

OOP com Python

Ideias chave

- O OOP é um paradigma de programação baseado no uso de classes, objetos e suas relações, análogo ao que existe no mundo real.
- Um objeto nada mais é do que um conjunto de variáveis (atributos) e funções (métodos).
- As classes são moldes para a criação de objetos. Na definição de uma classe, teremos os atributos e métodos que serão utilizados para criar os objetos dessa classe.
- Os construtores são métodos que dão um estado inicial às nossas cópias de classe. Eles podem ter parâmetros, em cujo caso, ao instanciar um objeto daquela classe, devemos passar os parâmetros especificados no construtor.
- UML é uma ferramenta para especificar sistemas de software. Tipos de diagramas padronizados nos ajudam a descrever e mapear visualmente o projeto e a estrutura de um sistema de software. Ele consiste em 14 tipos de diagramas para representar as relações entre os diferentes elementos de nosso programa.
- Herança é um conceito comum a todas as linguagens de programação orientadas a objetos. É semelhante à herança no mundo real. Uma classe de crianças herda seus atributos e métodos de uma classe de pais.
- O encapsulamento é uma forma de "proteger" os atributos e/ou métodos de nossas classes contra mudanças indesejadas.
- O conceito de polimorfismo implica que se um método particular de um objeto é invocado em um pedaço de código, diferentes resultados podem ser obtidos, dependendo da classe do objeto. Isto porque objetos diferentes podem ter um método com o mesmo nome, mas realizar operações diferentes.