Polymorphism – Articulate

Name: Roberto Soares

What is polymorphism and why is it important?

Your response must:

- Explain the meaning of Polymorphism.

- Highlight a benefit of Polymorphism.

- Provide an application of Polymorphism.

- Use a code example of Polymorphism from the program you wrote.

Polymorphism is the ability to take many forms. In programming, this principle is demonstrated when a line of code can have different behaviors depending on the context.

Polymorphism in C# is one of the fundamental principles of object-oriented programming. It allows objects from different classes to be treated uniformly through a common interface.

Polymorphism in C# promotes code reuse, facilitates maintenance and makes code more flexible and extensible. It is essential for building class hierarchies and for implementing concepts such as abstraction and encapsulation.

Polymorphism brings several benefits to software development in C# and other object-oriented languages. Here are some of the main benefits:

Code reuse:

Polymorphism allows methods and classes to be shared and reused in different contexts.

Extensibility: With polymorphism, it's easier to add new functionality to the system without modifying the existing code.

Simplified maintenance: Polymorphism promotes the principle of encapsulation, which means that changes made to a derived class do not affect the base classes or other derived classes.

Ex.

using System;

// Classe base

class Animal

{

public virtual void EmitSound()

{

Console.WriteLine("Som genérico de animal");

}

}

class Dog : Animal

{

public override void EmitSound()

{

Console.WriteLine("Latindo...");

}

}

class Cat : Animal

{

public override void EmitSound()

{

Console.WriteLine("Miando...");

}

}

class Program

{

static void Main(string[] args)

{

Animal[] animals = new Animal[2];

animals[0] = new Dog();

animals[1] = new Cat();

foreach (var animal in animals)

{

animal.EmitSound(); // Chama o método específico de cada tipo de animal

}

}

}