

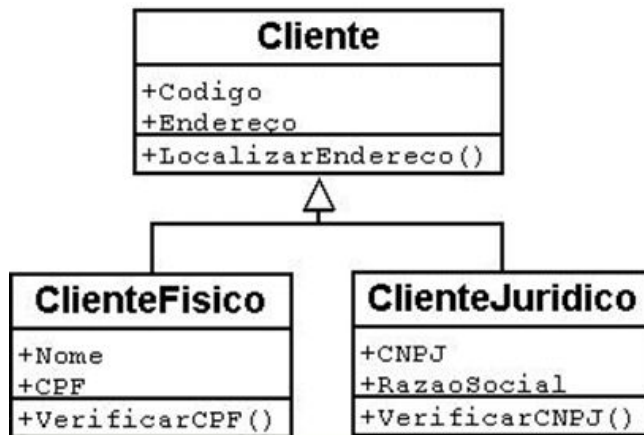
Programação Orientada a Objetos



Herança

Herança:

A herança é um mecanismo na programação orientada a objetos que permite a criação de novas classes baseadas em classes existentes. Ela promove a reutilização de código ao herdar atributos e métodos da classe pai (superclasse). A herança permite criar hierarquias de classes, compartilhar comportamentos comuns e especializar as classes filhas (subclasses) com novos atributos e métodos.



Classe mãe em C++

```
// Classe mãe (classe base)
class Animal
{
private:
    int numeroPatas;
    char nome[10];
    char especie[10];
    char som[10];

public:
    Animal(int numeroPatas, char* nome, char* especie, char* som)
    {
        this->numeroPatas = numeroPatas;
        strcpy(this->nome, nome);
        strcpy(this->especie, especie);
        strcpy(this->som, som);
    }

    void emitirSom()
    {
        printf("O %s faz %s!\n", especie, som);
    }
};
```

Classe filha em C++

```
// Classe filha (classe derivada)
class Cachorro : public Animal
{
public:
    Cachorro() : Animal(4, "Cachorro", "Canino", "Auau") { }
};
```

Classe mãe em C#

```
class Animal {  
    1 reference  
    int numeroPatas;  
    2 references  
    string nome;  
    1 reference  
    string especie;  
    2 references  
    string som;  
  
    3 references  
    public Animal(int numeroPatas, string nome, string especie, string som) {  
        this.numeroPatas = numeroPatas;  
        this.nome = nome;  
        this.especie = especie;  
        this.som = som;  
    }  
    1 reference  
    public void EmitirSom() {  
        Console.WriteLine("{0} faz {1}", nome, som);  
    }  
}
```

Classe Filha em C#

```
class Cachorro : Animal {  
    ↑ reference  
    public Cachorro() : base(4, "Cachorro", "Canis lupus familiaris", "Au au!") { }  
}
```

Múltipla Herança

A linguagem C++ nos permite trabalhar com múltipla herança, ou seja, uma classe pode herdar atributos de mais de uma classe mãe.

Múltipla Herança em C++

```
// Primeira classe mãe
class ClasseMae1 {
public:
    void metodoMae1() {
        cout << "Método da ClasseMae1" << endl;
    }
};

// Segunda classe mãe
class ClasseMae2 {
public:
    void metodoMae2() {
        cout << "Método da ClasseMae2" << endl;
    }
};

// Primeira classe filha
class ClasseFilha1 : public ClasseMae1, public ClasseMae2 {
public:
    void metodoFilha1() {
        cout << "Método da ClasseFilha1" << endl;
    }
};
```


Exercício:

Faça em uma linguagem de sua escolha, uma classe mãe com atributos e métodos próprios, e duas classes filhas, que tenham seus próprios atributos e métodos, além dos herdados da classe mãe.