

Programação Orientada a Objetos (POO)

Disciplina: Desenvolvimento de Aplicações



Conteúdo(s):

Programação Orientada a Objetos (POO): conceitos iniciais.

Habilidade(s):

Bloco 1

Como você está se sentindo hoje?



Programação Orientada a Objetos (POO)

A Programação Orientada a Objetos (POO) é um paradigma que utiliza objetos para representar entidades do mundo real e suas interações.

Características como encapsulamento, herança e polimorfismo promovem a organização e reutilização de código, tornando o desenvolvimento de *software* mais eficiente e flexível.



Pilares do POO

Abstração

Significa usar coisas simples para representar algo mais complexo. Todos nós sabemos como usar um computador, mas não precisamos saber como funcionam todas as suas peças para aproveitá-lo.

Herança

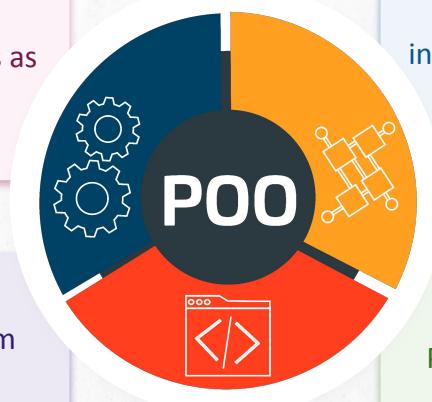
Permite criar novas classes baseadas em classes existentes, compartilhando características e comportamentos, promovendo a reutilização de código.

Encapsulamento

Refere-se à prática de esconder os detalhes internos dos objetos e fornecer uma interface controlada para interagir com eles.

Polimorfismo

Permite que objetos de diferentes classes sejam tratados de maneira uniforme, tornando o código mais flexível e adaptável.



Benefícios

Herança

Reutilização de código: evita duplicação de implementação.
Organização: hierarquia lógica de classes relacionadas.

Encapsulamento

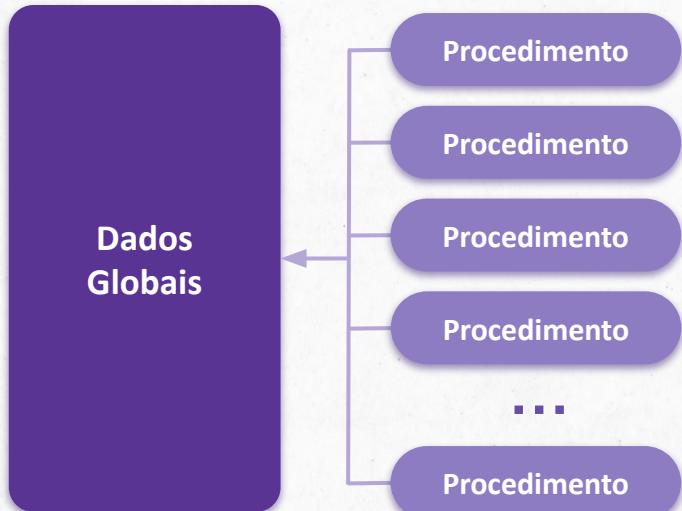
Melhora a **segurança** e a **modularidade** do código. Permite a alteração interna sem afetar o uso externo.

Polimorfismo

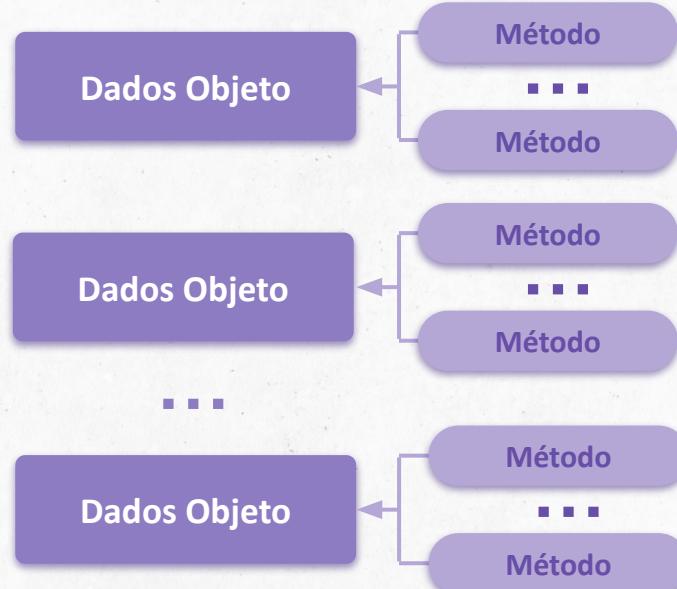
Flexibilidade: código genérico para tratar múltiplas classes.
Extensibilidade: adição de novas classes sem alterar o código existente.

POO x programação estruturada

Programação Estruturada



Programação Orientada a Objetos



Organização e reutilização de código

A POO promove a estruturação lógica do código em objetos, melhorando a compreensão e manutenção.

A herança e o encapsulamento permitem reutilizar classes existentes em novos contextos.



Vamos praticar?

Atividade

10 minutos

Pesquisem, no celular, um *software* que utilize POO em seu código e vejam como ele funciona. Em seguida, falem para a turma o que encontraram.



Discussões e reflexões

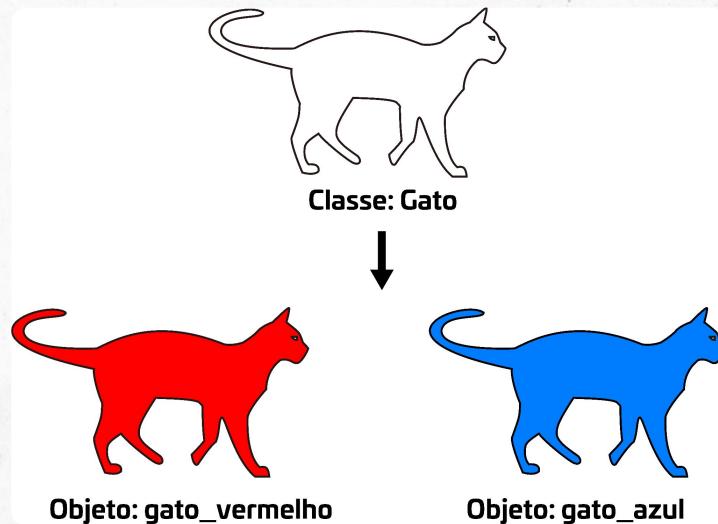


Bloco 2

Objetos

Na POO, objetos são representações concretas de entidades do mundo real.

Cada objeto é uma instância de uma classe. Uma classe é um modelo que define a estrutura e o comportamento de um objeto.



Atributos e comportamentos

Atributos

Os objetos possuem características que são representadas por seus atributos.

Atributos são variáveis associadas a um objeto.

Exemplos: nome, idade, cor, tamanho etc.

Objeto "Cachorro"

Atributos: raça, cor, tamanho.

Comportamentos

Os objetos podem executar ações que são representadas por seus métodos.

Métodos são funções associadas a um objeto.

Exemplos: ligar(), acelerar(), frear().

Objeto "Cachorro"

Comportamentos: latir(), correr(), dormir().

Tipos de objetos

Concretos

Objeto: *smartphone*.

Características: (modelo, marca, capacidade).

Comportamento: (ligar, desligar, clicar, rolarTela).

Estado: (ligado, desligado, travado).

Abstratos

Objeto: compromisso.

Características: (horário, comQuem, local).

Comportamento: (marcar, desmarcar, reallocar).

Estado: (marcado, cancelado).

Benefícios da abordagem orientada a objetos

- **Organização:**

A abordagem orientada a objetos organiza o código em unidades independentes e reutilizáveis.

- **Modularidade:**

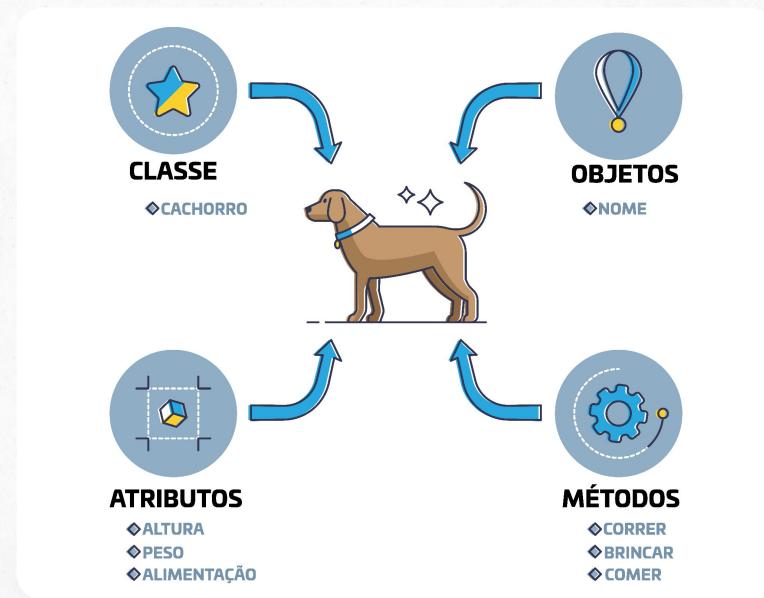
A criação de classes e objetos promove a modularidade, facilitando a manutenção e expansão.

- **Reutilização de código:**

Classes e objetos podem ser reutilizados em diferentes partes do programa.



Observe e comente!



Vamos praticar?

Atividade

10 minutos

Escolham um objeto e reproduzam a imagem anterior, informando classe, objeto, atributos e comportamento.



Discussões e reflexões



Bloco 3

Fofoca do bem

Contem para a turma, em forma de fofoca, um pouco sobre objetos.



Classes

As classes servem como modelos para a criação de objetos na POO.

Elas encapsulam atributos e métodos relacionados, facilitando a organização e reutilização de código.

Ao criar classes, você está **definindo a estrutura e o comportamento dos objetos** que serão instanciados a partir delas.



Classes e objetos

Na POO, uma classe é um modelo que define a estrutura e comportamento de um objeto. Uma classe serve como um plano para criar objetos.



Objetos são instâncias concretas de classes. Cada objeto tem atributos (características) e métodos (ações).

Sintaxe

java

 Copy code

```
class NomeDaClasse {
    // Atributos
    tipo nomeDoAtributo1;
    tipo nomeDoAtributo2;

    // Métodos
    tipoRetorno nomeDoMetodo1(parametros) {
        // Implementação do método
    }

    tipoRetorno nomeDoMetodo2(parametros) {
        // Implementação do método
    }

    // Mais atributos e métodos...
}
```

Exemplo: classe "Pessoa"

```
class Pessoa {  
    // Atributos  
    String nome;  
    int idade;  
  
    // Métodos  
    void apresentar() {  
        System.out.println("Olá, meu nome é " + nome + " e tenho " + idade + " anos.");  
    }  
  
    void envelhecer() {  
        idade++;  
        System.out.println("Aniversário celebrado! Idade atual: " + idade + " anos.");  
    }  
}
```

Benefícios do uso de classes

- **Organização:**

Classes permitem a organização estruturada de atributos e métodos relacionados.

- **Reutilização de código:**

Classes podem ser reutilizadas em diferentes partes do programa.

- **Abstração:**

Classes abstraem detalhes complexos e permitem o foco em funcionalidades específicas.



Vamos praticar?

Peguem uma folha para realizar a atividade.

Atividade

15 minutos

Criem um mapa mental sobre classes e sua sintaxe. Lembrem de detalhá-lo para lhes auxiliar no momento de criar um código.



Bloco 4

Atributos

Atributos são características ou propriedades que descrevem um objeto.

Uma classe "Carro" pode ter atributos como "modelo", "cor" e "ano".



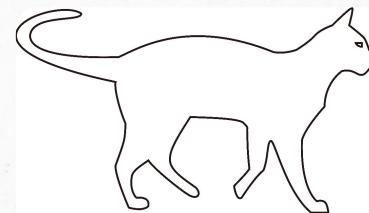
Valores

Definição

Cada atributo possui um valor que representa o estado atual do objeto. Os valores dos atributos determinam as características específicas do objeto.

Exemplo:

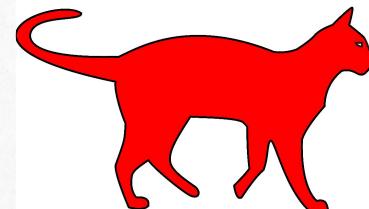
Se um carro tem o atributo "cor", o valor desse atributo poderia ser "vermelho", "azul" etc.



Gato gato_vermelho = new Gato();



gato_vermelho.Cor = "VERMELHO"



Objeto: gato_vermelho

Sintaxe

```
java
```

```
class NomeDaClasse {  
    tipo nomeDoAtributo1;  
    tipo nomeDoAtributo2;  
  
    // Construtor e métodos...  
}
```

Acesso

Acesso

Atributos podem ser acessados e modificados através de métodos.

Métodos de acesso

Getters: obtêm o valor de um atributo.

Setters: modificam o valor de um atributo.

java

```
class Carro {  
    String modelo;  
  
    // Getter para acessar o valor do atributo "modelo"  
    String getModelo() {  
        return modelo;  
    }  
  
    // Setter para modificar o valor do atributo "modelo"  
    void setModelo(String novoModelo) {  
        modelo = novoModelo;  
    }  
}
```

Importância dos atributos e valores

- **Caracterização:**

Atributos e seus valores caracterizam e distinguem objetos.

- **Armazenamento de informações:**

Atributos permitem armazenar informações importantes sobre objetos.

- **Estado dinâmico:**

Os valores dos atributos podem mudar ao longo do tempo, representando o estado dinâmico do objeto.



Vamos praticar?

Peguem uma folha para realizar a atividade.

Atividade

15 minutos

Criem um mapa mental sobre atributos e valores e sua sintaxe. Lembrem de detalhá-lo para lhes auxiliar no momento de criar um código.



Bloco 5

O que vem à sua mente quando
ouve falar sobre classes e objetos?



Declaração e instanciação

Declaração

Um objeto é uma instância de uma classe. A declaração envolve a criação de uma variável que representa o objeto.

Instanciação

Instanciar é criar uma cópia única da classe. Usa-se o operador *new* para alocar memória.

java

```
Classe objeto = new Classe();
```

Acessando atributos e métodos

Notação ponto

Usada para acessar membros de um objeto.
Formato: 'objeto.membro'

Atributos

Representam características do objeto. Acessados através da notação ponto.

Métodos

Representam comportamentos do objeto. Também acessados via notação ponto.

java

```
Carro carroA = new Carro();
carroA.modelo = "Sedan";
carroA.acelerar();
```

Vantagens da notação ponto

- **Organização:**

Mantém os membros de um objeto agrupados.

- **Leitura clara:**

Facilita a leitura e compreensão do código.

- **Clareza semântica:**

Indica a relação entre o objeto e o membro.

Vamos praticar?

Dividam-se em grupos para realizar a atividade.

Primeiro momento

20 minutos

Criem um pequeno programa que declare e inicialize objetos de diferentes classes, acessando seus atributos e métodos. Podem utilizar os mapas mentais já construídos para ajudar na atividade.

Segundo momento

5 minutos

Mostrem o código para a turma.



Bloco 6

Oi, faltoso!

Contem para a turma, como se estivesse falando para um amigo que faltou ao bloco passado, um pouco sobre conceitos de Programação Orientada a Objetos.



Vamos praticar?

Dividam-se em grupos para realizar a atividade.

Atividade

30 minutos

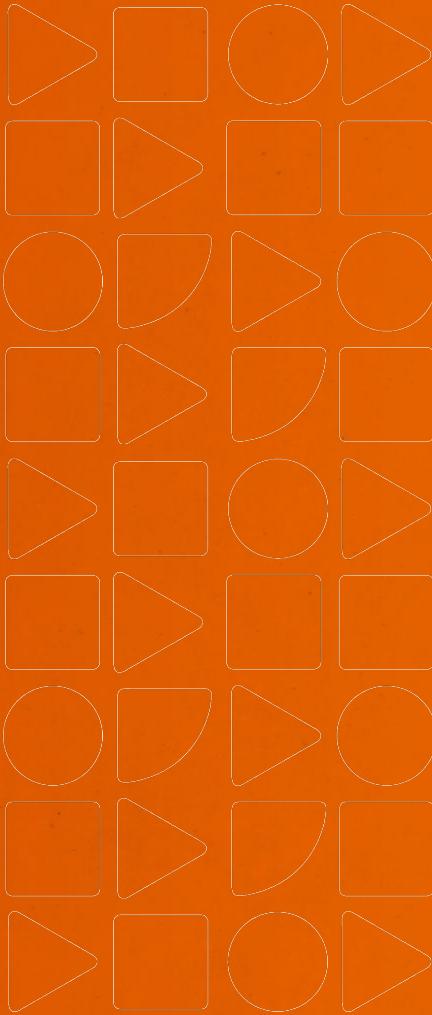
Criem um pequeno programa que tenha uma classe "Carro" com atributos como marca, modelo e método para ligar o carro. Podem utilizar os mapas mentais já construídos para ajudar na atividade.



Fechamento

Diga uma palavra sobre o que você achou da aula.





Referências Bibliográficas

PROZ EDUCAÇÃO. *Apostila de Desenvolvimento de Aplicações*. 2023.