PROGRAMAÇÃO ORIENTADA A OBJETOS

Polimorfismo

Prof. Emanoel Barreiros

Introdução ao Polimorfismo

- Suponha que desenvolvemos um programa que simule o movimento de diferentes animais
- Este sistema é usado para estudar peixes, sapos e pássaros. Para tal, você cria as classes Peixe, Sapo e Passaro
- Todas essas classes herdam da classe Animal, que possui um método mover() e atributos que mantém a posição de cada animal (x, y)
- Nosso sistema precisa manter um array de objetos do tipo Animal, e a cada segundo, executar o método mover() de cada um

Introdução ao Polimorfismo

- Um peixe pode nadar 1 metro, enquanto que um pássaro pode voar 5 metros e um salpo pode saltar 0,5 metros
- Cada objeto sabe como mover-se e atualizar a sua posição, mas o nosso sistema só deve chamar o método mover() de cada um
- O método mover é polimórfico, pois para cada objeto ele possui uma implementação diferente, inerente a cada tipo que a implementa
- Com polimorfismo podemos projetar e implementar sistemas facilmente extensíveis
  - Ex: se desejarmos criar uma nova classe Formiga, as outras classes de animais permanecem intocadas

Exemplos de polimorfismo

- Quadilateros
  - Se uma classe Retangulo extende a classe Quadrilatero, então a classe Retangulo é mais específica
  - Qualquer operação (.e.g. calcular a sua área) pode ser realizada no Retangulo
  - Essas operações também podem ser executadas em outros Quadrilateros (Quadrado, Paralelogramo, Trapezio)
  - O polimorfismo ocorre quando um programa invoca a execução de um método através de um objeto do tipo da superclasse e em tempo de execução, a versão correta do método na subclasse é chamada

Exemplo

- Exemplo no código!

Classes e métodos abstratos

- Classes abstratas nunca poderão ser instanciadas
- Classes abstratas tem como objetivo fornecer um projeto comum para suas subclasses
- Classes que podem ser usadas para instanciar objetos são chamadas de classes concretas
- Classes concretas são obrigadas a definir a implementação de todos os seus métodos, enquanto classes abstratas não são
  - Métodos sem implementação são chamados de métodos abstratos
- Nem todas hierarquias precisam de classes abstratas

Declarando um método abstrato

- Ex: public abstract void desenhar();

- O método abstrato é apenas isso, não existem chaves pois ele não possui implementação!
- Qualquer classe que defina pelo menos um método abstrato também deve ser abstrata
- Classes concretas que herdam de classes abstratas devem implementar todos os métodos abstratos

Exemplo de polomorfismo na vida real

- Drivers de dispositivos
  - O sistema operacional (SO) deve ser capaz de se comunicar com os mais variados dispositivos existentes. Tome como exemplo uma impressora. Para permitir a comunicação do SO com cada impressora, deve existir um driver para aquela impressora.
  - Um SO não sabe se comunicar com todos os dispositivos existentes no mundo, mas ele define um padrão de comunicação com drivers
  - Cada fabricante implementa o seu driver e o disponibiliza para o público, que instala em seus SOs
  - O SO precisa, por sua vez, apenas saber se comunicar de maneira padrão com drivers de dispositivos, e estes que vão saber como se comunicar com os dispositivos de verdade

Questionário rápido

- Se uma classe contém pelo menos um método abstrato ela é uma classe _____.
- Classes que podem ser instanciadas são chamadas de classes _____.
- Verdadeiro ou falso:
  - Todos os métodos em uma classe abstrata devem ser abstratos.
  - Invocar um método presente apenas em uma subclasse através de uma variável do tipo da subclasse não é permitido.
  - Se uma superclasse declara um método abstrato, a subclasse é obrigada a implementar aquele método.