Herança e polimorfismo

Herança: permite a criação de subclasses, e a partir disso, as subclasses pode herdar atributos e métodos de uma classe.

class Carro(Veiculo):

def \_\_init\_\_(self, marca, modelo, portas):

super().\_\_init\_\_(marca, modelo)

self.portas = portas

Aqui estamos criando uma subclasse carro, a partir da classe veiculo:

def \_\_init\_\_(self, marca, modelo, portas):

* Aqui estamos inicializando a classe

super().\_\_init\_\_(marca, modelo)

* Aqui está puxando o método da classe principal que é veículo, isso garante que a classe principal veiculo, faça a classe carro, funcionar corretamente.

self.portas = portas

* E aqui está adicionando um novo atributo na subclasse chamado “portas”.

Polimofismo:

class Moto(Veiculo):

def \_\_init\_\_(self, marca, modelo, cilindradas):

super().\_\_init\_\_(marca, modelo)

self.cilindradas = cilindradas

aqui nessa parte, estamos criando uma subclasse moto e herdando os atributos e métodos.

def mostrar\_informacoes\_veiculo(veiculo):

veiculo.mostrar\_detalhes()

A o polimorfismo entra nessa parte, por que?

Por que esse método, funciona do mesmo jeito, para qualquer veículo, ele aceita qualquer objeto que seja veículo. Ele pode se comportar diferentemente dependendo da classe do objeto.