendo que:

3.1.1 – Se este resultado der negativo ou 0 (zero), atribuir 0 (zero) à velocidade do carro e **retornar** a *String*:

<modelo do carro> --> **FREOU!!! O carro está PARADO!**

3.1.2 – Se este resultado der positivo, atribuir este valor ao atributo velocidade e **retornar** a *String*:

<modelo do carro> --> **FREOU!!! Velocidade atual: <velocidade> km/h**