blue

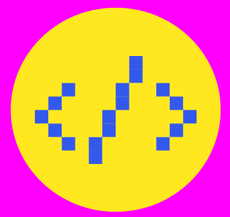
Introdução a Orientação a Objetos

Prof. Gustavo Molina

< thefutureisblue.me />

Orientação a objetos

Classes



Representação de uma classe criada em Python.

```python
class Pessoa:
    def __init__(self, nome, idade):
        self.nome = nome
        self.idade = idade

    def setNome(self, nome):
        self.nome = nome

    def setIdade(self, idade):
        self.idade = idade

    def getNome(self):
        return self.nome

    def getIdade(self):
        return self.idade
```

Definindo uma Classe

Classe

É um tipo que possui uma representação. Contém atributos e métodos.
É o nosso “molde”

Objeto

É a instância de uma classe e a concretização real e funcional de suas funcionalidades.

Definindo uma Classe

Atributos

São as características do nosso objeto.

Métodos

São as ações que os nossos objetos podem executar.

Classe

Classe: Herói.

Objeto: Capitão América, pois, ele é um herói.

Atributos: Nome, idade, peso.

Métodos: Engordar.

Classe: Pessoa.

Objeto: Steve Rogers.

Atributos: Nome, altura, peso.

Métodos: Emagrecer.

Classe

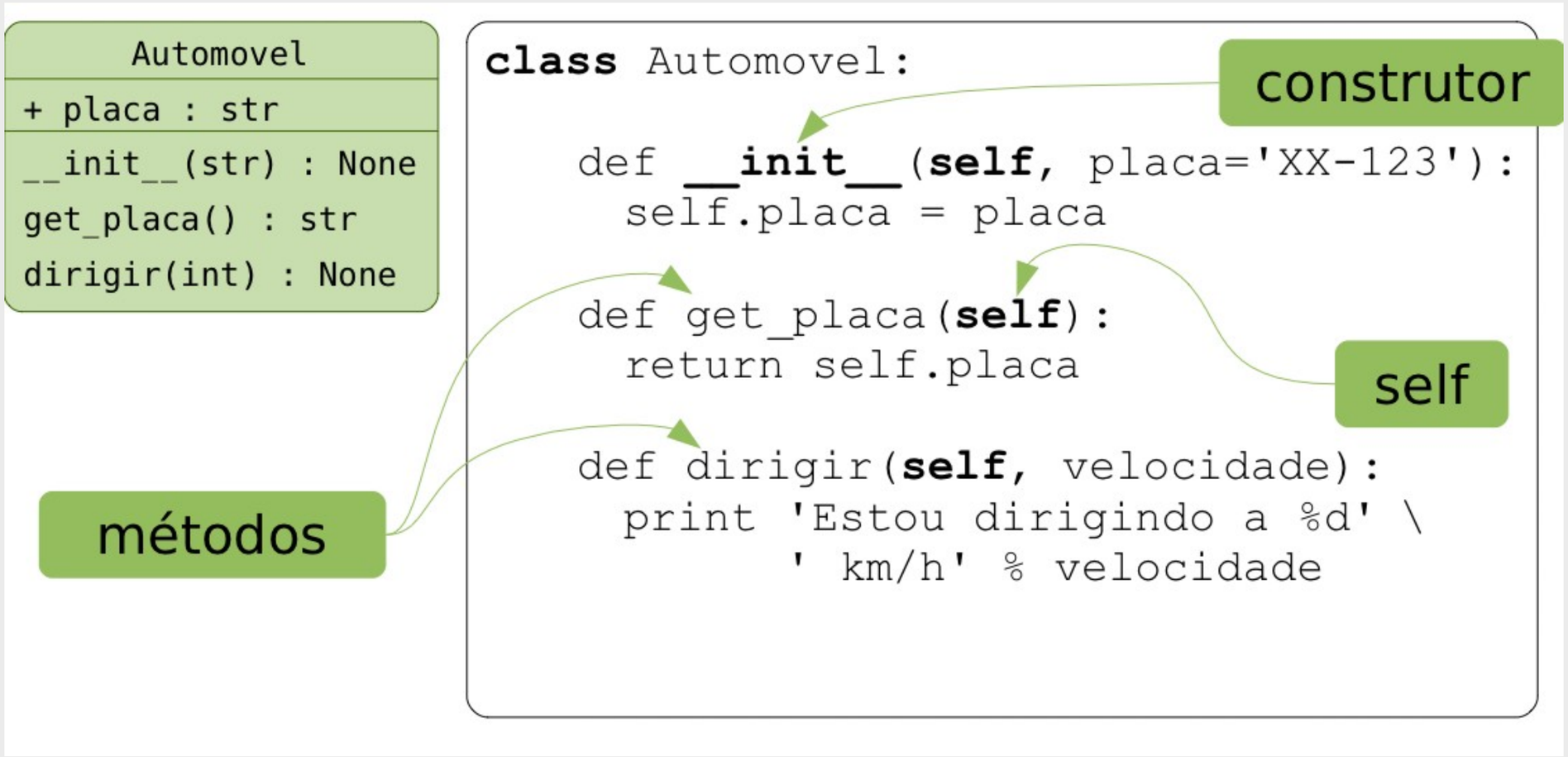
```python
class Herói():
    def __init__(self, nome, idade,peso):
        self.nome = nome
        self.idade = idade
        self.peso = peso

    def engordar(self, peso):
        self.peso += peso

a = Herói("Capitão América", 30, 85)
print(vars(a))
a.engordar(10)
print(vars(a))
```

```python
class Pessoa():
    def __init__(self, nome, altura,peso):
        self.nome = nome
        self.altura = altura
        self.peso = peso

    def emagrecer(self, peso):
        self.peso -= peso

a = Pessoa ("Steve Rogers", 1.80,90)
print(vars(a))
a.emagrecer(10)
print(vars(a))
```

Métodos em Classes

- Para definir um método em uma classe, basta incluir a definição da função seguindo o escopo do bloco da classe. Observem **o método engordar e o método emagrecer** que criamos.
- Em todos os métodos associados a instâncias, definidos dentro das classes, devemos ter o argumento **self** definido como **1º argumento.**
- Há geralmente um método especial *__init__* definido na maioria das classes.

Definição de uma Classe



blue

Aplicação de conteúdo!

Python na veia!

the future is blue

Exercícios de Fixação

1.a) Crie uma classe pessoa com os seguintes atributos: nome, sobrenome e idade.

1.b) Acrescente a classe criada um método para mostrar os dados de uma pessoa.

1.c) Crie um objeto do tipo pessoa para testar suas propriedades e métodos.

Exercícios de Fixação

2) Crie uma classe chamada Conta para simular as operações de uma conta corrente. Sua classe deverá ter os atributos Titular e Saldo, e os métodos Sacar e Depositar. Crie um objeto da classe Conta e teste os atributos e métodos implementados.

Resposta Exercício 1

Aula16_PessoaOO.py > ...

```python
class Pessoa:

    def __init__(self,nome,sobrenome,idade):
        self.nome=nome
        self.sobrenome=sobrenome
        self.idade=idade

    def mostrar_dados(self):
        print(f'Nome completo: {self.nome} {self.sobrenome}')
        print(f'Idade: {self.idade}')

#Testando a nossa classe:
pessoa=Pessoa('Gustavo','Molina',30)
print(pessoa.nome)
print(pessoa.sobrenome)
print(pessoa.idade)
pessoa.mostrar_dados()
```

Resposta Exercício 2

Aula16_ContaOO.py > ...

```python
class Conta:

    def __init__(self,titular,saldo=0):
        self.titular=titular
        self.saldo=saldo

    def depositar(self,valor):
        self.saldo+=valor

    def sacar(self,valor):
        self.saldo-=valor

conta=Conta('Gustavo Molina',500)
print(conta.titular,conta.saldo)
conta.depositar(700)
print(conta.titular,conta.saldo)
conta.sacar(1000)
print(conta.titular,conta.saldo)
```

Por hoje é só!

Obrigado! =)

Prof. Gustavo Molina
gmolina@thefutureisblue.me

the future is blue

blue