Paradigma Orientado a Objetos

Objetivo: juntar dados e funcionalidades.

1) A construção de programas torna-se muito mais natural e “fácil”, visto que podemos construir modelos de objetos do mundo real e colocar no nosso programa.

2) Cada objeto possui dados relacionados a ele, assim como funcionalidades. Além disso, podemos fazer diferentes objetos interagirem entre si.

Exemplo:

```
class Quark:
    def __init__(self, orientation):
        self.orientation = orientation

class Proton:
    def __init__(self):
        self.up_quark_1 = Quark('UP')
        self.up_quark_2 = Quark('UP')
        self.down_quark = Quark('DOWN')

class Atomo:
    def __init__(self, nome, numero_atomico, protons, neutrons, eletrons):
        self.nome = nome
        self.numero_atomico = numero_atomico
        self.protons = protons
        self.neutrons = neutrons
        self.eletrons = eletrons
        self.massa_atomica = self.protons + self.neutrons

class Molecula:
    def __init__(self, nome, atomos):
        self.nome = nome
        self.atomos = atomos
    def ligar(self, molecula):
        print(f'Ligando à molecula {molecula.nome}')
```

```python
class AcidoNucleico:
	def __init__(self):
		pass

class AcidoDesoxirribonucleico(AcidoNucleico):

	def __init__(self, adenina, citosina, timina, guanina):
		super().__init__()
		self.adenina = adenina
		self.citosina = citosina
		self.timina = timina
		self.guanina = guanina


	def realizar_mutacao(self):
		print('Mutação realizada')


class Humano:
	def __init__(self, dna_pai, dna_mae):
		self.idade = 0
		self.dna = dna_pai.realizar_mutacao() + dna_mae.realizar_mutacao()

	def comer(self, comida):
		pass


	def dormir(self, tempo):
		import time
		time.sleep(tempo)


class Planeta:
	def __init__(self, nome, tipo, estrela):
		self.nome = nome
		self.tipo = tipo
		self.estrela = estrela


class Terra(Planeta):
	def __init__(self):
		super().__init__('Terra', 'Rochoso', 'Sol')
		self.distancia_sol = 149_597_870


class Marte(Planeta):
	def __init__(self):
		super().__init__('Marte', 'Rochoso', 'Sol')
		self.distancia_sol = 208_650_000
```

```
class SistemaSolar:
	def __init__(self):
		self.estrela = 'Sol'
		self.planetas = []
		self.terra = Terra()
		self.marte = Marte()
		self.planetas.append(self.terra)
		self.planetas.append(self.marte)
		print('...')


class Galaxia:
	def __init__(self):
		self.sistemas = []
		self.sistema_solar = SistemaSolar()
		print('...')


class Universo:
	QUANTIDADE_PLANETAS_ROCHOSOS = 5_000_000_000_000_000_000_000_000_000
	QUANTIDADE_PLANETAS_GASOSOS = QUANTIDADE_PLANETAS_ROCHOSOS
	QUANTIDADE_TOTAL_PLANETAS = QUANTIDADE_PLANETAS_ROCHOSOS +
QUANTIDADE_PLANETAS_GASOSOS


	def __init__(self):
		self.galaxias = big_bang()


class Main:
	def __init__(self):
		hidrogenio = Atomo('H', 1, 1, 0, 1)

		print('...')
```