Name: Roberto Soares

**Encapsulation – Articulate**

What is encapsulation and why is it important?

Your response must:

* Explain the meaning of Encapsulation
* Highlight a benefit of Encapsulation
* Provide an application of Encapsulation
* Use a code example of Encapsulation from the program you wrote
* Thoroughly explain these concepts (this likely cannot be done in less than 100 words)

Encapsulation is one of the fundamental principles and plays a crucial role in building robust and secure systems in C#. In C#, encapsulation is achieved through the use of access modifiers such as public and private, which control access to a class's members.

Members marked as private are accessible only within the class itself. Keep in mind that when we talk about making data private or hiding it, we are not referring to encrypting sensitive data like credit card numbers to conceal it from users or hackers. Instead, we are limiting the parts of our program that can directly access a variable, effectively hiding it from other parts of the program.

Members marked as public are accessible from anywhere, allowing other classes to access them.

Encapsulation offers several benefits in object-oriented programming. Here are some of the key advantages, Security and Access Control, Private members provide a level of security by, restricting access to within the class.

Hiding Internal Details, Encapsulation allows the hiding of internal details, making it easier to understand and maintain the code.

Simplified Maintenance, Changes to the internal implementation of a class do not affect other parts of the code, simplifying maintenance.

Facilitates Business Logic Implementation, Encapsulation facilitates the implementation of business logic by exposing a well-defined interface while hiding complexity.

Code Reusability, Encapsulation promotes code reusability, as encapsulated classes can be used in different parts of the system without needing to understand their internal implementation.

It's important to note that when we talk about making data private or hidden, we are emphasizing limiting direct access rather than encrypting sensitive information. This distinction is crucial in understanding the role of encapsulation in maintaining code integrity and security.

Exemple :

public class Pessoa

{

// Campo privado encapsulado

private int idade;

// Propriedade pública para acessar o campo encapsulado

public int Idade

get { return idade; }

set