

PILARES DA POO: POLIMORFISMO		PILARES DA POO: POLIMORFISMO
O que é o Polimorfismo ?		Polimorfismo: é uma união de 2 palavras: Poli (Muitos) e Morfo (Formas), ou seja, a palavra Polimorfismo literalmente significa "De Várias Formas". Levado para o contexto de POO, Polimorfismo pode ser aplicado aos nossos métodos, onde um método com o mesmo nome pode assumir várias formas diferentes dependendo da aplicação á qual ele será aplicado.
O que é " Assinatura do Método "?		Assinatura do Método: se refere <i>á quantidade e tipo de argumentos que um método pode receber</i> , por exemplo, um método que recebe 2 argumentos como parâmetro e o tipo de dado destes argumentos é float, têm uma assinatura de: 2 argumentos do tipo float... método(2.3, 4.2) { } Já um método que recebe 3 argumentos, onde os 2 primeiros são strings e o último é inteiro, têm uma assinatura de: 2 argumentos strings e 1 inteiro... método (" String1 ", " String2 ", 3) { } OBS: Assinatura não se refere ao retorno do método, métodos podem ter assinaturas iguais, mas retornos diferentes.
O que são os Tipos de Polimorfismo ?		Podemos dividir o Polimorfismo em 2 áreas de abrangência: - Polimorfismo Universal: onde o Polimorfismo é aplicado de formas infinitas sobre o objeto que desejamos mutar. Dentro da área Universal temos os seguintes tipos de Polimorfismo: Override (Sobreposição) e o Parametric (Paramétrico); - Polimorfismo Ad-Hoc: onde o Polimorfismo é aplicado de forma específica sobre o elementos que desejamos mutar. Dentro do Ad-Hoc temos o seguinte tipos de Polimorfismo: Overloading (Sobrecarga) e o Coercion (Coerção);
Quais as características do Polimorfismo Override ?		Override: ou Sobreposição, é a forma mais utilizada de Polimorfismo - junto ao Overloading - onde nós sobrepomos um método abstrato, criando um método de sobreposição . Como já vimos anteriormente um método abstrato é um método criado sem implementação na Classe Abstrata com o objetivo de ser sobreposto dentro das nossas Sub Classes, quando fazemos isso estamos praticando o Polimorfismo Override. Polimorfismos Override sempre acontecem com uma herança, entre Super Classe e Sub Classe, sempre são com um método abstrato para um método de sobreposição e os métodos de sobreposição têm a mesma assinatura que os métodos abstratos.
Quais as características do Polimorfismo Parametric ?		Parametric: ou Paramétrico, é parecido com o Polimorfismo de Sobreposição, mas ele é mais aplicado a linguagens de tipagem forte, como o Java por exemplo, onde nós iríamos definir um método com um tipo genérico (um tipo que futuramente poderá receber qualquer tipo de dado, no Java usamos até a palavra reservada "generics" para definir esse tipo de dado) que poderá ser modificado futuramente assim que receber o dado . Esse tipo de mudança é o que chamamos de Poliformismo Paramétrico.
Quais as características do Polimorfismo Coercion ?		Coercion: ou Coerção, é o método usado principalmente em Linguagens de tipagem forte, esse método não precisa de usar herança, ele consiste simplesmente em mudar a tipagem dos parâmetros de um método para outro tipo de dado durante a execução do programa - é o que chamamos de Polimorfismo em Tempo de Execução. Geralmente utilizamos um "cast" para fazer a mudança da tipagem, é chamado de "Coersão" por que se certo modo estamos "Coagindo" o tipo de dado a mudar para áquele tipo de dado que queremos.
Quais as características do Polimorfismo Overloading ?		Overloading: ou Sobrecarga, é o método de pegar um método já existente dentro de uma classe e DENTRO DA PRÓPRIA CLASSE adicionar métodos com o mesmo nome, mas com assinaturas diferentes, onde, dependendo do valor que o usuário adentrar com ele, um determinado método da classe será utilizado em detrimento ao outro . Diferente da Sobreposição que implementa todo um novo comportamento ao método, o Overloading acrescenta apenas um argumento diferente e dentro da próxima classe. O Overloading também acontece em tempo de execução quando chamamos o método Overloading.