

# Python - Proway

Programação Orientada a Objetos (OOP) em  
Python

# Programação Orientada a Objetos em Python

- POO (OOP) um paradigma de programação fortemente baseada no conceito de objetos, que podem conter dados (atributos ou propriedades) e código (na forma de métodos).
- Um objeto é uma estrutura que reflete alguma entidade do mundo real que queremos modelar no nosso sistema

# Programação Orientada a Objetos em Python

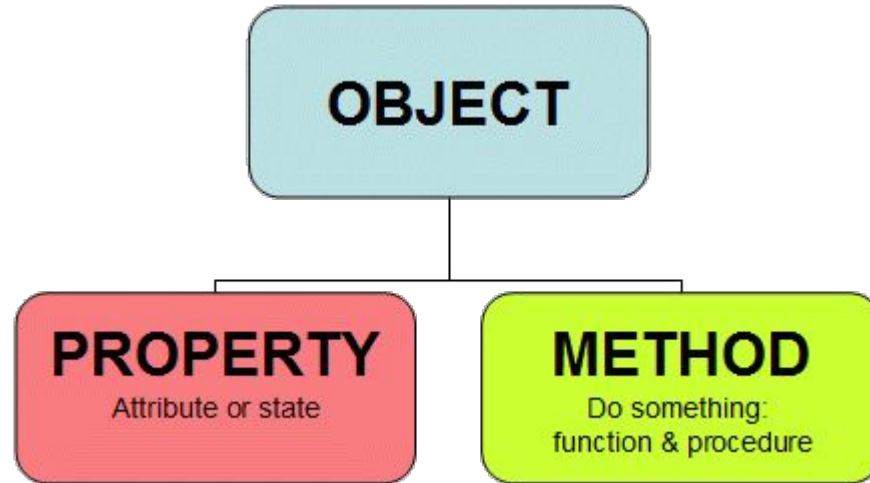


Imagem:  
<https://gist.github.com/mlimberg/43ca77ce4f6e6db1c0ff974a5fea4da5>

# Programação Orientada a Objetos em Python

Existem alguns termos centrais na programação orientada a objetos, que sempre farão parte do processo de análise e desenvolvimento dos programas.

# Programação Orientada a Objetos em Python

- Classe: É a modelo a partir do qual instanciamos (criamos) um objeto. Aqui definimos as características da classe (atributos) e os comportamentos dos objetos instanciados a partir dessa classe (métodos).

# Programação Orientada a Objetos em Python

- Objeto: É a instância de uma classe, ou uma entidade criada a partir de uma classe.

# Programação Orientada a Objetos em Python

- Atributos (ou propriedades): São características da entidade representada por uma determinada classe.

# Programação Orientada a Objetos em Python

- Métodos (ou ações): São as ações que um objeto pode realizar. Além disso, os métodos server pra alterar o estado do próprio objeto.



# Programação Orientada a Objetos em Python

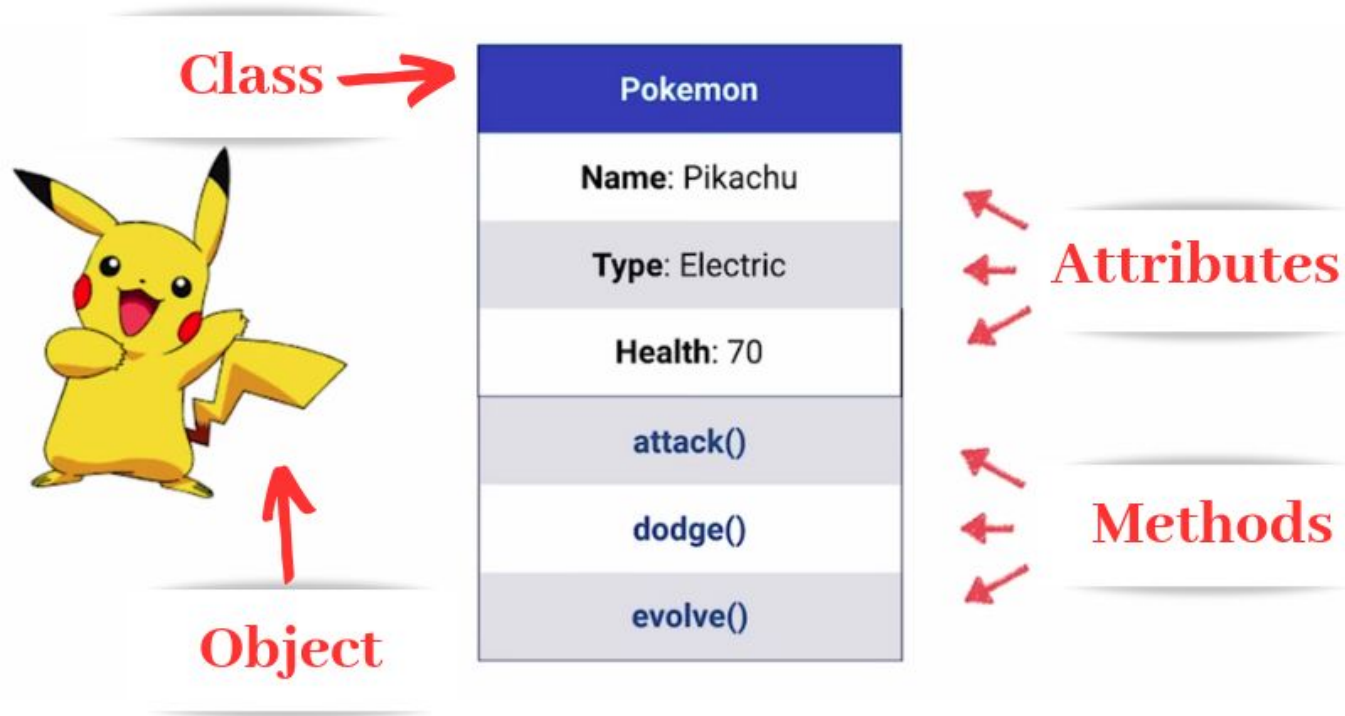


Imagem:

<https://medium.com/future-vision/intro-to-oop-with-python-39ba63967e45>

# Programação Orientada a Objetos em Python

Além da terminologia básica, existem “4 pilares” da Programação Orientada a Objetos, que são os seguintes:

# Programação Orientada a Objetos em Python

Abstração (ou generalização): É o processo de simplificação da definição de uma classe, focando apenas nas partes relevantes, reduzindo a complexidade.

# Programação Orientada a Objetos em Python

Encapsulamento: Utiliza a abstração como base, porém trabalha mais a nível de aplicação. É o processo onde “protegemos” determinadas propriedades ou métodos de serem “expostos”.

# Programação Orientada a Objetos em Python

Herança: É utilizada quando uma classe herda atributos e métodos de uma outra classe.

# Programação Orientada a Objetos em Python

Polimorfismo: Quer dizer “muitas formas”, ou seja, uma coisa que pode se comportar de mais de uma maneira diferente.

# Programação Orientada a Objetos em Python

Composição: Quando um objeto é composto por  
1 ou mais objetos