

Classes, encapsulamento e construtores

P. O. O.

Prof. Grace

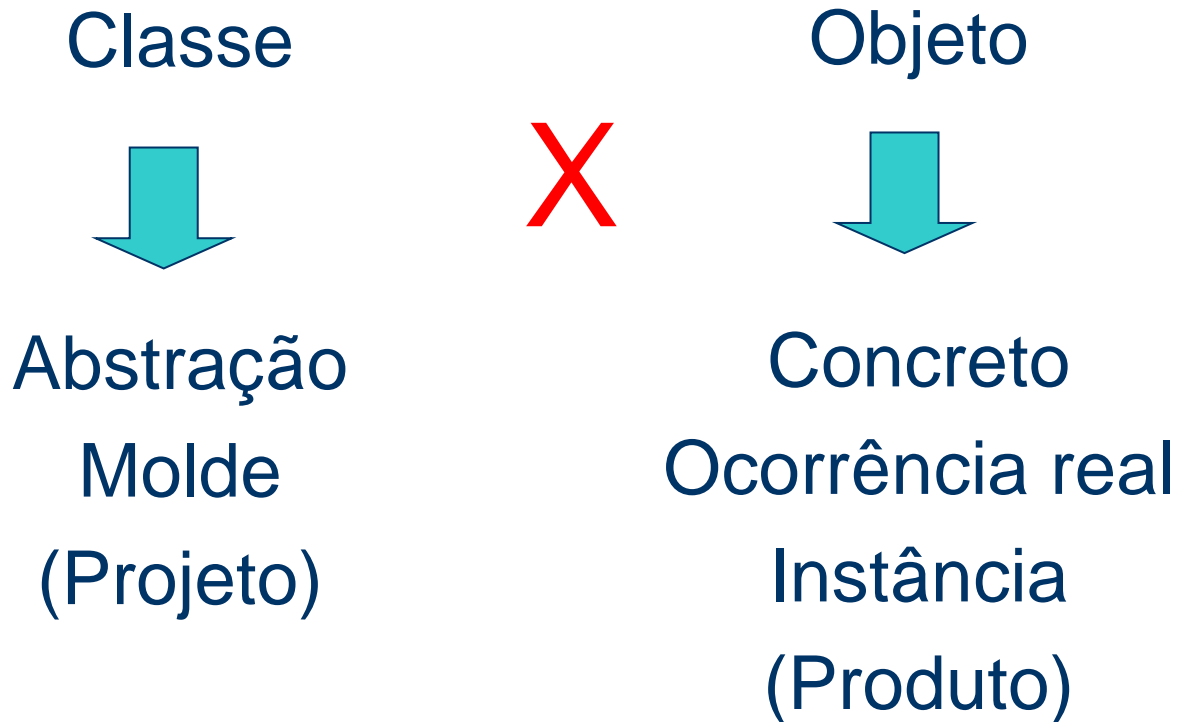
Aulas Anteriores

- Conceitos OO/ Plataforma Java/ Jgrasp
- Entrada e Saída de Dados
- Estruturas de decisão
- Estruturas de repetição

Agenda para Aula 04

- Modelagem e implementação de Classes
- Métodos e atributos
- Encapsulamento
- Construtor
- Exercícios

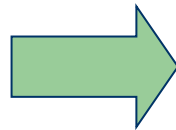
Classes e Objetos



Composição de uma Classe

Uma classe é composta por:

- Comportamentos ou operações
- Características ou atributos
(modelo)



- Métodos ou funções
- Dados ou variáveis
(implementação)

Exemplos usados até o momento

- Classe: **String**

```
String s;  
s = "abcd";  
char c = s.charAt( i );
```

- Classe: **Scanner**

```
Scanner entrada;  
entrada = new Scanner(System.in);  
int i = entrada.nextInt( );
```

Exemplos de Classes e Objetos

- No contexto de uma REDE SOCIAL, como modelar (dados/ atributos e operações/ métodos):

Usuário da rede social:

Dados (atributos)

Operações (métodos)



Exemplos de Classes e Objetos

- No contexto de um BANCO, como modelar (dados/ atributos e operações/ métodos):

Conta bancária:

Dados (atributos)

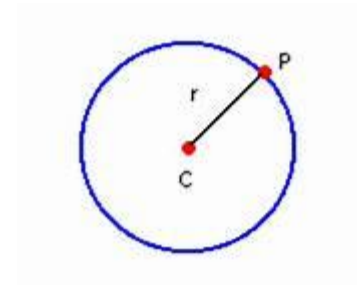
Operações (métodos)



Exemplo: Círculo

(Contexto: cálculos geométricos)

- Classe: Círculo
- Atributos (variáveis de instância)
 - Raio
- Métodos (tarefas)
 - Alterar (set)/ informar (get) raio
 - Calcular diâmetro
 - Calcular área
 - Calcular circunferência
 - Exibir Dados

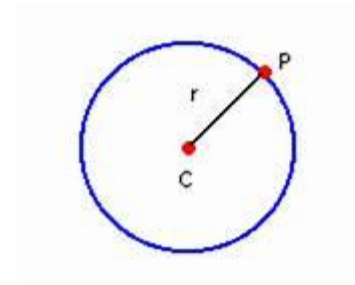


Dúvida: por que diâmetro, área e circunferência não são atributos?

Exemplo: Círculo

(Contexto: sw de desenho)

- Classe: Círculo
- Atributos (variáveis de instância)
 - Raio
 - Posição (x, y)
 - Cor borda
 - Espessura borda
 - Preenchimento
- Métodos (tarefas)
 - Desenhar
 - Copiar
 - Alterar (set)/ informar (get) raio (e demais atributos)



Implementando uma Classe Java

```
public class nome_classe
{
    // atributos
    // métodos
}
```

- Armazenar em um arquivo que tenha o mesmo **nome da classe**
- Extensão do arquivo => **.java**

Exemplo: Classe Circulo

```
//Declaração da classe Circulo.java
```

```
public class Circulo
```

```
{
```

```
    // atributo privado  
    private double raio;
```

```
    // método alterar raio  
    public void setRaio(double r)  
    {  
        raio = r;  
    }
```

```
    // método informar raio  
    public double getRaio()  
    {  
        return raio;  
    }
```

```
    // método exibir dados  
    public void exibeDados()  
    {  
        System.out.println("Raio: " + getRaio());  
    }
```

```
} // fim da classe
```

Atributos

Métodos
públicos

Chaves

Métodos

- Cabeçalho do método composto por:
 - Modificador de acesso (public, private)
 - Tipo de retorno (void, int, double, etc.)
 - Nome do método
 - Parênteses (argumentos, se tiver)
- Corpo do método deve ser descrito entre chaves “{ }”

Exemplos Implementados

```
// método alterar raio
public void setRaio(double r)
{
    raio = r;
}
```

```
// método informar raio
public double getRaio()
{
    return raio;
}
```

```
// método exibir dados
public void exibeDados()
{
    System.out.println("Raio: " + getRaio());
}
```

Implementando Classe Círculo

Tempo: 15 minutos

Material disponível no Teams

