

Orientação a objetos

Métodos e construtores

Métodos

- Primeira classe em Java com atributos e métodos

```
3 public class Carro {  
4     public String marca;  
5     public String modelo;  
6  
7     public void alugar() {  
8         System.out.println("Carro "+marca+" "+modelo+" alugado!");  
9     }  
10  
11     public void devolver() {  
12         System.out.println("Carro "+marca+" "+modelo+" devolvido!");  
13     }  
14 }
```

- Qual o nome da classe? A classe possui atributos e métodos? Quais? O que faz a classe? Ela é executável?

Métodos

```
3 public class Locadora {  
4  
5     public static void main(String[] args) {  
6         Carro c = new Carro();  
7         c.alugar();  
8         c.devolver();  
9     }  
10 }
```

Métodos

- Deu algo errado?
 - Não temos nada nos atributos...
- Jogo rápido:
 - Defina algum valor para os atributos!

Métodos

- Jogo rápido:
- Defina algum valor para os atributos!

```
5 public class Locadora {  
6  
7     public static void main(String[] args) {  
8         Carro c = new Carro();  
9         c.marca = "VW";  
10        c.modelo = "Fusca";  
11        c.alugar();  
12        c.devolver();  
13    }  
14 }
```

Jogo rápido

- Implemente a classe LocadoraVeiculos para fazer a leitura do modelo e marca de um carro, instanciar a classe Carro.
- Crie um método na classe Carro para exibir os dados do veículo.

Jogo rápido

```
3 public class Carro {
4     public String marca;
5     public String modelo;
6
7     public void alugar() {
8         System.out.println("Carro "+marca+" "+modelo+" alugado!");
9     }
10
11    public void devolver() {
12        System.out.println("Carro "+marca+" "+modelo+" devolvido!");
13    }
14
15    public void exibirDados() {
16        System.out.println("Carro: "+marca+"\nModelo: "+modelo);
17    }
18 }
```

Jogo rápido

```
5 public class Locadora {
6
7     public static void main(String[] args) {
8         Carro c = new Carro();
9         Scanner sc = new Scanner(System.in);
10        System.out.println("Digite a marca do carro: ");
11        c.marca = sc.nextLine();
12        System.out.println("Digite o modelo do carro: ");
13        c.modelo = sc.nextLine();
14        c.exibirDados();
15        sc.close();
16    }
17 }
```


Métodos - Retornos

- Os métodos podem ou não retornar um valor.
- Se um método retorna um valor, é preciso especificar qual é o tipo desse valor na declaração do método.
- O valor é retornado usando a palavra-chave `return` seguida do valor desejado.
- Por exemplo, um método que calcula a soma de dois números `double` e retorna o resultado pode ser definido da seguinte forma:

Métodos - Retornos

```
3 public class Calculadora {  
4     double a, b;  
5     public double somar() {  
6         double resultado = a + b;  
7         return resultado;  
8     }  
9 }
```

Métodos - Retornos

```
3 public class Principal {  
4  
5     public static void main(String[] args) {  
6         Calculadora c = new Calculadora();  
7         double res;  
8         c.a = 10;  
9         c.b = 15;  
10        res = c.somar();  
11  
12        System.out.println("O resultado da soma é: "+res);  
13    }  
14  
15 }
```

Métodos - Parâmetros

- Os parâmetros são valores que são passados para o método quando ele é chamado.
- Eles são usados dentro do método para realizar a tarefa desejada.
- Na declaração do método, é preciso especificar os tipos dos parâmetros e os nomes que eles terão dentro do método.
- Por exemplo, o método de soma que recebe dois parâmetros do tipo double, chamados a e b, conforme a seguir:

Métodos - Parâmetros

```
3 public class Calculadora {  
4     public double somar(double a, double b) {  
5         double resultado = a + b;  
6         return resultado;  
7     }  
8 }
```

Métodos - Parâmetros

```
3 public class Principal {  
4  
5     public static void main(String[] args) {  
6         Calculadora c = new Calculadora();  
7         double res;  
8         res = c.somar(10, 15);  
9  
10        System.out.println("O resultado da soma é: "+res);  
11    }  
12  
13 }
```

Classes e métodos

- Crie um método para atribuir valores para marca e modelo do carro, sendo que esses valores são recebidos pelo método através de parâmetros.
- Após, implemente a classe LocadoraVeiculos para instanciar a classe Carro, chamar o método correspondente e passar os parâmetros necessários. Além de um método para atribuir os valores, crie um método que retorna os valores atribuídos.
- O resultado da execução da classe LocadoraVeiculos deve ser:
 - Dados do carro: VW Gol

Classes e métodos

```
3 public class Carro {
4     public String marca;
5     public String modelo;
6
7     public void configuraDados(String marcaCarro, String modeloCarro) {
8         marca = marcaCarro;
9         modelo = modeloCarro;
10    }
11
12    public String retornaMarca() {
13        return marca;
14    }
15
16    public String retornaModelo() {
17        return modelo;
18    }
19 }
```


Classes e métodos

```
5 public class Locadora {
6
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9         Carro c = new Carro();
10        String ma, mo;
11        System.out.println("Digite a marca do carro: ");
12        ma = sc.nextLine();
13        System.out.println("Digite o modelo do carro: ");
14        mo = sc.nextLine();
15        c.ConfiguraDados(ma, mo);
16        System.out.println("Dados do carro:"+c.retornaMarca()+" "+c.retornaModelo());
17    }
18 }
```

Classes e métodos

- Jogo rápido: crie um classe chamada Moto com três atributos (marca, modelo e cilindradas) e dois métodos (atribuir valores e retornar valores).
- Na classe LocadoraVeiculos, crie um objeto do tipo Carro e dois objetos do tipo Moto, sendo que os objetos serão criados de acordo com a solicitação desses dados ao usuário, via linha de execução.
- Após a criação dos objetos, utilize o método para retornar valores e exiba na tela o conteúdo dos objetos criados.

```
15 public class Moto {
16     public String marca;
17     public String modelo;
18     public int cilindradas;
19     public void atribuiMarca(String marcaMoto) {
20         marca = marcaMoto;
21     }
22     public void atribuiModelo(String modeloMoto) {
23         modelo = modeloMoto;
24     }
25     public void atribuiCilindradas(int cilindradasMoto) {
26         cilindradas = cilindradasMoto;
27     }
28     public String retornaDados() {
29         return marca+" "+modelo+" "+cilindradas;
30     }
31 }
```

```
5 public class Locadora {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         String ma, mo;
9         Carro c = new Carro();
10        System.out.println("Digite a marca do carro: ");
11        ma = sc.nextLine();
12        System.out.println("Digite o modelo do carro: ");
13        mo = sc.nextLine();
14        c.configuraDados(ma, mo);
15        System.out.println("Dados do carro: "+c.retornaMarca()+" "+c.retornaModelo());
16
17        String mam, mom;
18        int cilin;
19        Moto m = new Moto();
20        System.out.println("Digite a marca da moto: ");
21        mam = sc.nextLine();
22        System.out.println("Digite o modelo da moto: ");
23        mom = sc.nextLine();
24        System.out.println("Digite cilindradas da moto: ");
25        cilin = sc.nextInt();
26        m.atribuiMarca(mam);
27        m.atribuiModelo(mom);
28        m.atribuiCilindradas(cilin);
29
30        System.out.println("Dados da Moto: "+m.retornaDados());
31    }
32 }
```

Exercícios

- 1) Crie uma classe Disciplina que contenha os atributos (nome, carga horária e nome do professor) e os métodos de atribuir e recuperar esses valores. Mostre na tela:
 - a) O conteúdo original dos atributos
 - b) A opção para o usuário inserir os valores
 - c) A exibição do novo conteúdo dos atributos
- 2) Crie uma classe chamada Livro que contenha os atributos Título, Autor, Ano de publicação, Gênero e Emprestado (bool). Crie método que retorne as informações do livro. Crie também um método responsável por fazer o empréstimo do livro e outro para fazer a devolução. Faça o controle disto utilizando o atributo Emprestado.
- 3) Crie uma classe (chamada RealizaConta) para realizar as quatro operações aritméticas (+,-,/,*) entre 2 valores. Nesta classe, você deve ter dois atributos do tipo double. Cada operação aritmética deverá ter um método e retornar o valor do cálculo. Os dois valores deverão ser lidos utilizando o método de leitura de teclado. Além disso, essa leitura deverá ser realizada dentro da classe principal do programa, chamada Calculadora. Ou seja, a classe principal (Calculadora) lê os dois. Esta, por sua vez, efetua os cálculos e retorna os valores para serem exibidos na classe principal.