



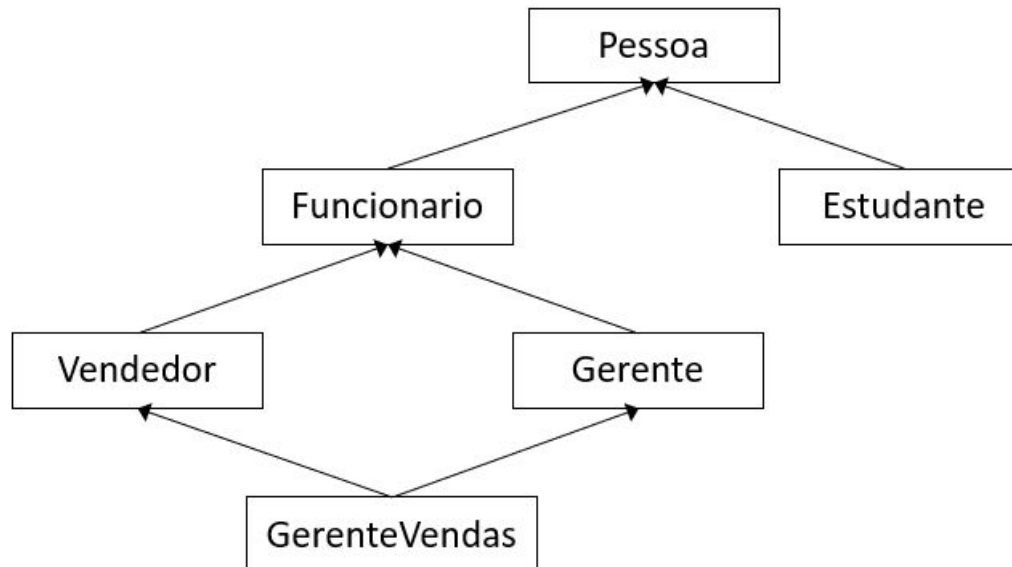
Programação Orientada a Objetos

Herança Múltipla

Prof. Paulo Vinicius Vieira
paulo.vieira@faculdadeimpacta.com.br

Herança Múltipla

- A herança múltipla ocorre sempre que uma subclasse possui duas ou mais superclasses
- Ou seja, uma classe é “filha” de mais de uma classe
- Através da herança múltipla é possível combinar as características de várias superclasses existentes como um ponto de partida para a definição de uma nova classe



Herança Múltipla

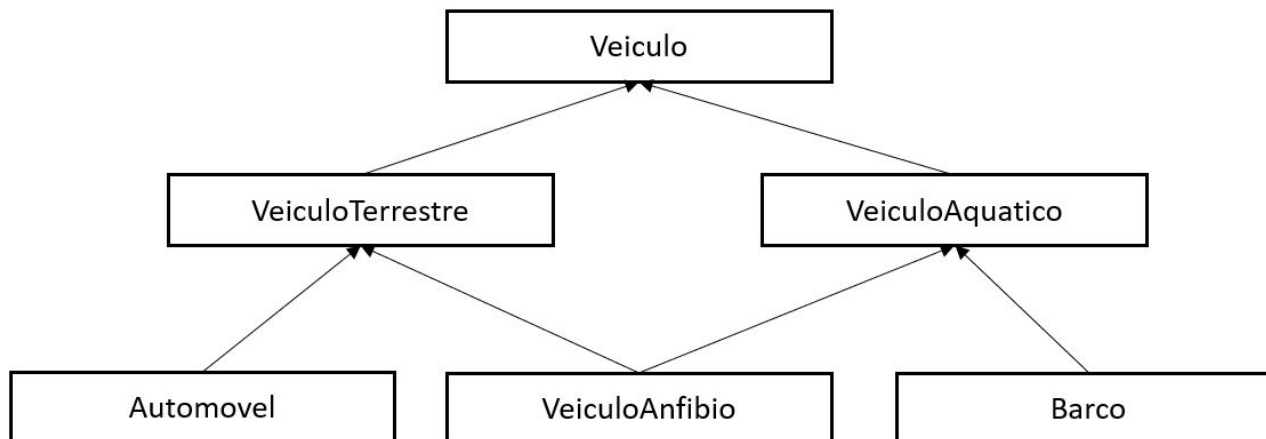
- Conceitualmente, a herança múltipla é necessária para modelar o mundo real de maneira mais precisa
- Na prática, ela pode levar a problemas de ambiguidade na implementação:
 - Métodos com mesmo nome em duas superclasses
- A herança múltipla deve ser utilizada com cuidado, pois o seu uso pode gerar problemas em uma estrutura de herança

Herança Múltipla

- Nem todas as linguagens Orientadas a Objetos suportam a herança múltipla.
 - Aceitam herança múltipla:
 - Python
 - Javascript
 - C++
 - Não aceitam herança múltipla:
 - Java
 - C#

Herança Múltipla

- Neste exemplo, a classe **VeiculoAnfibio** herda indiretamente duas vezes da classe **Veiculo**
- Suponha que desejamos utilizar um objeto da classe **VeiculoAnfibio** para invocar um método da classe **Veiculo**
 - Qual seria o caminho para atingir a classe **Veiculo**?
 - Qual das classes intermediárias seria utilizada?



Herança Múltipla

- A sintaxe é similar à herança simples:
 - Após a definição da classe derivada, listamos entre parênteses as classes a serem herdadas, separadas por vírgula

```
class VeiculoAnfibio(VeiculoTerrestre, VeiculoAquatico):  
    def __init__(self):  
        #função super chama a primeira classe herdada  
        super().__init__()
```