



Instituto Tecnológico de Aeronáutica

CES-22 - Programação Orientada a Objetos

Alunos: Rafael Studart Mattos Di Piero

Matéria: CES-22

Professor(a): Edgar Yano

Laboratório 4 - Documentação

1.) Questão 1

1.1) Modelagem do Problema

Ao analisar o enunciado do problema, pode-se desenvolver a modelagem mostrada no diagrama de classes, apresentado na **Figura 1** abaixo:

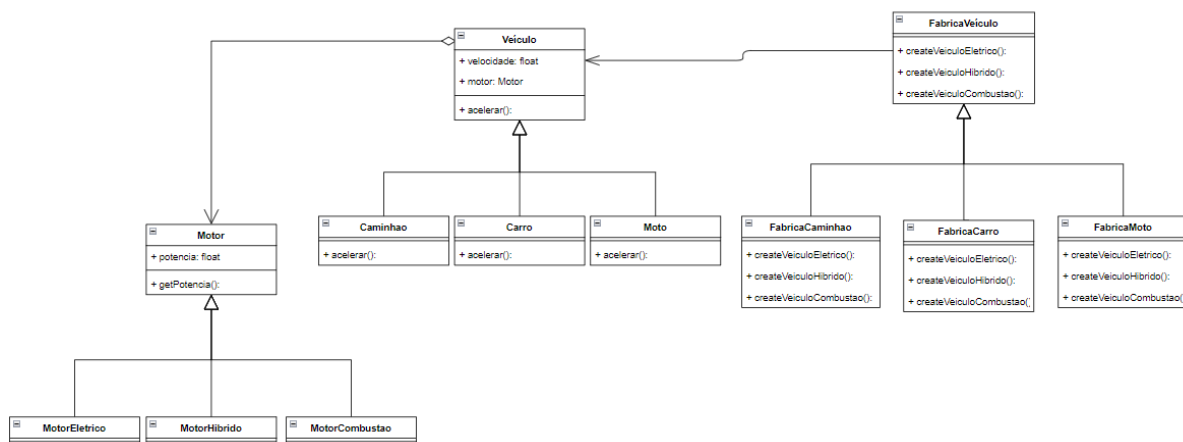


Figura 1: Diagrama de Classes com Modelagem do Problema

1.2) Execução do Código

Ao implementar a arquitetura mostrada, deve-se realizar testes para validar, chamando as fábricas, seus métodos de criação e a instanciação de objetos. As **Figuras 2,3 e 4** mostram o funcionamento do código:

```
CamEle = fabCaminhao.create_veiculo_eletrico()
CamEle.acelerar()
CamEle.get_info()
print('')
CamHib = fabCaminhao.create_veiculo_hibrido()
CamHib.acelerar()
CamHib.get_info()
print('')
CamCom = fabCaminhao.create_veiculo_combustao()
CamCom.acelerar()
CamCom.get_info()
print('\n')
```

main ×

C:\Users\paulo\AppData\Local\Programs\Python\Pyt

Sou um Caminhão, com velocidade: 10.5
Tenho um motor: MotorEletrico

Sou um Caminhão, com velocidade: 10.75
Tenho um motor: MotorHibrido

Sou um Caminhão, com velocidade: 11.0
Tenho um motor: MotorCombustao

Figura 2: Teste da Fábrica de Caminhões

```
CarEle = fabCarro.create_veiculo_eletrico()
CarEle.acelerar()
CarEle.get_info()
print('')
CarHib = fabCarro.create_veiculo_hibrido()
CarHib.acelerar()
CarHib.get_info()
print('')
CarCom = fabCarro.create_veiculo_combustao()
CarCom.acelerar()
CarCom.get_info()
print('\n')
```

main ×

Sou um Carro, com velocidade: 11

Tenho um motor: MotorEletrico

Sou um Carro, com velocidade: 11.5

Tenho um motor: MotorHibrido

Sou um Carro, com velocidade: 12

Tenho um motor: MotorCombustao

Figura 3: Teste da Fábrica de Carro

```
MotEle = fabMoto.create_veiculo_eletrico()
MotEle.acelerar()
MotEle.get_info()
print('')
MotHib = fabMoto.create_veiculo_hibrido()
MotHib.acelerar()
MotHib.get_info()
print('')
MotCom = fabMoto.create_veiculo_combustao()
MotCom.acelerar()
MotCom.get_info()
```

main x

```
Sou uma Moto, com velocidade: 12
Tenho um motor: MotorEletrico

Sou uma Moto, com velocidade: 13.0
Tenho um motor: MotorHibrido

Sou uma Moto, com velocidade: 14
Tenho um motor: MotorCombustao
```

Figura 4: Teste da Fábrica de Moto

2.) Questão 2

2.1) Modelagem do Problema

Ao analisar o enunciado do problema e o exemplo fornecido, pode-se desenvolver a modelagem mostrada no diagrama de classes, apresentado na **Figura 5** abaixo:

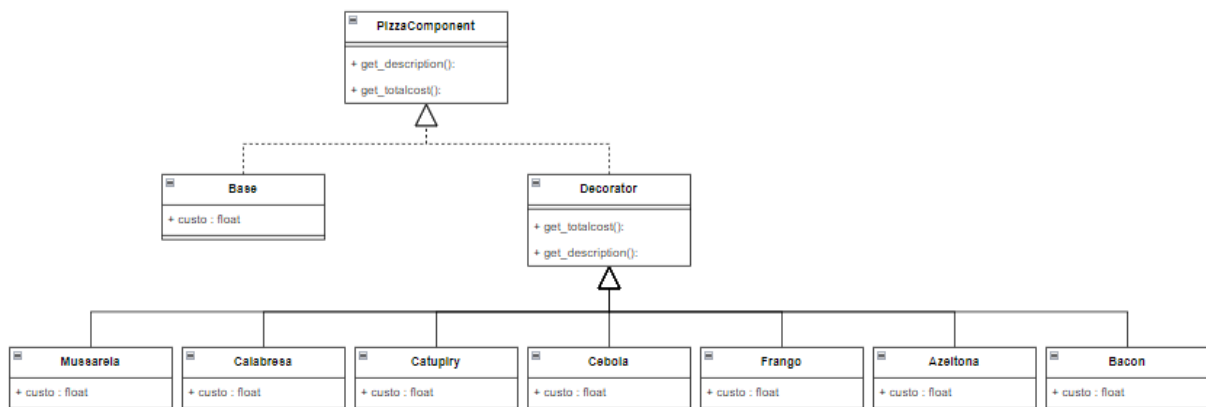


Figura 5: Diagrama de Classes com Modelagem do Problema

2.2) Execução do Código

Ao implementar a arquitetura mostrada, deve-se realizar testes para validar, se inspirando no exemplo mostrado para desenvolver os testes. Os testes e seus resultados estão mostrados na **Figura 6** abaixo:

```
calabresa = Calabresa(Cebola(Base()))
print(calabresa.get_description() + ":$" + str(calabresa.get_totalcost()))

toscana = Calabresa(Cebola(Mussarela(Base()))))
print(toscana.get_description() + ":$" + str(toscana.get_totalcost()))

calabacon = Calabresa(Cebola(Catupiry(Bacon(Base()))))
print(calabacon.get_description() + ":$" + str(calabacon.get_totalcost()))

frangocap = Frango(Cebola(Catupiry(Azeitona(Base()))))
print(frangocap.get_description() + ":$" + str(frangocap.get_totalcost()))

italiana = Calabresa(Cebola(Mussarela(Azeitona(Base()))))
print(italiana.get_description() + ":$" + str(italiana.get_totalcost()))
```

```
main x
C:\Users\paulo\AppData\Local\Programs\Python\Python39\python.exe C:/Users/pa
Base Cebola Calabresa:$7.0
Base Mussarela Cebola Calabresa:$7.0
Base Bacon Catupiry Cebola Calabresa:$12.0
Base Azeitona Catupiry Cebola Frango:$11.0
Base Azeitona Mussarela Cebola Calabresa:$9.0
```

Figura 6: Teste Implementação Decorators