

Classes e Métodos

Métodos

Assim como classes, para definirmos o **nome do método** existem algumas regras:

- ✓ Não pode conter espaços
- ✓ Deve ser iniciado por uma letra ou '_' ou '\$'
- ✓ Recomenda-se não utilizar acentos
- ✓ Pode conter números
- ✓ Não pode ser uma palavra reservada da linguagem Java

Um método não pode ser definido fora do corpo da classe.

Mecanismo de comunicação

Aumentam a facilidade!

- Reduzem a complexidade
 - ✓ Encapsulam a informação
 - ✓ Minimizam o tamanho do código

- Aumentam a manutenibilidade e a correção
 - ✓ Evitam duplicação do código
 - ✓ Limitam o efeito das mudanças
 - ✓ Promovem a reutilização do código

Diminuem o custo!

Métodos

- ✓ Mecanismo de comunicação
- ✓ Assinatura
- ✓ Sem retorno
- ✓ Com retorno
- ✓ Parâmetros
- ✓ Visibilidade

Método Construtor

É um método especial, responsável por definir o estado inicial de um objeto no momento de sua criação:

- ✓ A assinatura de um construtor diferencia-se das assinaturas dos outros métodos, pois não tem nenhum tipo de retorno (nem void).
- ✓ O nome do construtor deve ser o próprio nome da classe.
- ✓ O construtor pode receber argumentos, como qualquer método.
- ✓ Mais de um construtor pode ser definido na classe.

Sintaxe geral:

qualificador + nome da classe + (parâmetros)

Exemplo: `public Pessoa()`

Método Construtor

Exemplo

```
public class Automovel {  
    private String cor;  
    private double preco;  
    private String modelo;
```

Construtor padrão

```
public Automovel() {  
    System.out.println("Construtor padrão da classe Automóvel.");  
}
```

Construtor com 2 parâmetros

```
public Automovel(String modelo, double preco) {  
    this.cor = "PRETA";  
    this.modelo = modelo;  
    this.preco = preco;  
}
```

Construtor com 3 parâmetros

```
public Automovel(String cor, String modelo, double preco) {  
    this.cor = cor;  
    this.modelo = modelo;  
    this.preco = preco;  
}
```

```
}
```

Exercícios

1. Implemente uma classe chamada Pedido, contendo:
 - **Atributos:** número, nome do cliente e descrição.
 - Um **construtor** com os parâmetros número e nome do cliente.
 - **Métodos para:** alterar cliente, alterar descrição, retornar cliente, retornar descrição e apresentar em tela todos os dados do pedido.
2. Crie uma aplicação que instancie um objeto da classe Pedido e teste suas funcionalidades.
3. Faça uma classe chamada Virus contendo pelo menos 3 atributos e três métodos do tipo construtor diferentes. Elabore a classe UsaVirus para testar as funcionalidades da classe Virus.