

Atributos, métodos e construtores







Temas

1 Atributos

Construtores

3

Métodos



1 Atributos



Os atributos da classe Artigo

Os atributos de nossa classe são: descricao, preco_venda e estoque.

- descricao: String - preco_venda: Double - estoque: Int + Artigo(descricao: String, quantidade: int, preco: Double) + temEstoque: Boolean + consultarPreco: Double





Os atributos da classe Artigo em Java

Definimos os atributos como privados para preservar o encapsulamento.

```
public class Artigo{
    private String descricao;
    private double preco_venda;
    private int estoque;
}
```



2 Construtores



O Constructor da classe Artigo

- descricao: String - preco_venda: Double - estoque: Int + Artigo(descricao: String, quantidade: int, preco: Double) + temEstoque: Boolean + consultarPreco: Double





Os métodos da classe Artigo em Java

```
public class Artigo{
    private String descricao;
    private double preco venda;
    private int estoque;
         public Artigo(String descricao, int quantidade, double preco){
              this.descricao = descricao;
              this.preco venta = preco;
              this.estoque = quantidade;
```



```
public class Artigo{
    private String descricao;
    private double preco_venda;
    private int estoque;
```

public Artigo(String descricao, int quantidade, double preco){

```
this.descricao = descricao;
this.preco_venta = preco;
this.estoque = quantidade;
}
```

O construtor é um método que não possui nenhum tipo de dado, é chamado da mesma forma que a classe. Recebe como parâmetros os valores que deseja atribuir inicialmente aos atributos, ou seja, os valores iniciais. Ele pode ser usado para inicializar os atributos.





```
public class Artigo{
    private String descricao;
     private double preco venda;
     private int estoque;
          public Artigo(String descricao, int quantidade, double preco){
                                                Para diferenciar o atributo de
          this.descricao = descricao;
                                                descrição do parâmetro que tem o
         this.preco venta = preco;
                                                mesmo nome, usamos this.
          this.estoque = quantidade;
                                                Refere-se ao objeto ou instância com
                                                o qual você está trabalhando.
```

3 Métodos



Os métodos da classe Artigo

Artigo

- descricao: String
- preco_venda: Double
- estoque: Int
- + Artigo(descricao: String, quantidade: int, preco: Double)
- + temEstoque: Boolean
- + consultarPreco: Double

Os métodos de nossa classe são temEstoque () e consultarPreco()





Os métodos da classe Artigo em Java

```
public class Artigo{
     private String descricao;
    private double preco venda;
     private int estoque;
     public Artigo(String descricao, int quantidade, double preco){ ... }
     public boolean temEstoque(){
         return estoque;
    public double consultarPreco(){
         return preco_venda;
```



```
public class Artigo{
    private String descricao;
    private double preco venda;
    private int estoque;
    public Artigo(String descricao, int quantidade, double preco){ ... }
                                           Método temEstoque()
    public boolean temEstoque(){
                                           Retorna verdadeiro se o
         return estoque;}
                                           estoque for maior que zero
                                           e falso caso contrário.
    public double consultarPreco(){
         return preco venda;
```



```
public class Artigo{
     private String descricao;
     private double preco_venda;
    private int estoque;
     public Artigo(String descricao, int quantidade, double preco)
     public boolean temEstoque(){
         return estoque;}
     public double consultarPreco(){
                                              Método consultarPreco()
                                              Devolve o Preço de Venda
         return preco_venda;
```

DigitalHouse>