

- 1- Explique a diferença entre objeto e classe, comente quais suas funções em um programa. Dê exemplos de Classes e Objetos.

Classe é uma abstração, um modelo ou uma descrição do que seria o objeto em si. O objeto por sua vez é a representação de algo concreto, do mundo real, trazido ao mundo da programação. A classe permite organizar os atributos e comportamentos de um objeto, enquanto o objeto tem a função de representar algo físico no mundo da programação.

Exemplos:

Classe: Cachorro, Humano;

Objetos: Poodle, Collie, Jorge, Maria.

- 2- Qual a definição de generalização e especialização? Exemplifique as definições.

Generalização: Definição de uma classe genérica que contém comportamentos e atributos padrão para objetos que pertençam a um grupo genérico de classes que as herdem.

Especialização: Classes que herdam de um tipo genérico de classe, herdando atributos e comportamentos.

Exemplo: Criar uma classe genérica Veículo, com atributos velocidadeMaxima e quantidadePassageiros e o comportamento mover. Ao criar essa classe, geramos uma classe genérica, que pode originar diversas classes específicas, como Carro, Avião, Motocicleta, entre outros, que irão possuir outros atributos e comportamentos mais específicos, como por exemplo a classe Carro, que poderá ter o atributo quantidadePortas e o comportamento buzinar.

- 3- O que é polimorfismo e sobrecarga de métodos? Como implementar?

Polimorfismo é a capacidade de fazer com que um objeto se comporte como outro. Nesse caso, pode-se instanciar um objeto mais específico em uma classe mais genérica.

Sobrecarga é o conceito de permitir dentro de uma mesma classe vários comportamentos com o mesmo nome, mas com argumentos diferentes.

Para implementar polimorfismo primeiro é necessário ter uma classe genérica e outra classe especializada, que herde essa classe genérica. Feito isso, será possível instanciar um objeto proveniente de uma classe específica em uma classe genérica.

Para a sobrecarga, é preciso criar um comportamento e replicá-lo, contendo o mesmo nome e estando dentro da mesma classe, passando argumentos diferentes para cada comportamento.