

Orientação a Objetos 1

Classe String

Prof. MSc. Vinícius Camargo Andrade

vcandrade@utfpr.edu.br

Departamento Acadêmico de Informática
Universidade Tecnológica Federal do Paraná

Classe String

Classe String

*A Classe String serve para **representar e manipular** cadeias de caracteres.*

Classe String

A declaração é semelhante a *declaração de variáveis*:

```
String autor = "Vinicius de Moraes";
```

```
String autor = new String("Vinicius de Moraes");
```

Classe String

Há duas maneiras diferentes de declará-las, além de métodos para manipulá-las;

```
String autor = "Vinicius de Moraes";
```

```
String autor = new String("Vinicius de Moraes");
```

Classe String

*Os caracteres ficam **armazenados** em variáveis declaradas na classe e encapsuladas no objeto quando ele é criado.*

```
String autor = "Vinicius de Moraes";
```

```
String autor = new String("Vinicius de Moraes");
```

Classe String

O acesso se dá *indiretamente* por meio do objeto.

```
String autor = "Vinicius de Moraes";
```

```
String autor = new String("Vinicius de Moraes");
```

Métodos

Classe String

Métodos

Conforme mencionado, as classes nativas da linguagem possibilitam ao desenvolvedor utilizar funcionalidades já implementadas, como bibliotecas em outras linguagens.

Métodos

- *boolean equals(String s)*
- *char charAt(int n)*
- *String trim()*
- *int length()*
- *int indexOf(char c)*
- *int lastIndexOf(char c)*
- *String substring(int n1, int n2)*
- *String toLowerCase()*
- *String toUpperCase(char c1)*
- *String replace(char c1, char c2)*
- *String [] split(String s)*
- *static String valueOf(int n)*

Equals

*Retorna um valor booleano verdadeiro se a String que executa o método for **igual** a String “s”.*

Equals

```
public class Teste {  
    public static void main(String[] args) {  
        String saudacao = "olá";  
        String s1 = "olá";  
  
        if(s1.equals(saudacao)) {  
            System.out.println("são iguais");  
        } else {  
            System.out.println("não são iguais");  
        }  
    }  
}
```

Testes (run) ×	Git - POOEngenhariaEletronica - master	Browser do Reposi
run: são iguais CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo)		

CharAt

*Retorna o **caractere** correspondente a posição “n” da String.*

CharAt

```
public static void main(String[] args) {  
  
    String alfabeto = "abcdefghijklm...";  
  
    System.out.println(alfabeto.charAt(5));  
}  
}
```

- Testes (run) ×	Git - PO0EngenhariaEletronica - master	Browser do Repositório Git	Resultados da Pesquisa
run: f CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)			

Trim

Retorna uma nova String eliminando os espaços em branco que haviam antes e depois da String.

Trim

```
public class Teste {  
    public static void main(String[] args) {  
        String alfabeto = "    abcdefghijklm...    ";  
        System.out.println("<<" + alfabeto + ">>");  
        System.out.println("<<" + alfabeto.trim() + ">>");  
    }  
}
```

a - Testes (run) ×	Git - POOEngenhariaEletronica - master	Browser do Repositório Git	Resultados da Pesquisa
--------------------	--	----------------------------	------------------------

```
run:  
<<    abcdefghijklm...    >>  
<<abcdefghijklm...>>  
CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)
```


Length

Retorna o comprimento da String.

Length

```
public class Teste {  
    public static void main(String[] args) {  
        String alfabeto = "abcdefghijklmnopqrstuvwxyz";  
        System.out.println(alfabeto.length());  
    }  
}
```

a - Testes (run) ×	Git - POOEngenhariaEletronica - master	Browser do Repositório Git	Resultados da Pesquisa
run: 26 CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)			

IndexOf

Retorna a primeira ocorrência de um determinado caractere.

IndexOf

```
public class Teste {  
    public static void main(String[] args) {  
        String universidade = "Universidade Tecnológica Federal do Paraná";  
        System.out.println(universidade.indexOf('a'));  
    }  
}
```

- Testes (run) ×	Git - POOEngenhariaEletronica - master	Browser do Repositório Git	Resultados da Pesquisa
run: 9 CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)			

LastIndexOf

Retorna a última ocorrência de um determinado caractere.

LastIndexOf

```
public class Teste {  
    public static void main(String[] args) {  
        String universidade = "Universidade Tecnológica Federal do Paraná";  
        System.out.println(universidade.lastIndexOf('a'));  
    }  
}
```

run - PrimeiroProjeto (run) ×

run:
39
BUILD SUCCESSFUL (total time: 0 seconds)

Substring

*Retorna **uma parte** da String iniciando em $n1$ e terminando em $n2$.*

Substring

```
public class Teste {  
    public static void main(String[] args) {  
        String alfabeto = "abcdefghijklmnopqrtsuvwxyz";  
        String s1 = alfabeto.substring(5, 18);  
        System.out.println(s1);  
    }  
}
```

a - Testes (run) ×	Git - PO0EngenhariaEletronica - master	Browser do Repositório Git	Resultados da Pesquisa
--------------------	--	----------------------------	------------------------

```
run:  
fghijklmnopqr  
CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)
```


ToLowerCase

*Retorna a própria String com todos os caracteres convertidos para **minúsculo**.*

ToLowerCase

```
public class Teste {  
    public static void main(String[] args) {  
        String nomeUniversidade = "Universidade Tecnológica Federal do Paraná";  
        System.out.println(nomeUniversidade.toLowerCase());  
    }  
}
```

- Testes (run) ×	Git - POOEngenhariaEletronica - master	Browser do Repositório Git	Resultados da Pesquisa
------------------	--	----------------------------	------------------------

```
run:  
universidade tecnologica federal do parana  
CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)
```

ToUpperCase

*Retorna a própria String com todos os caracteres convertidos para **maiúsculo**.*

ToUpperCase

```
public class Teste {  
    public static void main(String[] args) {  
        String nomeUniversidade = "Universidade Tecnológica Federal do Paraná";  
        System.out.println(nomeUniversidade.toUpperCase());  
    }  
}
```

Testes (run) ×	Git - POOEngenhariaEletronica - master	Browser do Repositório Git	Resultados da Pesquisa
run: UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)			

Replace

Retorna a String com os determinados caracteres substituídos por outros.

Replace

```
public class Teste {  
    public static void main(String[] args) {  
        String nomeUniversidade = "Universidade Tecnológica Federal do Paraná";  
        System.out.println(nomeUniversidade.replace('a', 'A'));  
    }  
}
```

Saída × Resultados da Pesquisa

Console do Depurador × JavaApplication2 (run) ×

run:
UniversidAde TecnológicA FederAl do PARAná
CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)

Split

Retorna um vetor de String contendo substrings que foram divididas de um String principal. O critério da divisão é enviado por argumento para o método.

Split

```
public class Teste {  
    public static void main(String[] args) {  
        String universidade = "Universidade Tecnológica Federal do Paraná";  
        String [] stringDividida = universidade.split(" ");  
        System.out.println(stringDividida[0]);  
        System.out.println(stringDividida[1]);  
        System.out.println(stringDividida[2]);  
        System.out.println(stringDividida[3]);  
        System.out.println(stringDividida[4]);  
    }  
}
```

PrimeiroProjeto (run) ×

```
run:  
Universidade  
Tecnológica  
Federal  
do  
Paraná  
BUILD SUCCESSFUL (total time: 0 seconds)
```


static valueOf

*Retorna a representação de string do argumento
int/double/boolean/...*

static valueOf

```
2
3 public class Teste {
4
5     public static void main(String[] args) {
6
7         int numeroInteiro = 100;
8         double numeroDouble = 50.0;
9         boolean variavelBooleana = true;
10
11         String inteiroConvertido = String.valueOf(numeroInteiro);
12         String doubleConvertido = String.valueOf(numeroDouble);
13         String booleanaConvertida = String.valueOf(variavelBooleana);
14
15         System.out.println(inteiroConvertido);
16         System.out.println(doubleConvertido);
17         System.out.println(booleanaConvertida);
18     }
19 }
```

<

Properties Console X

<terminated> Teste (33) [Java Application] C:\Users\vinic\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe (12 de a

100
50.0
true

Exercício 1

Exercício 1

Dado uma entrada de dados do tipo String, o sistema deve:

- *Retornar o caractere da posição 3;*
- *Retornar o número total de caracteres;*
- *Imprimir a mesma String em letras minúsculas;*
- *Dividi-la em várias substrings. O critério de divisão deve ser o espaço " ".*
- *Solicitar para que o usuário informe um segunda String. Após receber esta entrada de dados, o sistema deve compará-la com a primeira, exibindo uma mensagem se forem iguais.*

Exercício 2

Exercício 2

Solicite ao usuário os seguintes valores:

- *Um valor do tipo **inteiro**;*
- *Um valor do tipo **double**;*

*Após esta etapa, converta-os para variáveis do tipo **String** e os apresente em tela.*