

Orientação a Objetos com Python

Classes Abstratas e Biblioteca ABC



Classes Abstratas e Biblioteca ABC

- ▶ Uma **classe abstrata** pode ser considerada como um modelo para criar outras classes.
- ▶ Uma classe abstrata precisa ter um ou mais métodos/propriedades abstratos que devem ser implementados pelas classes filhas.
- ▶ Python tem a biblioteca **ABC** (**Abstract Base Classes**) que fornece a funcionalidade de classes abstratas.
- ▶ A classe abstrata precisa herdar de ABC e métodos abstratos são marcados com o *decorator* **@abstractmethod**.
- ▶ As classes abstratas são usadas para definir a **API** de suas classes filhas.

```
from abc import ABC, abstractmethod

class ClasseAbstrata(ABC):
    @abstractmethod
    def metodo_abstrato(self):
        pass

    @property
    @abstractmethod
    def propriedade_abstrata(self):
        pass

class ClasseDerivada(ClasseAbstrata):
    # sobrescrevendo o método abstrato
    def metodo_abstrato(self):
        print("Boa noite")

    # sobrescrevendo a propriedade abstrata
    @property
    def propriedade_abstrata(self):
        return 'Noite'
```

