



Aluno: André Luiz N. Carneiro De Castro RA: 92854

ATIVIDADE - POO



Atividade Extra.

Prof. Dr. Mauricio Acconcia Dias

FUNDAÇÃO HERMÍNIO OMETTO

ARARAS/SP

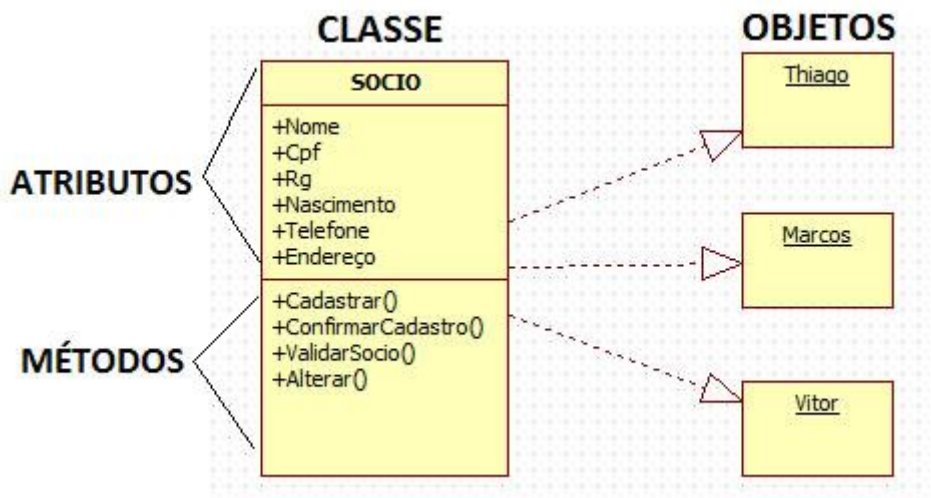
05/2021

- O que é classe em POO?

Uma classe é uma forma de definir um tipo de dado em uma linguagem orientada a objeto. Ela é formada por dados e comportamentos. Para definir os dados são utilizados os atributos, e para definir o comportamento são utilizados métodos.

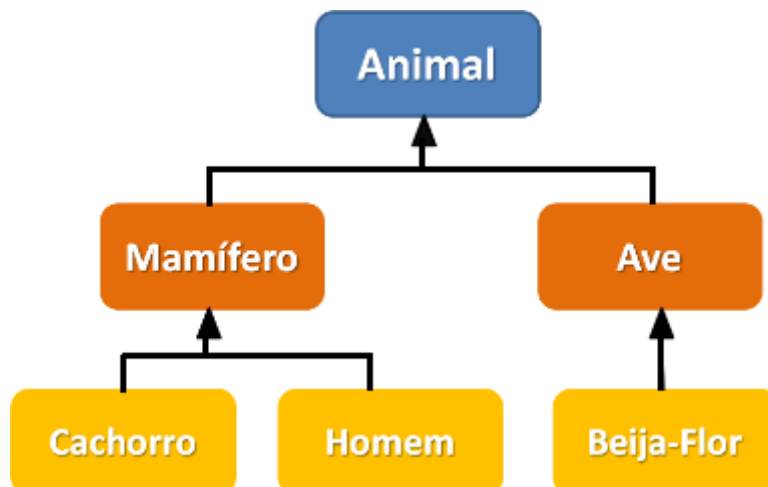
- O que é objeto em POO?

Um objeto é um elemento computacional que representa, no domínio da solução, alguma entidade (abstrata ou concreta) do domínio de interesse do problema sob análise. Objetos similares são agrupados em classes. No paradigma de orientação a objetos, tudo pode ser potencialmente representado como um objeto.



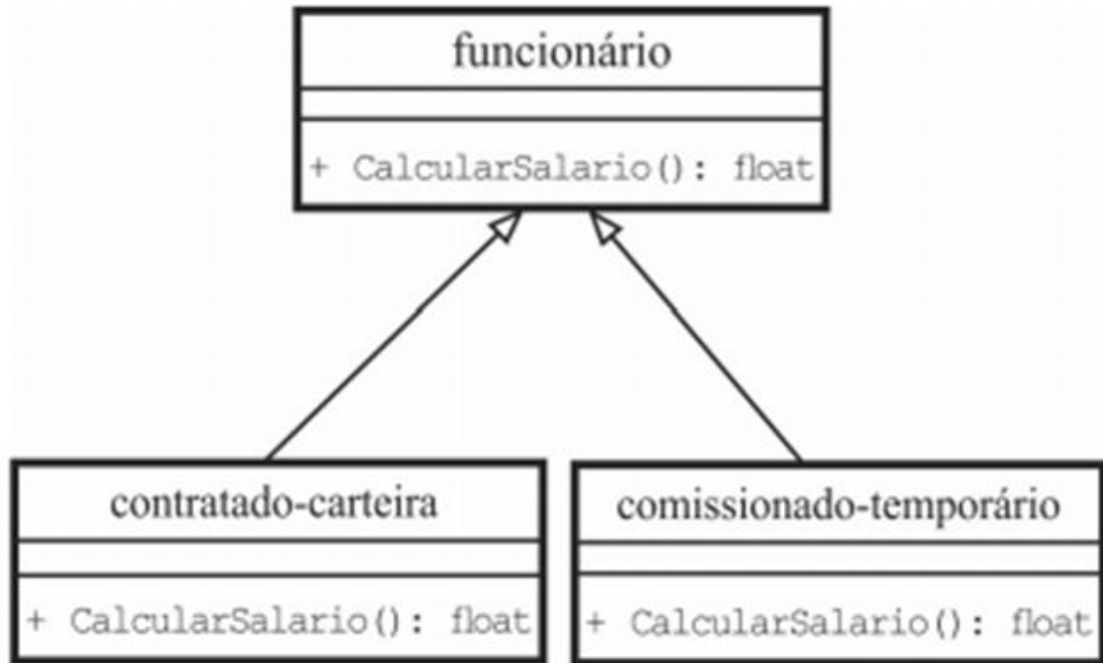
- O que é herança em POO?

Herança é um princípio de orientação a objetos, que permite que classes compartilhem atributos e métodos, através de "heranças". Ela é usada na intenção de reaproveitar código ou comportamento generalizado ou especializar operações ou atributos. O conceito de herança de várias classes é conhecido como herança múltipla.



- O que é polimorfismo em POO?

Em programação orientada a objetos, polimorfismo é o princípio pelo qual duas ou mais classes derivadas da mesma superclasse podem invocar métodos que têm a mesma assinatura, mas comportamentos distintos



Exemplo de **HERANÇA**; [c++]

```
#include <bits/stdc++.h>
using namespace std;
```

```
class Pais
{
public:
    int id_p;
};
```

```
class crianca : public Pais
{
public:
    int id_c;
};
```

```
//main
int main()
{
```

```
    crianca obj1;
```

```
// Um objeto da classe filho possui todos os membros de dados
// e funções de membro da classe pai
```

```
obj1.id_c = 7;  
obj1.id_p = 91;  
cout << "Crianca id sera " << obj1.id_c << endl;  
cout << "Pais id sera " << obj1.id_p << endl;  
  
return 0;  
}
```

Nota-se que uma classe derivada não herda o acesso a membros de dados privados. No entanto, ele herda um objeto pai completo, que contém todos os membros privados declarados por essa classe.

Saída de dados:

Crianca id sera 7

Pais id sera 91

Exemplo de classe [c++]

```
class MinhaClasse  
{  
public:          // Modo de acesso  
    int meuNum;    // Atributo (int)  
    string minhaString; // Atributo2 (string)  
};
```

Exemplo de Objeto criado a partir de uma classe [c++]

```
#include <bits/stdc++.h>  
using namespace std;  
  
class MinhaClasse  
{  
public:          // Modo de acesso;  
    int meuNum;    // Atributo (int)  
    string minhaString; // Atributo2 (string)  
};  
  
int main() {  
    MinhaClasse meuObj;  
  
    // Acessando atributos e valores  
    meuObj.meuNum = 11;  
    meuObj.minhaString = "POO OBJ";  
  
    cout << meuObj.meuNum << "\n";  
    cout << meuObj.minhaString;  
    return 0;  
}
```

Exemplo de Polimorfismo [c++]

A herança nos permite herdar atributos e métodos de outra classe. O polimorfismo usa esses métodos para realizar diferentes tarefas. Isso nos permite realizar uma única ação de maneiras diferentes.

```
#include <iostream>
```

```
using namespace std;
```

```
class Animal {  
public:  
    void somAnimal() {  
        cout << "O animal faz um determinado som; \n" ;  
    }  
};
```

```
class Cavalo : public Animal {  
public:  
    void somAnimal () {  
        cout << "O cavalo faz\n" ;  
    }  
};
```

```
// Derived class  
class Gato : public Animal {  
public:  
    void somAnimal () {  
        cout << "O Gato faz \n" ;  
    }  
};
```

```
int main() {  
    Animal meuAnimal;  
    Cavalo cavalo;  
    Gato gato;  
  
    meuAnimal.somAnimal ();  
    cavalo.somAnimal ();  
    gato.somAnimal ();  
    return 0;  
}
```