

# Documentação

---

Uma das maiores características da linguagem Java é que desde suas primeiras versões tínhamos em nossas mãos uma documentação rica e detalhada dos recursos da linguagem.

Conforme site oficial, podemos compreender e explorar todos os recursos organizados por pacotes e classes bem específicas sem nem mesmo escrever uma linha de código.

Hoje costuma-se afirmar que para se tornar um desenvolvedor nível avançado, é um requisito imprescindível adquirir a habilidade de compreender a documentação oficial da linguagem e dos frameworks que são incorporados nos projetos atuais.

Aqui temos o link da documentação de uma das principais classes da linguagem Java.

{% embed url="https://docs.oracle.com/javase/7/docs/api/java/lang/String.html" %} Documentação da classe String na versão 7 da linguagem {% endembed %}

## Tags

---

Mas e quais as informações que obtemos através de classes documentadas na linguagem ? Java Documentation é composto por tags que representam dados relevantes para a compreensão da proposta de uma classe e os conjunto de seus métodos e atributos conforme tabela abaixo:

Tag	Descrição
@autor	Autor / Criador
@version	Versão do recurso disponibilizado
@since	Versão / Data de início da disponibilização do recurso
@param	Descrição dos parâmetros dos métodos criados
@return	Definição do tipo de retorno de um método
@throws	Se o método lança alguma exceção

Abaixo iremos ilustrar a classe Calculadora com um exemplo de documentação destacando as **tags** mais utilizadas.

{% tabs %} {% tab title="Código" %}

```

/**
 * <h1>Calculadora</h1>
 * A Calculadora realiza operações matemáticas entre números inteiros
 * <p>
 * <b>Note:</b> Leia atentamente a documentação desta classes
 * para desfrutar dos recursos oferecidos pelo autor
 *
 * @author Gleyson Sampaio
 * @version 1.0
 * @since 01/01/2022
 */
public class Calculadora {
    /**
     * Este método é utilizado para somar dois números inteiros
     * @param numeroUm este é o primeiro parâmetro do método
     * @param numeroDois este é o segundo parâmetro do método
     * @return int o resultado deste método é a soma dos dois números.
     */
    public int somar(int numeroUm, int numeroDois) {
        return numeroUm + numeroDois;
    }
}

```

{% endtab %}

{% tab title="Guia" %}!{% endtab %}{% endtabs %}

## Tipos de comentários

{% tabs %}{% tab title="One Line" %}

```
// Olá, eu sou um comentário em uma única linha
```

{% endtab %}

{% tab title="Mult Line" %}

```

/* Olá,
 * Eu sou um comentario
 * que posso ser mais detalhadod
 * quando necessário
 */

```

{% endtab %}

{% tab title="Documentation" %}

```
/**
 * Estas duas estrelinhas acima
 * é para identificar que você
 * pretende elaborar um comentário
 * a nível de documentação.
 * Que incrível !!!
 */
```

```
{% endtab %} {% endtabs %}
```

{% hint style="danger" %} Um comentário não possui a finalidade de **amenizar** um algoritmo não estruturado conforme as convenções da linguagem. {% endhint %}

```
/*
 * Este método foi elaborado as pressas
 * por isso eu abreviei o nome das variáveis
 * mas olha, ele tem a finalidade somar ou multiplicar
 * dois números
 * /
public int somaMultiplica (int n, int x, String m){
    int r = 0; // r é igual ao resultado
    if (m == "M"){ // M= multiplicação
        r= n * x;
    }else{
        // se não soma mesmo
        r = n + x
    }
    return r;
}
```

## Javadoc

**Javadoc** é um gerador de documentação criado pela [Sun Microsystems](#) para documentar a [API](#) dos programas em [Java](#), a partir do [código-fonte](#). O resultado é expresso em [HTML](#). É constituído, basicamente, por algumas marcações muito simples inseridas nos comentários do programa.

Este sistema é o padrão de documentação de classes em Java, e muitas dos [IDEs](#) desta linguagem irão automaticamente gerar um Javadoc em [HTML](#).

```
{% embed url="https://pt.wikipedia.org/wiki/Javadoc" %}
```

Criando nossa documentação no formato html para disponibilizar via web.

```
// No terminal execute o comando abaixo

javadoc -encoding UTF-8 -docencoding ISO-8859-1 -d ../docs src/*.java
```

# #