**4 Pilares da Programação Orientada a Objeto**

* **Abstração**

A abstração consiste em representar um objeto no sistema, destacando apenas suas características essenciais. Os principais aspectos desse conceito são:

**- Identidade:** Cada objeto deve ter uma identidade única dentro do sistema, que o diferencie de outros. Para isso, utilizamos o conceito de Namespace, que garante que a identidade de um objeto não seja repetida dentro do mesmo espaço de nomes.

**- Propriedades:** São as características do objeto, que definem seus atributos ou estados.

**- Métodos:** Representam as ações que o objeto pode realizar no sistema. Eles definem o comportamento do objeto.

* **Encapsulamento**

O encapsulamento protege os dados internos de um objeto, restringindo o acesso direto a eles. Para isso, utiliza:

- **Propriedades Privadas:** São atributos que não podem ser acessados diretamente de fora do objeto.

- **Getters e Setters:** São métodos especiais usados para obter (get) ou definir (set) os valores das propriedades privadas, garantindo maior controle e segurança sobre os dados.

* **Herança**

A herança permite que um objeto (chamado de classe "filha") herde características de outro objeto (a classe "pai"). Isso inclui tanto as propriedades quanto os métodos, facilitando o reaproveitamento de código e a organização do sistema.

* **Polimorfismo**

O polimorfismo consiste em alterar o comportamento de um método herdado da classe pai. Isso significa que, ao sobrescrever o método na classe filha, podemos adaptar sua funcionalidade para atender às necessidades específicas do novo objeto.