Gabriela de Melo Freitas PWBE

**Polimorfismo**

Polimorfismo pode ser definido como a capacidade de um método ou função ser aplicado de formas diferentes em classes ou objetos distintos. Assim, os objetos em que o polimorfismo foi aplicado respondem da forma adequada ao método, mas de maneiras distintas, que são associadas características específicas da sua classe particular.

Na programação, é normal ter classes que são derivadas de uma “classe mãe”. Elas compartilham os mesmos métodos, mas tem funções específicas e executam tarefas diferentes umas das outras. É fundamental para a flexibilidade do código que um objeto seja capaz de assumir. Com o polimorfismo, é possível que um objeto assuma diversas formas, dependendo do contexto em que ele está sendo usado.

**Herança**

A Herança é a capacidade de as classes compartilharem atributos, métodos e outros membros da classe entre si. Para que a ligação entre as classes aconteça, a herança adota um relacionamento hierárquico.

Na Herança, existem dois tipos principais de classes: a Classe Base, que é a que dá as características para a outra classe e a Classe Derivada, que é quem herda as características da Classe Base.

Um exemplo de Classe Base é uma classe Pessoas, que contém informações como nome e idade. A Classe Derivada desse exemplo poderia ser a classe Funcionário, que herdaria os campos de nome e idade da classe Pessoas, mas teria novas informações específicas dela, como um campo cargo.

**Referências**

https://www.devmedia.com.br/conceitos-e-exemplos-polimorfismo-programacao-orientada-a-objetos/18701

https://www.devmedia.com.br/conceitos-e-exemplos-heranca-programacao-orientada-a-objetos-parte-1/18579