**Definição e Estrutura de uma Classe em Java**

* **Classe**: Um modelo para criar objetos que contém dados (atributos) e comportamentos (métodos) relacionados.
* **Estrutura de uma classe**:
  + **Nome da Classe**: Nome único da classe, que começa com letra maiúscula por convenção.
  + **Atributos**: Variáveis que armazenam dados associados à classe ou ao objeto.
  + **Métodos**: Funções que definem os comportamentos dos objetos, sendo capazes de manipular os atributos ou realizar operações.
  + **Construtor**: Método especial que é chamado para inicializar um objeto quando ele é criado. Em Java, o nome do construtor é igual ao nome da classe.

**Exemplo de estrutura básica de classe em Java:**

Copiarpublic class Carro {  
 // Atributos String cor;  
 int ano;  
  
 // Construtorpublic Carro(String cor, int ano) {  
 this.cor = cor;  
 this.ano = ano;  
 }  
  
 // Métodopublic void ligar() {  
 System.out.println("O carro está ligado.");  
 }  
}

**O que São Atributos em Java**

* **Atributos**: São as variáveis associadas a uma classe. Eles podem ser de dois tipos:
  + **Atributos de Instância**: São específicos para cada objeto criado a partir da classe. Cada instância pode ter valores diferentes para esses atributos.
    - Exemplo: String cor; (cada carro pode ter uma cor diferente).
  + **Atributos de Classe (ou Estáticos)**: São compartilhados por todas as instâncias de uma classe. Todos os objetos têm o mesmo valor para esses atributos.
    - Exemplo: static int rodas = 4; (todos os carros têm o mesmo número de rodas).

**Exemplo com Atributos de Instância e de Classe:**

Copiarpublic class Carro {  
 // Atributo de instância String cor;  
   
 // Atributo de classe (estático)static int rodas = 4;  
  
 // Construtorpublic Carro(String cor) {  
 this.cor = cor;  
 }  
}

**Resumo:**

* **Classe**: Define a estrutura e comportamento dos objetos.
* **Atributos**:
  + **Instância**: Dados específicos de cada objeto.
  + **Classe (Estático)**: Dados compartilhados por todos os objetos de uma classe.

Se precisar de mais algum exemplo ou explicação, só avisar!

**Definição e Estrutura de uma Classe em Java**

* **Classe**: Um modelo para criar objetos que contém dados (atributos) e comportamentos (métodos) relacionados.
* **Estrutura de uma classe**:
  + **Nome da Classe**: Nome único da classe, que começa com letra maiúscula por convenção.
  + **Atributos**: Variáveis que armazenam dados associados à classe ou ao objeto.
  + **Métodos**: Funções que definem os comportamentos dos objetos, sendo capazes de manipular os atributos ou realizar operações.
  + **Construtor**: Método especial que é chamado para inicializar um objeto quando ele é criado. Em Java, o nome do construtor é igual ao nome da classe.

**Exemplo de estrutura básica de classe em Java:**

Copiarpublic class Carro {  
 // Atributos String cor;  
 int ano;  
  
 // Construtorpublic Carro(String cor, int ano) {  
 this.cor = cor;  
 this.ano = ano;  
 }  
  
 // Métodopublic void ligar() {  
 System.out.println("O carro está ligado.");  
 }  
}

**O que São Atributos em Java**

* **Atributos**: São as variáveis associadas a uma classe. Eles podem ser de dois tipos:
  + **Atributos de Instância**: São específicos para cada objeto criado a partir da classe. Cada instância pode ter valores diferentes para esses atributos.
    - Exemplo: String cor; (cada carro pode ter uma cor diferente).
  + **Atributos de Classe (ou estáticos)**: São compartilhados por todas as instâncias de uma classe. Todos os objetos têm o mesmo valor para esses atributos.
    - Exemplo: static int rodas = 4; (todos os carros têm o mesmo número de rodas).

**Exemplo com Atributos de Instância e de Classe:**

Copiarpublic class Carro {  
 // Atributo de instância String cor;  
   
 // Atributo de classe (estático)static int rodas = 4;  
  
 // Construtorpublic Carro(String cor) {  
 this.cor = cor;  
 }  
}

**Resumo:**

* **Classe**: Define a estrutura e comportamento dos objetos.
* **Atributos**:
  + **Instância**: Dados específicos de cada objeto.
  + **Classe (Estático)**: Dados compartilhados por todos os objetos de uma classe.